SESSION

KNOWLEDGE DISCOVERY AND LEARNING

Chair(s)

Dr. Raymond A. Liuzzi
Raymond Technologies
USA
Drs. Peter M. LaMonica and Todd Waskiewicz
US Air Force Research Lab
USA

Collaborative Information Analysis & Visualization for Knowledge Discovery

P. Benjamin, K. Madanagopal, C. Wu, B. Gopal, and J. Hamilton
[1]Knowledge Based Systems, Inc., College Station, TX, USA

Abstract - *This paper describes the methods and an application architecture for knowledge discovery using collaborative information analysis and visualization. First, the challenges that motivate this research are outlined. The requirements on a solution are then described. Next we outline a solution method. Architecture examples are provided to help illustrate how automation support is provided for the activities detailed in the method. Finally, opportunities for further research and development are summarized. The main benefits of this research include (i) an improved ability to rapidly gain shared situational awareness through the use of collaborative analysis method, and; (ii) a reduction in the time needed to generate decision-enabling information from large, multi-source data.*

Keywords: Collaborative Information Analysis, Knowledge Discovery, Semantic Technologies, Visualization Methods, Visual Data Analytics.

1 Motivations

Decision makers are slowly drowning in a flood of human and computer generated information. The "paperless office," the World Wide Web, and advances in document management and analysis technologies are making it increasingly difficult to identify and utilize key information. Improvements in knowledge discovery, semantic modeling, natural language processing, and collaboration may provide significant leverage to address this problem.

The knowledge discovery process is a guided search in which the knowledge of the domain experts and an accurate understanding of the operational characteristics of the domain are critical. Today's decision-makers use tools that rely upon creating knowledge "on the fly" that ignore or are unable to take advantage of the accumulated collection of knowledge (this exists in a variety of forms that are often difficult to access). There is a need for methods that can cost effectively leverage and re-use domain knowledge and integrate this knowledge with the capabilities of analytical search methods (particularly in the context of re-occurring decision situations). Data analytics models typically involve dealing with aspects of the domain that are often hidden or invisible to the casual observer. Hence, a crucial aspect of the knowledge discovery process involves communication between the domain expert and the data analytics expert. Because of the inherent inefficiency of this communication process, knowledge discovery activities are often time and effort intensive.

The act of making a decision is an iterative and interactive process that employs knowledge, various information exploration strategies, and a degree of creativity. Within this process, many theories and paths of reasoning will be employed, altered, proven and disproven until a conclusion is reached. Often, the productivity and effectiveness of this discovery process is dampened due to the high degree of complexity of the analytical model development process. The process is also strongly hindered by the lack of formal methods in combining the various sources of knowledge that are available to "mine" and "fuse." Data sources range from the qualitative and quantitative domain knowledge that is available with the domain experts, to the automated multi-dimensional sources of data that often exhibit discovery-inhibiting factors such as data redundancy, missing information, counter-intuition and mis-information. A primary advance in the ease, efficiency and effectiveness of data analytics can be achieved through techniques that allow for combining the knowledge discovery from these multiple sources.

Because information analysts and decision makers seldom work in isolation, a shared understanding of decision goals and the subsequent sharing of knowledge and effort can significantly improve information analysis outcomes. Intelligent discovery mechanisms such as "search by example" may be used to derive an understanding of analysts' goals directly from their information analysis and search activities. Such an understanding may be used to connect collaborators working on similar problems or to alert members of a group about progress made by others in related areas.

Further, information analysts will sometimes repeat significant work performed by other analysts or miss key insights that are available to collaborators working on related tasks. Mechanisms that enable robust information exchanges between information analysts with similar goals would yield a transformative improvement in group productivity and the quality of the resulting information products.

2 Requirements on a solution

2.1 Support for exploratory data analyses

Exploratory and analytic visualization requirements may be grouped into three categories [1]: (i) direct mapping of data into visual display; (ii) mapping of sampling, aggregation, and generalization of data in a manner that summary information is preserved; and (iii) mapping computionally extracted patterns of data to reflect the underlying phenomena.

This type of classification scheme paves the way for applying visualizations from the perspectives of data, computing, target of visualization, and iteraction techniques at multiple levels of abstraction. Furthmore, these abstractions map to the manner in which humans perform reasoning. The following table shows the abstaction level from data, summary and pattern to knowledge (Table 1).

Table 1: Knowledge Discovery Abstraction Levels Mapped to Visualization Mechanisms

	Visualization relevant methods			
	Characteristic	*Computing*	*Visualization*	*Interaction*
Data	Types; Structures; Properties; Relations	Derived attributes; Smoothing; Interpolation; Projection;…	Data item	Query Focus Select
Summaries	Types; Properties;	Derivation of discrete aggreagtes, fields, flows, summaries…	Summary	Drill-down; Dysnamic re-aggregation; Query;…
Patterns	Types; Structures; Semantics; Perception	Extration of intial forms of patterns	Intial forms of patterns	Filter Arrange Combine;….
Knowledge	Cognition; Reasoning		Analysitic results	Knowledge Externalization and Reaging

The technical challenges associated with collaborative visualization are summarized in Table 2.

Table 2: Collaborative Work and Visualization Challenges

Aspect	Challenges in Collaborative Visualization
users	multiple participants with different roles
tasks	collaborative activity-centric
cognition	foraging and sense-making
results	consensus, shared insight
interaction	dynamic interaction techniques
visual presentations	transformation from inputs to visual forms
evaluation	social interaction

2.2 Support for sense making

The sensemaking model from [2] describes sensemaking as a three-phase process: (i) information foraging; (ii) sensemaking; and (iii) problem solving (Figure 1).

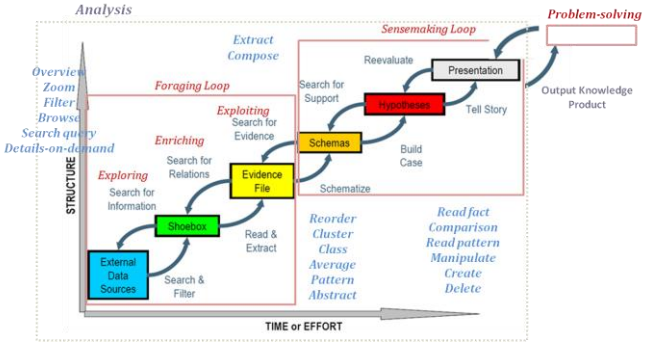


Figure 1: A Depiction of Sensemaking [2]

2.2.1 Information foraging

Foraging for food, an activity often performed by social packs of animals, inspires our idea of collaborative information foraging [2]. The idea is that collaborators will pool their findings (e.g., the discovery of relevant information) and support notification updates and information retrieval. Challenges include formalizing contributions, such as identifying trends or outliers of interest and positing explanatory hypotheses, and providing retrieval mechanisms by which others can access the contributions. Additional possibilities lie in analyzing and displaying activity traces to facilitate social navigation [3]--metaphorically similar to the scent trails left by ants foraging for food.

2.2.2 Sensemaking

The next phases of sensemaking focus on the tasks of the design and the population of information schemata. This could be conducted in a general form by enabling discussion among collaborators. One challenge is to synthesize the results of discussion into more accessible forms, such as summaries of arguments and evidence. The cost structure of these tasks could be further reduced and the integration of contributions facilitated by providing additional shared artifacts or external representations [4] for structuring group work. For example, an analytic sandbox of [6] provides a visual environment for spatially organizing hypotheses and positive and negative evidence, while [3] describes a system for the collaborative use of analytic evidence matrices.

2.2.3 Problem solving

The final phases of sensemaking involve problem solving and action. This may or may not take place within the collaborative analysis environment. Findings gained from analysis may serve as input to collaboration in other media, suggesting the need to both facilitate external access to the contents of the visual analysis environment and extracting content for use in other systems. If problem-solving and decision making are conducted within the system, aforementioned issues regarding communication, discussion, and consensus must be addressed.

3 Overview of solution concept

3.1 Exploratory data analytics and sensemaking

'Visual analytics' seeks to marry techniques from information visualization with techniques associated with the computational transformation and analysis of data. Information visualization itself forms part of the direct interface between user and machine. Information visualization amplifies human cognitive capabilities in several different ways as outlined in the following list.

- Increasing cognitive resources –
  - High-bandwidth hierarchical interaction
  - Parallel perceptual processing
  - Offload work from cognitive to perceptual system
  - Expanding working memory
  - Expanding storage of information
- Reducing search –
  - Locality of processing
  - High data density
  - Spatially-indexed addressing
- Enhancing the recognition of patterns –
  - Recognition instead of recall
  - Abstraction and aggregation
  - Visual schema for organization
  - Value, relationship, and trend
- Perceptual inference –
  - Visual representation makes some problem obvious
  - Graphical computations
- Perceptual monitoring of a large number of potential events, and
- Providing a manipulable medium that, unlike static diagram, enables the exploration of a space of parameter values.

Visual analytics comprises different mechanisms for combining analytics and visualizations, as outlined below.

1. Analytical reasoning techniques that let users obtain deep insights that directly support assessment, planning, and decision making;
2. Visual representations and interaction techniques that exploit the human eye's broad bandwidth pathway into the mind to let users see, explore, and understand large amounts of information simultaneously;
3. Data representations and transformations that convert all types of conflicting and dynamic data in ways that support visualization and analysis; and
4. Techniques to support production, presentation, and dissemination of analytical results to communicate information in the appropriate context to a variety of audiences.

3.2 Combining collaboration with visual analytics

[5] provides a characterization of 'collaborative visual analytics' by using a space-time matrix for classifying collaborative applications. The time axis indicates synchronous or asynchronous, and the space axis distinguishes same location or remote site (Figure 2). From our perspective, the team intends to address the importance of providing viewer-based, scalable visualizations. By adding the third axis, scale, to the matrix, a new scale-space-time matrix can reflect what collaborative visual analytics should address (Figure 2).

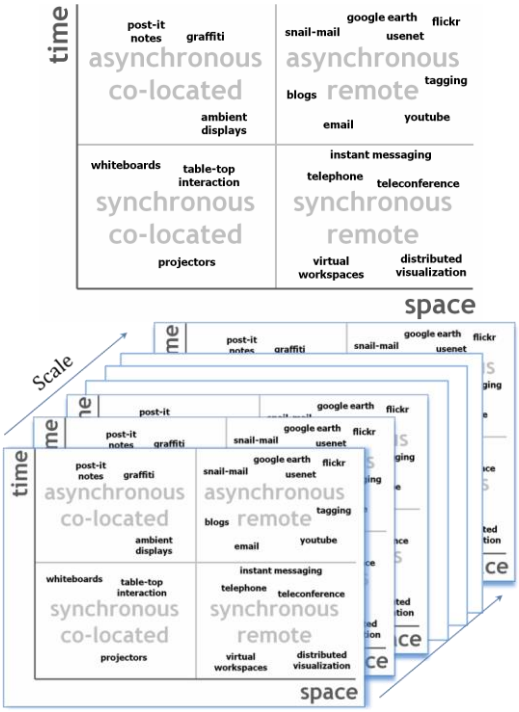


Figure 2: Key Idea: Scaleable, Collaborative Visual Analysis

4 A collaborative information analysis method

This section outlines a method for collaborative information analysis in support of knowledge discovery. Activities that are performed for collaborative information analysis and their inter-relationships are illustrated in Figure 3.

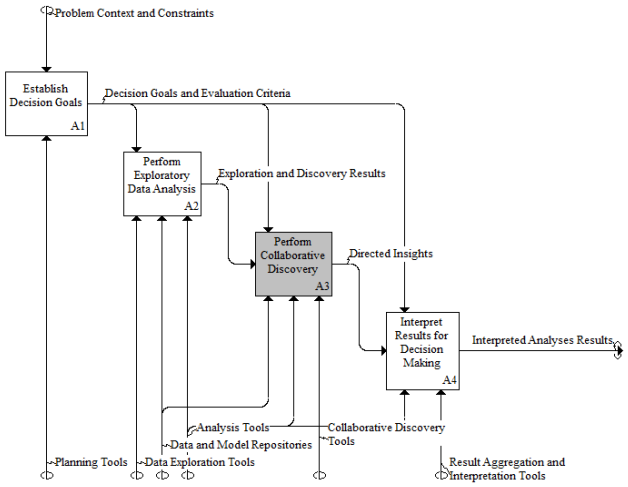


Figure 3: Collaborative Information Analysis Method

The method is now described in greater detail.

4.1 Establish decision goals

The first step is to define a set of decision goals. The decision goals often seek to generate information to enable the solving of a problem or to support better understanding of a situation. The decision goals provide the basis for analyzing and establishing information requirements in support of decision making.

4.2 Perform exploratory data analysis

This activity refers to a set of activities used to understand and analyze data that contains the information and knowledge needed to address the decision goals.

This activity involves multiple inter-related tasks, including (i) accessing and loading the data, (ii) pre-processing and transforming the data, (iii) validating the data, and (iv) visualizing the data for better understanding.

Broadly, data may be categorized as being structured (numeric, sensor) or unstructured (text). With structured data, one must address the 'legacy data' problem. A legacy system is most often dispersed in different tables (and databases). There is a need to Extract, Transform, and Load (ETL) data in a format suitable for downstream processing / analytics from these sources. Very often, this will involve accessing multiple data sources, setting references between multiple tables, and creating different views of data. Extract, Transform, and Load (ETL) methods are used to get this activity completed properly.

With unstructured data, a first step is to convert the native source format into a format that is suitable for automated text processing. XML-based formats are most common and lend themselves well to automated processing. The next step is to perform 'sentence boundary detection,' which results in the creation of sentence boundary marker tags. After this, 'tokenization' is performed: the creation of tokens (tags) for different groups of words to facilitate better downstream text processing.

After the data acquisition and preprocessing steps are completed, the data must be verified, validated, repaired, and transformed before it can used by data analytics and decision support applications. Errors in the input data could result from misspecification, omission, corruption, approximation, etc. In the context of data analytics and natural language processing applications, even a few incorrect data points could drastically change the intended pattern that these systems are expected to learn, resulting in poor model performance. Data validation involves data repair and data transformation.

4.3 Perform collaborative discovery

This important activity involves (i) generating semantically tagged information, (ii) performing semantic querying to generate decision enabling information, and (iii) providing collaboration and visualization aids to support better human cognition and sensemaking.

4.3.1 Semantic querying

A key capability is to allow end users to browse, query, and search the results of applying semantic information processing methods on multi-source data. An advanced query and natural language search interface is provided that runs on 'semantically tagged' data produced by a set of semantic data processors (using data analytics and natural language processing techniques). Two key features distinguish our approach to semantic information analysis: (i) ontology-driven query in which application-domain ontologies are used to contextualize the queries/search, leading to higher precision and recall, and (ii) example-based search in which searches are based on free form text ('example text') rather than being restricted with the use of a finite set of key words as the input. Figure 3 shows a conceptual architecture that shows an implementation that provides ontology-driven, example based semantic query and search functionalities.

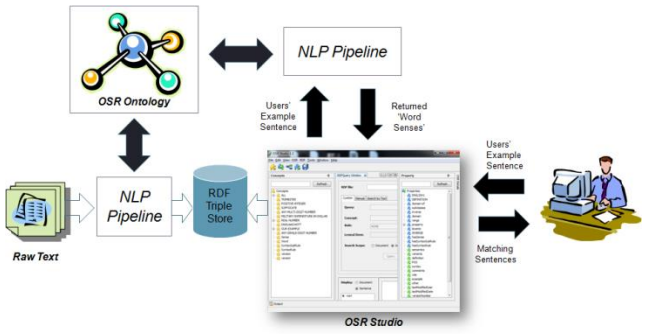


Figure 4: Using Semantic Querying to Support Sensemaking

The semantic querying capability may be used to support the generation and visualization of information to support

decision making. Specifically, analysis and visualization mechanisms are provided in order to help end users assess the results of the SPARQL queries applied to the tagged (RDF) data. For example, the generated analysis information might include reports about locations, people of interest (humans), key concepts, and frequency analysis of documents (Figure 5).

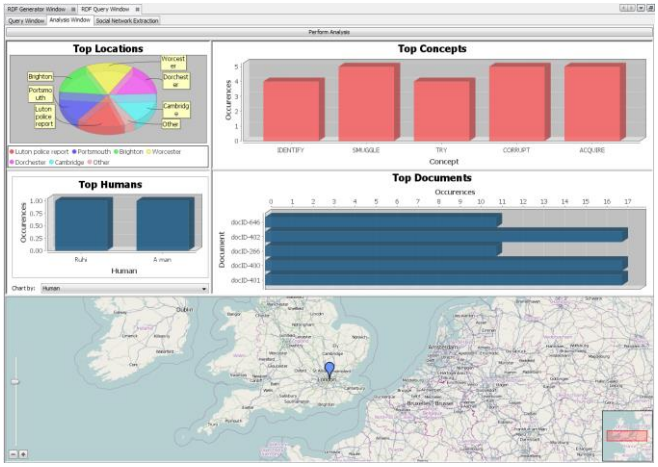


Figure 5: Example Semantic Query-based Analyses Displays

As part of the 'semantic analysis' performed within the text processing module, different 'senses' for the nouns are identified and disambiguated. For example, the terms 'a man' and 'Tom' will both be tagged with the label 'HUMAN' during the semantic information processing activity. The geo coordinates for the locations identified by this capability are extracted from the DBpedia data repository using the publicly available SPARQL endpoint. The DBpedia repository was produced through a community-wide initiative that extracted structured information from Wikipedia and made this information available on the web. DBpedia enables information retrieval from Wikipedia using SPARQL, and also enables cross linking from other web-based data sets to the Wikipedia data repository.

4.3.2 Social network extraction

With the disambiguated senses produced by the semantic analysis process, additional meaningful information ('semantic information') is available for different types of applications. With meaningful information stored in the form of RDF, the data can be analyzed in many different ways. For example, mechanisms are provided for the automated extraction, analysis, and visualization of social networks from the semantically tagged data (Figure 6).

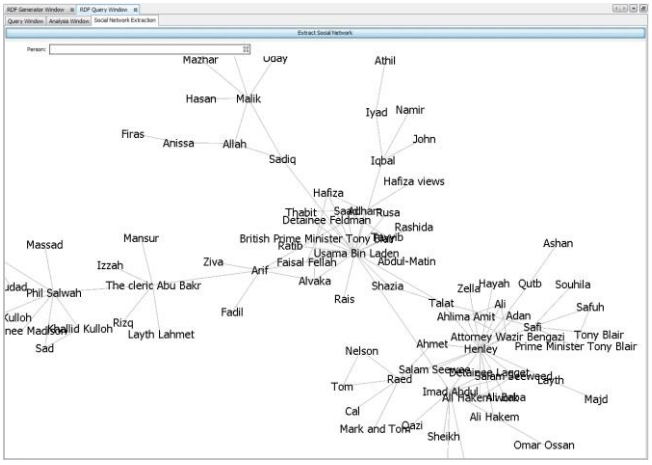


Figure 6: Example Social Network Display

4.3.3 Collaboration support for information analysis

Collaboration mechanisms are used to support situations involving multiple analysts/end users working on a common dataset with a set of shared tasks and objectives. The collaboration support essentially allows analysts to share semantic analyses-based information effectively. For example, suppose that Analyst A and Analyst B are using a shared dataset to perform analysis to support a common task. Suppose that Analyst A finds some interesting results based on a query and would like to share these results with Analyst B. He/she can directly share the search term and the associated results using automated collaboration mechanisms. This idea of 'collaborative semantic analysis' is shown in Figure 7.

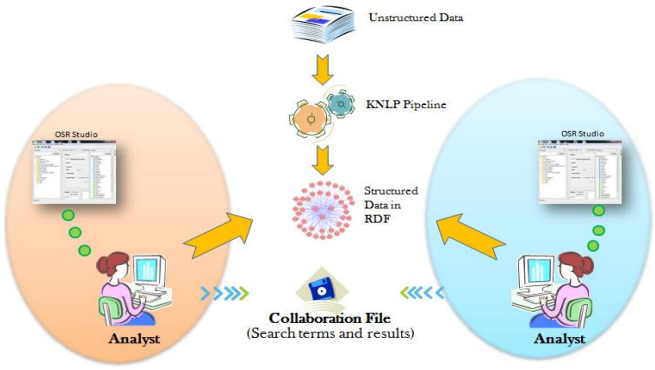


Figure 7: Collaborative Information Analysis

4.3.4 Visualization support for knowledge discovery from semantically tagged data

Collaborative visualization and analytics have been observed to be useful in supporting sense making from multi-source data. Decision making often involves multiple participants with different roles, expertise, and needs. Some of them may be map-based information experts that perform spatial information analysis; others could be information analysts focused on hostile activity timeline tracking. The users may or may not work at the same time and in the same location. The collaboration activity must facilitate the

generation of shared insights that must inform both work in progress and the final result. These results need also to be presented in intuitive, visual formats that can support further evaluation and collaborative foraging. During the process, interaction techniques can enhance productivity by providing various views from different perspectives. Ultimately the final results will be presented to decision makers in a streamlined form with/without duplicating the entire process (Figure 8).

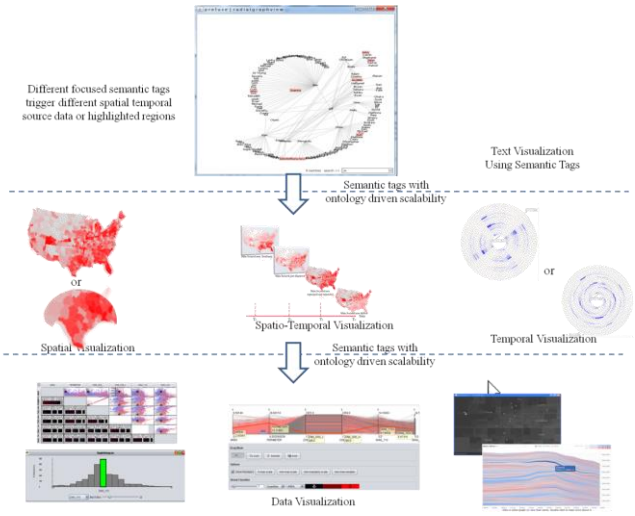


Figure 8: Semantic Tagging Enables Collaborative Visual Analytics

Whether users access synchronously or asynchronously to collaborative visualizations at the same location or different locations will greatly affect the design of information acquisition, presentation and interaction, and analytic results storage, retrieval, and sharing. In addition, analysts working asynchronously should be able to access up-to-date information without reputing their previous efforts. More importantly, any analytic results need to be scalable to meet the different roles of users. For instance, decision makers/managers often prefer visual presentations of 'final results.' In contrast, analysts often pay attention to detailed information, compare the notes with each other, and then perform sense-making judgments.

One of the common activities in any of kind of analysis is making sense of the body of data. Different cognitive and external resources are required in the process of sense making and these cognitive resources play a key role in reducing the cost of operations in information processing tasks. There are four key elements that are central the event related information discovery process:

- People
- Time of Event
- Location of Event and type of event
- People, time, location and event.

Visualization is commonly thought of as "forming a mental image" and is often used interchangeably with sense making. Picking the right set of visualizations and presenting it to the user within the context greatly impact the sense making process. We have incorporated various visualization techniques such as brushing and linking in order to help the user answer task specific questions. For instance, in the screenshot given below, the user would be able to see the time, social network and the location information in a single view and also would be able to filter information from any of the other views.

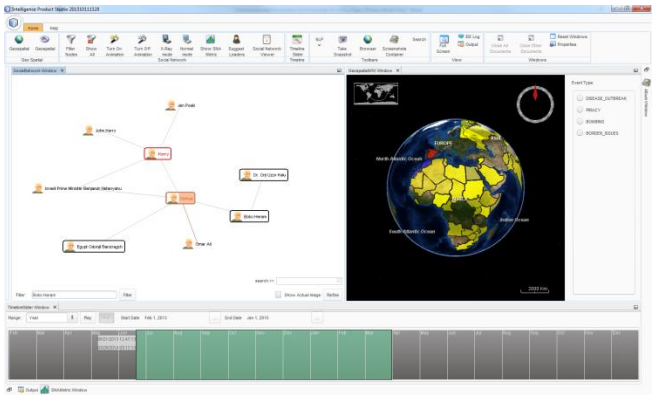


Figure 9: Information Discovery Visual Interface

One of the key features of our technique is capturing the reflection-in-action while the user is working with the data. This is done using the annotation capability. The user is able to take snapshots of the information and drop them in the annotator, which can be used for the 'reflection-on-action' process at the end of the analysis.

The human mind is powerful, but it often needs to be provided with the right set of resources to trigger it. Simulating control is one such feature that triggers the human mind to make powerful inferences that are difficult to do using computational methods. After collecting the relevant facts for the information discovery task, the simulation ('enactment') mechanism helps to step the user through the events sequentially. The simulation also helps in identifying 'patterns' of events. The example display in Figure 10 shows the state of a computer branch network at different snapshots in time. The top row of the figure represents the branch network and the bottom row represents the network in the main datacenter. The link in the top row represents the connection from the branch network to the main data center with the link strength representing the number of connections. The third snapshot on the top represents the branch that makes a large number of connections (thereby overloading the branch network). The bottom network represents the network diagram at the main datacenter. The third snapshot shows details of the nodes in the main datacenter that are accessed.

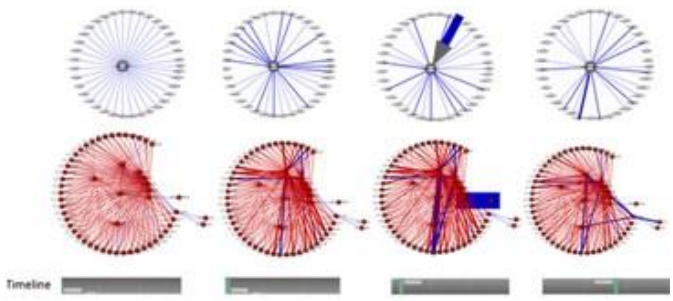


Figure 10: Simulation of Network Events

We also have a semantic document reader that is powered by a natural language processing engine. The semantic document reader enables the user to search through the document while disambiguating between possible 'senses' and highlighting the relevant content in the document. Unlike other search engines that only show the relevant sentences to the user, this capability displays the relevant document 'in context.'

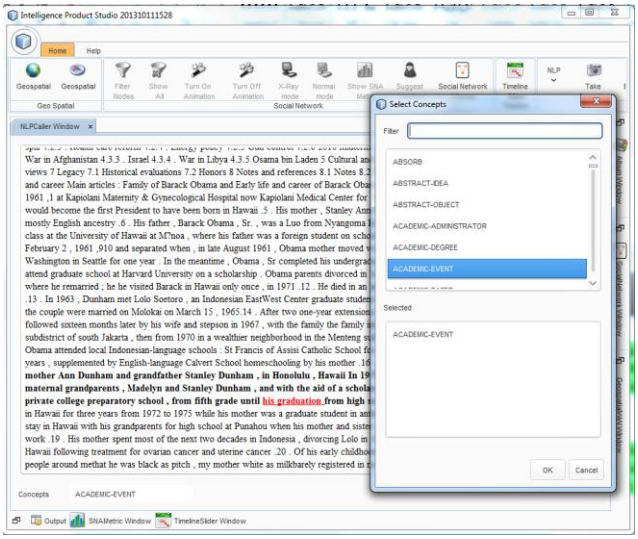


Figure 11: Semantic Document Reader

For example, in the example shown above, the user searched for instances of "ACADEMIC-EVENT," and the reader highlighted the sentence related to "graduation."

4.4 Interpret results for decision making

Once the collaborative information analysis is complete, the following tasks are performed: (i) integrate the analysis results from multiple sources, (ii) compile the integrated results into a format that is suitable for decision makers, and (iii) interpret the results for decision making. Integrating the analysis results from multiple sources often involves information fusion techniques. Decision makers often prefer to evaluate information in summary form or as structured reports, with mechanisms to 'drill-down' as needed. Interactive visualization mechanisms help decision makers obtain better cognitive understanding for the conveyed information, leading to timely and higher quality decisions.

5 Summary and opportunities for further R&D

This paper describes the motivations and a method for collaborative information analysis and visualization to support knowledge discovery. The ideas expressed in the paper have been implemented in multiple research and development initiatives. The following areas will benefit from continued research and development: (i) Reasoning Mechanisms to Support Collaborative Information Analysis; (ii) Information Fusion Methods to Support Information Fusion from Multi-modal Data; (iii) Truth Maintenance to Support Collaborative Information Analysis; and (iv) Automation Mechanisms to Assist Interpretation of Analysis Results to Support Decision Making.

6 References

[1] [Andrienko 2008] Natalia Andrienko, Gennady Andrienko, Peter Gatalsky, "Exploratory spatio-temporal visualization: an analytical review", Journal of Visual Languages and Computing 14 (2003) 503–541,

[2] [Pirolli 2006] Ed H. Chi, Peter Pirolli, "Social Information Foraging and Collaborative Search", HCIC Workshop; 2006 February.

[3] [Dourish 1994] Dourish P., Chalmers M., "Running Out of Space: Models of Information Navigation", Human Computer Interaction (HCI'94) 1994.

[4] [Russell 1993] Daniel M. Russell, Mark J. Stefii, Peter Pirolli, Stuart K. Card, "The Cost Structure of Sensemaking", 24-29 April 1993 INTERCHI'93.

[5] [Heer 2008] Jeffrey Heer, Maneesh Agrawala, "Design Considerations for Collaborative Visual Analytics", Information Visualization Journal, 7(1), 49–62, 2008,

[6] [Schroh 2006] Wright W., Schroh D., Proulx P., Skaburskis A., Cort B., "The sandbox for analysis: concepts and evaluation. In Proc.", ACM CHI 2006. 2006.

Artificial Intelligence and Radar Target Tracking

Gerard T. Capraro
Capraro Technologies, Inc., 401 Herkimer Road, Utica, NY 13502 USA

Abstract

An airborne ground looking radar sensor's performance may be enhanced by selecting and modifying algorithms adaptively as the environment changes. A brief presentation of an airborne intelligent radar system (AIRS) is provided. A description of the knowledge based tracker portion of AIRS is emphasized. Many tracking algorithms discard all information about a track once a track is dropped. Our approach maintains this information to enhance the algorithm's learning ability.

Introduction

The desire to anticipate, find, fix, track, target, engage, and assess, anything, anytime, anywhere (AF2T2EA4) by the US Air Force (USAF) will require changes to how we modify, build, and deploy radar and sensor systems. The US Air Force Research Laboratory (AFRL) is attacking these issues from a sensor and information perspective and has generated a way forward in their defining of layered sensing [1].

How can the US Air Force system of the future detect and identify threats and meet the implicit requirements of this scenario in a timely manner? We must, as a first step to full automation, implement the following ground breaking changes: place more compute intensive resources closer to sources of the information gathering – e.g. assign tasks to sensors to look for "triggers" created from intelligence surveillance and reconnaissance (ISR) sources, provide for the analysis of intelligence data automatically and without human involvement, move the human sensor operator from managing data - to managing actionable knowledge and sensor aggregation, and develop these "triggers" and rules for automatic assignment and management of heterogeneous sensors to meet dynamic and abstract requirements. This paper will address an airborne ground looking radar and how to use artificial intelligence (AI) to enhance its tracking performance.

Sensor performance may be enhanced by selecting algorithms adaptively as the environment changes. It has been shown [2-12], that if an airborne radar system uses prior knowledge concerning certain features of the earth (e.g. land-sea interfaces) intelligently, then performance in the filtering, detection and tracking stages of a radar processing chain improves dramatically. As an example the performance of an intelligent radar can be increased if the characteristics and location of electromagnetic interference, terrain features [9], mountainous terrain [10], and weather conditions are known. The Sensors Directorate of the USAF Research Laboratory conducted and sponsored research and development in the use of prior knowledge for enhancing radar performance, as did the Defense Advanced Research Project Agency (DARPA) under the Knowledge Aided Sensor Signal Processing Expert Reasoning (KASSPER) program.

One design of an intelligent radar system that processes information from the filter, detector, and tracker stages of a surveillance radar, investigated by the USAF and under the KASSPER program, was specifically designed for an Airborne Intelligent Radar System (AIRS). Futuristic advanced intelligent radar systems will cooperatively perform signal and data processing within and between sensors and communications systems while utilizing waveform diversity and performing multi-sensor processing, for reconnaissance, surveillance, imaging and communications within the same radar system. A high level description of AIRS is shown in Figure 1 and is described in detail, [8, 11], in the literature. This work has been extended to include metacognition and is illustrated in Figure 2. See [13, 14] for a more detailed description.

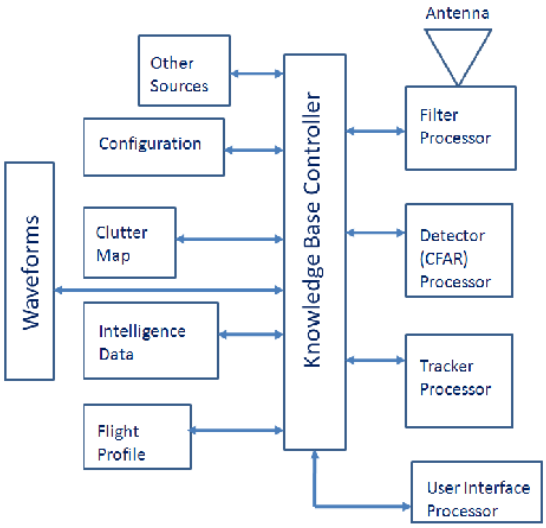


Fig.1. Airborne Intelligent Radar System (AIRS)

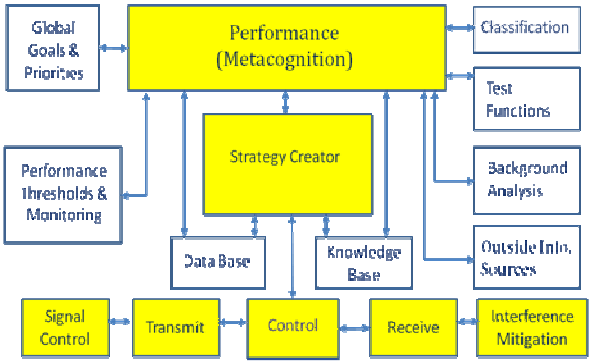


Fig. 2 Cognitive Radar Software Architecture

In this paper we wish to investigate the tracking portion of AIRS. See Figure 1. In [13] we addressed the filter and detector portions. This paper will extend our work of an AIRS architecture. We will present a AI overview of this tracking algorithm and some of its AI rules e.g. maneuver or obstacle rules and shadow rules. An AI logic structure for implementing these rules is discussed next and some additional rules for our AIRS design are provided.

The logic structure is independent of any tracking algorithm and can address aircraft or ground moving targets. It is compatible with the overall AIRS design and is modifiable. The thrust of this logic structure is to utilize as much auxiliary data (e.g. maps, other sensors, target kinematics, and radar platform characteristics) as possible to maintain individual identifiable tracks. With today's tracking algorithms if a track is dropped and another track is formed there is minimum effort expended to determine if the two tracks were formed from the same target. If a track is dropped algorithms, for the most part, do not investigate why and then use this information in enhancing the overall signal processing performance. Algorithms do not learn based upon their previous performances. They are memoryless once a track is dropped. The proposed logic structure presented herein addresses these issues and investigates the potential for building an AI based tracking algorithm.

Our current tracking algorithm has three separate instantiations. There is an uncoupled two state alpha beta filter with position and velocity component states, an uncoupled three state Kalman filter with position, velocity, and acceleration component states, and an extended four state Kalman filter with both x and y position and velocity component states. The tracker gathers reports, evaluates the reports and correlates them with known tracks, forms a correlation matrix and distance matrix, performs an association logic based upon nearest neighbor and oldest track, and performs track maintenance i.e. update extant track states, spawn new tentative tracks with unused reports and drops tracks with a state value of zero. A diagram illustrating the state logic is shown in Figure 3. A new tentative track is given a state of 1. If its projected position is detected again on the next coherent processing interval (CPI) it is given a state of 2, and so on. Once the target is in state 4 it is considered in a firm state as long as it is still detected for each subsequent CPI. Once in the firm state, if there are four consecutive CPIs in which the target is not detected (i.e. a Miss) then the track is dropped. It is our contention that once a tentative track exists then we should maintain its history even if it receives one or more misses. This is important in order to correlate false or dropped tracks with roads, or jammers, discretes, shadow regions, etc. This information is needed to feed back to the Knowledge Base Controller (KBC) shown in Figure 1.

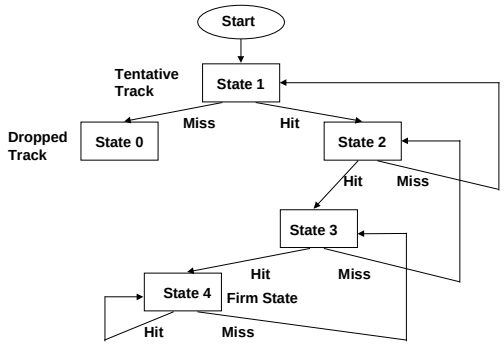


Figure 3. Integrating AI Rules

The following is a preliminary design of a logical structure to capture AI rules for the tracking portion of AIRS. It is by no means complete and does not address each of the numerous attributes for tracking any specific type of target (e.g. aircraft, Unattended Air Systems (UAS), ground vehicles, missiles) for all its possible scenarios embedded in all possible environments or clutter. It is constructed to work with a radar tracking filter such as alpha beta or Kalman. The logical structure is shown in Figure 4. It is an abstract model and will require numerous detail level designs before it can be coded and tested. The logic is described using alpha characters to indicate where in the structure we are referring. Throughout the description the use of outside data sources is illustrated and the addition or verification of data sources is presented.

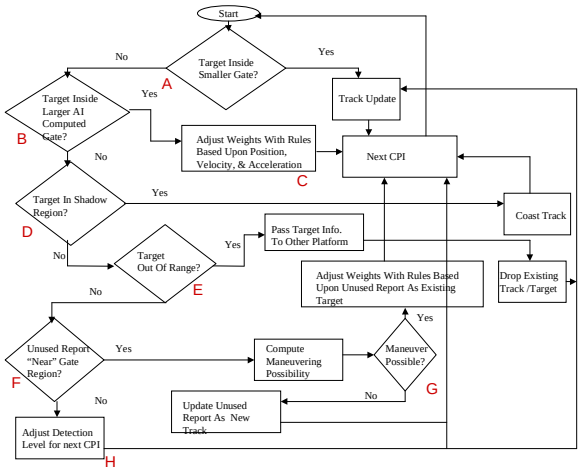


Figure 4. Logical Structure

Section A

Within this decision block (A) we are asking whether a detected target is within the gate of a known and therefore projected track. If the answer is yes then we simply update the track using the tracking filter of choice (e.g. Kalman). If however a target is detected and it is not within any projected track's gate (i.e. an unused report) then we need to determine whether it lies in a larger AI computed gate. The idea of using more than one size or variable size gate is discussed in the literature. Skolnik [15] suggests "The size of the small gate would be determined by the accuracy of the track. When a target does not appear in the small gate, a larger gate would be used whose search area is determined by the maximum acceleration expected of the target during turns." Brookner [16] states while discussing the g-h filter

"However, aircraft targets generally go in straight lines, rarely doing a maneuver. Hence, what one would like to do is use a Kalman filter when the target maneuvers, which is rarely, and to use a simple constant g-h filter when the target is not maneuvering. This can be done if a means is provided for detecting when a target is maneuvering. In the literature this has been done by noting the tracking-filter residual error, that is, the difference between the target predicted position and the measured position on the nth observation. The detection of the presence of a maneuver could be based either on the last residual error or some function of the last m residual errors. An alternative approach is to switch when a maneuver is detected from a steady-state g-h filter with modest or low g and h values to a g-h filter with high g and h values, similar for track initiation. This type of approach was employed by Lincoln Laboratory for its netted ground surveillance radar system. They used two prediction windows to detect a target maneuver. If the target was detected in the smaller window, then it was assumed that the target had not maneuvered and the values of g and h used were kept ... If the target fell outside of this smaller 3 sigma window but inside the larger window called the maneuver window, the target was assumed to have maneuvered."

Section B

These references were provided to indicate that the radar community has tried different approaches for varying the gate sizes for tracking maneuvering targets. The Kalman filter is more suited for maneuvering targets. However, a universal method for choosing a larger gate size because of a maneuver is not well established. If the larger gate is too large then multiple targets may occur within them. The maneuverability of a target is target dependent and may be human dependent and very unpredictable. What we are proposing is that the larger gate be built using AI techniques. Let the history of the target's flight and a priori knowledge about a potential target dictate how to compute the larger AI gate, e.g. a UAS versus a B-52 aircraft.

Since we are building an intelligent surveillance system we will have data obtained from sources outside our radar system, e.g. map data, intelligence data, and other sensors. We can assume we know what type of targets we are tracking, such as helicopters, tanks, scud launchers, surveillance aircraft, fighter aircraft, and missiles. If so then we know something about their kinematics, i.e. their minimum, maximum and average velocities for different altitudes, their maximum gravitational (G) force turn they can withstand and at what radius, and their maximum acceleration. Using these data we can construct rules that will compute the larger size gate based upon a degree of belief given the type of target, e.g. helicopter or a fighter aircraft. This degree of belief can be computed using information from outside data sources, its previous kinematics data (velocity, location, etc.), radar cross section, and altitude amongst other factors such as the type of mission, its position in the scene, and sensitive locations or targets.

A simple rule is to take the maximum velocity for the target type that has the highest belief and compute the maximum distance it could have traveled from the previous position on the last CPI. This allows us to compute a semi-circle around the vector the target was heading. See Figure 5. This approach may be fine for a target like a surveillance aircraft, but not for a tank or track vehicle or scud launcher. For example,

a tank which can easily turn 180 degrees, a circle may have to be drawn with radius equal to the maximum distance that can be traveled within the time between CPIs. The more we know about the targets we are tracking the more intelligent we can be in designing our rules and estimate our gate sizes.

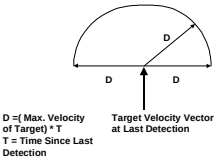


Figure 5. Example AI Computed Gate

Section C

If the target is detected in the larger gate then we need to adjust the weights of our tracking filter. Indicated in block C we can adjust the weights with rules based upon position, velocity and acceleration. These rules can be simple, e.g. if the target was detected in the larger gate then set the weights for the next CPI as if the target were detected the first time. This will eliminate any memory or smoothing that the filter had performed and start off with a larger gate size. More sophisticated rules can be employed and should be investigated further, dependent upon the tracking filter used.

Section D

If the target was not found in the smaller or the larger gate then we need to determine if it is being shadowed from our radar, possibly by terrain. Our logic is assuming that the radar system has a priori data that are available such as terrain data containing elevation attributes, roads, and bridges. With this information we can compute whether or not given the elevation of the radar and the last position of the track if there is terrain obstructing the radar's illumination of the target. If there is an obstruction then we should be able to project, given the last known velocity of the track and the changing position of the radar, how many CPIs the track will be obstructed. Based upon these computations we can then coast the track until the next CPI. For each coasted CPI we should also look for new unused reports that can occur due to our coasted track changing its projected velocity while it is being obscured. See Figure 6. If this does occur and a new track is initiated we should "flag" this track that it may be the coasted track. Once we compute when or which CPI the original track should be visible and if it isn't, even after two additional CPIs, we should then revisit the new track. During this revisit we need to compute whether or not the dynamics of the target/track were capable of maneuvering to the position that the radar detected the target. (See paragraphs F and G for more details.) If it is shown possible, then the new track should be updated as being the old track with some degree of belief. If however, the original track is detected after it has moved beyond the obstruction then we should go back to the new track that was initiated and remove the "flag" indicating the possibility that this was a firm track that was coasted.

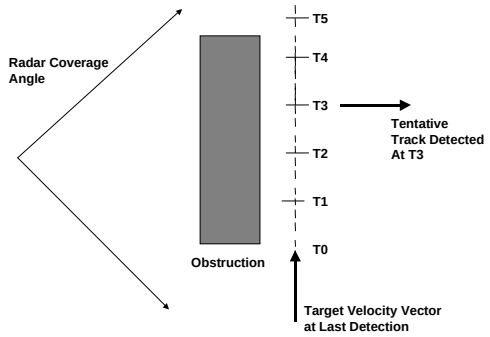


Figure 6. Track Obstruction

Section F

If the target is not in either gate and it is not shadowed then maybe the target is out of range. This is easy to compute given its last position relative to the radar. If it is out of range then we should pass this information to another sensor platform along with the track data we have acquired. The knowledge of when a target is going to reach this point can be predicted earlier than the last CPI. However, the point in space when a target will be out of range is a variable dependent upon the radar and the target's movements.

The information that can be passed to the other platform can contain the time of the first acquisition, its history path, velocity range, hypothesis of type of target, and any other kinematics or knowledge that has been gathered throughout its track. This data can be used by the message receiving platform in assigning degrees of belief about the target's maneuverability, type of target, and identification.

Section F

If the target is not in either gate, not shadowed, and not out of range then what happened to it? Maybe our knowledge about its kinematics was incorrect? Maybe our sensor and filtering model has more error variation than we thought? Maybe the target

maneuvered and its radar cross section (RCS) is too low and therefore not detected. Maybe the clutter is too large and we can't detect the target?

What we can do is determine if there are any unused reports. If unused reports exist then maybe one of these are our target. First we need to perform a quick culling to determine if at maximum velocity (Vmax) our target could have traveled from where we last detected it to where the unused report was detected, a distance of D. If Vmax times T (time between the two detections) is less than D then this unused report can't possibly be due to the same track. If all unused reports result in the same finding then we conclude that there are no unused reports that may be due to our track. If however, one or more computations show that the distance to the possible reports could have been traveled by the target then we need to compute its possibility and assign a degree of belief to each report.

Section G

A simple algorithm for computing the possibility of an A/C maneuverability is illustrated in Figure 7. D is the distance between the last detection and the position of an unused report. The different radii (R1 and R2) represent the different radius that one can construct that can pass a circle or arc through the two end points of the chord of length D. If we assume that the acceleration is a maximum then we can assume that the velocity is our last estimated velocity or its maximum velocity. Each assumption has a certain amount of error. We can compute different values of R by the following:

$Rest = (Vlast)^2/Accmax,$

$Rmax = (Vmax)^2/Accmax.$

For different values of R and D we can compute the distance of the arc connecting the end points of the chord D. It can be shown from Figure 7 that:

Theta = 2(arcsin((D/2)/Rest)) or
Theta = 2(arcsin((D/2)/Rmax)).

The distance along the arc is 2*Pi*Rest/(Theta/360) = Darcest. Therefore if at (Vlast)*T is less than Darcest then the maneuver is not possible. Similarly if (Vmax)*T is less than Darcmax then the maneuver is not possible.

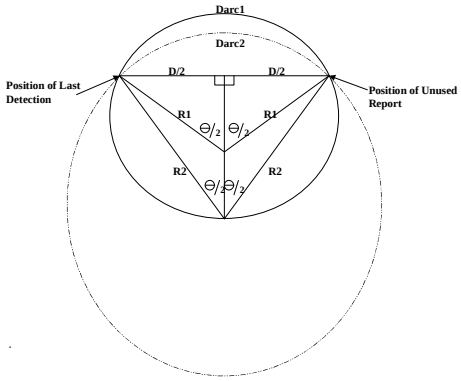


Figure 7. Maneuver Possibilities

Similar rules can be developed for different targets and their kinematics to determine the best rules for each. The developed rules can be verified and modified by consulting with experts who are aware of a target's kinematics.

Section H

If the target is not in either gate, not shadowed, not out of range, and our kinematics is verified then what happened to the target? It may have maneuvered such that its RCS decreased. If it's a ground slow moving target it may have stopped. It may be hidden by a tunnel. The level of detail for examining why a target track is undetectable needs to be perused dependent upon the target, the environment, the amount of detail a priori data available, and the scenario under investigation. For this iteration of our AI logic structure we have elected to halt our level of investigation and to coast the target. The algorithm would request the KBC to reduce the detection level for the location which we lost the target and the locations where we project the track to be for the next four CPIs. We should identify that the track is potentially dropped and treat the track as a coasted track. If after four CPIs it cannot be correlated with a detection then the tracking filter will drop the track.

Summary

This paper has provided a brief overview of a hypothesized integrated end-to-end radar signal and data processing chain. The majority of the paper described a tracking algorithm and proposed an AI logic structure for incorporating rules for different targets, environments, and scenarios. The driving force of this logic structure is to use AI to learn about each track and to analyze each track completely before it is dropped. The logic structure is independent of any tracking algorithm, environment, target type, or scenario.

The AIRS architecture is new and revolutionary. Its potential is great. It is one element in a bigger program dealing with waveform diversity and sensors as robots.

References

[1] Bryant, M., Johnson, P., Kent, B. M., Nowak, M, and Rogers, S., "Layered Sensing Its Definition, Attributes, and Guiding Principles for AFRL Strategic Technology Development"

[2] W. Baldygo, M. Wicks, R. Brown, P. Antonik, G. Capraro, and L. Hennington, "Artificial intelligence applications to constant false alarm rate (CFAR) processing", Proceedings of the IEEE 1993 National Radar Conference, Boston, MA, April 1993.

[3] R. Senn, "Knowledge Base Applications To Adaptive Space-Time Processing", AFRL-SN-TR-146, Final Technical Report, July 2001.

[4] P. Antonik, H. Shuman, P. Li, W. Melvin, and M. Wicks, "Knowledge-Based Space-Time Adaptive Processing", Proceedings of the IEEE 1997 National Radar Conference, Syracuse, NY, May 1997.

[5] M. C. Wicks, W. J. Baldygo, JR., and R. D. Brown (1996), Expert System Constant False Alarm Rate (CFAR) Processor, U. S. Pat. 5,499,030.

[6] Multi-Channel Airborne Radar Measurement (MCARM) Final Report, Volume 1 of 4, MCARM Flight Test, Contract F30602-92-C-0161, for Rome Laboratory/USAF, by Westinghouse Electronic Systems.

[7] C. T. Capraro, G. T. Capraro, D. D. Weiner, and M. Wicks, "Knowledge Based Map Space Time Adaptive Processing (KBMapSTAP)," Proceedings of the 2001 International Conference on Imaging Science, Systems, and Technology, June 2001, Las Vegas, Nevada.

[8] A. Farina, H. Griffiths, G. Capraro, and M. Wicks, "Knowledge-Based Radar Signal & Data Processing", NATO RTO Lecture Series 233, November 2003

[9] Capraro, C. T., Capraro, G. T., Bradaric, I., Weiner, D. D., Wicks, M. C., Baldygo, W. J., "Implementing Digital Terrain Data in Knowledge-Aided Space-Time Adaptive Processing", IEEE Trans, on Aerospace and Electronic Systems, Vol. 42, No. 3, July 2006.

[10] C. T. Capraro, G. T. Capraro, and M. C. Wicks, "Knowledge Aided Detection and Tracking", Proceedings of the IEEE 2007 National Radar Conference, Boston, MA, April 2007

[11] G. Capraro and M. Wicks, "An Airborne Intelligent Radar System", Radar 2004, International Conference on Radar Systems, Toulouse France, 2004.

[12] W. L. Melvin, M. C. Wicks, and P. Chen (1998), Nonhomgeneity Detection Method and Apparatus For Improved Adaptive Signal Processing, U. S. Pat. 5,706,013.

[13] G. Capraro and M. Wicks "Metacognition for Waveform Diverse Radar", *International Waveform Diversity and Design Conference*, Kauai, Hawaii, January 2012

[14] G. Capraro, "Cognitive RF Systems and EM Fratricide – Part II", Proceedings of the 2013 International Conference on Artificial Intelligence, Volume I, pp 172-176, Las Vegas, Nevada USA, July 22-25, 2013

[15] M., L., Skolnik, "Introduction to Radar Systems", McGraw Hill, New York, 1980.

[16] E., Brookner, "Tracking and Kalman Filtering Made Easy", Wiley, New York, 1998.r\

Deep Collaborative Data Fusion in a Cloud Framework

Aaron Wheeler
3 Sigma Research, Indialantic, Florida, USA

Abstract – ***Data fusion algorithms often require having all the data available. This presents a problem for Big Data in cloud environments that we must process on multiple nodes running in parallel on individual chunks of the complete data set. MapReduce nodes need mechanisms for deep collaboration in order to successfully perform distributed data fusion on massive data sets. We identify and discussed several strategies for deep collaboration in distributed data fusion algorithms and an agent architecture for controlling distributed data fusion in cloud frameworks. The overall control architecture enables data fusion analyst to act as a "fusion pilot" to control a larger number of data fusion activities. The collaboration architectures lend themselves to existing cloud frameworks for both static and streaming data. The work presented here allows us to adapt existing cloud deployments for deep collaboration of distributed data fusion, thereby leveraging the data and computational power contained within existing cloud infrastructure. Furthermore, our design makes it straightforward to add multi-level security as part of the collaboration.***

Keywords: data fusion, cloud, map reduce, machine learning, intelligent agents

1 Introduction

Data fusion systems learn and update data models as they process data. This holds for both bulk static data and streaming data. Each fusion level processes all the data so that updated models and discovered relations get immediately applied to the fusion task. Legacy fusion systems that perform these tasks break down when applied to Big Data available in cloud architectures. Multiple such systems running in parallel do not share information or work from a common data model. This form of parallelization presents a challenge for fusion algorithms and machine learning because they typically require having all information present at once to make the required associations.

Each node processes a chunk of a large data set but may require information contained in another chunk. Typical cloud data fusion algorithms do not allow nodes to collaborate with respect to sharing needs and information. Furthermore, continual data enrichment adds to the data available for learning and discovery. Data models used by nodes must get updated then applied possibly to the entire data set to find false negatives.

Processing nodes in a MapReduce framework do communicate and coordinate for the purposes of replication, load-balancing, and other related activities. However, the MapReduce model does not generally support communication and coordination from within a particular MapReduce job running on separate nodes, or between separate MapReduce jobs.

Thus, an opportunity exists to implement data fusion algorithms for cloud environments that can maintain data models, support machine learning, and perform fusion at all levels. We seek to identify strategies to enable deep collaboration between computation nodes in cloud architectures. We also want to enable intelligent agent-based control of the bulk of the multi-level distributed data fusion process, with people acting as "fusion pilots" to provide evaluation, feedback, and control.

This paper describes our efforts to date. Section 2 evaluates options for enabling communication between nodes. In Section 3 we identify and discuss strategies for introducing collaboration at various points in the cloud. Section 4 presents an architecture for agent control of the collaborative distributed data fusion architecture. Section 5 makes some concluding remarks about our work and future directions.

2 Collaboration workspace

Deep collaboration between processing nodes in a cloud requires a mechanism for exchanging messages and information.

Message queues provide a method for distributed processes to exchange information for collaboration. Distributed message queues run in cloud clusters and allow for interaction with cloud processes. Kestrel, original developed for Twitter [1] and Apache Kafka [2] represent two widely used distributed message queues that work well with Hadoop [3] and Storm [4].

While suitable for some types of collaboration, we do not believe message queues best serve the needs of collaboration for distributed data fusion. Collaboration may require communication with a large but unknown number of other nodes. We can configure a message queue as a fan-out queue that replicate messages to many other queues. However, this requires each node having its own queue. Furthermore, message queues place the burden of message/data persistence on the receiver. While not a great burden, it seems more

efficient to centralize storage of fusion data models and other supporting information.

A blackboard represents another design pattern for collaboration [5]. Blackboards provide a centralized access point for distributed processes to exchange messages and information. For this reason, we believe a blackboard architecture offers the best advantage for collaboration during distributed data fusion.

We have chosen to implement our blackboard using the Accumulo database. Accumulo provides a relaxed consistency triple store database and runs on the Apache Hadoop distributed file system. The National Security Agency (NSA) developed Accumulo as a modification of the Google BigTable design. [6] Among the modifications include cell-based access control and programmatic server-side record modification [7]. The critically important issue of information security does not fall within the scope of the present discussion, but we have planned ahead by choosing Accumulo as the collaboration blackboard.

3 Strategies for deep collaboration

In this section we identify a number of strategies for introducing deep collaboration between processing nodes in distributed data fusion applications in cloud frameworks.

3.1 Sequence of MapReduce jobs

This strategy allows for collaboration between nodes in a sequence of MapReduce jobs. Collaboration occurs by nodes in a job generating output used as input by the next MapReduce job. One job runs to completion before the next job begins. Figure 1 shows the architecture.

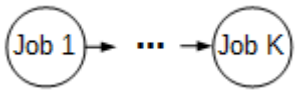


Figure 1. Collaboration between nodes in sequential MapReduce jobs.

3.2 Iterated MapReduce jobs

We can enable collaboration between nodes by implementing MapReduce jobs as a single iteration of a data fusion algorithm. Nodes exchange information at the end of an iteration and use this information to either continue with another iteration or terminate the process. We can achieve this kind of collaboration using either counters or a blackboard.

Counters

Hadoop MapReduce jobs have the concept of counters [8]. Map and reduce tasks may increment or decrement these counters. The MapReduce client can examine these counters to make decisions about the next course of action. Map and reduce tasks may update counters to indicate a change in the state of their model or update deltas to parameters of a shared data model. Figure 2 shows an architecture for collaboration between MapReduce nodes using counters.

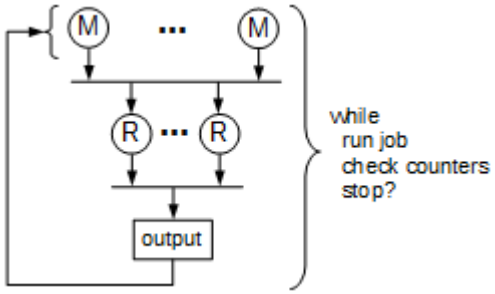


Figure 2. Counters for iterative MapReduce node collaboration.

Blackboard

Counters allow for a simple exchange of information between nodes, but do not provide an efficient way to communicate new information to other nodes. Instead, a blackboard architecture provides greater flexibility for collaboration. Figure 3 shows a blackboard used in an iterative data fusion MapReduce job.

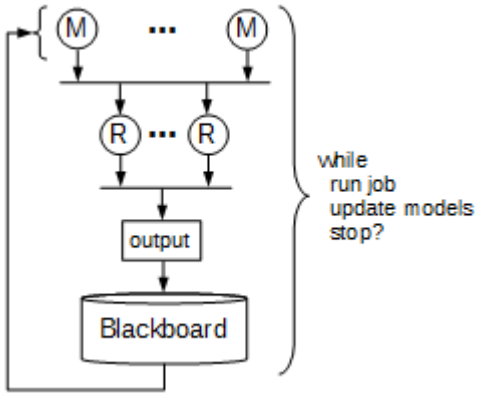


Figure 3. Blackboard for iterative MapReduce node collaboration.

3.3 Reducer re-iteration

Reducer re-iteration provides a limited way to repeatedly process the same data within a single MapReduce job. This strategy involves a programmatic deviation from the standard MapReduce flow by introducing into the reduce tasks the iterative step in the data fusion algorithm. Between iterations, each reducer posts updated information to a blackboard and pulls relevant new information from the blackboard. The blackboard will also contain a message for reducers to exit. Figure 4 shows the reducer re-iteration collaboration architecture.

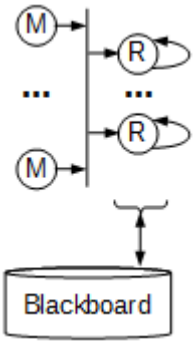


Figure 4. Reducer re-iteration collaboration architecture.

Map tasks run on individual key/value pairs read from an input data source. Reduce tasks run on individual keys and all associated values. The mapper processes individual records (e.g. lines in a file) within a chunk of data. A mapper could update its data model prior to processing a record. However, it could not then revisit previous records and apply the new data model to them. This means the mapper might have missed important information or at least has produced data evaluated under different criteria. Reducers process keys and all values associated with that key, and can iterate multiple times over these values. Once a reducer has processed a key, it cannot revisit this key.

After mapping, get a shuffle/sort phase where results get allocated to reducers, sorted, then processed by the reducer. All keys of a given value get partitioned and directed to the same reducer. For a given key, the reducer iterates over all values. After iterating, the reducer optionally emits data to the output file then receives the next key and values, or exits.

This approach requires some design considerations. Each reducer must receive values for only one key because it cannot re-iterate over keys. Not all data fusion problems will yield to this restriction. Because nodes collaborate in real-time, more than one reducer can receive the same key. We must set the number of reducers in the job configuration. In some cases, we might need a custom partitioner to distribute key/value pairs from the mappers to the correct reducer.

3.4 Blackboard collaboration

We can achieve asynchronous collaboration between nodes in separate MapReduce jobs via a blackboard. This collaboration architecture allows for jobs executing data fusion at different fusion levels to share information. It also provides a straightforward way for collaboration between static and streaming data fusion processes. Figure 5 shows this design for MapReduce jobs.

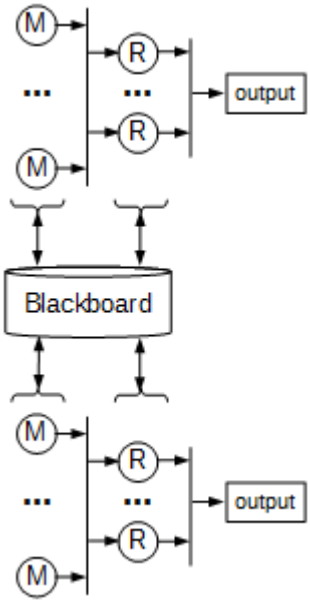


Figure 5. Blackboard collaboration architecture.

The MapReduce jobs run independently, according to the needs of the information analysts. Output of these jobs get placed in a common data store so that they become available for other MapReduce jobs. The collaboration between nodes in separate MapReduce jobs does not happen in real-time, but through a shared data source that each may use to refresh their data models before execution and update during their execution.

Accumulo has relaxed consistency and so does not have the guarantee of ACID (atomicity, consistency, isolation, durability). This means simultaneous queries may produce different results and that transactions lack atomicity [6]. Without the ACID guarantee, fusion models may become inconsistent or a situation may occur where mappers or reducers simultaneously read a data model from Accumulo, but receive different versions of it. We can avoid this issue by having a single process read the necessary data models from Accumulo and right them to a location accessible to the particular MapReduce job or topology.

4 Agent controller

Deep collaboration requires reasoning and decision-making. Often, we can delegate a significant portion of the reasoning and decision-making to intelligent agents that act on behalf of human data fusion analysts, leaving them free to focus on the most difficult fusion tasks and for controlling multiple fusion tasks.

Figure 6 shows an architecture that includes human data fusion analyst (fusion pilot) in the role of controlling agents responsible for distributed data fusion of both static and streaming data using Hadoop, Storm, and Accumulo.

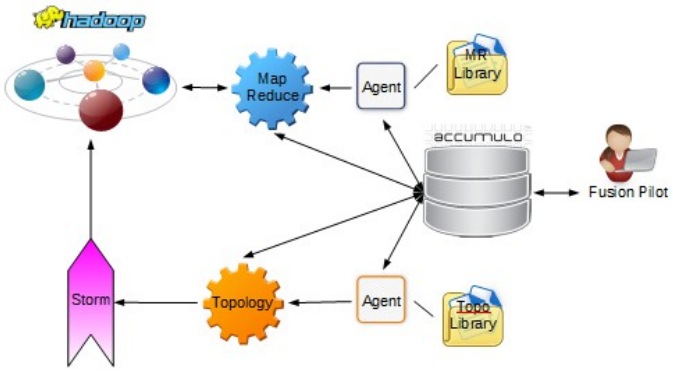


Figure 6. Agent control architecture for deep collaboration of distributed data fusion.

In Figure 6, MapReduce jobs perform data fusion on static data stored in the Hadoop file system. Real-time data fusion occurs with Storm topologies, with the data ultimately getting archived in the Hadoop file system for future use. Agents act directly to control the data fusion process, with the fusion pilot overseeing results and providing feedback via the Accumulo blackboard.

Intelligent agents can act more quickly, but human agents can provide additional insight and judgment not incorporated in the intelligent agent model. Foo and Ng [9] note that explicit human assessment represents a valuable and often necessary component of the data fusion process. Without extensive real-world experience and common-sense reasoning, unsupervised machine learning algorithms could produce unrealistic and incorrect models. Intelligent agents act on behalf of people and nothing prevents people from acting as their own agents. Thus, the agent controller allows for human agents in the loop as well.

5 Conclusion

We have identified and discussed several strategies for deep collaboration in distributed data fusion algorithms and an agent architecture for controlling distributed data fusion in cloud frameworks. The overall control architecture enables data fusion analyst to act as a "fusion pilot" to control a larger number of data fusion activities. The collaboration architectures lend themselves to existing cloud frameworks for both static and streaming data, with Hadoop and Storm as primary examples. This allows for us to adapt existing cloud deployments for deep collaboration of distributed data fusion. Furthermore, the choice of Accumulo as the collaboration blackboard means we enhance collaboration with multi-level security, thus allow more organizations to contribute to the fusion effort with due consideration for the security concerns of each participating organization. In the near future, we intend to demonstrate our deep collaboration architecture on a multi-level data fusion problem.

6 References

[1] Kestrel: Simple, Distributed Message Queue System. https://github.com/twitter/kestrel

[2] Apache Kafka: A High-Throughput Distributed Messaging System. http://kafka.apache.org/documentation.html

[3] Apache Hadoop. http://hadoop.apache.org/

[4] Storm: Distributed and Fault-Tolerant Realtime Computation. http://storm.incubator.apache.org/

[5] D.D. Corkill (1989). Design Alternatives for Parallel and Distributed Blackboard Systems. In Blackboard Architectures and Applications V. Jagannathan, Rajendra Dodhiawala, and Lawrence S. Baum, editors, Academic Press. http://dancorkill.home.comcast.net/~dancorkill/pubs/parallel-distributed-chapter.pdf

[6] Jeremy Kepner, Christian Anderson, William Arcand, David Bestor, Bill Bergeron, Chansup Byun, Matthew Hubbell, Peter Michaleas, Julie Mullen, David O'Gwynn, Andrew Prout, Albert Reuther, Antonio Rosa, Charles Yee (2013). D4M 2.0 Schema: A General Purpose High Performance Schema for the Accumulo Database. MIT Lincoln Laboratory. http://www.mit.edu/~kepner/pubs/D4Mschema_HPEC2013_Paper.pdf

[7] Apache Accumulo. http://accumulo.apache.org/

[8] Iterative MapReduce and Counters. http://hadooptutorial.wikispaces.com/Iterative+MapReduce+and+Counters

[9] Pek Hui Foo and Gee Wah Ng (2013). High-level Information Fusion: An Overview. Journal of Advances in Information Fusion, Volume 8, Number 1. http://www.isif.org/sites/isif.org/files/398_1_art_8_20128.pdf

Functions as Conditionally Discoverable Relational Database Tables

A. Ondi and T. Hagan
Securboration, Inc., Melbourne, FL, USA

Abstract - *It is beneficial for large enterprises to have an accurate and up-to-date picture of the status of the enterprise, yet the information, though available, is rarely accessible from one place and in a unified manner. The research presented here introduces conditionally discoverable relational database tables (CDTs) that present a subset of the information they contain via links to special input columns. Such CDTs can be used to present functions and web-methods in a unified relational manner, thus easing the burden on information discovery. CDTs can be used in conjunction with each other or with other traditional database tables via SQL SELECT queries.*

Keywords: Conditionally discoverable table, Virtual table, Relational database, Information discovery

1 Motivation

It is beneficial for large enterprises to have an accurate and up-to-date picture of the status of the enterprise, yet the information, though available, is rarely accessible from one place and in a unified manner. Having to know what piece of the big picture is located where hinders putting together a comprehensive overview.

The goal of the research presented here is to help in the understanding of the current status of large enterprises where the information needed is disparately located, by offering a unified view of the available data at a single location.

2 Background

Relational databases [1] have long been used to store, retrieve and manipulate information. One of their beneficial qualities is that the information is presented in a unified manner: all data is stored as rows of tables, the columns of which representing the various aspects of the data, referred to by the name of the column. The database is broken up into a number of tables, each representing a coherent view of a subset of the data. Relationship between the views offered by the tables can be further established by linking columns of the different tables (and sometimes within one table). As all aspects of the data (up to access rights) can be linked, users with a good understanding of the layout of the data can craft queries that can answer many questions about the data (e.g. "Which manufacturers are behind schedule for parts used in the construction of product X?"). The name of the table columns, the type of the data stored in the columns and possible restrictions on the data (ensuring its integrity) stored in the tables are described in schemas, which allow discovery of the data. It is important to note that the view of the data in relational databases is "static", similarly to procedural programming languages: the data is accessed and manipulated by external processes (in this case, queries).

Objects in object-oriented programming languages [2] store data adhering to the principles of abstraction, encapsulation, inheritance and polymorphism. Moreover, the view of the data is dynamic: the processes to access and manipulate the data captured by the objects are associated with the objects themselves (in the form of member functions), ensuring the integrity of the state of the objects they are associated with. Relationship between objects are established directly by encapsulation or indirectly by linking aspects (exposed via member functions) of different objects.

Due to their differences in what data is accessible, how they are linked and the ways to ensure the integrity of the data, relational databases and object oriented languages had a hard time working together. Either the objects had to be "opened up" to enable storage in relational databases (thus violating the abstraction and encapsulation principles); or the objects had to be stored in object databases [3] that allow discovery of information in only ways the objects were specifically designed for (e.g. following one-way pointers to other objects).

To alleviate the object-relational impedance mismatch, object-relational databases [4] were created that offer a hybrid approach between pure relational and pure object databases by allowing the storage of objects in relational tables that preserve their compound types (along with inheritance), while still allowing queries similar to those in pure relational databases.

3 Approach

The data is presented as (virtual) relational database tables, thus our system can be viewed as a mostly read-only relational database server, supporting a large subset of the SELECT section of the Standard Query Language (SQL)

(Reddy, 2008). The virtual tables can be backed by real database tables, web services or any other means the data is available.

3.1 Functions as Tables

Whereas presenting data in a virtual database table is straightforward when it is backed by other database tables, the situation is somewhat more complicated when the virtual table is backed by functions (e.g. in the form of web methods of a web service).

From a mathematical point of view, functions are relations between their input values and their output values; therefore a table can be formed by listing the values of the function for each input. For example, if the function to be represented is *"f(x:int):int=2*x"*, the corresponding table representation is as listed in Table 1.

Table 1 : Table representation of *f(x:int):int=2x*

x:int	*f(x)*:int
…	…
-1	-2
0	0
1	2
2	4
…	…

Immediately, we can see that constructing the whole table is troublesome for at least two reasons. First, the range of the input values can be quite large (compounded for functions with multiple input values). Second, knowing the domain of the function is non-trivial.

Both of these issues can be circumvented if we require the input values to be provided for the virtual table. For example, the above *f(x)* can be easily computed and represented if the values -1, 0, 1, 2 are provided for *x*, yielding the result listed in Table 2.

Table 2 : Table representation of *f(x:int):int=2x* restricted to $x \in \{-1, 0, 1, 2\}$

x:int	*f(x)*:int
-1	-2
0	0
1	2
2	4

This requirement turns the virtual table into a *conditionally discoverable table* (CDT): in order to discover the contents of the table, one has to provide a subset of the table columns. Note that such conditionally discoverable tables are equitable to functions in imperative languages – indeed, it was precisely our desire to represent imperative functions that drove the conditions imposed on the table. The restrictions of conditionally discoverable tables are easily satisfiable: one has to provide the required input values as a table in the first part of a *JOIN* expression (possibly with column aliasing to match the name of the input value), where the second part of the *JOIN* is the desired conditionally discoverable table. In situations where the same name is used for different input values (e.g. two different functions *f(x)* and *g(x)* are backing two tables used in the construction of the result table) the conflict is resolvable by using sub-queries for each CDT, followed by possible column aliasing to avoid column name conflict, just as with the classic use of *JOIN*. Note that tables that are backed by functions that do not require input values (e.g. constants) have an empty condition and are *unconditionally discoverable*, thus the *JOIN* can be avoided.

Although this conditionality theoretically restricts the discovery of the whole table, it does not impose a real restriction as we are able to discover all the rows of the table that are relevant to our purpose.

3.2 Objects as Tables

Another issue we have to address in order to use object-oriented web methods is the representation of objects in our tables. Although objects in the object-oriented paradigm fulfill a number of requirements (abstraction, encapsulation, inheritance and polymorphism), it will be sufficient to view objects here as "plain old data" with the possibility of inheritance, utilizing the shallow and static nature of objects used in web methods.

To represent compound data values (objects) in tables, we need to decompose them recursively into columns following the member values. Note that the traditional dot notation used in imperative programming languages uniquely identifies the member values, up to object identity. Using this scheme and using the parameter names (for input parameters) and the returned type name (for return value) as the base name of the columns, the function *getPersonnelInfo(employeeId:int): PersonnelInfo* that returns personnel information given the ID value of an employee can be represented as a CDT as listed in Table 3.

Note that although in the above example the input parameter (*employeeId*) is superfluous as it appears as part of the returned values, this is not necessarily the case: a function *getAddress(employeeId:int):Address* would only return the address associated with the employee ID, but not he ID itself. For the sake of generality, we will not differentiate between these cases and include the input parameters in the function's table representation.

Table 3 : Table representation of function *getPersonnelInfo(empoyeeId:int):PersonnelInfo* – columns represented as rows for easier readability

employeeId:int	123456	789012
personnelInfo.employeeId:int	123456	789012
personnelInfo.firstName:String	Jane	John
personnelInfo.lastName:String	Dow	Smith
personnelInfo.address.street:String	123 Apple St.	45 Orange Ct.
personnelInfo.address.city:String	Melbourne	Melbourne
personnelInfo.address.state:String	FL	FL
personnelInfo.address.zip:int	32901	32901
personnelInfo.phone:String	000-123-4567	000-890-1234

The choice of base column name for the return value is arbitrary and can be custom-set for each function-backed virtual table. This helps in cases where the return value is a simple type, like *int* that has little meaning to a human.

3.2.1 Handling collection values

Function return values and members of objects can have collection types (e.g. lists, sets). The straight-forward approach is to represent all collections by replicating the associated rows for each value of the collection; however there are problems with is approach: if the collection is empty, it will lead to leaving out all other member value data as well. A refined approach is to follow the 2NF recommendations of relational database design [1] and factor the collection representations into their own tables and connect to the rest of the object by an artificial ID value (the name of which is arbitrary, as long as it does not conflict with other column names). To demonstrate, assume a modified version of the above *getPersonnelInfo* function that returns a list of known addresses and phone numbers for each employee and that Jane Dow has two known addresses (123 Apple St., Melbourne, FL 32901 and 6 Date Ln., New York, NY 10453), and John Smith has two known addresses (45 Orange Ct., Melbourne, FL 32901 and 78 Mango Ave., Chicago, IL 60007) and two phone numbers (000-890-1234 and 001-567-8901). The resulting tables are listed in Tables 4-8.

Under this scheme, the various sub-tables have to be JOIN-ed to connect the non-collection parts to their corresponding collections; however care must be taken as JOIN-ing on empty tables will lead to the same issues as the straight-forward approach.

3.2.2 Handling of *null* values

Occasionally, functions return *null* values or objects that have null members, often representing processing errors or missing information. In such cases, each column related to the *null* value can be set to *null*. Although this is against the strict guidelines of traditional database design [1], the alternatives are no better: either leave out the entire object or use known error values. The first approach might be acceptable for *null* return values, but it can hide useful information if only some members of the returned value is *null*; the second approach requires that no backing function returns values that use the designated values representing error, which is usually an unrealistic requirement as this restriction is rarely known to the developers of the backing functions.

3.3 Temporal Tables

The last issue to remedy is the possibility of temporal functions: functions that return different values at different invocation times, for the same input values. Although the reason for such temporality can be many-fold (e.g. change of underlying hidden state, dependence on current time, use of random numbers), the exact cause is irrelevant for our considerations.

To resolve the issue, we extend the resulting table with a new column (taking care to avoid conflicts with existing column names) representing the time of invocation of the underlying function. Note that since this extra column can introduce unnecessary complications (e.g. triggering a false-positive in change detection of the returned data when only the time-column has changed), this extension can be used per-table, as needed. Of course, the extra column could also be easily removed with a projection in the query.

3.4 Dependencies between Tables

It is often tedious to write queries by hand, especially when the input columns of conditionally discoverable tables are numerous (as can be the case when the input parameters of the backing function are complex objects themselves). In cases when some of the non-input columns of other tables correspond to input columns of a conditionally discoverable table, these two tables can be linked. These links then can be used to automatically supply the necessary components for a *JOIN* between the two tables. Moreover, when there is only one table that is linked to another's input columns, the first table can be automatically added to the *JOIN* when the second table's input columns are not fully specified.

Table 4 : Main-table representation of the modified *getPersonnelInfo* function

employeeId:int	*personnelInfo.employeeId*:int	*personnelInfo.firstName*:String	*personnelInfo.lastName*:String
123456	123456	Jane	Dow
789012	789012	John	Smith

Table 5 : Sub-table of the modified *getPersonnelInfo* function, representing the address values

personnelInfo. address_id:int	*personnelInfo. address.street*:String	*personnelInfo. address.city*:String	*personnelInfo. address.state*:String	*personnelInfo. address.zip*:int
1	123 Apple St.	Melbourne	FL	32901
2	6 Date Ln.	New York	NY	10453
3	45 Orange Ct.	Melbourne	FL	32901
4	78 Mango Ave.	Chicago0	IL	60007

Table 6 : Sub-table of the modified *getPersonnelInfo* function, representing the phone values

personnelInfo.phone_id:int	*personnelInfo.phone*:String
1	000-123-4567
2	000-890-1234
3	001-567-8901

Table 7 : Sub-table of the modified *getPersonnelInfo* function, representing the connection between the main table and the address sub-table

employeeId:int	*personnelInfo.address_id*:int
123456	1
123456	2
789012	3
789012	4

Table 8 : Sub-table of the modified *getPersonnelInfo* function, representing the connection between the main table and the phone sub-table

employeeId:int	*personnelInfo.phone_id*:int
123456	1
789012	2
789012	3

The dependency relations between tables are stored in an internal knowledge base, identifying the tables linked and the column alignment (with possible aliasing).

To demonstrate the effects of table dependencies, consider a scenario, where information about items in a warehouse inventory system can be accessed via the following functions:

- *getWarehouses(zip:int):[int]*

- *getInventory(warehouseId:int):InventoryItem{itemId:int, itemCount:int}* and

- *getItemInfo(id:int):ItemInfo{id:int, priceCents:int, manufacturerId:int}*,

where brackets (‘*[]*’) represent lists, braces (‘*{}*’) denote their associated compound types and *getWarehouses* returns a list of warehouse IDs belonging to the given ZIP code. The corresponding virtual tables are named after their backing functions and the base name of the *getWarehouses* return column names is set to *“warehouseId”*.

There are two dependency relationships:

- *getInventory* depends on *getWarehouses* with the column *getInventory.warehouseId* aligned with *getWarehouses.warehouseId* and

- *getItemInfo* depends on *getInventory* with the column *getIemInfo.id* aligned with *getInventory.inventoryItem.itemId*

Listing 1 : Unassisted SQL query assessing the total price of items stored in warehouses in the 32901 ZIP area

```
SELECT SUM(getInventory.inventoryItem.itemCount * getItemInfo.itemInfo.priceCents)
FROM (SELECT DISTINCT 32901 AS zip FROM tables) AS tmp
    JOIN getWarehouses ON getWarehouses.zip = tmp.zip
    JOIN getInventory ON getInventory.warehouseId = getWarehouses.warehouseId
    JOIN getItemInfo ON getItemInfo.id = getInventory.inventoryItem.itemId
```

Listing 2 : Modified query of Listing 1 using column alignment and aliasing

```
SELECT SUM(getInventory.inventoryItem.itemCount * getItemInfo.itemInfo.priceCents)
FROM (SELECT DISTINCT 32901 AS zip FROM tables) AS tmp
    JOIN getWarehouses
    JOIN getInventory
    JOIN getItemInfo
```

Listing 3 : Modified query of Listing 1 using singleton dependency information on tables *getInventory* and *getItemInfo*

```
SELECT SUM(getInventory.inventoryItem.itemCount * getItemInfo.itemInfo.priceCents)
FROM (SELECT DISTINCT 32901 AS zip FROM tables) AS tmp
    JOIN getItemInfo
```

An unassisted query assessing the total price of the items stored in warehouses in the 32901 ZIP area would look like as presented in Listing 1.

Note that the inner *SELECT* expression is a work-around for the fact that *getWarehouses* is a CDT, and therefore needs its input columns satisfied. Were it unconditionally discoverable, we could have used the traditional approach with a *WHERE* clause. The table *tables* is guaranteed to be present and non-empty (listing the available tables in the database, although that is irrelevant for the above query).

Using the dependency information, the query can be written as presented in Listing 2. The system recognizes the special dependency relationship between the tables *getInventory*/*getItemInfo*; and re-writes the *JOIN* expression to use the column alignment and aliasing information associated with the dependencies. The input columns of *getWarehouses* and *getInventory* are fulfilled by rows of the expected name and type present.

The query can be even further simplified, as presented in Listing 3, if the only table *getInventory* depended on was *getWarehouses* (and similarly for *getItemInfo* and *getInventory*). The system identifies that there are unfulfilled input columns of *getItemInfo* and that there is a unique dependency on table *getInventory* (similarly for *getInventory* and *getWarehouses*), thus re-writing the query to the previous form (which will be further re-written to the original).

4 Conclusion

In this paper we presented a database system utilizing conditionally discoverable virtual relational tables that can be used to represent web-methods in a relational manner, thus allowing information discovery of an enterprise in a unified manner.

Although any function can be used to back a CDT, care must be taken to minimize the use of functions that modify system state (e.g. storing information that other processes or functions utilize), restricting their use to functions for information discovery. The reason for this is the mental discrepancy between a *query* (which is associated with discovering information without modifying it) and an *update* (which is associated with modification of information): users of the system will expect that *SELECT* queries do not modify the state of the database or have other external impact.

Relational database management systems often support the notion of stored procedures: query statement templates that are pre-constructed and can be invoked by providing a number of parameters. Stored procedures ease the burden of using the system by not requiring the user to understand how to craft the required valid query statement, nor to understand how the data is structured in the database – these are done by the administrator who creates the stored queries. This idea can be easily implemented for our system as well, by capturing the stored procedures in the internal knowledge base.

5 Future Work

The system presented in this paper is a work in progress, which we are planning to improve in a number of ways.

First, if information about the backing functions is programmatically discoverable (e.g. via source code, reflection or WDSL), the dependency information between tables can be partially automated: a discovery process can list all other functions/tables that can be used in satisfying the input columns of the CDT, thus easing the burden on the human administrator of the system.

Second, although collecting information about an enterprise in one place is certainly beneficial for gaining an up-to-date picture, it is essential to ensure only people with appropriate rights can access the system. Access rights can be established to restrict access to individual tables or to combinations of tables (as combining information is sometimes more damaging than access to only parts of the picture).

6 Acknowledgement

This work was supported in part by the Air Force Research Laboratory, Contract No. AF131-054

7 References

[1] C. J. Date. “Database Design and Relational Theory: Normal Forms and All That Jazz”. O'Reilly Media, 2012

[2] B. Pierce. “Types and Programming languages”. MIT Press, 2002.

[3] W. Kim. “Introduction to Object-Oriented Databases”. MIT Press, 1990.

[4] M. A. Stonebreaker. “Object-Relational DBMSs: The Next Great Wave”. Morgan Kaufmann Publishers, 1996.

[5] M. Reddy. “Structured Query Language (SQL)”. Encyclopedia of geographic information science, 457-459, 2008.

Temporal Modeling of Twitter Posting Behavior: An Empirical Study

Matthew Michelson
InferLink Corporation
2361 Rosecrans Avenue, Suite 348
El Segundo, California 90245

John Bumgarner
U.S. Cyber Consequences Unit
P.O. Box 690087
Charlotte, NC 28227-7001

***Abstract*—This paper describes our approach for temporally modeling the posting behavior of users on the Twitter micro-blogging service. While other researchers have analyzed general trends of Twitter behavior (for instance, few users contribute most of the posts), we instead focus on modeling the temporal behavior of individual users. Specifically, we determine whether we can model the day-of-the-week posting behavior (e.g., this person is a "weekend warrior") and the within-day posting behavior (e.g., "night owl" or "morning person") for individuals. This "individual life pattern" analysis is useful for medical and public health applications. For instance, significant deviations from normal behavior might indicate that a person is potentially ill. Our contribution focuses on probabilistically modeling the posting behavior and then using Kullback-Leibler divergence to demonstrate how well the model captures a user's posting behavior.**

I. Introduction

With the growing popularity of sharing on social networks comes the increasing possibility of using social network data for social science and public health research (e.g., [1], [2], [3], [4]). However, much of this work has focused on research of aggregate behavior.

In this paper, we instead focus on modeling "individual life patterns" of social media users, specifically focusing on the Twitter micro-blogging service. Intuitively, these patterns capture aspects such as person's tendency to post information during the weekend or early in the morning. Our intention is that these individual life patterns capture the baseline behavior, identifying normal activities (e.g., usually posting messages at lunch), which we can then use to uncover deviations in normal behavior. Such deviations, in turn, could potentially identify life-changing events such as major illness, which might then warrant additional investigation (e.g, by a health provider, public health analyst, etc.). In fact, at large such an approach could potentially help signal early indications of community illness if a significant number of users have deviations in behavior.

Previous work has identified other such life-pattern information, such as geolocating Twitter users by analyzing the content of their Tweets [5] (instead of their geo-tagged posts which may be error prone). In this paper, we focus on the temporal aspect of the posting behavior, which can be combined with the previous approach to locate a user in space and associate the temporal behavior for improved health monitoring.

Specifically, our goal is to model users temporal posting behavior, noting the preferred days of the week and times within a day in which a user tends to post his or her messages. We model a specific user's posting behavior at two levels of granularity. First, we model day-of-the-week (DoW) behavior, which reflects a user's preference for posting on different days of the week. For instance, a user may do most of their posting during the week (when they have access to their work computer) or on the weekend (when he or she has more free time). Second, at one level of granularity deeper, we model the within-the-day (WtD) behavior, that is, the time of day that a user prefers to post. This captures the intuition that some people are morning posters, while others are "night owls." In this study we focus on modeling the behavior probabilistically and one of our key results is exploring Kullback-Leibler divergence as a means to characterize how well the model captures the user's behavior behavior.

The rest of this paper is organized as follows. Section II describes our probabilistic model for a user's posting behavior. Section III details our empirical study. Section IV outlines future research, and Section V presents our conclusions.

II. Modeling a User's Posting Behavior

Our main contribution is to model a user's posting behavior. At its core, our approach builds a *distribution* representing the probability that a Twitter user will send a message at a point in time (at some level of granularity). For instance, we model the probability that someone will send a Tweet on Monday, Tues., Wed., etc.[1] We model the behavior probabilistically (e.g., instead of rule-based) to capture the uncertainty for any particular data point in a user's history (for instance, someone may simply not Tweet on a day because he or she gets stuck in traffic).

Formally, given a set of granular time periods $t_i \in T$, we first define the set of observations (e.g, Tweets) for each time period as:

$$o_t = \sum o_i^t; \quad t \in T$$

We then can define the full set of observations as:

$$O_T = \sum o_t; \quad t \in T$$

[1]We then test this model on hold-out data in our empirical study to verify that the model indeed can capture the characteristic distribution in the hold out data as well.

And then we can define the distribution (model) as

$$P(O_T) = \left\{ \frac{o_t}{o_T}; \quad t \in T \right\}$$

We build this behavioral distribution for each user, noting that we can flexibly define the time periods T (e.g., they could be weekdays, hour periods within a day, etc.).

III. Empirical Study

Here we describe an empirical study of our modeling of posting behavior. We gathered data on 10,621 Twitter users who sent at least 100 messages during our collection period of 8/11/2010 through 2/8/2013. From this data, we built two models, a day-of-the-week model (DoW), which models user posting behavior for each day of the week, and a within-the-day model (WtD), which characterizes behavior during the course of a given day of the week. To construct the WtD model, rather than model each hour in a day, we instead define three disjoint time periods, "morning," "afternoon" and "night." The posts are then bucketed into each of these categories according to their posting time. The model then defines the distribution over these time periods rather than individual hours in the day. By binning the data into these groups, rather than individual hours, we felt that the model was more intuitive (the distribution reflects a "night owl" which is easier to interpret than preferring "9pm" and "11pm") and we could improve the accuracy since less data points can sufficiently capture the distribution.

We have two main motivations in this paper. First, we develop a probabilistic model of posting behavior, and second we explore the use of Kullback-Leibler divergence [6] as a means for understanding how well the model characterizes the user's posting behavior.

To that end, our procedure starts with an individual user's posts (e.g., observations), O_T^u and separates the posts into "historic" data (O_T^{hist}) and "hold out" data, which we mark as O_T^{test} (using "test" simply in order to avoid confusion with "h" in the superscript). We then build the behavioral model using the observations from O_T^{hist}, as defined above (e.g., $P(O_T^{hist})$) and measure how well that distribution describes the observations in O_T^{test} (e.g., how well $P(O_T^{hist})$ describes $P(O_T^{test})$). If it describes the distribution well, then the model can be thought to have sufficiently described the behavior.

More specifically, we run Monte Carlo simulations to build the model by randomly sampling (without replacement) from O_T^u to construct O_T^{hist}, and then using $O_T^{test} = \{O_T^u \setminus O_T^{hist}\}$ as the hold out set. These random samples then provide the observations to construct $P(O_T^{hist})$ and $P(O_T^{test})$. We note that in this study we ran 1,000 simulations per user.

We then examine how well $P(O_T^{hist})$ characterizes $P(O_T^{test})$ using the Kullback-Leibler divergence (KL) between the distributions. KL measures how well one distribution ($P(O_T^{hist})$ in this case) approximates another ($P(O_T^{test})$), and can intuitively be thought of as the information lost (in information theory terms) when you try to do the approximation. If the value of KL is zero, then $P(O_T^{hist})$ perfectly approximates $P(O_T^{test})$, so values closer to zero mean that the measured distribution from the historic data generalizes well in explaining the distribution from the hold out data.

Kullback-Leibler divergence is defined as:

$$KL(P(O_T^{test})||P(O_T^{hist})) = \sum_{t_i \in T} \log\left(\frac{P(O_{t_i}^{test})}{P(O_{t_i}^{hist})}\right) * P(O_{t_i}^{test})$$

To gain a more intuitive understanding of KL, consider Figure 1 which shows three example distributions A, B and C. The distributions A and B are constructed to be quite similar. As such, their KL value is quite small since the deviations between the distributions are pretty minimal. On the other hand, distribution C is a mirror image of distribution A. Therefore, as shown in the figure, it has a much higher value for KL divergence.

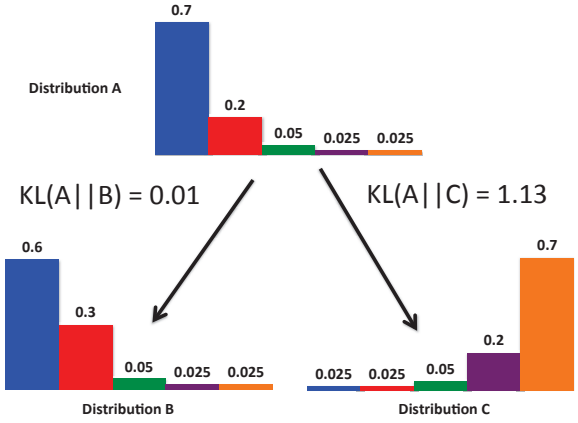


Fig. 1. Different distributions and their KL divergences

Given this experimental procedure, we then built models for each user and explored how well those models characterized the behavior. Our first experiment focused on users who posted frequently, to ensure that we had sufficient data for our analysis (later in this section we relax this condition to analyze the effects of the amounts of data). Specifically, we sub-sampled our users down to those that posted at least 5,000 times during our collection period. This resulted in 555 distinct users for the analysis. Then, we computed the percentage of those users whose distribution had an average KL below a specific threshold. For instance, for a threshold of 0.05 (quite small), we measure the percentage of the users whose KL between $P(O_T^{hist})$ and $P(O_T^{test})$ is less than 0.05 on average, across the 1,000 Monte Carlo trials. Further, we vary the size of O_T^{hist} to measure its effect as well. For instance, if we set the size of O_T^{hist} to be 30%, this means that 30% of O_T^u was used for O_T^{hist} in each of the 1,000 runs (and 70% for O_T^{test}). We tested sizes of 10% through 70% (as shown in the figure), and with 1,000 simulations per user, per threshold. The results for this DoW model are shown in Figure 2

At its peak, the models appear to explain almost 85% of the users' behavior with a KL at or below 0.1. This means that the model is accurate for a large percentage of the users.

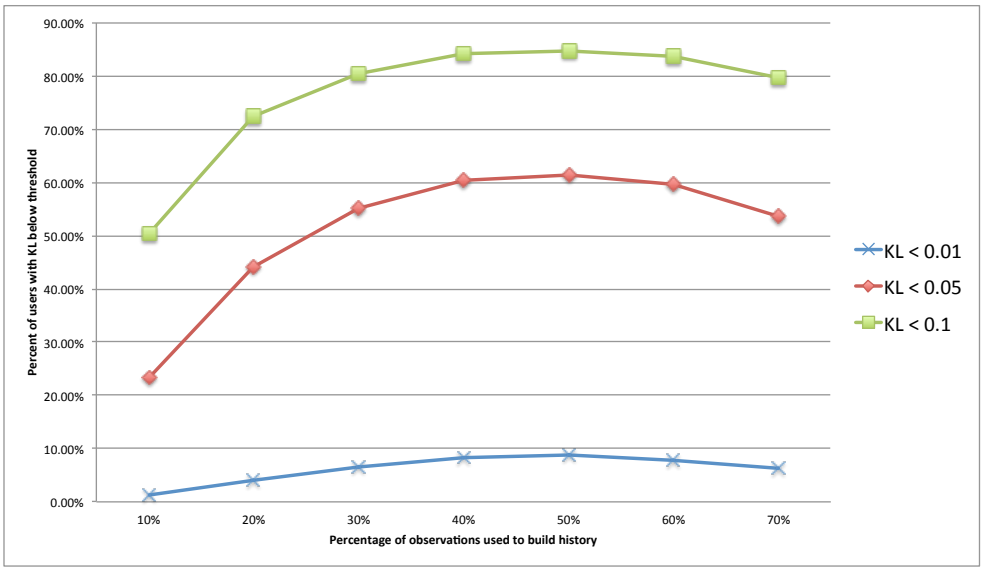


Fig. 2. Fitting distributions of posting behavior for a DoW model

TABLE I
COHORT DEFINITIONS

Cohort Membership	Number of Users	% of overall users
$\geq$ 5,000 posts	555	5.2%
$\geq$ 2,500 posts	903	8.5%
$\geq$ 1,000 posts	1,636	15.4%
$\geq$ 500 posts	2,775	26.1%
$\geq$ 250 posts	4,875	45.9%
$\geq$ 100 posts	10,621	100.0%

Interestingly after 50% the model begins to degrade. It seems that after 50% of the training data is used for modeling, the uncertainty around the data points when there is less data starts being reflected in the training data (which is why it climbs to a point at 50%) and then begins to decline as the less data in the test set begins to exhibit more variation and uncertainty. Nonetheless, we believe that these results indicate that the model captures the main characteristics of the posting behavior.

As we described above, another aspect of this study is to not only understand the effects on the model of the amount of history for each user, but also to understand how the model is affected by absolute size of a user's overall posts. Intuitively, we aim to understand whether the model holds for users who might post less frequently overall. To that end, we divided our users into (overlapping) cohorts, defined by the minimum number of posts he or she made during the collection period. At the highest end, we have our 555 users who made at least 5,000 posts during that period. At the lowest end, we have 10,621 users who made at least 100 Tweets (e.g., everyone in our collection). The full set of cohorts and their data description are given in Table III.

We then ran the same procedure and analysis as before. For each user in the cohort, for each amount of historic data, we ran 1,000 simulations and recorded the average KL for the constructed DoW model. The results are shown in Figure 3. We note, for clarity in the figure we only show the results for each cohort with the KL threshold set to 0.1.

The first result to note from the figure is that the more data we have overall for a user, the better we can model the distribution. This is fairly obvious and is reflected by the fact that the cohorts align in order vertically in the figure (e.g., the model is the best with the cohort whose users have a minimum of 5,000 posts and the worst when the cohort only requires at least 100). Perhaps less obvious are that the gaps between 2,500 and 5,000 are much smaller than the other gaps between cohorts. So it appears that after some point, a sufficient amount of data is seen to be considered "large." Also, when a user has at least 500 posts, the results seem to begin to become useful in a practical sense (for instance in this case we can characterize more than 50% of the users with this low KL).

Finally, we demonstrate that we can build within-the-day (WtD) models as well. Figure 4 shows the results for the same procedure as above, except for that the time granularity is set for the WtD model, rather than the DoW model. Interestingly, not only can we build models that explain the WtD behavior, these models outperform the DoW models. This is especially clear for the cohorts with smaller amounts of data. This is likely due to the fact that we are modeling fewer categories (three time periods instead of five days), so we have more observations for fewer categories. Regardless, this type of WtD analysis is quite useful and our results demonstrate that it is possible.

IV. FUTURE WORK

This current effort reflects our early work in temporally modeling the posting behavior on social media. We currently model at daily units (e.g., a particular day of the week or

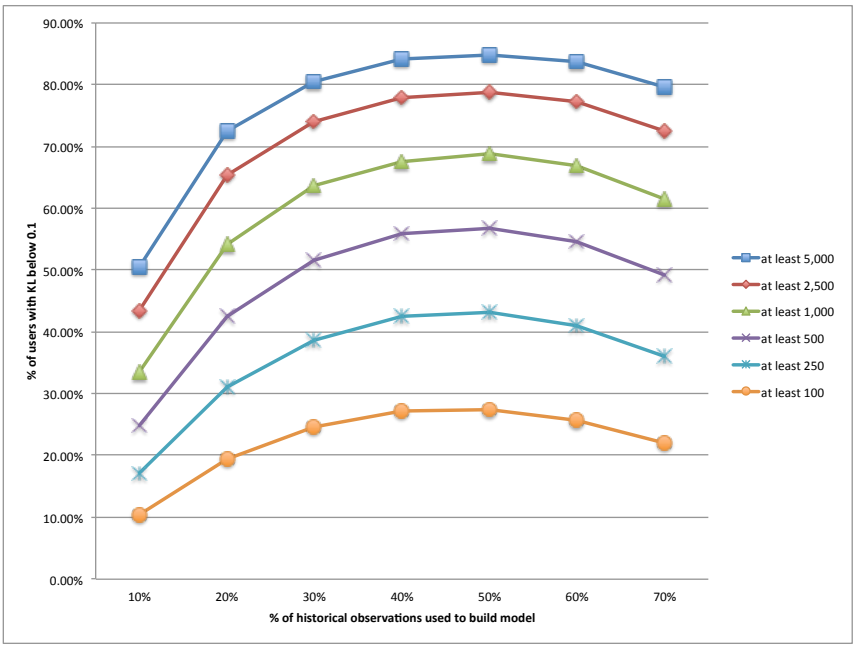


Fig. 3. Cohort analysis for fitting the DoW model

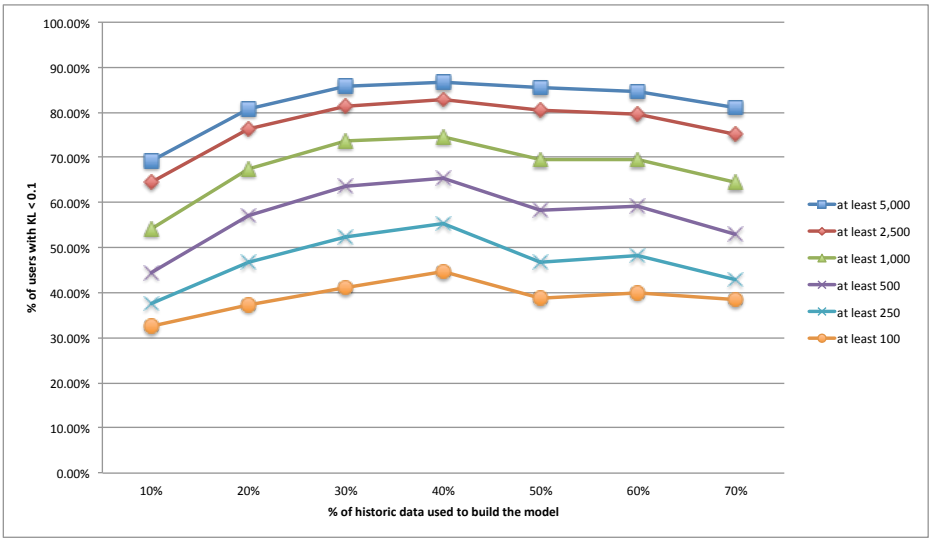


Fig. 4. Fitting distributions of posting behavior for WtD model

a period within the day), but we do not yet differentiate between specific days at larger scale (e..g, particular day(s) of the month) or special days, such as holidays or historic events (such as during a natural disasters) which should be treated specially. Also, we didn't yet investigate high levels of granularity (e.g., weeks within a month or months within a year) or much lower levels (e.g, minutes within a day).

Therefore, in the future we intend to build more sophisticated models that provide deeper insight through various levels of temporal granularity. For instance, we may model day of the week, within the day, day of the month, and month of the year, all at once, and incorporate them into a single model. However, since we have to incorporate time series data at different levels, some of which may be parse, this is a statistical challenge to model. Further, this model should take into account the special days and model the behavior accordingly. However, these future enhancements could help build a more robust individual life pattern to provide deeper insights.

V. Conclusion

In this paper we demonstrated how to build models of user posting behavior on the micro-blogging service Twitter. This type of analysis is useful for modeling an individual life

pattern for a user which can then be used for applications such as determining when someone might be sick, as deviations from this pattern become obvious. Our contributions in this paper were two-fold. First, we demonstrated a probabilistic model for posting behavior, modeling the posting time as a distribution. Second, we performed an empirical study to understand how well the model explains the user's behavior and also studying how the amount of data used to build the models effects its overall performance.

REFERENCES

[1] T. Sakaki, M. Okazaki, and Y. Matsuo, "Earthquake shakes twitter users: Real-time event detection by social sensors," in *Proceedings of the 19th International Conference on World Wide Web*, 2010.

[2] S. A. Macskassy and M. Michelson, "Why do people retweet? anti-homophily wins the day!" in *Proceedings of the International Conference on Weblogs and Social Media*, 2011.

[3] D. Scanfeld, V. Scanfeld, and E. L. Larson, "Dissemination of health information through social networks: Twitter and antibiotics," *American Journal of Infection Control*, vol. 38, no. 3, pp. 182 – 188, 2010.

[4] C. Chew and G. Eysenbach, "Pandemics in the age of twitter: Content analysis of tweets during the 2009 h1n1 outbreak," *PLoS ONE*, vol. 5, no. 11, p. e14118, 11 2010.

[5] Z. Cheng, J. Caverlee, and K. Lee, "You are where you tweet: A content-based approach to geo-locating twitter users," in *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, 2010.

[6] S. Kullback and R. A. Leibler, "On information and sufficiency," *The Annals of Mathematical Statistics*, vol. 22, no. 1, pp. 79–86, 03 1951. [Online]. Available: http://dx.doi.org/10.1214/aoms/1177729694

Case Studies: Big Data Analytics for System Health Monitoring

Dinkar Mylaraswamy[1], Brian Xu[1], Paul Dietrich[1] and Anandavel Murugan[2]

[1] Honeywell Aerospace, Golden Valley, MN, USA

[2] Honeywell Technology Solutions Limited, Madurai, India

Abstract. *This paper describes a case-study where we built and exercised a cloud computing framework with machine learning (ML) algorithms to improve the accuracy of Auxiliary Power Units (APU) health monitoring. An APU is a small turbo machine that flies on all commercial transport airplanes. The paper describes the objective of our study, sources of available data, the ETL scripts to populate the underlying HBase tables and two examples. In one example machine learning algorithms operating on multiple data sources produce useful insights to increase our ability to predict APU wear from 39% to 56%. In the second example, it increased our ability to predict shutdown events from 19% to 60%. This case-study illustrates the effectiveness of big data analytics and tools to discover additional insights that can further reduce operational interrupts arising from airborne equipment problems.*

Keywords: Big Data Analytics, Machine Learning, System Health, Case Studies, Cloud Computing.

1 Introduction

In recent years, the aviation industry has witnessed a steady increase in more data being collected by Aircraft Condition Monitoring Systems (ACMS). Data volumes ranging from 5 ~ 10 megabytes per flight hour (each aircraft) are routinely collected by onboard recorders and sent directly over airport Wi-Fi and GSM wireless networks without incurring the costs associated with ACARS messaging. Advances in IT and software that allow secure movement of data from airplanes provide an ideal framework for embedding statistical machine learning algorithms that can discover sweet-spots in global operations can feedback into day-to-day actions. This cloud computing based information network created by these connected aircrafts (a part of Industrial Internet) hold the potential of providing valuable knowledge needed to maintain profitability in an economically challenged civil aviation industry.

Technically, one of our study areas that can benefit most from current big data analytics is to reduce maintenance cost of high-value aerospace assets. For example, a recent GE article estimates a $250M savings in engine maintenance cost [1] is possible using insights gained from machine learning (ML), data mining and knowledge discovery. While the actual savings depend on specific aftermarket business policies, such case studies have been widely reported, clearly indicating the potential of data mining methods for discovering useful business insights from big data. In this paper, we describe our approach to data mining using data collected from auxiliary power units (APUs).

Our primary objective is to develop Predictive analytics. Specifically predict events and failures in an APU before they cause an operational interrupt. Our approach is to analyze past historic data available from two distinct sources: data collected while the APU is operational and data when the APU is being repaired. Currently these data sources are separated and in many cases, the design schemas within these databases are poorly documented. Further, the schema is not uniform when dealing with different APU makes – since these products were introduced at different time periods. The repair records are aggregated by three global centers—each of which uses different tools that introduces site-specific biases. The challenge lies in making multiple queries, retrieving the correlated data and developing predictive analytics. Data quality (missing and incorrect entries) makes this problem difficult—using a common APU serial number and timestamp is not sufficient. It is this problem statement we try to address using big data tools.

This paper is organized as follows: Sections 2 introduces the problem along with a brief description of the APU. Section 3 describes our cloud-based ML framework and its key components. Section 4 describes the Parser used for data ETL to populate data into our HBase and HDFS. Analysis and results are presented in Section 5. Conclusions and significance of this work is presented in Section 6.

2 Problem Statement

An auxiliary power unit (APU) is a small gas turbine engine that provides pneumatic and electrical power to the airplane. The main functions of an APU are listed below:

1. High pressure bleed air for starting the main propulsion engines
2. Provide air to the environmental control system and cabin pressurization
3. Drive a generator to provide electric power for the airplane

Though an APU does not provide propulsion, the internals are as complex as large jet engines. A typical APU for commercial transport aircraft is broken up into three main sections – the *power section*, the *load compressor* and the *gearbox*.

Two ongoing programs within Honeywell are relevant to the discussion presented in this paper.

1) Predictive Trend Monitoring and Diagnostic (PTMD) program which reports data from APUs while they are operational
2) Product In-Service Performance (PIPS) program which reports data from APUs after they are removed from the airplane and sent to the repair shop.

Honeywell's Predictive Trend Monitoring and Diagnostic (PTMD) services provide APU usage and sensory data downloaded through the Aircraft Communication and Addressing Reporting System (ACARS). Stated simply, it provides sensory data while the APU is operational and installed on the aircraft.

The general description of PTMD is given in [7]. However it suffices to mention the following three key outputs from the PTMD system:

A. **AHRS**: This is called the APU usage hours or AHRS. This is similar to the odometer reading in a car, albeit it measures the cumulative "time duration" rather than distance.
B. **EGT margin**. Within the APU fuel is converted to mechanical energy. The exhaust gas from the APU is the energy that is not converted to useful work. EGT margin measured in degree Celsius is a measure of the thermodynamic efficiency of the APU.
C. **PB margin**. In order to start the main propulsion engine, bleed air provided by the APU must meet certain pressure and flow conditions. The bleed margin (abbreviated as PB margin) measures the APU's ability to meet these constraints.

As the APU ages, both the EGT and PB margin decreases steadily with AHRS and at some point, the APU operation is no longer economically viable and hence it is removed from the aircraft and sent to the repair shop for repairs.

When the APU is received at the repair shop, Honeywell's Product In-Service Performance System (PIPS) captures actions performed at the repair shop. The general description of PIPS is outside the scope of this paper. However it suffices to mention the following two three annotations from the PIPS:

a) **Symptom**: This is an enumerated text describing the reason for removing an APU, typically provided by the airline operator. Examples include: no-start (APU is not able to start), auto-shutdown (the APU is shutting down due to some internal problem), and high-wear (the APU efficiency has decreased beyond its economic threshold.
b) **Description**. This is a free-form text provided as a summary of the repairs performed on the APU. It includes parts replaced, damage observed, and possible primary cause and secondary effects.

Our primary objective is to develop analytics aimed at predicting (1) severe wear, and (2) erratic behavior arising from auto-shutdowns (the APU switches itself off to protect internal damage). Figure 1 summarizes our problem statement.

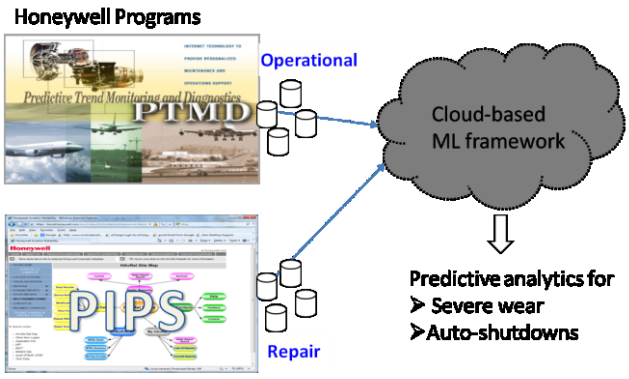


Figure 1: Summary of data available for applying ML methods and mining

Next we briefly describe the cloud-based ML framework used to develop these predictive analytics.

3 Cloud-based ML Framework

Details of the cloud-based ML framework are described in [2]. Here we present a short description. The cloud is a Hadoop cluster of Linux machines. Salient features shown in Figure 2 are described below. Additional details are provided in [2].

1. Our data storage strategy uses HDFS—distributed file system built on top of Hadoop
2. Mahout provides machine learning algorithms for clustering, classification tasks. The list also

includes algorithms classifying textual repair descriptions.

3. The HBase has two technical components: (a) Convenient base classes that support Hadoop MapReduce jobs and functions with HBase tables; and (b) Query predicate pushes down via server side scan and gets filters that will select related data for track management systems.
4. A user can plug-in algorithms based on domain knowledge. Our initial focus is on algorithms developed in Mathwork's Matlab and the R language.

Information from the two sources (PTMD and PIPS) arrives as *reports*. We considered two aspects while arriving at the storage strategy needed for analytics: (1) Reports need to be logically grouped, based on typical access pattern, (2) File should also be organized in a way to support efficient map reduce jobs. When we initially stored our reports in native report format, map reduce jobs took more time than the desktop execution. The reason for this behavior is as follows: map-reduce jobs are efficient in handling files smaller than the HDFS block size. We found typical report size (from both PTMD and PIPS) was less than few MBs while the HDFS block size is 64MB. This not only slows down map reduce jobs, it is also not good for HDFS.

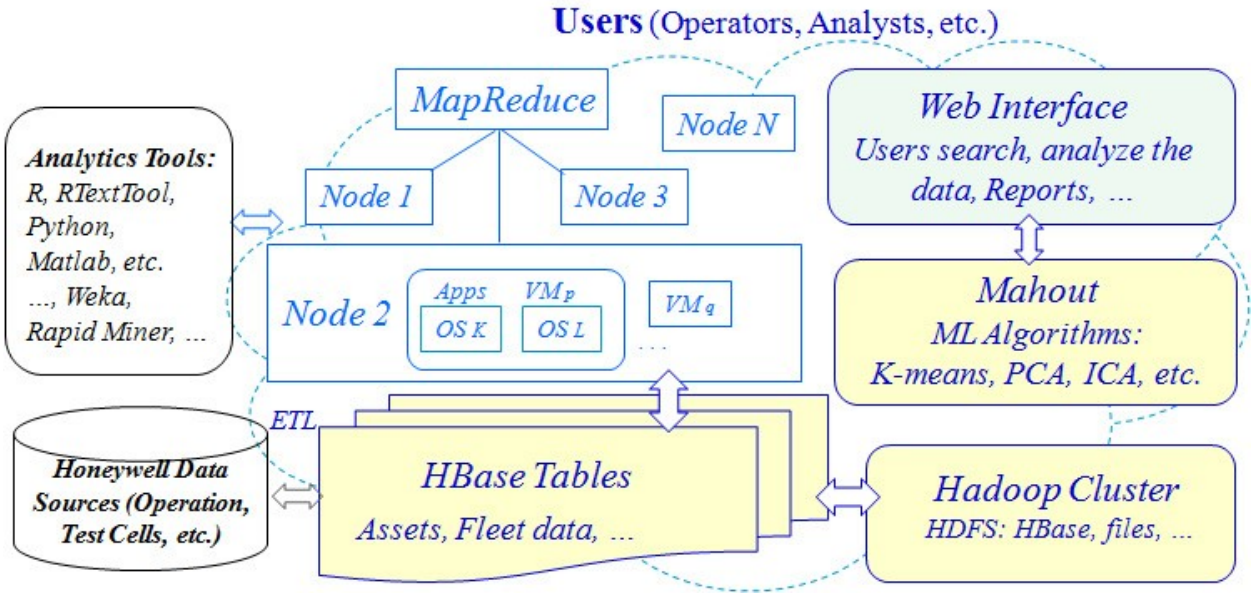


Figure 2:
Architecture and Components of the Cloud-based ML Framework

Honeywell has been supplying APU for the commercial aerospace industry for several decades. As a result a large amount of legacy data was available for predictive analytics development. Organizing this legacy data was an important step in our process. The legacy data ETL is described in the next section.

4 Data ETL to HBase and HDFS Using Parser

An important first step in our process was to import legacy APU reports. The format of these reports range from comma-separated-values ASCII text files to binary files with specific encoding. The import process into HBase and HDFS is carried out using our Python package (*importer*.py). The importer module is a script with a number of different options to control the source of input data (SQL database, CSV files, etc.), its specific format and the destination for the data. This information is supplied in a configuration file. The HappyBase library [3] is used for access to HBase (via the Python Thrift gateway) and the Pydoop package [4] is used for access to HDFS. Our importer script has a valuable debugging feature that allows us to see exactly what would be added to any HBase table without committing any changes. Figure 3 shows the screen shot illustrating the usage of this script for importing legacy data from the PIPS source.

The destination for the imported data extracted from legacy reports are a series of HBase tables. The HBase table design is described next.

```
·(thrifty)tt-a664-4 [1250]%
·(thrifty)tt-a664-4 [1250]%
·(thrifty)tt-a664-4 [1250]%
·(thrifty)tt-a664-4 [1250]% ./importer/importer.py --index-file=./PIPS/PIPS.APU13
·3.2013_06_17.csv --field-separator=',' --dry-run --leave-source PIPS
```

Figure 3:
A Screenshot of importing PIPS data into our HBase and HDFS

An important consideration in HBase table design is row key design. Each row in HBase table is identified by a random unique key. Best practice is to make this key a composite key matching the querying pattern. A composite key is created by adding multiple attributes of data stored in the row. In our case, users would search reports based on the asset model, serial number and operator. They would then filter the reports based on a time range. So row key we chose had the following format:

<model>:<serial>:<operator>:<timestamp>

To keep the row key unique, random characters are appended to the row key. With this row key format, a scan can be done with a row key filter. Row key filter based scan just looks up the row keys and hence it is faster. Random characters appended to the key also help in avoiding region server hotspotting. Region server hotspotting is a case when more data is accumulated in one region server when the row keys are logographically closer. This overloads one region server when the other region server is underutilized.

The other aspect of HBase table design is the column and column family design. In our case, for all the report tables, we have two column families. One stores the report data and other stores the asset details. The column name for these families is listed in Figure 4.

```
    'ast:mod',      # APU Model
    'ast:ser',      # APU Serial Number
    'ast:ow',       # APU Owner
    'ast:OC',       # APU Ownercode
    'rpt:mis',      # Report Mission
    'rpt:pth',      # Report Path
    'rpt:tm',       # Report Date
```

Figure 4: A Screenshot of the column families

The next section describes the analysis and the results obtained by applying ML methods.

5 Analysis and Results

As stated earlier our objective was to explore ML methods that could discover (1) factors that indicated severe wear on the APU, and (2) factors that have led to an auto-shutdown event. The end goal of this exercise was to encode the resulting understanding as a predictive analytic algorithm for continuous monitoring of APU health. The analysis performed on the large volume and variety of data being collected by the PTMD and the PIPS program. Our intent in this section is to summarize the effectiveness of the ML methods.

A fundamental concept in our analysis is called a lifecycle point (LCP). LCP for an APU is the time interval between its n^{th} repair shop visit and the $(n+1)^{th}$ repair shop visit. $n = 0$ is a special case when the APU leaves the manufacturing shop and enters the operating fleet for the first time. The LCP is explained pictorially in Figure 5.

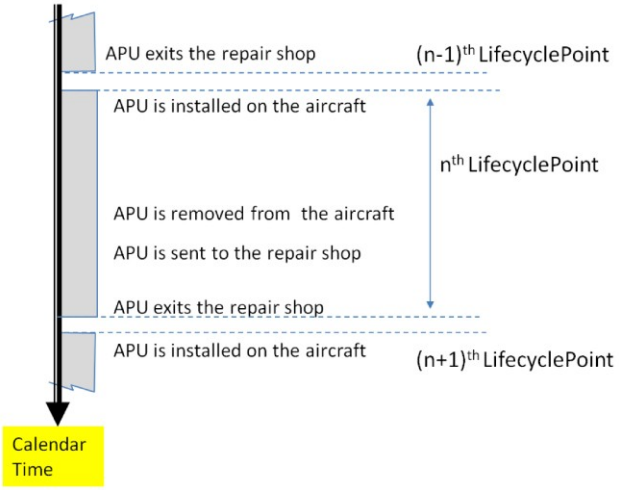


Figure 5: LifeCyclePoint is the 'atomic unit' for applying ML methods

The primary indicators available for ML analysis while the APU is in the repair shop are the symptoms and free-form text description of maintainer actions and observations. The free-form text was processed using a combination of Bayesian and Regression text-mining methods to produce target categories/classes that summarized airline operator describing the reason for removing the APU and the repair performed. Two specific categories of interest in this study were those related to severe wear and auto-shutdown.

In section 2 we described the three primary indicators available for ML analysis while the APU is operational. These are: EGT margin, PB Margin and

Usage Hours (AHRS). As the APU ages, the both the EGT and PB margin decreases steadily with AHRS. Typical trend of EGT Margin within a lifecycle point is shown in Figure 6.

It is well known in the industry that the EGT Margin is a good indicator of APU general wear [5]. The margin becomes smaller (approaches zero) as the APU wears and becomes negative as the APU wears severely. Further, an APU that has been operating longer is more likely to develop internal faults and hence more likely to exhibit auto-shutdowns. In addition to the numeric values of EGT and PB margin at specific AHRS values, we pre-processed the trend lines to retain the following features.

1. Slope – this measured the rate at which the EGT and PB margins changed with respect to AHRS
2. Jumps – this measured large changes in the EGT and PB margins between two consecutive flights
3. Slope changes – this measured inflexion points in the EGT and PB trend line where the slope changed values.

Figure 6: EGT margin as a function of APU usage.

A total of $3 + 2 \times 3 = 9$ features were available for the clustering and classification ML algorithms.

Both the PTMD and the PIPS programs have been tracking APU lifecycles for several years. They span three APU models used by more than 100 global airline operators. Lifecycle points available for ML are summarized in Table 1.

Table 1 : Scope of ML analysis

APU	# of Lifecycle points	Features per lifecycle point
Model A	1001	9
Model B	764	9
Model C	629	9

We applied three ML classification methods—Random Forest (decision trees), Support Vector Machine and Naïve Bayes. We assume the reader is familiar with three standard ML algorithms. The choice of these methods was motivated by the fact that they represent non-overlapping approaches for capturing the decision boundaries. We used a uniform weighted averaging (weight = 1/3) to fuse the output generated from each of the method. In addition to quantifying the accuracy of detection, we were also interested in understanding how much each of the nine features contributed to our ability to predict severe wear and auto-shutdown events.

Figure 7 shows the results for the severe wear category. It is well known in the industry that the EGT Margin is a good indicator overall APU wear [5]. However, as shown in Figure 7, it can only predict he severe wear event only 39-out-100 times. The ML methods discovered three additional features that can increase the effectiveness in predicting APU severe wear from 0.39 to 0.56. These additional features were: (1) the slope of the EGT margin trend, (2) the slope of the PB margin, and (3) jumps in the EGT trend line.

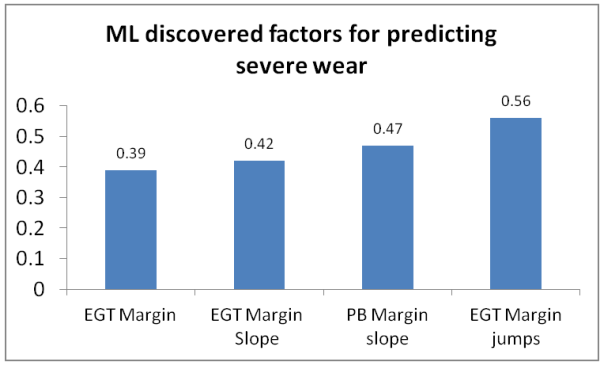


Figure 7: New features discovered by the Cloud-computing ML framework for predicting APU wear.

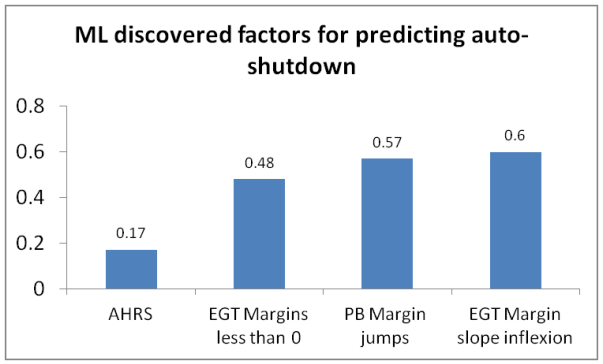


Figure 8: Features discovered by the Cloud ML framework for predicting APU auto-shutdown.

Figure 8 shows the results for the second example—predicting APU auto-shutdown events. While the APU age characterized by the AHRS can only

provide 17% prediction accuracy, the framework discovered three new features that increased the accuracy to 60%. As shown in Figure 8 these features were: (1) EGT margins becoming negative, (2) jumps in the PB trend line, and (3) inflexions in the EGT margin trend line.

6 Conclusions

In this paper, we described the analysis and the results we obtained using our cloud-based ML framework. Specifically we described the use-case where we used the framework to gain additional insights for improving the accuracy of Auxiliary Power Units (APUs) health monitoring. The paper described the Honeywell's' data source we used, and the customization of our previously developed cloud-computing framework [2] with respect to data ETL and the HBase table design. Among various ML algorithms, we selected three algorithms to investigate how big data analytics can be scaled and applied to solve real aviation industrial problems.

We illustrated the effectiveness of the cloud computing ML framework using two examples aimed at predicting (1) severe wear, and (2) erratic behavior arising from auto-shutdowns. In the wear example, the ML framework discovered three new insights that can increase the accuracy from 30% to 56%. In the auto-shutdown case, the ML framework discovered three insights that improved the accuracy from 19% to 60%.

Both these examples clearly illustrate the value-provided cloud computing tools and big data analytics can provide to the aerospace industry. For example, insights gained by analyzing decades of historical data can be converted into predictive analytics for system health monitoring. The resulting service would provide alerts to an airline maintainer and minimize operational interrupt. This analysis can be done in real-time using streaming data from more connected airplanes.

Our future work will focus on expanding our big data analytics for developing data-driven predictive analytics for other aerospace assets.

7 References

1. Evans, P. C., & Annunziata, M. (2013). Industrial Internet: Pushing the Boundaries of Minds and Machines. http://files.gereports.com/wp-content/uploads/2012/11/ge-industrial-internet-vision-paper.pdf
2. B. Xu, D. Mylaraswamy, and P. Dietrich. A Cloud Computing Framework with Machine Learning Algorithms for Industrial Applications. WorldCom ICAI, July 2013, Las Vegas, USA.
3. HappyBase, http://happybase.readthedocs.org
4. Pydoop, http://pydoop.sourceforge.net/docs/
5. Honeywell PTMD: https://commerce.honeywell.com/webapp/wcs/stores/servlet/eSystemDisplay?catalogId=10201&storeId=10651&categoryId=42999&langId=-2

Identifying and Resolving Conflicts in Multi Source Data

J.M. Brown, C.G. Fournelle
Boston Fusion, Corp., 1 Van de Graaff Drive, Burlington, MA

John Spina
AFRL/RIEA, Rome, NY

Abstract*—While access to data is continually increasing, it is unclear whether access to reliable information is increasing at close to the same rate, if at all. Data inconsistencies are ubiquitous; as data arrives from multiple sources, analysts and other data consumers must determine what data to accept and what to reject. Today's data consumers need the ability to automatically detect when conflicts occur and examine these inconsistencies in context—using an understanding of temporal evolution to consider how different conflicts may be related, and how the underlying sources factor into the credibility of the information.*

In response to this need, we are developing inference algorithms that detect and resolve conflicting information in multi-source text data by placing the potentially conflicting information in context, to suggest resolutions and to understand the social connections behind different pieces of conflicting information. We embed our techniques in a system called REsolving Differences via Inference (REDI). Our approach uses statistical and graph-based characterizations of data to describe expected structures, and identify combinations of data that deviate from them. Using the expected structures, we then propose resolutions to the data conflicts.

Keywords*—Conflict detection, resolution, context-based inference, NLP integration, anomaly detection*

1. Introduction

Today, we have access to ever expanding collections of rapidly evolving data stores. Analysis and exploitation of that data affects society through many avenues ranging from military intelligence analysts defending national security to journalists covering current events, and from commercial data analysts driving business decisions to ordinary users of the internet searching for information. Unfortunately, not all data or information is credible or trustworthy. Identification of conflicting information across multiple sources (news stories, reports, blogs, etc.) is a challenging task. New capabilities are needed to: (1) examine inconsistencies in context; (2) understand temporal evolution of data and data conflicts; (3) consider how different conflicts may be related to each other; and (4) factor how the underlying sources influence the credibility of the information.

In response to this need, we designed and implemented a proof-of-concept conflict detection and resolution system called REsolving Differences via Inference (REDI). REDI is targeted primarily for intelligence analysts although the techniques can be more broadly applied. The ultimate goal of REDI is to create an analysis system component that will automatically identify and assess conflicting information in multi-source data, and propose resolutions to conflicts when appropriate. REDI outputs can inform analysts of potential issues in their data, and automatically tag conflicting data for subsequent processing by other automated tools.

For the purpose of this paper, we define "conflicting" data as ***two or more data elements that are unlikely to simultaneously be true when reviewed in their proper context***. This perspective shifts the focus from the simplistic "these two strings don't match" view to the more meaningful "these data don't make sense" for conflict detection and resolution.

In prior research, the natural language processing (NLP) community's efforts on conflicting data analysis have focused on determining whether conflicts exist between pre-identified pairs of sentences, as in [5]. These approaches to analyzing potential conflicts do not, however, automatically identify meaningful (i.e., conflicting) pairs of sentences; rather, they assume the sentences as input. Other NLP work in defining conflicting data determined that significant inference would be required to identify conflict [7,8]. Prior NLP research ultimately settled on a simpler task, which was to identify redundancies in the data, relying on *entailment* to formalize the redundancy: *A* entails *B* if *B* being true forces *A* to be true. This notion enabled technology developers to consider how some reported information may be more specific than others, or updated. This approach fails to tackle the harder problem of deciding when the truth of *A* makes *B* impossible or highly unlikely, since making such judgments requires consideration of many different pieces of information considered in their proper temporal and locative context. In contrast to existing approaches, REDI both *identifies and resolves* conflicts in data.

In the sections that follow, we discuss the REDI system we developed to detect and resolve conflicts. We provide an overview of each of the system's main components, describe a test data set (the MUC 3 / MUC 4 corpus), and

report our preliminary test results. We summarize our findings, and discuss ideas for future improvements.

2. System Overview

The REDI framework consists of five main components to extract and associate information from multi-source data, identify potential conflicts based on statistical and graphical properties, and propose resolutions to the identified conflicts (Figure 1).

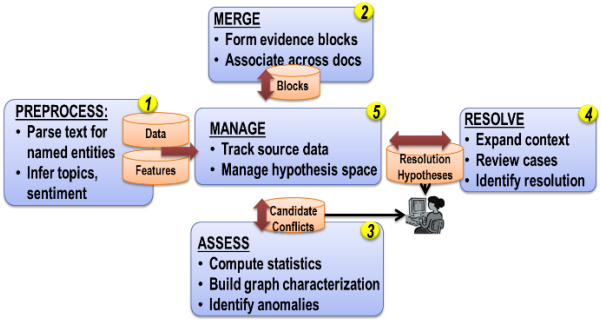


Figure 1. System overview

Each of the four analysis components (*Preprocess*, *Merge*, *Assess*, and *Resolve*) accesses data from the others, and tasks/updates the other components to further its own progress via the fifth component (*Manage*). Specifically, these components perform the following functions:

Preprocess prepares the available data sources for conflict resolution by invoking natural language processing tools for feature extraction.

Merge ingests the extracted text and unstructured text sources and merges them into "blocks," where each block consists of one or more distinct entities or events that have been (temporarily) associated and are awaiting further processing.

Assess reviews the blocked data set, and uses characteristics of the data to deconflict entity blocks and highlight other conflicts in the data.

Resolve performs a case analysis on the nominated conflicts, and attempts to provide a resolution to the conflict, if possible.

Manage maintains the common data for the system. This data includes the working collection of entities and events, and the features computed for them, as well as the conflicts that REDI has identified.

3. Preprocess

Preprocess prepares the available data sources for conflict resolution. We include a plug-and-play ability to incorporate existing NLP technologies for topic and sentiment modeling, pedigree management, and secondary processing as needed for later context reasoning.

Table 1 lists some of the NLP tools we integrated into the REDI system.

Table 1. Open source language processing tools and related features in REDI

Tool	Features
Stanford Core NLP [13]	Named entity recognition; Coreference resolution; Date & time standardization; Syntactic dependency trees; User-defined entities and entity class extraction
Semafor [14]	Provides semantic role labels
FrameNet [10]	Provides synonyms, lists of related terms
MALLET [12]	Latent Dirichlet Allocation-based topic modeling [1]
Linguistic Inquiry and Word Count, LIWC [11]	Document-level sentiment features

In addition to the feature types described in Table 1, we incorporated additional language-based features by using the extensible capabilities of Stanford's NLP toolkit. With this toolkit, we were able to define special classes of entities with particular operational interest, to ensure they were part of our structured data set. These included concepts such as *Attack* (attack, kill, murder, kidnap, bomb), *Military* (soldier, colonel, general, brigade, infantry), and *Infrastructure* (bridge, highway, building).

4. Merge

Merge ingests the extracted text and unstructured text sources and merges them into a single data set according to a common data model. In the merging process, we take an aggressive strategy that is likely to "over associate" individual data—i.e., we use a high recall approach to creating individual entities, at the risk of including data from other distinct individuals. We refer to these post-merge, pre-deconfliction entities as "blocks", where each block consists of one or more distinct entities or events that have been (temporarily) associated and are awaiting further processing. This approach is consistent with many state-of-the-art entity resolution approaches that exist, and allows us to place our emphasis on the subsequent deconfliction to create appropriate distinct entities and events, as they track all (even loosely) related information about the entity or event in question.

Merge creates a graph structure from the blocks. There are three primary types of nodes—entity nodes, event nodes, and metadata—and links represent extracted or inferred

associations between nodes. In the case of the entity graph, links include co-occurrence at the document or sentence level as well as labeled semantic relationships. See Figure 2.

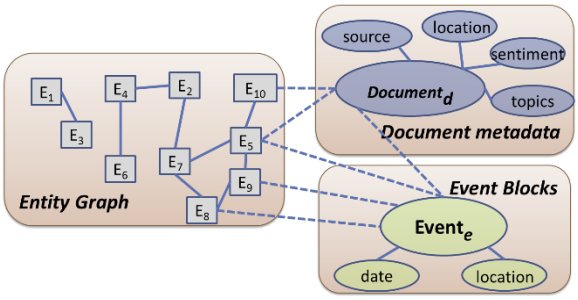


Figure 2. Blocked data in graph representation

Note that we also persist features of the primary nodes as secondary nodes in the graph, with links back to the primary nodes. For example, in Figure 2, locations mentioned in documents become nodes in the graph, with links back to their source documents.

To make decisions about whether to associate two data elements to a common entity block, we make a series of decisions about their similarity, invoking criteria based on Levenshtein distance [2], phonetic transformation (via Metaphone [15]), expansions of potential compound word entities, and named entity recognition class. If the candidate pair of elements scores above a threshold, we merge them into a common block for subsequent assessment.

To form event blocks, *Merge* segmented documents according to likely event boundaries, and grouped all entities within (sub-document) boundaries into unique events. The *Merge* algorithm then calculated the likelihood of two entities randomly occurring in the same event block given their frequency throughout the corpus. This method downgrades co-occurrences of common entities which are more likely to occur together purely by chance. The event merge algorithm also considered whether the event dates match with potential conflicts noted for later review.

Finally, *Merge* identifies multiple types of relationships between entities based on co-occurrence criteria, and on the semantic content of the documents in which they appear.

Figure 3 and Figure 4 show a graph of the entities extracted from a single document in the MUC 3 /MUC 4 data set, before and after *Merge* processing. Note that the processing refines the overall graph of the dataset, and identifies chains of connections that are not apparent prior to *Merge*.

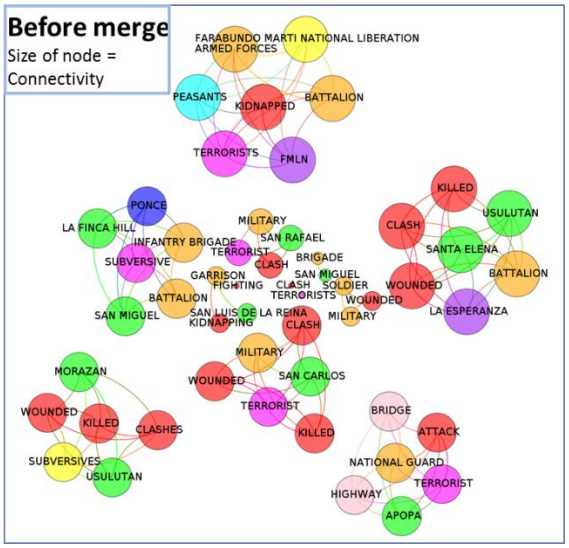


Figure 3. Entities extracted from a single document in the MUC 3 / MUC 4 data set prior to Merge processing, where connections represent co-occurrence in sentences.

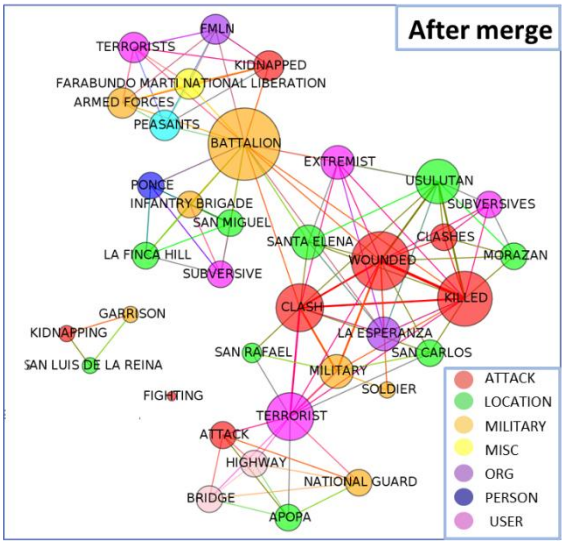


Figure 4. The same entities as in Figure 3 after Merge processing.

5. Assess

Assess reviews the blocked data set, and uses characteristics of the data to deconflict entity blocks and highlight other conflicts in the data. *Assess* computes graph-based and statistical characterizations of the data to identify anomalies and unexpected data combinations. After filtering low-priority, low-value conflicts, this component nominates a set of candidate data conflicts for further investigation in the *Resolve* component.

Assess analyzes the data to identify contradictory or unlikely combinations of data, including:

- ***Statistical outliers***, based on a combined statistical description of the data that includes available data fields and derived data descriptors; and
- ***Graph outliers***, i.e., individuals whose graph characteristics are unusual for the current population.

Assess detects conflicts and anomalies via a two stage process, where the first stage is responsible for identifying factual problems with the entity blocks and forwarding them to be resolved. The second stage uses the cleaned and resolved entity blocks to look for more abstract anomalies by leveraging their extended semantic representation.

We compute multiple statistics over the data set to determine characteristics such as uniqueness of names, relationship types, and other individual and aggregated pieces of information. In developing the REDI system, we considered a simple set of statistics to characterize block content.

Entity level statistics include:

- Number of names for the entity;
- Number of different relationships associated with the entity's name;
- Number of documents that reference the entity;
- Number of distinct sources that reference the entity;
- Distribution (histogram) of timestamps for mentions in newswire; and
- Number of distinct topics that appear in the documents containing mentions of the entity.

Event level statistics include:

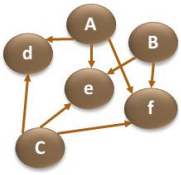
- Number of entities per document;
- Number of entities in the document that do not appear in other documents;
- Number of topics in the document; and
- Number of different dates appearing in the document.

Complementing the data statistics, we also compute graph characteristics, which are key to analyzing the context surrounding the entity blocks. Graph characteristics provide insight into the expected social structure and connectivity patterns, highlighting unexpected or unusual data combinations. We investigated two different methods of characterizing graphs: normative graph pattern (NGP) analysis [1,4,] and cosine similarity. We focused our analysis on extracted features from unstructured data, which limited the utility of the NGP analysis; however, we believe NGP analysis will make a greater contribution towards characterizing data from structured data bases. Cosine similarity demonstrated immediate value even on the limited semantics of our extracted data.

Cosine similarity for pairs of entities in a graph identifies entities with similar neighbors by determining the cosine similarity of the relationship vectors for entity pairs. This method proved excellent for determining normative connections (e.g., which entities are most connected). Formally, we define a measure of cosine similarity of nearest neighbors such that: Σ

$$\frac{\sqrt{\Sigma} \quad \sqrt{\Sigma}}{}$$

where *a* is the number of connections between entity 1 and potential neighbor *i* and *b* is the number of connections between entity 2 and *i.* Figure 5 shows a sample graph to illustrate how cosine similarity of nearest neighbors is calculated. Note that with this measure, we see that A and C (with identical neighbor sets) are scored as more similar than B and C, which exhibit overlapping, but non-identical neighbor sets.



cos$\sqrt{}$ $\sqrt{}$ between A & C: = = 1

cos$\sqrt{}$ $\sqrt{}$ between B & C: = = 0.82

Figure 5. Cosine similarity is higher for pairs of nodes with more similar neighbor sets

After flagging the anomalous graph structures, REDI uses the expected graph structure to propose resolutions.

6. Resolve

Resolve performs a case analysis on the nominated conflicts, and attempts to provide a resolution to the conflict, if possible. This component assembles additional information about the conflicts, possibly tasking the extraction technologies for secondary information about the data and/or sources.

The essence of conflict detection and resolution requires that the data be viewed in its appropriate context. To resolve conflicts in data, we must make inferences on (1) the underlying knowledge, or (2) biases that could have led to the conflicts. To date, we have demonstrated early progress against (1), and have laid the framework for (2) to reach back to the source network surrounding the conflict and consider the consistency, reliability, and dynamics among underlying sources.

We identified several outliers in the entity blocks based on cosine similarity measures. The following are examples of resolutions the REDI system proposed on our test data (see Section 8) using similarity measures alone:

- REDI correctly identified several extracted entities, including *William Walker*, *Mr William Walker*, *William*

Walter, *William Walkler*, and *Walker*, to be a common entity, despite two misspellings.

- REDI determined that *FBI* and *U.S. FBI* are the same extracted entity, and correctly rejected the near-string match entity *FBIS*, despite the close similarity and sound of the two names.
- REDI resolved that the two "distinct" extracted countries, *Uruguay* and *Uraguay*, must reference the same location, despite the misspelling of the country name.
- REDI correctly recognized that the two organizations, *FSLN* and *FMLN*, which were initially merged into a single entity block because of their proximity in names, must be, in fact, different organizations because of their network structure.
- REDI correctly associated the entity block "*Rivera*" with the entity block "*Irias Rivera*" and correctly rejected merging with "*Arturo Rivera y Damas*"; that is, REDI could recognize which association to make for information about *Rivera* based on network information, without relying on heuristics about surname association that would have mistakenly associated this information with both (distinct) entities.

7. Manage

Manage maintains the common data for the system. This data includes the working collection of entities and events, and the features computed for them, as well as the conflicts that REDI has identified.

Our experience in this research demonstrated that conflict detection and resolution sometimes require sophisticated inference techniques over many different pieces of information considered in their proper temporal and locative context. Refining this process is part of our ongoing research efforts.

8. Test Data

To test our approach on real data, we used the MUC 3/MUC 4 data set. This dataset consists of newswire collected circa 1998 by the Foreign Broadcast Information Service (now The Open Source Center). The MUC 3/MUC 4 data provides us with the ability to draw information from multiple text documents which haven't been "normalized," reflecting the inconsistencies typical of multisource data, and focuses on terrorism events occurring in Latin America.

As Figure 6 shows, the corpus is highly interconnected, with common entities and topics forming a densely connected graph over the documents in the corpus.

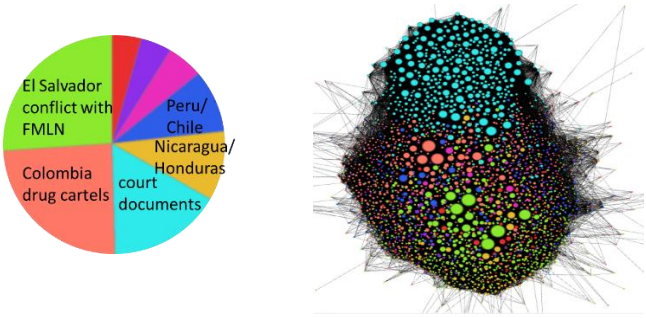


Figure 6. Distribution of most common topics in MUC 3/ MUC 4 data (left), and interconnectedness of topics across documents (right).

9. Evaluation

We conducted a manual review of system output, to determine the accuracy of a random sample of 100 out of the ~4000 entity blocks in the MUC data for correct or incorrect merges after two points in the Resolve process: (1) using only the fuzzy string match criteria, and (2) after invoking *Resolve*'s graph-based analysis. From this set of entity blocks, REDI produced ~700 candidate matches, which we subsequently labeled by hand to establish "ground truth" for this test set. REDI first applies a fuzzy string match, and then a graph-based cosine similarity measure, and assigns each candidate match to one of three bins: ***accept***, ***potential***, and ***reject***.

After applying fuzzy string match, the precision and recall of the *accept* matches were high (P=96% and R=88%). For the 368 merges we categorized as *potential* (almost 50% of all candidate merges), REDI's fuzzy string match criteria was unable to make a clear determination. The graph-based cosine similarity criteria determined that ~200 of the merges should be rejected, reducing the number of unclear potential merges to a more manageable 135 candidates. Also, the precision and recall of the *reject* were excellent (P>99%, R=98%) following the graph-based cosine similarity match.

Table 2 and Table 3 show evaluation results of a random sample of 100 entity blocks and their associated merges out of the ~4000 entity blocks in the MUC 3/4 dataset. Table 2 shows the statistics after the fuzzy string match, while the statistics after both fuzzy string match and then graph-based resolution are in Table 3.

Table 2. Resolve results using fuzzy name matching only

Fuzzy string	Predicted Accept	Predicted Potential	Predicted Reject
Actual Accept	*28.9*	*33.5*	*4*
Actual Potential	*14*	*14*	*3*
Actual Reject	*1.1*	*320.5*	*325*

Table 3. Resolve results using graph-based information about entity pairs

Graph-based	Predicted Accept	Predicted Potential	Predicted Reject
Actual Accept	*38.4*	*14.5*	*12.5*
Actual Potential	*16*	*12*	*4*
Actual Reject	*4.6*	*108.5*	*523.5*

10.Next Steps

As we extend our research into more complex and interconnected conflict sets, we recognize the need for more sophisticated data management techniques, including multiple hypothesis management to track potential conflicts, recommended resolutions, and associated uncertainty. This approach will enable us to analyze the complex interactions of sequences of resolutions, and to consider such multiple hypotheses without necessitating aggressive, too-soon pruning techniques because of the associated hypothesis space explosion. Additionally, we will extend the context of the conflict to consider the source(s) of the contradiction, and the biases and/or inconsistencies they have provided historically. We intend to achieve this goal by invoking analysis of linguistic features (including sentiment analysis) and graph-based analysis.

The technology we are developing will be beneficial to operational customers, particularly for the essential process of building intelligence from various data sources. The ability to identify conflicting, suspicious and inconsistent information across those data sources would help broaden the analyst's situation awareness as part of the Collection, Processing, Exploitation and Dissemination (CPED) process. One of the challenges that the operational community still faces today, with regards to data analytics, distinguishing between contradictory or conflicting data and trivial data, which this research addresses. From the analyst's standpoint, it would be helpful to have the ability to distinguish between data that was intentionally corrupted (deception) and data that could be deemed as trivial, unrelated or unimportant. Being able to resolve conflicting, suspicious, inconsistent and intentionally deceptive information, specifically when the contradictions emerge within the analyst's trusted sources, would help produce a more accurate picture of conditions. Currently, many conflicts are left unresolved because there has been no way of informing the analyst; deploying this technology to the operational community shows promise for addressing this problem.

11. Summary

In this paper, we presented a formal method for identifying potential conflicts based on statistical and graphical properties of multi-source data. We introduced capabilities to assess both individual- and event-level conflicts. We generated a prototype design for a conflict detection and resolution system that will support data preprocessing, conflict detection, assessment, and resolution, and demonstrated its effectiveness on real-world data. Our results demonstrate that the combination of statistical and graph-based approaches can not only identify anomalies in multi-source data, but can be leveraged to resolve those conflicts based on the expected structure of the data.

12. Acknowledgements

This research was sponsored by the U.S. Air Force Research Labs (AFRL) under contract FA8750-13-C-0191. We gratefully acknowledge the support of AFRL and our technical monitor, Mr. John Spina. The opinions expressed in this paper are those of the authors and do not necessarily represent the views of the U.S. Air Force, AFLR Lab, or the United State Government.

13. References

1. Blei, David. *Probabilistic topic models*. Communications of the ACM, 55(4):77–84, 2012.
2. Cohen W.W., P. Ravikumar, and S. E. Fienberg, "A comparison of string distance metrics for name-matching tasks", In Proc. of IIWEB, pages 73-78, 2003.
3. D. Cook and L. Holder (2000). Graph-Based Data Mining. *IEEE Intelligent Systems*, Vol. 15, No. 2.
4. D. Cook and L. Holder (Editors). *Mining Graph Data.* John Wiley and Sons, 2006.
5. De Marneffe, Marie-Catherine, Anna N. Rafferty, and Christopher D. Manning. "Finding Contradictions in Text." *ACL*. Vol. 8. 2008.
6. W. Eberle and L. Holder (2007). Anomaly Detection in Data Represented as Graphs. *Intelligent Data Analysis*, Vol. 11, No. 6.
7. Harabagiu, Sanda, and Andrew Hickl. "Methods for using textual entailment in open-domain question answering." *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational*

Linguistics. Association for Computational Linguistics, 2006.
8. Hickl, Andrew, et al. "Recognizing textual entailment with LCC's GROUNDHOG system." *Proceedings of the Second PASCAL Challenges Workshop*. 2006.
9. McCallum, A., Corrada-Emmanuel, A.,Wang, X.. The Author-Recipient-Topic Model for Topic and Role Discovery in Social Networks, 2004.
10. framenet.icsi.berkeley.edu
11. www.liwc.net
12. mallet.cs.umass.edu
13. nlp.stanford.edu/software/sutime.shtml
14. www.ark.cs.cmu.edu/SEMAFOR/
15. www.php.net/manual/en/function.metaphone.php

SESSION

THE INTERNATIONAL WORKSHOP ON INTELLIGENT LINGUISTIC TECHNOLOGIES 2014, ILINTEC'14

Chair(s)

Dr. Elena Kozerenko
Russian Academy of Sciences
Russia

Semantic identification of text in the class of tasks of information retrieval

O.L. Golitsina, N.V. Maksimov, O.V. Okropishina, and A.E. Okropishin
System Analysis Department, National Research Nuclear University MEPhI (Moscow Engineering Physics Institute), Moscow, Russia

Abstract - *As part of the ontological approach to the identification of the information methods for constructing of document search images are proposed. These images includes two types of components: a list of terms and network structures for identification of semantic content of the scientific text within the objectives of full-text information retrieval. Procedures for the allocation of the terms and relations between them are based on the formalization of semantic and structural characteristics of the text. Proposed methods for constructing search images are used not only for evaluation of potential usefulness of the found document, but also to reformulate queries by feedback on relevance, adaptation and enrichment of user queries, as well as carrying out multiaspect search.*

Keywords: semantic information retrieval; identification of content; ontologies; semantic search image; NLP.

1 Introduction

Man uses information retrieval and automated information systems for a relatively long time. However, almost all industrial systems, including those that are referred to the intellectual, implements the search of text (data) rather than information retrieval (IR) – a new meaning that will change the state of the subject in a problem situation. Interaction in such systems is unidirectional: the system reflects user's problematic situation on its "knowledge".

In human primary activity knowledge, information and data exists naturally – so that have the appropriate forms and related stable associations between them and existing set of methods and means for its obtaining. In the informational activity these objects (or more precisely, their images) in modern conditions of informatization will be presented in a single having semiotic nature numerical form and retain only those connections that will be considered as important during the formation of native image. That is, we can say that the unification of the presentation of any objects (things, processes, knowledge, symbols, etc.) in computing environment leads to *impersonality*: the loss of diversity of real connections transforms knowledge into separate messages.

Another feature is the fact that the transformation of the information is sequential filtering of information: reducing the freedom of expression degree (and thus subsequent perception) by fixing the semantic content, i.e. the imposition of part of meaning to the metainformational component. For example, a scientific report implies that domain of interest is known to the receiver (i.e., it is fixed in the mind of the receiver, and only enough to call it); document (published report) suggest fixing the representation way of gist of object through the allocation and linguistic binding of concepts, whose meanings are defined and fixed in a corresponding, existing quite independently, conceptual and terminological system (glossaries, ontologies); retrieval image of document suggests fixing of the semantic context of the terms, whose features of use for a given domain of interest are defined in the thesaurus. I.e. using of abstractions of different orders and using of objects of another level (or environment), which are considered as implicitly or explicitly specified context determines the possibility to simplify the description of both queries and documents, reducing it to a few concepts.

This, in particular, explains the fact that the retrieval dialog request typically includes 3-5 concepts that obviously cannot provide a construction of the IR results that corresponds the real information needs perfectly, predetermining the need of subsequent user's evaluation of the real value of each retrieved document. Until recently it was quite rational: the required completeness of selection was achieved with an acceptable amount of viewing. The situation changes significantly when the IR is conducted in the global polythematic data storages or in specialized historical databases containing tens and hundreds millions of documents: volumes for viewing are significantly surpassing the opportunities of a human. The use of "smart" information retrieval means, more likely, creates the appearance of effectiveness, that is highlighting the proverbial (actually, "burying" the new deeper) and often distracts from the subject of search.

Improving the selection efficiency involves the use in retrieval images more precise and specific concepts, as well as semantic relationships. Obviously symbolic constructions in these cases will be more cumbersome (wordy), syntactically and semantically more complex and therefore perhaps of little use in the traditional query-answer mode (when formulating a query expression is hard enough not only to remember the exact "construction" of the term, but even correctly enter it in the query string of information retrieval system). However, due to the high specificity, they will certainly be useful in

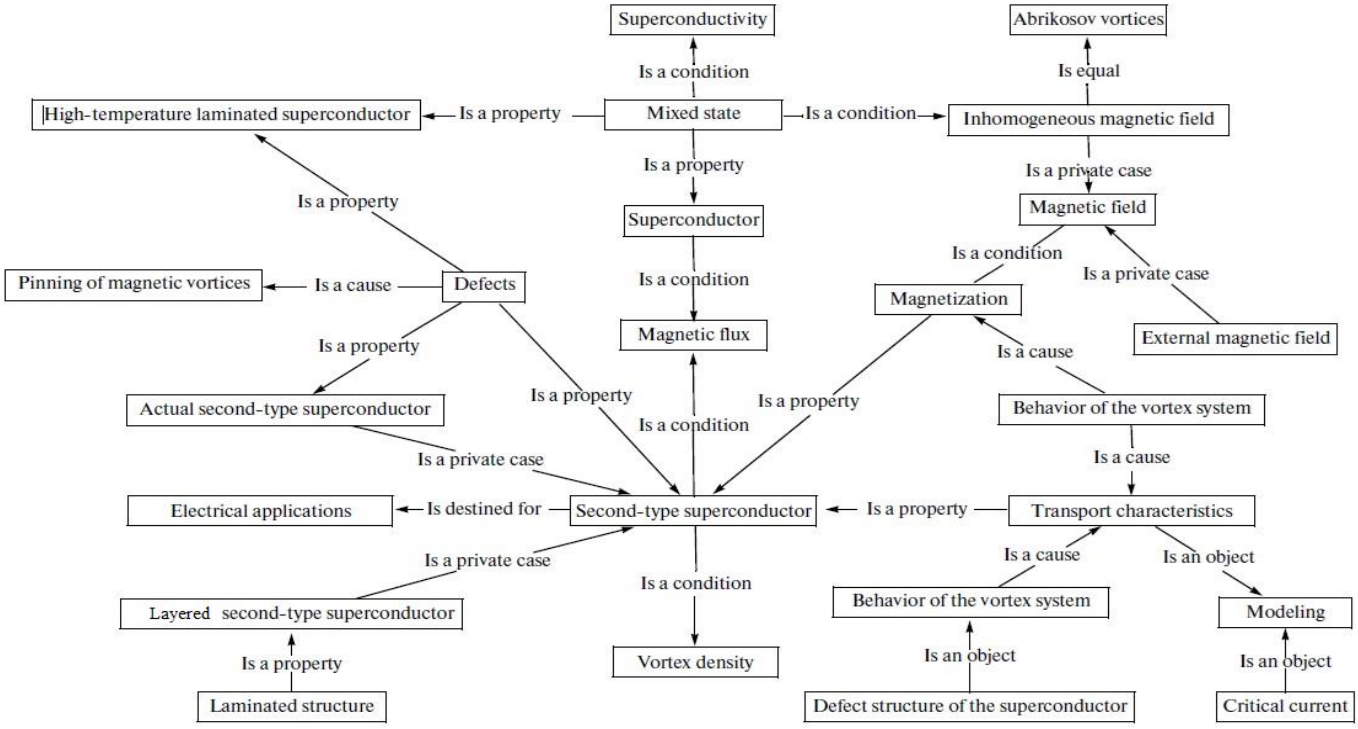


Fig. 1. A fragment of the multigraph of the ontology functional system

evaluating the potential usefulness of a retrieved document based on its semantic image, and in the procedures of IR using the technology of reformulation of query by feedback on relevance.

To construct well-structured document images – basic knowledge carriers providing multidimensional IR, an ontological approach to analysis of content based on the formalization of semantic and structural characteristics of the text is proposed.

2 Ontological approach to the information identification

The definition of an ontology as a document image, reflecting essential concepts and semantic links and a set of operations within a task of document identification have been proposed by the authors in [1]. Consider its briefly.

An ontology of a domain of interest is formally defined as system $O =< S_f, S_c, S_t, \equiv >$, where:

S_f - is a functional system (the "workspace interface" of the ontology in the activity of the subject);

S_c - is a conceptual system (the logical and semantic base of an ontology);

S_t - is a term system (signs that are used to fix ontology on the carrier);

$\equiv$ - is the operation of comparing the elements in different systems at the level of signs, which provides their equality in the functional, conceptual, and term systems.

The functional system is defined as, $S_f =< M_f, A_f, R_f, Z_f >$, where M_f - is a set of symbol descriptions of the domains of interest objects; A_f - is a set of characteristic attributes; R_f – is a set of functional relations; Z_f – is the composition law, in accordance with which a specific system base was selected $\{M_f, A_f, R_f\}$.

The system of concepts is defined as, $S_c =< M_c, A_c, R_c, Z_c >$, where M_c - is a set of symbol descriptions of the domains of interest concepts; A_c - is a set of characteristic attributes of symbol descriptions; R_c – are genus-species and associative relations; Z_c – is the composition law according to which the specific system base $\{M_c, A_c, R_c\}$ was selected.

The terminology system: $S_t =< M_t, A_t, R_t, Z_t >$, где M_t - is a set of the domains of interest terms; A_t - is a set of characteristic attributes of terms; R_t – is the relationships of equivalence and inclusion; Z_t – is the composition law according to which the specific system base $\{M_t, A_t, R_t\}$ was selected.

From the point of view of structure (for the implementation of operations on ontologies) a functional system can be a marked, weighted, and directed multigraph and a conceptual system with a marked, weighted, and directed graph. A term system is described by an *n*-linked graph, where each connection component is a complete graph (equivalency), a tree (inclusion), or the result of combining complete graphs and trees (if there are common vertices).

An ontology that is defined this way also has the properties of a system (in terms of general systems theory), i.e., one can construct, for a particular domain of interest (including its single elements), as many ontologies as the number of the aspects of the presentation of the domain of interest $O^i =< S^i_f, S^i_c, S^i_t, \equiv >$.

Based on general systems theory [2] and semiotic approach (Frege's triangle) are proposed to systematically

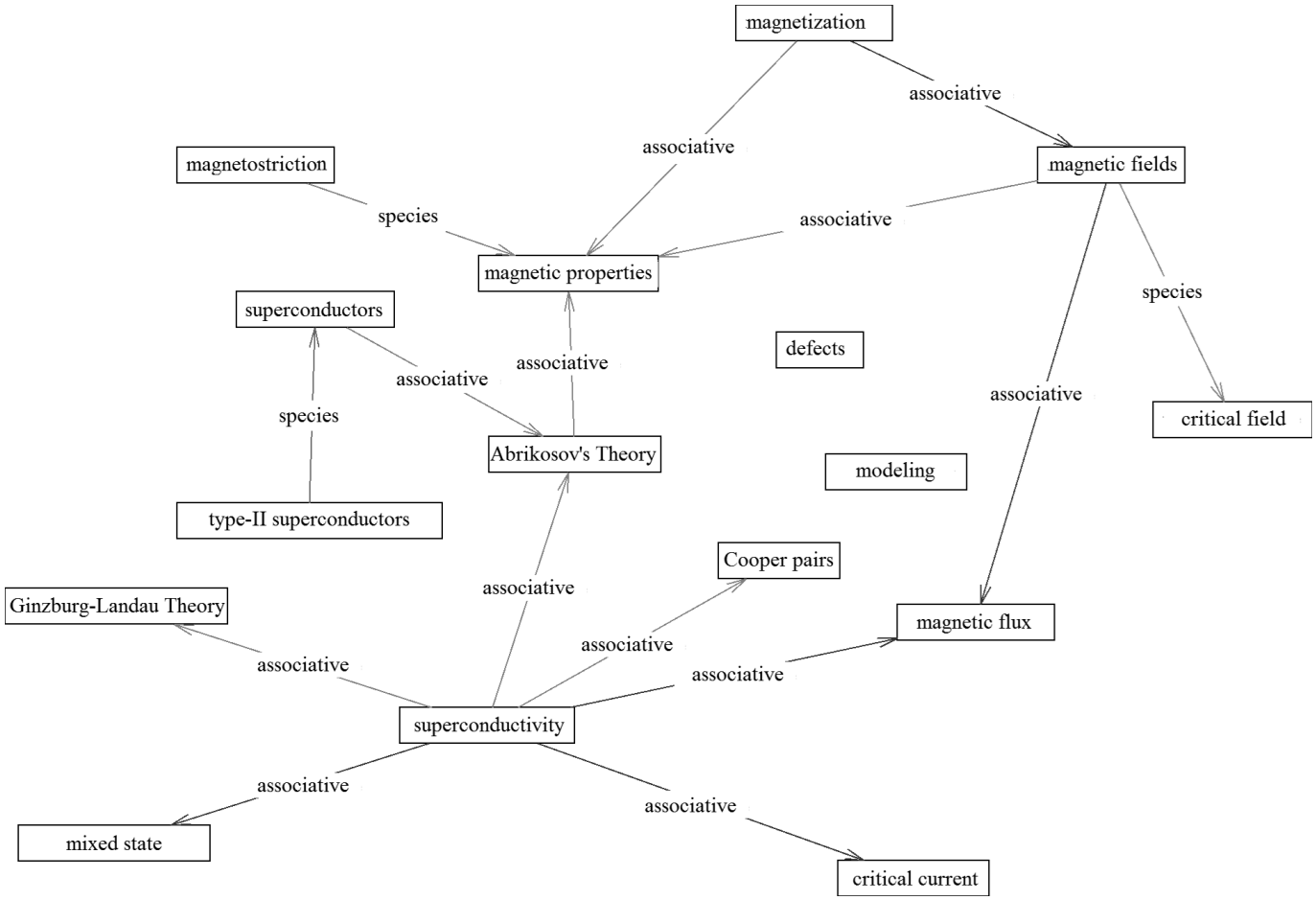


Fig. 2. Fragment of the graph of the ontology conceptual system

consider the semantic content of the document (description, in particular, scientific research) on three levels causing each other: domain of interest, conceptual and symbolic. At each of these levels, the image of the document can be represented as a system - a combination of elements that are defined on the set of characteristic features and associated relationships (respectively, *functional system - objects and relations of subject-level*, *conceptual system - concepts and conceptual relationships*, *symbolic system - lexemes and linguistic relations*) within the law of the composition (for example, within the selected scientific paradigm or aspect of representation)[1].

In the tasks of information communications, in particular, document retrieval the essential feature is that image that reflects object properties and behavior (relations) must be represented in sign system. Thesauri are traditionally used for the formation of IR images - network structures, which nodes are signs (names of concepts), and the arc is a fixed immanent connection (the main type of connections are generic, which meet the basic method of concepts defining - by kind and species differences). Functional system of ontology, which is usually identified with the representation of domain of interest being networked structure too, fixes its own (situational) relations of domain of interest. Thus, a thesaurus is a model (represented as a sign system) of instrument of cognition in a particular field of knowledge, and ontology - a model of a specific object, which represents a *situational knowledge about the domain of interest*, that reflects the variety of main entities and typical relations between them usually at the level of individual solution or compact domain of interest.

Fig. 1 shows a fragment of semantic image of document - multigraph of functional system of ontology built on the text of the dissertation [7].

Accordingly, subgraph of thesaurus shown in Fig. 2 which nodes correspond to the set of terms of a functional system of document forms its conceptual system.

3 Methods of semantic image formation

Formation of semantic image basically reduces to allocation of semantic relationships (functional relationships) and terms (individual words and phrases), representing the basic concepts.

[1] I.e. under another law of the composition may be a different view of the same domain of interest in which there are as coincident with the first presentation, and distinct elements and relationships.

Let R - functional relationship, A and B - terms. Then $T =< A, R, B >$ is called a semantic triple.

Document semantic image includes components of two types - a list of terms (keywords) and a list of semantic triples, representing functional system.

Terms are extracted via statistical and linguistic text processing procedures. Terms corresponding to sectoral or specialized dictionaries can be specified depending on requirements to accuracy of identifying of terms - or in accordance with the propositional structure of sentences, or only on the basis of grammatical properties of words[2]. The greatest interest in both cases represents the concepts that are expressed by noun phrases [3].

Propositional sentence structure represents as a semantic network, whose vertices are containing lexical units (symbols-terms), expressing some of the concepts and the arcs correspond to linguistic relations between them[3] [4].

Each symbol is mapped to concept which is denoted by this symbol (the concept of a concept name [5] or more concepts in the case of homonyms). Each concept corresponds to the class of objects of domain of interest, the boundaries of which are wider than the boundaries described by domain of interest.

Linguistic relations between words are clustered in accordance to proximity of values of these relations and then clusters that belongs to the group of relations of constriction / expansion of concepts volume are considered.

Fig. 3 shows the semantic network of propositional structure of sentence "The calculation of the influence of defects concentration on the dependence of critical current is conducted".

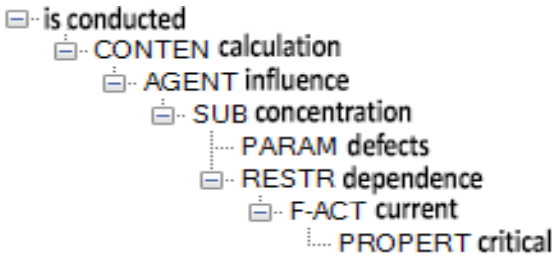


Fig. 3. An example of semantic network of propositional sentence structure

The terms "defect concentration" and "critical current" can be derived from combining the words related with linguistic relations from group of relations of narrowing the scope of the concepts: PARAM and PROPERT.

In the method of clarification of terms, excluding propositional structure of sentence for the terms of the document included in the thesaurus, built noun phrases in which the dominant word is combined with all its dependent words on base of its lexical and grammatical properties. The resulting phrases are the names of concepts and designate object classes of domain of interest reflected in the analyzed text.

For example, for the ontology shown in Fig. 1, the terms "high-temperature layered superconductor" and "defect structure of the superconductor" obtained by refinement of thesaurus concepts "superconductors" and "structure" and the terms "layered second type superconductor" and "actual second type superconductor" - as result of refinement of the thesaurus term "superconductors the second kind".

It should be noted that the Incorporation of the propositional structure of sentences in terms formation process allows to reduce chance of appearing of incorrect constructions.

According to [4, 6] actions and their participants in the text are indicated by lexical units constituting the propositional structure of sentences. Semantic predicate has the property to subdue some semantic actants of situation expressed (verbalized) by the predicate. The binary relations between the predicate and each of its actant (participant in situation) are linguistic relations.

On the other hand, the sentence of natural language as a constructive sign correlates to the situation described by it as its denotate (examples of classification - UDC, GRNTI etc.), i.e. represents fragment of reality not just as a collection of individual objects of domain of interest, but as the system. Thus, objects of domain of interest may qualify as participants of situation (interaction elements), related by some (situational) relation and performs a specific function within an interaction.

The transition from the linguistic relationship to functional occurs based on fuzzy binary matching function:

$f : L \mapsto P$, where L - the set of linguistic relations, P - set of functional relations.

Depending on requirements to relations establishing accuracy, binding of terms using functional relations is held either based on propositional structure of sentences or using templates.

In case of determining the functional relationships taking into account propositional structure of sentence the function *f* is a set of rules of correspondence between linguistic relations and functional relations.

For example, the functional relation "Is a condition" (in the semantic network shown in Fig. 3) for the linguistic relationship RESTR is determined based on the following rules:

Let X, Y, Z - words. If (R(Z,Y) = RESTR) and (R(Y,X) = F-ACT) then (Z ' Is *a condition*' X), that allows you to create a semantic triple

T=<'defect concentration', 'Is a condition', 'critical current'>

If the function *f* can be represented by a set of patterns, then the definition of the functional relations occurs by comparing the elements of the text with templates of signal structures indicating the presence of a functional relation (of

[2] In this paper, all statements and examples made for the Russian language.

[3] Some examples of linguistic relations: PROPERT(*magnetic*, field); PARAM(*temperature*, melting); BELNG(*vortex*, pinning).

particular type) between a pair of terms adjacent to signal structure. The template defines the composition of the signal structure (terms, words, numbers, punctuation), the relative positions of the signal structure elements in the text, their grammatical features.

Consider the sentence:

"As a result, it is the behavior of the vortex system and its interaction with the defect structure of the superconductor that has an impact on the transport characteristics of high-temperature superconductors".

The way of refinement of thesaurus terms, excluding propositional sentence structure gives the following noun phrases:

- "behavior of the vortex system" - based on the words "behavior" and "system" belonging to the thesaurus concepts;
- "interaction of the vortex system" - as a result of disclosure of anaphoric reference, expressed by personal pronoun "her" and refinement of the concept "vortex system";
- "defect structure of superconductor" - based on the words "superconductors" and "structure" belonging to the thesaurus concepts;
- "transport characteristics of high-temperature superconductors" - based on the words "transport", "characteristics", "high-temperature" and "superconductors" belonging to the thesaurus concepts.

Template of the signal structure "effect on" having the form

RELATION[N:Is a cause]('effect' 'on'); SOURCE [C:sub]; TARGET,

with a parameter that specifies the morphological feature of the term - subjective case - allows you to select the functional relationship "Is a cause".

On the basis of the pattern of the signal structure "with" having the form

RELATION[N: Is an object,DIR:reverse,THRESOLD: 2] ('with'); SOURCE;TARGET,

wherein the distance between the signal structure and terms (source and target future functional relation) is set - not more than two words, the functional relation "Is an object" is determined.

After identifying the functional relationships can be built following semantic triples:

T1 = <'behavior of the vortex system', 'Is a cause', 'transport characteristics'>

T2 = <'interaction of the vortex system', 'Is a cause', 'transport characteristics'>

T3 = <'defect structure of the superconductor', 'Is a subject', 'the interaction of the vortex system'>

As a result of post-processing of resulting list of semantic triples (checking the presence of connections between the parts built "term-concepts" on the basis of representation of these parts as separate "term-concepts"), the term "transport characteristics of high-temperature superconductors" can be converted into following semantic triple:

T4 = <'transport characteristics', 'Is a property', 'high temperature superconductors'>

4 Using semantic images in information retrieval (as a conclusion)

Physically in IR Indexes components of semantic image of document of both structural types - terms and semantic triples - are equally represented by separate terms (including the names of relations) that actually allows you to use the classic IR engines, based on binary logic. Joint use of methods to extraction of semantic elements from text, that are based on full complementary approaches (lexical, statistical, linguistic) allows you to create a multiple images for a single document (groups of terms) differently identifying the content: domain of interest special common terms (eg thesaurus), specific terms that are constructed by refining content, etc. While keeping the statistics of occurrence of the term inside / outside the document allows to identify features of novelty of the corresponding concept. In particular, for the document [7] allocated next groups:

- *Concepts which reflect the novelty*: "the method of calculating the magnetic reversal", "annihilation wave", "calculation of the hysteresis loss".
- *Basic concepts*: "layered HTSC," "critical current", "vortex filament", "volt-ampere characteristic," "pinning center", "columnar defect," "pinning potential".
- *General concepts*: "magnetic field", "Monte Carlo method", "potential well", "energy loss", "electromagnetic interference", "phase transition".

The experiments[4] showed that the use of semantic images significantly improves IR quality by increasing the recall and precision, while reducing the number of search iterations. This is achieved, in particular, due to the fact that at the initial stage (query forming) semantically "simplified" terms are used. This provides a relatively representative, although not exact, issuing documents, which true relevance (or rather, the potential utility) due to more complete and accurate IR image of document will be evaluated by the user more adequately. In case of continuing of IR using the technology of reformulating search query by feedback by relevance, system will use already highly specific terms that will determine the best selectivity of the selection process.

Application of ontological representation of conceptual system of domain of interest (usually, thesaurus) allows the use of projection and scaling operations. This creates the conditions for the introduction of technologies of "problem-oriented" enrichment of query by thesaurus vocabulary, where

[4] The experiments were performed in information retrieval system xIRBIS [8] on an array comprising 6000 full texts of dissertations.

the descriptors relations of which will fit most domain of interest ontology will be used for the expansion of the query expression.

In general, this approach to information retrieval tasks in processes of dynamic reformulation and correlation IR semantic images of queries and documents will allow not only take into account the diversity of points of view, but also to identify potential areas for the development of the domain of interest.

5 References

[1] Golitsyna, O.L., Maksimov, N.V., Okropishina, O.V., and Strogonov, V.I., The ontological approach to the identification of information in tasks of document retrieval, *Autom. Docum. Math. Ling.*, 2012, vol. 46, pp.125–132.

[2] Urmantsev, Yu.A., General Theory of Systems: State. Applications and Development Perspectives, in Sb. sistema, simmetriya, garmoniya (Coll. Papers 'System, Symmetry, Harmony'), Moscow: Mysl, 1988, pp. 38–124.

[3] Russian grammar / Edited by N.Yu. Shvedova and others. – M.: Nauka, 1980.

[4] Leontyeva Nina N. ROSS: Semantic Dictionary for Text Understanding and Summarization // META. - 1995.-40, No 1.- p.178-184.

[5] Meighen S.V., Schroeder J.A. Methodological aspects of the classification theory // Problems of Philosophy. - 1976. - № 12. - P.67-79.

[6] Popov E.V. Communication with the computer in natural language / E.V. Popov. – M.: Nauka, 1982.

[7] Odintsov D.S. Modeling the transport characteristics of high-temperature superconductors. Dissertation for the degree of Cand. Sci. Sciences, Moscow Engineering Physics Institute, 2008.

[8] Maksimov, N.V., Golitsyna, O.L., Okropishin, N.V., and Okropishina, O.V., Documental information analytical Subsystem. Svidetel'stvo o gosudarstvennoi registratsii programmy dlya EVM, No. 2011611694, 2011.

Building unified knowledge resources for multiple applications

Max Petrenko
NTENT
mpetrenko@ntent.com

Gavin Matthews
NTENT
gmatthews@ntent.com

Erin MacMurray van Liemt
NTENT
evanliemt@ntent.com

Abstract – *The paper discusses the techniques of maintaining large-scale knowledge resources supporting multiple applications based on different types of data. Issues of granularity, interoperability, and scalability that arise from the need to reconcile functional differences of multiple applications are discussed, based on specific examples.*

Keywords: information retrieval, ontology engineering, disambiguation, search engines

1 Introduction

When building large-scale and domain-independent knowledge resources that are accessed by a number of user-facing and backend applications, each of which carries specific, output-driven, requirements as to the nature and type of representations of knowledge, the considerations of granularity, domain specificity and interoperability of the underlying knowledge resources come to the fore. The difficulty of selecting the right grain size, the breadth and width of ontological and lexical coverage is exacerbated by the fact that the overall number and nature of the applications that would have to be supported by the knowledge resources might not be available to the knowledge engineering team at early stages of research and development. In many such cases, knowledge engineering decisions are made semi-intuitively: an optimal (often for unspecified reasons) level of grain size and domain coverage is set with the expectation that, quite circularly, because those levels are optimal, the potential applications would align their functionality accordingly.

Significant progress has been made in the areas of ontology engineering, maintenance and evaluation. Current literature offers extensive theoretical and methodological guidance on building scalable ontological resources for single or multiple applications (see [11] for an example of ontology engineering methodology). In-depth analysis of existing ontology engineering environments is also available (e.g. [7]). Additionally, methods of cost/benefit analysis of the overall ontology life cycle have been developed, which rely on parameters such as domain complexity, reusability, personnel costs and project costs (see [10] for details and an example of a well-developed cost model of ontology engineering).

Even a cursory look at the existing literature indicates that the field of ontology engineering has reached considerable maturity. The paper aims to address an issue, which, while being undoubtedly widespread across ontology engineering enterprises, has been insufficiently addressed in the literature: how does one ensure the unity of an already built and actively deployed large-scale ontology? In the above formulation, being large-scale and already deployed by specific applications are critical features that render many ontology engineering guidelines inapplicable: in the time-sensitive, goal- and product-oriented environment of business projects, rebuilding the resources from scratch is simply not an option. Ensuring the unity of the knowledge resources and the continuity of the pre-existing functionality are serious considerations: in light of novel, unanticipated, applications the issues of granularity, domain specificity and interoperability must be addressed with minimal disruption to the already created architecture.

As noted before, insufficient guidance is offered in the literature with respect to the issue above. In their discussion of ontology specifications, [11] recognize the risks of discovering "gaps" in the original specifications during later stages of development, but do not go beyond acknowledging the cyclic nature of the ontology engineering process. By the same token, the argument that multiple ontologies should be built for multiple applications because "no ontology fits all purposes" (as presented in [1]) is often inapplicable for mature enterprises which have spent years developing a single resource and integrating it with income-

earning technologies. Methodologies must be developed that allow for versatile, risk-free adjustments to the already existing and actively used large-scale ontology in order to support multiple applications.

The paper presents a case study of how the issue of building unified knowledge resources for multiple applications was addressed based on the proprietary large-scale knowledge resources built and maintained at NTENT (formerly known as Vertical Search Works). While the paper does not attempt to offer an all-purpose guidance on the identified issue, it nevertheless hopes to provide a comprehensive illustration of choices made to insure the integrity of knowledge resources in order to support three different applications with different functionalities.

The structure of the paper is as follows. First, a brief overview of the knowledge resources maintained at NTENT will be provided. Each subsequent section will focus on three user-facing applications – a search engine, an ad matching platform, and a video recommendation system. For each application, the paper will outline its functionality, discuss specific requirements it imposes on the precision, recall, the grain size, domain coverage and the nature of ontological representations, and discuss various techniques used at NTENT in order to accommodate the requirements while maintaining the integrity of the knowledge resources.

2 Knowledge resources at NTENT

The theoretical, methodological underpinnings of knowledge resources maintained at NTENT were outlined in [6]. The knowledge resources consist of a language-independent ontology and multiple layers of language-specific sets of expressions. Supplementary components include verticals (i.e. domain-specific clusters of ontological concepts), and facets (i.e. topic-specific classifications of documents). A standalone supplementary component of the knowledge resources is a hierarchy of tags, i.e. conceptual labels that support article-advert matching based on semantic proximity (see Section 4). In addition to the top-level branching into entities, processes, relations and properties, as well as a strong distinction between classes and individuals, NTENT ontology supports collections of various order, depending on the number of instantiation steps from classes to individuals (for more details on higher-order collections in ontology engineering and Cyc, see [5]). In particular, second-order classes (i.e. classes whose instances are classes, e.g. "football player by position") are commonly used as an efficient way of creating collections of entities (e.g. "quarterback", "running back") grouped by their attribute values (e.g. "football position").

The difference between classes and individuals is expressed through the membership in two disjoint higher-order classes. The difference between individual instances and collection instances is expressed through external marking through quoted collections and subsumption link sequences. Chains of subsumption links also control the determination of "ancestor" categories used in disambiguation and in search.

The set of ontological relations expresses meronymy, locality, group membership, event roles, and additional case-specific entity-entity relations like authorship of conceptual works or roles performed in movies. The behavior of relations is controlled through the constraints on their arguments. Argument constraints are also used in detecting disjointness violations, as each individual assertion of a lateral link is defined to hold between inherently disjoint entities.

Language-specific expressions are annotated through a set of attributes, which, based on the appropriate part of speech tag, control the morphology and syntax of each expression in standard context. Expression attributes also drive tokenization, stemming and the behavior of the lexical disambiguation module.

As a core component of NTENT's semantic technology, the knowledge resources support and drive the indexer. The indexer relies on expression attributes and ontological relations to "latch" input tokens to underlying concepts by (1) identifying acceptable token matches, and (2) determining the "latches" based on the ontological proximity between concepts expressed in the input. The determination of ontological proximity is largely performed within the spreading activation model (see more details in [2], [9], also see [3] for a similar implementation).

The ontology is used by the indexer in a number of different ways. First, lexical entries allow expressions in documents to be associated with ontological concepts, a preliminary step to disambiguation. Second, the labelled digraph of the ontology supports spreading activation. Third, the

subsumption hierarchy is used to generalize the topics of the document to more general topics (hypernyms), a form of relevance blurring to increase recall. The lexicon and ontology subsumption are also used to support document classification rules (known as facets), detecting mentions of places, people, etc. Fourth, ontological subsumption and specialized links are used to derive overall topic areas (known as verticals) from the salient concepts of the document. This is used to support vertical-specific interpretation of search queries, among other applications. Fifth, the ontological relationship of concepts within a document is also used to measure the coherence and depth of a document's subject matter, which affects how the document is weighted in ranking.

3 Using NTENT's knowledge resources to support the web search engine

NTENT maintains an in-house semantic search engine called Excalibur. Traditional search engines take natural language words from the user's query and return documents that contain those words. Excalibur's behavior differs substantially. This works well in many cases, but can suffer from a number of issues:

- The words in a query may not represent isolated individual concepts, but may bind tightly into multi-word expressions. Documents that merely contain the individual words may not be relevant, and hence reduce precision.
- A given word can have many possible interpretations, only one of which was intended by the user. Documents using the word in other senses may not be relevant, and hence reduce precision.
- Documents may be relevant to a user's information need without necessarily using the exact words used by the user. Typically this involves synonyms and hyponyms. Failure to bring back these documents reduces recall.

In contrast, Excalibur is a semantic search engine. Documents are analyzed (see a section on indexing below) to extract the concepts that forms the topics of the discourse, and these concepts are provided as part of the searchable index. When a user enters a natural language query, it is interpreted into concepts, and the search engine returns documents that are relevant to those concepts.

User queries are too short to extract concepts in the same way as for longer documents. They are commonly not formed from coherent language constructs and words are often omitted from terms. A specific search interface may be configured to be biased towards some specific domain of discourse such as food, health, finance; this is known as the *vertical*. Excalibur's patented ([13]) approach to extracting concepts from user queries uses the ontology in two different ways, one direct and the other indirect:

- Under some circumstances, sequences in the user query may be converted into concepts on the basis of direct lexicon matches. Unless the user employs meta-query syntax, this is usually constrained to match the entire query against the lexicon for concepts specific to the vertical.
- If direct matching does not eliminate all natural language terms from the query, then the natural language (or possibly hybrid) query is executed against a vertical-specific corpus. This query is performed in much the same way as for a traditional search engine, but it is not intended to generate results to be shown to the user. Instead, the documents are used as an expanded context for the user's terms to select the best semantic interpretation of the query. This interpretation may succeed in matching a fragmentary expression of a concept in a way that could not be done simply from a lexicon match. (e.g. "harry book phoenix")
- The resulting interpretation is then executed as a search to generate results for the user. The interpretation of the user's query may be entirely conceptual, but Excalibur may choose to leave natural language terms alone if the conceptual interpretation is unclear, or if the concept is unattested. Unlike the first search, this final search uses the full corpus biased towards the vertical, to ensure recall.

The Excalibur search engine tends to favor recall over precision for two reasons. The first reason has to do with the granularity of indexing: because the primary goal of the indexer is the identification and ranking of document topics and not the construction of sentence representations, the ontological differences between descendants of disjoint top-level classes are

not useful for many applications as long as both senses pertain to the same document topic. For example, both documents discussing the service of installing a plumbing system (a subcollection of the top-level "event" class) and documents discussing plumbing systems (a subcollection of the top-level "entity" class) would be appropriate results for a query "plumbing costs", despite the fact that both are descendants of disjoint ontological classes and are expressed through very different grammatical categories, "plumb" as a verb and "plumbing" as a noun in English.

The second reason has to do with the general policy on document retrieval, which stipulates that result documents address topics mentioned in the query in more specific terms. For example, a document that mentions one or more dog breeds could be relevant to a query only mentioning "dog".

As described above, a benefit of semantic search is an improved precision resulting from the disambiguation of senses of homonymous expressions. Unfortunately, there are circumstances where concepts are not only homonymous, but are also strongly related and commonly collocated in the corpus. For example, "plumbing a washing machine" uses "plumbing" as a verb referring to an event, whereas "repairing the plumbing" uses "plumbing" as a noun referring to an artefact. Common ontological considerations require events and artefacts to be disjoint, but it could be argued that both examples have both concepts as topics; a search would likely not benefit from forcing the document and the query to settle on one of the two interpretations. This causes a tension between ontological coherence and adequate recall in the search application.

To mitigate this issue, the ontology has a special link labeled CPROX ("conceptually proximate"), which is asserted between concepts that have homonymous expressions in a specific language but likely co-topics. The effect is to suspend the usual disambiguation "competition" between "latch" candidates and instead latch both concepts as topics wherever they both match the text. Another mitigation strategy used is to allow co-latching between unrelated concepts where there is insufficient evidence to disambiguate between them. In this case, one of the concepts will still be the "primary latch", and will be accorded greater weight in ranking.

In both these cases, Excalibur errs on the side of recall during indexing, making the facially incorrect claim that one natural language expression refers to two distinct concepts. This allows a search application to select a single interpretation of the user's query without losing documents, while allowing the ontology to avoid disjointness violations.

A brief discussion is perhaps in order of why disjointness violations should be avoided and why avoiding them by collapsing the offending classes into a single non-disjoint node is a poor choice. Consider the example of defining the verb "plumb" and the noun "plumbing" as mapped to disjoint descendants of "event" and "entity". First, conflating the event sense of "plumbing" and the physical entity sense of "plumbing" into a single node, which, is say, a subcollection of just "domestic appliance" makes it impossible for the system to establish the distinct senses of "plumbing" in cases where the two senses are not just mentioned, but explicitly juxtaposed (e.g. "the plumbing in the old house started to clank; it was so old that plumbing just the first floor took two weeks"). Each such sense entails a specific syntactic behavior and semantic properties that could be useful for a variety of applications. Applications that require that level of lexical precision will likely suffer from this crude treatment of polysemy, Second, one of the fundamental purposes of creating top-level disjoint classes in an ontology is to help represent and justify polysemy. Representing polysemy through mappings to disjoint top-level classes serves as a key criterion for splitting a lexical sense into several senses. The failure to properly utilize disjoint top-level ontological classes in lexical semantics results in unjustified sense conflation or sense proliferation, both of which can often be observed in human-readable dictionaries (see [8] for a discussion of inconsistencies in sense determination in human-readable dictionaries).

4 Using NTENT's knowledge resources to support the ad matching platform

The advert matching application has the goal of automatically assigning adverts to articles that are viewed by a user. Based on what is known about the user, the application must select the most relevant adverts to show on an article page. A significant part of what is known is that the user has chosen to look at the article in question, which is where the semantic

processing and knowledge resources are involved. In a similar process to that of Excalibur's semantic search engine, the semantic indexer processes both documents and adverts to extract the concepts that are the most salient. Articles are treated as queries for which the search corpus is the total of available adverts and their corresponding landing pages, and other meta-information. There a few key differences, however, that distinguish the indexing process for advert matching from the process for web search:

- Indexing of Web documents traditionally uses a pipeline architecture where throughput is more important than latency. In advert matching, the latency is much more important and an on-demand indexing architecture is therefore required.
- The natural language input for adverts not only includes the conventional web page pointed to ("landing page"), but also the text displayed in the advert ("hook") and other hidden information supplied by the ad team.
- On the modern media-rich web, adverts often end up with a small quantity of natural language text, which may be ungrammatical and employ deliberately ambiguous or misleading language. Many of the phrases commonly used in adverts (e.g. "free delivery", "in your area") supply no useful information regarding topic.

The most salient concepts are then mapped into a vertical-specific tagging hierarchy in order to derive a semantic signature for each document. The article's signature can then be compared to those of the adverts to find the closest matches.

Adverts are organized into certain advertising verticals. Each industry or trade that could be of interest to a user is captured in a vertical associated with a hierarchy of tags. This tag hierarchy is informed by business needs to reflect the range of relevant products and services. The organization is partly subsumptive and partly topical. More specifically, the tagging hierarchy maps ontological nodes that may be emitted by the indexer (including concepts, facets, and ontological verticals) to tags. When an ontological concept is found on an article or advert, it is assigned a tag that corresponds to the category where that concept is defined in the hierarchy. By extension, these categories indicate the over-arching vertical to which an article or advert belong. This layer provides a key insulation between the general-purpose ontology and the application- and vertical-specific task of advert matching.

Section 3 described how the indexer stores hypernyms of direct latches. These also assist contextual advert matching. For example, in figure 1 below, a web document for a recipe for “wild mushroom and vegetable gnocchi” is attributed an ontological concept “gnocchi”. However, if an insufficient number of adverts are returned for this concept, the system will use the knowledge resource to identify the parent of “gnocchi” as the higher level concept “pasta”. This will then allow the advert matching application to attribute a tag of “pasta” to the article and place adverts with this parent tag accordingly.

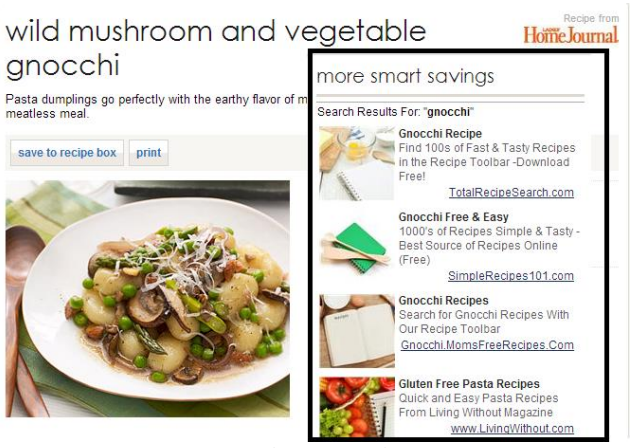


Figure 1. adverts displayed for webpage from recipe.com

All the extracted information (concepts and their mapping) creates a semantic signature for each article and advert. The Ad Matching engine uses this signature to rank each advert as relevant for a single article. Those adverts with a high rank are then displayed as being relevant, as shown in the example above. The algorithm used to compute the proximity between an article and adverts is roughly similar to cosine distance. It has been modified to use the hierarchy. In this way, similar to processing adverts for “gnocchi” shown above, an article associated with a high-level tag can be a weak match for adverts associated with child tags. For example, an article may be about cheese (or possibly about many types of cheese), but an advert may be about a specific type of cheese.

As explained above, the Ad Matching process integrates an external resource, the tagging hierarchy that uses an OPML format to map ontological nodes to verticals. The mapping is not stored in the unified knowledge resource. The knowledge resource is used

to communicate with the tagging hierarchy in order to indicate which ontological nodes are appropriate for which vertical-specific tags. In other words, the knowledge resource does not provide the verticals and their specific tags directly and requires this further step in the process.

There are several reasons for maintaining the tagging hierarchy as an external resource. First, non-ontological relationships cannot be expressed without distorting ontological purity. Second, the rank of a concept for a given vertical may require more targeted adjustment. Third, the Ad Matching platform could make use of external product taxonomies (such as Google Product Taxonomy[1]).

Non-ontological relationships are those associations which cannot be expressed by the controlled vocabulary of the knowledge resource. In most cases these relationships are associations important to a particular client or vertical with the goal of generating revenue. Articles about certain anchor products often provide an opportunity to sell accessories or associated services, even where they are not a direct topic of the article. For example, anyone interested in reading about iPhones might be a potential buyer of an iPhone case or app. Such relationships do not express a property or function inherent to an individual or collection in the ontology. A mapping that would indicate junk food adverts as appropriate on articles tagged for alcoholic beverages, such as beer, is not an association that is readily available in the knowledge resource and should be more appropriately defined in the tagging hierarchy.

Among the adjustments made to accommodate the preferences of the Ad Matching platform are ontological classes that help establish useful distinctions that might otherwise be encoded lexically. Examples include concepts such as "pet fish", which was created to help avoiding placing adverts with fishing gear and fish-based food on articles discussing pet fish. Concepts like "pet fish" are often discussed in the literature as examples of the conflation of ontological types and roles, or a violation of the rigidity principle (see [12] for a detailed discussion, and [4] for more examples of similar violations). The decision to create a questionable ontological collection was largely pragmatic: this, arguably pinpoint, grouping did not require large-scale refactoring of the knowledge resources or any component of the architecture, and provided a low-cost, risk-free and effective support for a tagging platform. As can be seen, any accommodating adjustment in the knowledge resources should be preceded by an analysis of benefits sought and costs incurred, as well as the analysis of available alternatives.

5 Using NTENT's knowledge resources to support the video recommendation system

NTENT's video recommendation system ("Vidady") is a user-facing application (currently implemented as a browser extension or a publisher widget) that offers relevant videos next to web documents. The documents are converted into search queries, which are in turn run against a corpus of annotated videos. The resulting pool of selected videos is then filtered by source, authority and additional business-related criteria (see Figure 2).

The video recommendation system has many requirements in common with web search, in terms of the way that documents are indexed. There are several respects in which its requirements differ.

Web search is primarily concerned with finding the top-ranking documents for some term or set of terms. Once an interpretation has been selected for a query, web search is not affected by the presence of other concepts in the documents. In contrast, video recommendation is concerned with the salient concepts in a single document and they can potentially interfere with each other.

A consequence of this is that the mitigations described in section 3 to co-latch distinct concepts (either because disambiguation is impossible or because they are strongly co-topical) may have the effect of introducing noise into the query used by a video recommendation system. Where there are multiple competing interpretations, only the one considered primary by the indexer should be used.

The policy of adding subsumption ancestors to documents to improve recall in search also affects video recommendation, but in ways that are both positive and negative. If a document mentions many breeds of dog (e.g. poodle, terrier) then it is useful to

[1] Google categorizes products in their search results based on various factors. Google Product Taxonomy is a tree of categories that helps a business generate values for the Google Product Category attribute, increasing the chances that a user's product will be included in relevant category and displayed by Google.

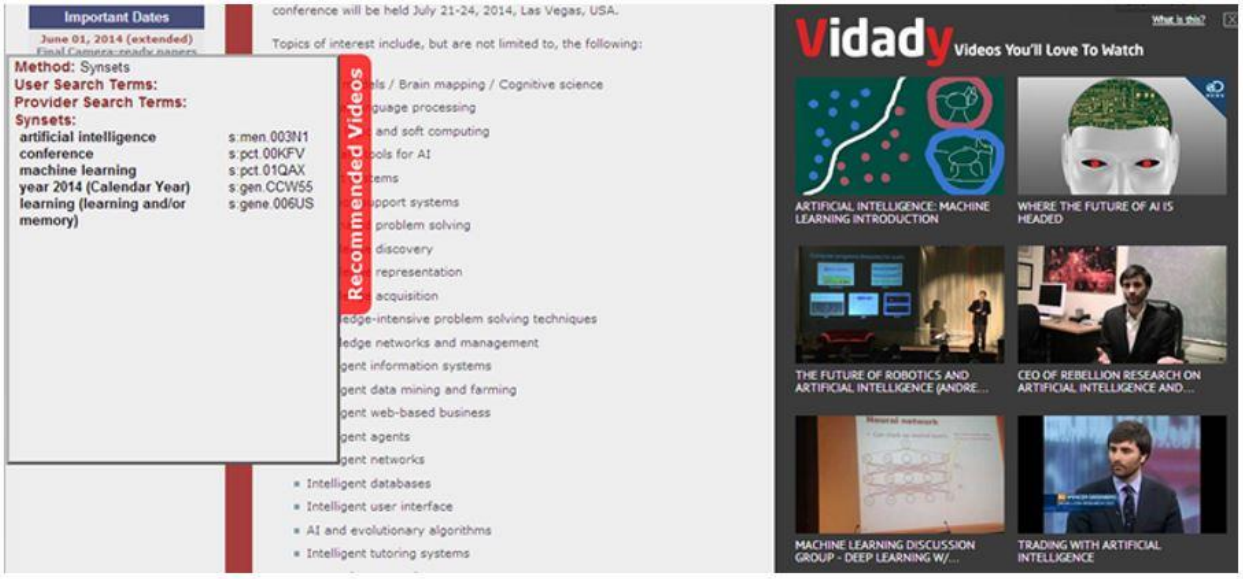


Figure 2: "Vidady" video recommendation system using the ICAI 2014 program page as an example. On the left can be seen an internal debugging widget that shows the salient concepts from this document, together with their primary expressions. The user-facing Vidady widget can be seen on the right (in expanded form), showing the results of searching for those salient concepts.

be able to find videos related to the concept of dogs in general rather than being forced to pick a specific breed. On the other hand, it is often not helpful to end up with both a parent and child (strictly concepts that lie on a common ancestor chain) in the results. For example, it is needless to include both "hurricane" and "Hurricane Sandy", which have both ontological and lexical commonality.

Another difference is that, while web search is concerned with recall, it is generally acceptable for it to neglect low-quality documents. If a document is not a plausible search result, then it does not matter if the disambiguator is unable to determine its concepts. For a document-focussed application like video recommendation (or contextual advert matching), the indexer must come up with usable results for every document presented. This affects the way that the ontology and disambiguation network is used, especially for short documents.

While search interfaces do need to be able to generate natural language text for concepts in order to be able to describe query interpretations and drilldowns, for video recommendation this is not only more important, but has different requirements. A search interface wants expressions that convey concepts clearly and, to the greatest extent possible, preclude ambiguity. Video recommendation requires expressions that correspond to the most likely usage in document text, even if that introduces great ambiguity. For example, a search interface might want to refer to "US Senator Johnny Reid Edwards" and "Affordable Care Act", whereas a video recommendation system would prefer "John Edwards" and "Obamacare".

To resolve this tension, a hybrid statistical and manual approach is employed. The indexer has access to statistics about how often specific synonyms of concepts are latched to those concepts, so it is able to identify the most popular expression in all cases. There are a few cases where this turns out to be inappropriate, so a mechanism is provided that allows an ontologist to override the statistical approach with selective manual annotation. A client of indexing services is therefore free to select between the two approaches to expression selection, known as the "primary expression" and "popular expression". In practice, they will often coincide.

6 Conclusion

This paper discussed some of the challenges of developing, maintaining, and exploiting a large-scale language-independent ontology that must meet the disparate requirements of multiple commercial user-facing applications. Specific cases were given of where the needs of different applications present a tension not only between applications, but also with fundamental principles of good ontology design. Key issues included considerations of disjointness, grain-size, and generation of natural language text. All three

applications were concerned with topic identification in text. Web-scale search requires good recall and moderate precision, which is mitigated by allowing contradictory conclusions, and hyponymy. Context-sensitive advertising and video recommendation require good precision, but also require adequate recall in short documents. Both context-sensitive advertising and video recommendation can be affected both positively and negatively by hyperonymic expansion. Context-sensitive advertising has strong business drivers concerning the salient concepts of a field, their appropriate non-ontological organization, and the asymmetric relationship between article and advert content. This is mitigated by inserting a layer to bridge the gap between ontological representations and business needs. Video recommendation involves building natural language queries for conventional search engines, which requires consideration of corpus frequency in synonym usage, rather than simple clarity.

7 References

[1] Cimiano, P., Unger, Ch., and J. McCrae. Ontology-based Interpretation of Natural Language. Synthetic Lectures on Human Language Technology #24. Morgan & Claypool Publishers, 2014.

[2] Crestani, F. Aplication of Spreading Activation Techniques in Information Retrieval. *Artificial Intelligence Review* (11) 1997, pp. 453-482.

[3] Fahlman, S.E. Marker Passing Inference in the Scone Knowledge-Base System. Knowledge, Science, Engineering and Management. Lecture Notes in Computer Science. Vol. 4092. 2006, pp. 114-126

[4] Fellbaum, Ch., and P. Vossen. Challenges for a Multilingual Wordnet. *Language Resources & Evaluation* (46) 2012, pp. 313-326.

[5] Foxvog, D. Cyc. In: Theory and Applications of Ontology: Computer Applications. R. Poli, Healy, M., and Kameas, A. (eds.) Springer, 2010, pp. 259-278.

[6] Matthews, G, and M. Petrenko. Knowledge-based and vertical-driven information retrieval. *ILINTEC'13: Workshop on Intelligent Linguistic Technologies*. Part of Proceedings of ICAI: International Conference on Artificial Intelligence. Las Vegas, 2013, pp. 70-76

[7] Mitoguchi, R., and R. Kozaki. Ontology Engineering Environments. In: Handbook on Ontologies. S. Staab and R. Studer (eds.) Springer, 2009, pp. 315-337.

[8] Nirenburg. S. and V. Raskin, Ontological Semantics. Cambridge, MA: MIT Press, 2004.

[9] Norvig, P. Marker Passing as a Weak Method of Text Inferencing. *Cognitive Science*, 13(4) 1989, pp. 569-620

[10] Simperl, E., and Ch. Tempich. Evaluating the Economical Aspects of Ontology Engineering. In: Handbook on Ontologies. S. Staab and R. Studer (eds.) Springer, 2009, pp. 337-361.

[11] Sure, Y., Staab, S. and R. Studer. Ontology Engineering Methodology. In: Handbook on Ontologies. S. Staab and R. Studer (eds.) Springer, 2009, pp. 135-153

[12] Welty, Ch., and N. Guarino. Supporting ontological analysis of taxonomic relationships. *Data and Knowledge Engineering* (39) 2001, pp. 51-74

[13] Convera Corporation (2010). Search System and Method. US Patent us 7,668,825

A Rule-Based Approach to Free Word Order Languages

Anton V. Zimmerling[1]
[1]Institute for Modern Linguistic Research, Sholokhov Moscow State University for the Humanities, Moscow, Russian Federation

Abstract – *This paper is addressed the problems of modeling a rule-based approach to free word order languages. Grammar-based parsing for free word languages is problematic. Functionalist models of free word order are based on context-sensitive rules which cannot be parsed automatically. Generative models based on Chomskyan and Stablerian minimalist grammars undergenerate word orders licensed by natural free order languages. I argue that word order alternations in Russian can be predicted by Linear-Accent Transformations (LAT) linking together pairs of sentences with the same numeration but different information structure. LAT theory is compatible with the postulate on basic word order. Each LAT rule amounts to a pair of operations <Active movement; Remnant movement>, which makes it possible to reset LAT rules as mildly context-sensitive. Deaccenting of sentence categories in Russian can be analyzed as remnant movement*.

Keywords: syntax, information structure, movement, scrambling, free word order, mildly-context-sensitive rules

1 Introduction

Natural language processing interacts as a research field with the theory of formal grammars, cf. [7], [16] and formal frameworks in linguistics, cf. [2], [6], [11]. Generative capacity of native speakers, i.e. their ability of generating/recognizing well-formed structures and sorting out ill-formed structures is usually interpreted as proof for the hypothesis that the grammars of all natural languages share a core corresponding to some class of formal grammars. However, it is an open issue, whether languages with free word order, where syntactic trees can be linearized in more than one way, can be effectively recognized by grammar-based parsers. Empirically adequate descriptions of free word order languages include context-sensitive rules reordering already generated trees. Context-sensitive languages cannot be generated and parsed automatically, but in recent decades an attempt of eliminating context-sensitive rules was made. So called tree-adjoining grammars [7] and minimalist grammars [16] generate and parse syntactic trees according to a down-to-top principle and operate with mildly context-sensitive rules. The most consistent and elaborate formal grammars based on mildly context-sensitive rules are Stablerian minimalist grammars, cf. [16], [17], [18]. Stablerian grammars have a number of restrictive constraints on movement and adjunction of sub-trees. They can be extended with a scrambling operator, i.e. operator responsible for word-order alternations which makes them a suitable tool for parsing free word order languages. Still, lifting basic minimalist constraints on movement and adjunction, in a combination with a scrambling operator leads to derivational crash, at least in a sub-class of free-word languages known as languages with unbounded scrambling [8], [4], [14]. Formal linguistic frameworks, such as Chomskyan Minimalist Program [2], [3] operate with locality conditions on movement and adjunction similar or identical to the constraints of Stablerian minimalist grammars. A number of authors have addressed the diversity of scrambling types in natural languages like Russian, cf. [9], [6], [1], [21], [28]. There is general consensus that productive scrambling patterns have a communicative motivation. Generative models of Russian syntax capitalize the idea that word orders licensed by minimalist grammars represent the core of Russian grammar, while those not licensed represent its periphery and are only possible with given prosodic markings and communicative status, e.g. with contrastive focus or in thetic sentences, cf. [1], [6]. Still, these models are too rigid, sort out many well-formed Russian sentences and do not always predict the correct mapping of word order to information structure, so the undergeneration/completeness problem remains unresolved. Functionalist models of Russian syntax capitalize the idea that linearization of syntactic trees and assignment of communicative status are triggered in Russian by the same set of transformational rules called Linear-Accent-Transformations (LAT rules). LAT rules changing both the placement of sentence elements and their communicative status/prosodic markings correctly predict the diversity of word orders associated with one and the same numeration, which has been demonstrated in [9], [11], [21]. However, LAT theory in its current shape has a substantial drawback, since LAT rules are context-sensitive. Therefore, word order calculus based on LAT rules is impossible.

I aim at combining the advantages of the generative and functionalist models and offer a transformational approach to free word order languages like Russian, based on modified LAT rules. The paper has the following structure. In section 2, I render the notions of scrambling, conditions on movement and argue that Russian is a language with unbounded scrambling and direct prosodic marking of communicative status. In section 3, I argue that word order calculus based on LAT rules is feasible, if one adopts a postulate on basic word

order and defines LAT as mildly context-sensitive by resetting each rule as a pair <Active movement; Remnant movement>. Russian can be described by a small set of unidirectional LAT rules linking together pairs of sentences with the same numeration but different information structure. Deaccenting of sentence elements can be analyzed as remnant movement. LAT rules in Russian make use of 5 syntactically relevant prosodic markings, 4 of which belong to a strictly limited inventory of Russian tonemes, 'intonation constructions'.

2 Scrambling and formal grammars

The term 'free word order' is metaphoric, since all natural languages are restrictive: no language allows all possible linear orders or sentence categories in 100 % of sentences. It is reasonable to think that linearization constraints are present in all word order systems. However, there is a general consensus that free word order is a condition, when sentence categories can be linearized in two or more different ways at least in some well-formed sentences of a given language. This condition is also known as scrambling of predicate arguments and/or other sentence categories. It has become customary to classify natural languages into a class of languages with a fixed order of lexical sentence categories and a class of scrambling languages. For instance, an English sentence like *Pete ate a tomato* does not have a linear variant **A tomato ate Pete*, since this language blocks for OVS orders[1]. The class of scrambling languages can be defined in two ways – either as a) languages displaying a number of diagnostic movement patterns responsible for alternations like SVO > VSO, SVO > OSV, SVO > OVS, SVO > SOV; or b) languages completely lacking any fixed order of diagnostic sentence categories, say S and O or S, O and V. Both approaches proceed from the assumption that the same numeration, i.e. tree structure with a given number of positions filled by identical elements, may be linearized differently. A movement approach to scrambling languages capitalizes the idea that there is a unidirectional relation between different linear variants of the same numeration, one of the variants being the source of the other (s), cf. the presumably base-generated order in Rus. […] *Pet'a s''el pomidor* and the derived order [*Pomidor*$_i$] *Pet'a s''el* t_i: the symbol *t* marks the original placement of the moved category before the reordering, and the brackets […] mark the target position of the movement. A non-movement approach to scrambling denies the idea of a fixed order of sentence categories in a scrambling language and treats all linear variants as representing the same level of derivation. The domain, where categories scramble may be called scrambling domain. In the standard case illustrated by the Russian examples above, argument scrambling is bounded with a single clause, while all scrambled arguments S, O..U..W belong to one and the same verbal head v^o:

(i) $[_S\{_{\text{SCRAMBLING DOMAIN}} \ldots S \ldots v^o \ldots O \ldots\}]$.

Fig. 1. Local Scrambling.

Scrambling of the type (i) is called local or bounded; it does not pose big problems for linguistic theory with either non-movement or movement analysis, since all positions available for a scrambled category are in one and the same domain. Natural languages also have unbounded scrambling, where the permuting arguments may belong to different verbal heads v^1, v^2.. v^n. This has been proven in [15] for German, where unbounded argument scrambling takes place in complement clauses in the domain between the complementizer and the verbal complex, cf. (ii). Note that the verbal heads themselves are placed in a rigid order, so that the scrambling domain is more narrow than the complement clause:

(ii) $[_{CP}$ Comp $\{_{\text{SCRAMBLING DOMAIN}}$ $A^1 + B^2 + C^3\}$ $[_{VP}$ $[v^3, [v^2, [v^1]]]$ AUX].

Fig. 2 Unbounded Scrambling in German.

Stablerian minimalist grammars [16] and generative grammars based on Chomskyan Minimalist Program [2] generate ordered trees. Grammars of this type are mildly context-sensitive [4]. They can be adjusted for parsing scrambling languages, if their formalism is extended by a special Scrambling operator in addition to standard Merge and Move operators responsible for merging and moving sub-trees [14].

Under movement analysis, the scrambling type (local vs unbounded) is established in the end positions of the scrambled elements, not in their original positions before the reordering. There is a different tradition, where scrambling is understood as a characterictics of original domains. J.Baylin [1] distinguishes 'short' scrambling, when an element moves to a target position in the same clause and 'long-distance scrambling', when an element is extracted into a higher clause. However, extraction entails scrambling in the end domain only if the extracted element has more than one available target position in the higher clause. In the most simple case, under local scrambling, the elements remain in the same clause, so the original and the end domains match. This matching does not hold for unbounded scrambling. Therefore, it would be better to reserve the term 'scrambling' only for the distinction 'local vs unbounded' but replace it by the term 'movement' in Bailyn's opposition of 'short vs long-distance scrambling'.

Minimalist grammars (MGs) fail to parse languages with unbounded scrambling, cf. [8]. Both Chomskyan and Stablerian MGs are tree-adjoining grammars with a Move-operator, Merge-operator, Scramble-operator, Adjoin-operators and a number of locality conditions on movement

[1] A sentence like *A tomato ate Pete* will be proven well-formed if we assume that carnivorous vegetables exist, but this sentence won't get a linear variant *Pete ate a tomato* used in the same bizarre meaning "A human has been eaten by a vegetable". Consequently, the ungrammaticality of the SVO > OVS alternation in English does not depend on ontological assumptions about carnivorous vegetables and hungry humans.

and adjunction, known as Shortest Move Condition, Specifier Island Condition, Adjunct Island Condition etc. [3], [4]. MGs generate mildly-context-sensitive languages, while Chomskyan MGs aim at describing all natural languages as mildly-context-sensitive. The most salient condition affecting computational efficiency of a MG is the Shortest Move Condition (SMC)[2]. It requires that an element moves to the closest available target (Specifier of some P). Unbounded scrambling is a severe violation of the SMC. Blind application of the SMC to linear orders would mean that certain well-formed structures attested in languages like Russian or German would not be recognized/generated by an MG. Lifting the SMC and preserving the Specifier Island Constraint (SPIC) leads to Turing-equivalent grammars, i.e. to a derivational crash [8]. Moreover, MGs with unbounded scrambling (i.e. with scrambling without the SMC) and a single indiscriminating barrier[3] make the recognition problem NP-hard[4] [14].

2.1 Russian as an unbounded scrambling language

In this section I show that Russian is an unbounded scrambling language. Sentences with three scrambled NPs A^1, B^2, C^3 linked with three hierarchically arranged verbal heads are rare. Sentences with two scrambled NPs A^m, B^n, linked with two hierarchically arranged verbal heads v^m, v^n are wide-spread. One of the common cases of long-distance unbounded scrambling is triggered by non-projective embedding of a constituent or its element into a higher clause. Let A° B° C° D° E be the basic word order, A° B° C° D° be lexical heads and each next head be a dependent of the preceding one. It gives a projective structure (1), where blocks DE, CDE, BCDE, ABCDE are embedded constituents:

(1) [A ° [B° [C° [D °E]]]].

Moving the blocks DE, CDE and embedding the heads A°, B° into lower constituents one can get orders like $[CDE]_i$ A°B° t_i, $[[DE]_j$ C° $t_j]_i$ A°B° t_i, $[[DE]_j \ldots$ A°$_k$ … C° $t_j]_i$ t_k B° t_i, …A°$_k$ …$[[DE]_j$ C° $t_j]_i$ t_k B° t_i , where $t_{i, j, k}$ – traces of the moved heads or blocks. An illustration is provided in Fig. 3.

(1') Ru. *Arbitry*[1] *ne imeli prava*[1] [$_{IP}$ *fiksirovat'*[2] [*pobedu*[2] *«Triumfa»*]]][5].

'The referees[1] had no right[1] to fix[2] the victory[2] of 'Triumph''.

	Pattern	Word orders
Basic word order	[A° [B° [C° [D° E]]]]	(1a) *Arbitry*[1] *ne imeli prava*[1] [$_{IP}$ *fiksirovat'*[2] [*pobedu*[2] *«Triumfa»*]].
Derived orders	$[CDE]_i$ A°B° t_i,	(1b) ⇒ [$_{IP}$ *Fiksirovat' pobedu «Triumfa»*] $_i$ *arbitry ne imeli prava* t_i.
	$[[DE]_j$ C° $t_j]_i$ A°B° t_i,	(1c) ⇒ [*Победу «Triumfa»*] $_j$ [$_{IP}$ *fiksirovat'* t_j] $_i$ *arbitry ne imeli prava* t_i.
	$[[DE]_j \ldots$ A°$_k$ … C° $t_j]_i$ t_k B° t_i	(1d) ⇒ [*Pobedu «Triumfa»*] $_j$ *arbitry*$_k$ [$_{IP}$ *fiksirovat'* t_j] $_i$ t_k *ne imeli prava* t_i.
	…A°$_k$ …$[DE]_j$ C° $t_j]_i$ t_k B° t_i	(1e) ⇒ *Arbitry*$_k$ [*pobedu «Triumfa»*] $_j$ [$_{IP}$ *fiksirovat'* t_j] $_i$ t_k *ne imeli prava* t_i.

Fig 3. Long-Distance Unbounded Scrambling in Russian

2.2 Undergeneration problem

Since it is impossible to get the set of well-formed sentences of an unbounded scrambling language L by combining a scrambling operator with the SMC, all MGs undergenerate languages with unbounded scrambling and do not license some sentences which native speakers treat as well-formed. Bailyn's account of Russian short scrambling crucially relies on the SMC. He assumes that the basic word order in Russian is SVO, but with a wide class of transitive verbs every argument NP is equidistant from the preverbal position. This correctly predicts that OVS orders can be as discourse-neutral as SVO orders [1]. However, the architecture of his model urges him to make a wrong prediction that scrambled orders with a fronted verb and two-three post-verbal arguments should be unacceptable. In fact, sentences like (2) are well-formed in Russian.

(2) Dal učitel' včera knigu mal'čiku V-S-O-IO <a mal'čik eje zabyl posmotret'>

gave teacher-NOM book-ACC boy-DAT yesterday

"The teacher gave a book to the boy yesterday. <but the boy forgot to look it through.> "

T.L.King assumes that the basic word order in Russian is VS(O), while all other orders arise due to topicalization and focalization [6]. That means that Russian VS-sentences are thetic and base-generated, while Russian SVO ~OVS ~ SOV ~ OSV sentences are categorical and derived by overt movement. There are multiple issues with this approach. Russian thetic sentences are compatible both with VS and

[2]. Earlier called the Minimal Link Condition.

[3]. A non-discriminating barrier is a barrier not sensitive to the type of syntactic category the movement of which it blocks.

[4]. An NP-hard recognition can be fulfilled in a polynomial time, only if the language is NP-hard. This possibility cannot be eliminated as such, but practically it means that natural languages with unbounded scrambling are unparsable for MGs.

[5] For the sake of simplicity I treat the predicate *imet' pravo* 'have a right' as a single element.

with SV order [20], [30], while many Russian verb-initial sentences are categorial, non thetic, cf. [21], [28]. MG-based models still can be a useful, if they explain how the unparsable residue of well-formed sentences is derived from the alleged MG-compatible core. In the current generative research this problem is not solved yet.

2.3 Syntax vs information structure

Functionalist models of free word order capitalize the idea that scrambling is triggered by the same mechanisms that assign communicative status to sentence categories. The most elaborate formal model of the syntax-to-information structure interface in Russian was proposed by I.I.Kovtunova [9] and developed in [11], [12], [21], [28]. Kovtunova and her followers claim in Russian well-formed sentences with the same numeration but different constituent order and communicative status of elements are generated by a set of transformational rules called Linear-Accent Transformations. They also raise a claim that communicative categories are intrinsic features of sentences :

> (iii). Topic, focus, contrast and other communicative categories are intrinsic features of sentences, which take the same value by the speaker and the addressee.

A well-formed Russian sentence with topic and focus elements must have characteristic phrasal accents associated with topics and foci [10], [22], [24]. If topic, focus and other kinds of communicative status are intrinsic characteristics of sentences and Russian marks them prosodically, it is natural to assume that there is only one correct way to associate communicative structure and phrasal accents it correlates with.

.

> (iv) Russian is a language with direct prosodic marking of communicative status. Any correct interpretation of a well-formed Russian sentence implies the reader/addressee's ability to reconstruct the path from communicative semantics to phrasal accentuation, and vice versa. Communicative and prosodic structure are mapped to each other in a one-to-one correspondence in Russian.

Note that according to (iv) the information structure of a sentence may be reconstructed either from the verbal context or from phrasal prosody. The claim that Russian has direct prosodic marking of communicative status needs clarification. The two main intonation constructions, IC-1 ('↘') and IC-3 ('↗') are double-loaded: the falling pattern IC-1 marks either the focus in declarative sentences or the non-question constituent in yes-no questions, while the raising pattern IC-3 marks either the topic in declarative sentences or the question component in yes-no questions [5], [21]. Therefore, the claim that communicative and prosodic structure are mapped to each other in a one-to one correspondence in Russian can only mean that a) a sequence of Russian phrasal accents bears all the necessary information for the reconstruction of topic-focus articulation in a sentence, cf. [21], b) each relevant phrasal prosody in Russian can be interpreted as topic, focus marker etc. if we get the whole sequence of phrasal accents or learn the type of a sentence, whether it is a declarative, a question etc. The exact number of Russian phrasal tonemes and the limits of the allophonic variation in the realizations of Russian ICs is still an open issue, cf. different approaches in [24], [10]. I nevertheless argue that one does not need the whole alphabet of Russian tonemes for the purposes of word order calculus and that it is possible to build a LAT grammar using tags for just four phonologically distinct patterns: IC-3, IC-6, IC-1 and IC-2. The contrast in the pairs of two rising accents (IC-3 vs IC-6) and two falling accents (IC-1 vs IC-2) can probably be accounted for in terms of timing [28], but I simply state here that all these four accents are perceptually different.

IC	Tag	Communicative load
IC-3	↗	1) Topic. 2) The question component.
IC-6	↱	1) 2nd, degraded topic. 2) Left edge of a dislocated focus.
IC-1	↘	1) Focus. 2) The non-question component.
IC-2	↘↘	1) Focus. 2) The non-question component.

Fig. 4 Phonologically and Syntactically relevant Tonal Prosodies in Russian.

I will of speak of IC-3, IC-6, IC-1 and IC-2 as *non-zero accentual markings*. In addition I introduce an extra marking which does not correspond to any phonologically relevant prosody: it denotes a deaccenting operation. A tag like '$_0$X' reads 'constituent X got deaccented'[6].

IP	Tag	Prosodic cues
*	$_0$X	- (even tone/ absence of a tonal accent)

Fig. 5 Syntactically relevant 'eliminated accent'

I place the basic accentual tags before the constituent they attach to. This is done on several reasons, one them being the need to distinguish the basic tag for a given IC and the additional tag for its discourse-driven allophones[7].

[6]. Phonetic details, such as absolute pitch level (High, Semi-High, Semi-Low, Low), dynamic range, phonation type, tempo etc. do not bear for syntactic purposes — only the absence of a tonal movement is relevant.

[7] E.g. the two falling focal accents '↘X' or '↘↘X' are sporadically replaced in coherent speech by a rising tone marking the incompleteness of a text fragment: the latter option is tagged 'X↗'.

3 C-paradigms and LAT grammar

This section contains a list of transformational rules that derive accentually marked Russian sentences and form the *C*(ommunicative) paradigm of a sentence. I define a *C*-paradigm as a set of sentences sharing the same numeration i.e. constituent structure and given amount of lexical categories but having different constituent order and/or accentual tags. I define *LA–transformations* as rules, which both change constituent order and accentuation of at least one communicative constituent. Several issues have to be clarified. 1) LAT-rules establish the linkage and derivation vector between well-formed sentences with the same numeration. They do not create new syntactic positions. 2) Contrary to the original claim made in [11], LAT-rules are non-synonymic and can change the boundaries of communicative constituents. 3) The shift of accentual-marking from ↗X ~ ↘X ~↘↘X to $_0$X is a transformation**.** 3) LAT-rules are context-sensitive but can be reset as mildly context-sensitive [28]. 4) LAT-rules must be accounted for in terms of overt movement, not adjunction. 5) LAT-rules do not cover the so called afterthought elements. If LAT-rules are set as context-sensitive and all variants in a C-paradigm are mutually derivable, as [11] and [21] suggest, no word order calculus is possible. However, one can modify LAT-theory, combine it with the postulate on basic word order and reset LAT as mildly-context sensitive unidirectional rules. I am suggesting a procedure based on several interface principles. Topic and Focus are analyzed as communicative phrases headed by Topic Proper and Focus Proper. In the basic LAT-variant of a declarative sentence the boundaries of the TopicP and grammatical subject overlap. The FocusP consists of a Focus Proper phrase, where the main focus accent is located, and a transitional zone (Transition). The transitional elements are analyzed as belonging to FocusP, not to TopicP.

	TopicP		FocusP	
Communicative structure		Topic proper	Transition	Focus proper
Syntactic structure	Grammatical subject		Grammatical Predicate	
	External argument		Verbal head	Complements (Internal arguments & adjuncts)

Fig. 6 Prototypic mapping of information structure and syntactic structure for a verb with internal arguments

The previous research has shown that all previously LAT-rules have communicative motivation, be topicalization, focalization, splitting of FocusP etc. At the same time, all previously described LAT-rules also have predictable side effects, such as deaccenting of elements crossed by other elements undergoing leftward movement. E.g., a derived verb-initial sentence ↱*Posadil*$_i$ $_0$*ded* t$_i$ ↘*repku* 'Gramps planted a turnip' is generated from a basic SVO sentence [$_{TopicP}$ ↗ *Ded*] [$_{FocusP}$ *posadil* [$_{FocusProper}$ ↘*repku*]], where both the topical subject ded 'gramps' and the focal VP have their characteristic prosodic markers. In the derived V1 sentence ↱*Posadil*$_i$ $_0$*ded* t$_i$ ↘*repku* splitting of FocusP has a side effect in the deaccenting of the subject *ded.* The deaccenting of *ded* in such structures is not the primary communicative goal of the LAT-rule, but a consequence of the fact that the dislocated verb *posadil* 'planted', which lacked a tonal prosody in the basic SVO variant, was moved, got a special accent marking '↱' characteristic of dislocated elements, and crossed the node *ded* on its way. I suggest that LAT-rule 'Verb dislocation' examplified by this pair of sentences, can be reset as a pair of operations <Active movement (dislocation of the verb) & Remnant movement (deaccenting of the subject)>. Remnant movement is kind of compensatory effect responding to active movement. Russian data prompt that active movement always correlate with acquiring new non-zero prosodic markings, which is generalized in (v).

> (v) LAT-rules in Russian can be reset as pairs of operations <Active movement; Remnant movement>. Active movement puts sentence elements into target positions, where they get non-zero accent markings: '↗X', or'↘X', or '↘↘X', or'↱X'). Operations, which put sentence elements into positions with zero accent $_0$X, instantiate Remnant movement.

In order to make LAT grammar of Russian feasible, one has to adopt one more further postulate :

> (vi). Russian thetic sentences are derived through LAT-rules from categorial sentences with the same numeration, but non vice versa.

I argue that all Russian thetic sentences irrespective of their surface order (SV, VS, VSO, SVO) are derived from categorial sentences by deaccenting their theme. Topic daccenting results from Left Focus Movement i.e. an operation moving a postverbal complement X which bears the focus accent (schematically - ↘X) to the left for its governing verbal category. The moved element gets a reinforced focus accent (↘↘X): [$_{VP}$ V° ↘ X] ⇒ ↘↘X$_i$ …V° t$_i$. Russian does not allow post-focal accented themes [22]. Therefore, if a focal element moves outside VP and crosses the position of an accented thematic subject marked with IC-3 (↗X), the subject gets deaccented (↘$_0$X): [$_{NP}$ ↗S°] [VP V° ↘X] ⇒ ↘↘X$_i$ **[$_0$S]**…V° t$_i$. This explains why Russian thetic sentences can be realized both with SV and VS-orders since ↘↘S$_0$V structures like *Babuška spit* 'Grandma is asleep' are just inverted variants of $_0$V↘S structures, cf. the derivation in (4). Neither King's nor Bailyn's analysis fits Russian VS-

sentences since VS-orders apart from marking theticity can also mark three types of categorial sentences. The inverted verb can be a) the theme (↗V) b) the rheme (↘V) c) part of the dislocated rheme, schematically marked as (↱V). In all these cases the verb gets different accent markings.

(3) <Pocemu tak malo narodu?> $_0$Direktor $_0$p'at' $_0$sotrudnikov v [$_{FocusProper}$ ↘ ↘ *komandirovku*$_i$] $_0$poslal t_i.
<'Why so few people here?> 'The director has sent five workers to *a business trip*'.
Thetic, S-DO-IO-V.

(4a) ↗ Babuška ↘ spit. ⇒ (5b) $_0$Spit [$_F$ ↘babuška]. ⇒ (5c) [$_F$ ↘ ↘ Babuška]$_i$ spit t_i.
grandma-Nom sleeps sleeps grandma-Nom

I introduce 4 basic symbols for communicative phrases — F (Focus), T (Topic), Tr (Transition), SF (Dislocated Focus component. Symbol '↦' stands for 'movement of X to the right from its base position', 'right movement' symbol '↤' reads 'movement of X to the left from its base position', ' left movement', symbol '↧' reads 'deaccenting of X'. Formula X/F ↦ ' reads 'right movement of X to a clause-final Focus position', formula ↤ X/T' reads 'left movement of X to the clause-initial Topic position', formula '↤ Tr/T' reads 'left movement of X from the base position of Tr(ansition) to the clause-initial Topic position' etc. The symbol '&' is inserted between the formulas of the Principal and Remnant movement (cf. X/T & Y/F) and signals a determinist relation between Active and Remnant movement patterns. A formula like X/F↦ & ↧ T, which describes Right Focus Movement, reads 'Right movement of X to the clause-final Focus position; deaccenting of the node T is due to the fact that X crosses node T'.

	Rule	Operation	Active movement	Remnant movement
1.	Right Focus Movement	X/F ↦ & ↧ T	X/F ↦	↧ T
2.	Left Focus Movement	↘F/↘ ↘F ↤ & Tr	↘F/↘ ↘F ↤	Tr
3.	Verb Topicalization	Tr/T ↤ & ↧T	Tr/T ↤	↧T
4.	Dislocation	Tr/SF ↤& ↧T	Tr/SF ↤	↧T
5.	Verb Focalization	Tr/F ↤ & ↧T , ↧F	Tr/F ↤	↧T, ↧F
6.	Topic-Focus Inversion	F/T ↤& T/F ↦	F/T ↤	T/F ↦

Fig. 7. Linear-Accent Transformations in Russian.

4 Conclusions

Word order in scrambling languages like Russian can be predicted by Linear-Accent Transformations changing both the placement of sentence elements and their communicative status. Each LAT rule amounts to a pair of operations <Active movement; Remnant movement>, which makes it possible to reset LAT as mildly context-sensitive rules.

5 Acknowledgements

The paper is written with financial support from the Russian Foundation of Sciences, project RSCF 14-04-18-03270 'Word order typology, communicative-syntactic interface and information structure in world's languages'.

6 References

[1] John F. Bailyn. "Generalized inversion"; Natural Language and Linguistic Theory, 22, 1-49, 2004.

[2] Noam Chomsky. "A Minimalist Program for Linguistic Theory" ; The view from building 20. /Hale, K. S.L.Keyser (eds). MIT Press, 1993.

[3] Noam Chomsky. "Three factors in language design" ; Linguistic Inquiry, 36: 1–22, 2005.

[4] Hans Martin Gärtner & Jens Michaelis. "Some remarks on the Locality Conditions and Minimalist Grammars" ; Interfaces + Recursion = Language? Chomsky's Minimalism and the View from Syntax and Semantics. Mouton de Gruyter, 162-195, 2007.

[5] Nina Y. Shvedova et alii (eds.). Russkaja grammatika. Moscow: Nauka, 1982.

[6] Tracy H. King. "Configuring Topic and Focus in Russian". CSLI, 1995.

[7] Aravind K. Joshi. "Tree adjoining grammars: How much context-sensitivity is required to provide reasonable structural descriptions?"; D. R. Dowty, L.Karttunen, and A.M. Zwicky (eds.), Natural Language Parsing. Psychological, Computational, and Theoretical Perspectives, 206–250. Cambridge University Press, 1985.

[8] Gregory M. Kobele and Jens Michaelis. "Two type 0-variants of Minimalist Grammars"; Jäger, G., Monachesi, P., Penn, G., Wintner, S., eds. FG-MOL 2005. Proceedings of the 10th conference on Formal Grammar and the 9th Meeting on Mathematics of Language, Edinburgh, Scotland (2005).

[9] Irina I. Kovtunova. "Sovremennyj russkij yazyk. Por'adok slov i aktual'noe členenie predloženija." ['Modern Russian. Word Order and Communicative Perspective' – in Russian]. Moscow: Nauka, 1976.

[10] Cecilia Odé. "Perspektivy opisanija i transkripcii russkoj intonacii v korpusah zvučasčix tekstov" ['Perspectives of describing and transcribing Russian

intonation in Russian sound corpora' – in Russian]; Problemy i metody ekxperimental'no-fonetičeskix issledovanij. K 70-letiju professora L.V.Bondarko. Sankt-Petersburg, 2003.

[11] Elena V. Padučeva. "Kommunikativnaja struktura predloženija i ponjatie kommunikativnoj paradigmy " [= 'Communicative structure and the notion of communicative paradigm' – in Russian]; Naučno-texničeskaja informacija, Series 2. N 10, 25-31, 1984.

[12] Elena V. Padučeva. "Vyskazyvanie i ego sootnesennost' s dejstvitel'nost'ju" ['Utterance and its reference' – in Russian]. Moscow: LKI, 2008.

[13] J.B.Pierrehumbert. "The Phonology of English Intonation". M.I.T. Doctoral dissertation, 1980.

[14] Alexander Perekrestenko. "Minimalist Grammars with unbounded scrambling and non-discriminating barriers are NP-hard"; Martin-Vide, Carlos, Friedrich Otto, Henning Fernau (eds.), Language and Automata Theory and Applications [Lecture Note on Computer Science], Springer, 321-432, 2008.

[15] Owen Rambow. "*Formal and Computational Aspects of Natural Language Syntax"*. PhD thesis. IRCS Technical Report, University of Pennsylvania, 1994.

[16] Edward P. Stabler. "Derivational minimalism"; Christian Retore, ed., Logical Aspects of Computational Linguistics. Springer, p. 68–95, 1997.

[17] Edward P. Stabler. "Acquiring languages with movement"; Syntax, 1: 72–97, 1998.

[18] Edward P. Stabler. "Remnant movement and complexity"; G. Bouma, G.-J. M. Kruijff, E.Hinrichs, and R. T. Oehrle, (eds.), Constraints and Resources in Natural Language Syntax and Semantics, pp. 299–326. CSLI Publications, 1999.

[19] Edward P. Stabler. "Recognizing head movement"; P. De Groote, G. Morrill, C. Retoré (eds.). Logical Aspect of Computational Linguistics, Springer, 245-260, 2001.

[20] Tatyana E. Yanko. "Kommunikativnaja struktura s neingerentnoj temoj" ['Communicative structure with a non-inherent theme' – in Russian]; Naučno-texničeskaja informacija, Series 2., 25-32, 1991.

[21] Tatyana E. Yanko. "*Kommunikativnye strategii russkoj reci v sopostavitel'nom aspekte"* ['Communicative strategies of Russian speech in a comparative aspect' – in Russian]. Moscow, 2001.

[22] Tatyana E. Yanko. (2008) "*Intonacionnye strategii russkoj reci v sopostavitel'nom aspekte*." ['Intonation strategies of Russian speech in a comparative aspect' – in Russian], Moscow, 2008.

[23] Tatyana E. Yanko. "Accent placement principles in Russian"; Computational linguistics and intellectual technologies, vol. 10 (17), 712-724, 2011.

[24] Olga Yokoyama. "Neutral and non-neutral intonation in Russian: A reinterpretation of the IK system"; Die Welt der Slaven. XLVI. 1-26, 2001.

[25] Anton V. Zimmerling. "*Tipologičeskij sintaksis skandinavskix yazykov*." ['Typological Scandinavian Syntax' - in Russian]. Moscow: Yazyki slavyanskoj kul'tury, 2002.

[26] Anton V. Zimmerling. "Topic-Focus Articulation, Verb Movement and the EPP in Russian"; Slavic Linguistic Conference SLS 2. Berlin, 2007.

[27] Anton V. Zimmerling. "Locative Inversion and Right Focus Movement in Russian". Moscow, 2008.

[28] Anton V. Zimmerling. "Local and global rules in syntax" ; Computational linguistics and intellectual technologies, vol. 7 (14), 551-562, 2008.

[29] Anton V. Zimmerling. "Scrambling types in Slavic languages" ; Computational linguistics and intellectual technologies, vol. 10 (17), 750 - 766, 2011.

[30] Anton V. Zimmerling. "Word order, intonation and thetic sentences in Russian" ; Linguistics Association of Great Britain, LAGB 2013 Annual Meeting. Abstracts. University of London, School of Oriental and African Studies, 28-31 August, 2013. P. 67.

On the method for automatic determination of semantic similarity of the document texts

Victor N. Zakharov[1], **Alexey A. Khoroshilov**[2], **and Alexandr A. Khoroshilov**[2]
[1]Institute of Informatics Problems of the Russian Academy of Sciences, Moscow, Russia
[2]Center of Information Technologies and Systems of Executive Power Organs, Moscow, Russia

Abstract - *The paper describes the methods for automatic generation of the formalized semantic document description and determining of semantic proximity of document texts. These methods are based on the use of semantic-syntactic and conceptual analysis procedures providing the identification of the conceptual text content and the assignment of the characteristics to the concept names, corresponding to their semantic role in the text. Automatic thematic text content similarity assessment is made by comparison of the conceptual text content.*

Keywords: Semantics, text similarity, natural language documents, information processing

1 Introduction

1.1 Problems of Text Information Processing

At present in connection with ever-growing amount of information resources the access of the users to data of their interest becomes more and more difficult.

The main problem of text information processing is the difficulty of automatic generation of the formalized description of document semantic content and, as a consequence of this, the difficulty of determining the semantic relations between different documents. This is because the same situations in different texts can be described in terms of various degrees of generality and with the help of different linguistic means. And only a man analyzing the documents in accordance with his understanding of the document content and means for expression of this content, and on the base on his professional knowledge and experience, can determine the degree of semantic similarity of the documents being analyzed. The most part of automatic text processing systems operating now are unable to solve these problems in full measure.

In this connection, it is necessary to develop effective methods of automatic analysis of the document content. The distinctive feature of the proposed methods is that they are based on the modern perception of conceptual text structure and unique procedures for semantic-syntactic and conceptual analysis.

1.2. Text Comparison Methods

At present a lot of text comparison methods are used for solving different problems of the text information analysis [5-8], but the most commonly used methods are TF, Opt Freq, Lex Rand, Log_Shingle, Megashingles, Long Sent, Descr Words. The study of the potentials of these methods was discussed in the paper [5], in which the wide-scale experiment on the comparative analysis of above mentioned methods is described. The aim of this work was to estimate the quality of best-known, various and computationally efficient determination algorithms for fuzzy duplicate detection. It was proposed to compare algorithms with respect to completeness and accuracy, to determine the correlation between them and the combined coverage of the initial set of pairs of fuzzy duplicates by different algorithm combinations. The web collection of ROMIP documents (about 500 thousand documents) was used as a test collection.

Various statistically processed text fragments (fixed sequences of significant words ("shingles"), frequency word dictionaries, etc.) were used in the algorithms under study as one of the document similarity measure parameters. The most accurate results were shown by the algorithms based on the use of longer text fragments. The algorithms based on the use of shorter text fragments provided higher completeness, but showed lower comparison accuracy.

It should be noted that in all considered algorithms the text is regarded as a set of isolated words. Various operations carried out in the process of duplicate text search were performed with words and strings of these words. But the text is not a set of words and sequences of these words, and, first of all, it is necessary to compare the text semantic units in determining the semantic similarity of documents. It is also necessary to take into account the variation of presentation forms in the text of the same sense.

Recently, the semantic methods of text comparison received widespread usage in foreign authors' works. The so-called deep semantic-oriented approach is used in the work [13-15] for detection of documents similar in sense (duplicates). This method is based on the use of semantic networks obtained with the help of semantic and syntactic analyzer. In this case both lexical and semantic relations in the text are taken into consideration. The processing

complexity of incorrect, homonymous, and negative phrases was detected in the use of this method.

A similar approach is used in the works [16-18]. The authors use the WordNet electronic thesaurus as a tool for determination of semantic relations. One of the original ideas set out in this work is that the semantic profiles of words are expressed in terms of implicit, explicit, or salient concepts. This solution allows to pass from the sparse word-space to a richer and unambiguous concept space. It allows to determine relations of semantic similarity of concepts. The standard cosine method is used for determining the text similarity measures.

2 Procedures for Document Semantic Analysis

The main purpose of automatic text processing procedures and technologies is to solve such problems as structuring and formalization of the semantic content of the text, definition of the conceptual content of the subject field, determination of paradigmatic, syntagmatic and associative relations between concept names and determination of their context environment. In what follows we consider the main procedures for text information processing. [1–4] The basic procedure is the procedure of morphological analysis.

2.1 Morphological Analysis of Words

The morphological analysis of natural language words is intended to determine word structure and assign grammatical features necessary for execution of various automatic text processing procedures, such as procedures for morphological synthesis of words, text parsing, syntactic synthesis of texts and conceptual analysis.

The morphological analysis procedure developed by the authors of this study is based on the ingenious algorithms and methods of generation of machine dictionaries and unique methods for high-speed searching in them.

The procedure for word normalization (lemmatization) - the procedure of reduction of the word text forms to the normalized forms is developed on the base of the morphological analysis. Usually, the normalized (canonic) form of the word means its form that is usually given in dictionaries.

2.2 Semantic and Syntactic Analysis of Texts

The semantic and syntactic analysis of texts is carried out with the purpose of formalized representation of their structure - extraction of semantic units and determination of relations between them. The text structure may be interpreted in many ways and described in various formalized languages. For solving specific problems, such as the problems of concept text model building, the text components (text segments), which denote concepts: words, word combinations, phrases, superphrasal unities should be extracted from the text.

2.3 Conceptual Analysis of Texts

The conceptual analysis of texts is intended to determine the conceptual text structure, define the conceptual content of the texts and determine the semantic relations between the concept names. In more restricted sense, the conceptual analysis can be considered as the procedure for determination of the concept names in the texts. This problem is complicated and it can't be solved only by the analysis of syntactic text structure. It is also necessary to use semantic features for solving this problem. The authors of this study developed several variants of solving this problem.

3 Conception of Semantic Processing of Text Information

3.1 Conception of Formalized Semantic Description of Documents

The solution of the problem of comparing the semantic content of the documents can be provided by presentation of this content in the formalized form. In our study a combination of normalized concept names and relations between them is considered as the formalized semantic description of the document.

The formalized semantic description of the document must include the concept names along with a factor determining the level of their semantic significance in the text. Therefore, in building the formalized document description it is necessary to determine a number of description elements and assign a weighting factor to each element. For this purpose it is necessary to detect informative words or word combinations in the text under analysis on the base of their formal characteristics among which are: their frequencies in the subject field and in the particular text, word combination lengths (in words), their belonging to categories of geographical names or personal names, as well as their availability or unavailability in the standard conceptual dictionary and their availability or unavailability in the stopword list.

Each element of the formalized semantic document description consists of a pair of concept names - actants, related by the concept - predicate.

Thus we can formulate the following definition of the formalized semantic document description (FSDD), by which we mean the ordered set

$F = \{Su_i \mid i \in [1, n_F]\}$, where

n_F - number of elements in the formalized semantic document description;

$Su_i = (Nc_i, w_i, R_i)$ - i-th element of the FSDD;

Nc_i - concept name;

w_i - weighting factor corresponding to the concept name;

R_i - set of links relating to given element of the FSDD.

3.2 Determination of Weighting Factors for Concept Names

For specifying the semantic significance of the concept name it is necessary to assign a weighting factor to each concept name in the formalized semantic document description.

The following characteristics of the concept name have impact on the weighting factor:

- Number of concept name occurrence in the subject area (global frequency)
- Number of concept name occurrence in the text (local frequency)
- Concept name length (in words)
- Belonging of the concept name to the proper name category

In assigning the weighting factors we will use the formula proposed by us:

$$W_{ij} = \begin{cases} (p_{ij} + fg_{ij}) \cdot f_{ij} \cdot l_{ij} & l_{ij} \le k_{\max} \\ (p_{ij} + fg_{ij}) \cdot f_{ij} \cdot k_{\max} & l_{ij} > k_{\max} \end{cases}.$$

Where, p_{ij} - coefficient increasing the degree of concept name significance depending on its belonging to the categories of proper names, geographical names etc.

l_{ij} - number of words in the word combination, which expresses the j-th concept in the i-th text;

f_{ij} - frequency of occurrence of the j-th concept in the i-th text;

fg_{ij} - normalized frequency of occurrence of the j-th concept name in the subject field (We assign a rank of q to concept names in accordance with the predefined ranges of values of concept name occurrence in the subject field. This frequency is taken from the frequency dictionary of concept names, compiled on the basis of text corpus of given subject field)

$k_{\max}$ - experimentally determined coefficient corresponding to the maximum word combination length, after which it won't have effect on resulting concept name weight.

4 Automatic Estimation of Document Semantic Similarity

4.1 Description of the Process of Automatic Determination of Document Semantic Similarity

The procedure for automatic determination of document semantic similarity is performed by comparison of the formalized semantic content of two documents [9-12]. Since, in this work, the problem consists in determination of semantic similarity of the thematically related documents covering the same subjects, but maybe in a somewhat different aspect, we can simplify the formalized semantic description and, in this case, eliminate the links between objects from it. Then the formalized semantic document description will take the following form:

$F = \{Su_i \mid i \in [1, n_F]\}$, where

n_F - number of elements in the formalized semantic document description;

$Su_i = (Nc_i, w_i)$ - i-th element of the FSDD;

Nc_i - concept name;

w_i - concept name weight.

To perform a task of automatic determination of the document semantic similarity it is necessary to establish formal criteria determining the numerical characteristic of the degree of their similarity.

The computational formula for the coefficient of the thematic content similarity of p-th and q-th texts can be written as follows:

$$K_{\text{sim}} = \frac{\sum_{j=1}^{n_\cap} w_{\cap j} \cdot \sum_{j=1}^{n_p} f_{pj}}{\sum_{j=1}^{n_p} w_{pj} \cdot \sum_{j=1}^{n_q} f_{qj}}$$

$w_{\cap j}$ - j-th component of the vector of weighting factors of concept names contained in both texts, with weights being taken from the formalized semantic description of the q-th text.

w_{pj} - j-th component of the vector of weighting factors of concept names contained in the p-th text.

f_{pj} - j-th component of the vector of frequencies of concept names contained in the p-th text.

f_{qj} - j-th component of the vector of frequencies of the concept names contained in the q-th text.

$n_\cap$ - length of the vector of concept names contained in both texts.

n_p - length of the vector of concept names contained in the p-th text.

n_q - length of the vector of concept names contained in the q-th text.

With increasing of the coefficient K_{sim}, the degree of thematic similarity of the thematic text content increases. The experiments performed with Russian texts confirmed the reliability of the proposed methods [10-

12]. This confirms the hypothesis that if $K_{sim} = 1$, the main contents of the texts under analysis are very similar and sometimes identical.

5 Experiment on Determination of Semantic Similarity of Documents from Science Magazine Collection

The objective of the experiment was to automatically cluster the semantically similar English-language documents on the base of the proposed methods. The collection of English-language texts from the Science Magazine including 1584 documents on a wide range of subject areas was used as the initial data for the experiment. The results of the preliminary analysis of the text collection are given in Table 1.

Table 1. Results of preliminary analysis of text collection

Number of texts	1835
Total text size (in symbols)	33 million
Number of word combinations	7 million
Number of different words	887 thousand

The frequency dictionary of word combinations was automatically compiled using this text collection. The fragment of this dictionary is given in Table 2. Each dictionary entry consists of frequency, normal form and text form of the word combination.

Table 2. Fragment of frequency dictionary

*00001018 health service * health services*
*00001017 gene transfer * gene transfer*
*00001017 natural resource * natural resources*
*00001015 risk assessment * risk assessment*
*00001013 fusion protein * fusion proteins*
*00001009 blood pressure * blood pressure*
*00001009 guinea pig * guinea pigs*
*00001006 aids patient * AIDS patients*
*00000996 somatic cell * somatic cell*
*00000994 cell body * cell body*
*00000994 muscle cell * muscle cells*
*00000989 national security * national security*
*00000985 growth hormone * growth hormone*
*00000970 protein structure * protein structure*
*00000958 communication skill * communications skills*
*00000955 nuclear weapon * nuclear weapons*
*00000944 mental health * mental health*
*00000943 natural science * natural sciences*
*00000942 human chromosome * human chromosomes*
*00000935 rat brain * rat brains*
*00000935 salary range * salary range*
*00000925 clinical research * clinical research*
*00000923 skeletal muscle * skeletal muscle*
*00000919 research grant * research grants*
*00000916 cell proliferation * cell proliferation*
*00000915 serum albumin * serum albumin*
*00000914 dna polymerase * DNA polymerase*
*00000905 membrane potential * membrane potentials*
*00000901 electron density * electron density*
*00000901 molecular size * molecular size*
*00000900 hela cell * HeLa cells*
*00000886 synthetic peptide * synthetic peptides*
*00000869 low temperature * low temperatures*
*00000868 high energy * high energy*
*00000868 lymph node * lymph nodes*

The conceptual dictionary was compiled on the base of this dictionary. The dictionary size was about 1 million word combinations. Then this dictionary was converted into machine form and was used in the experiment on determination of semantic similarity of the documents**.**

The following procedures were used as a tool for the study:

1. The procedure for FSDD generation from the text of the English-language document. The concept names were extracted using the standard dictionary which was also created earlier within this study.
2. The procedure for comparison of the FSDDs of two documents (the second document in relation to the first one), visualization of matched elements of the FSDDs, counting the number of matched elements of the FSDDs, summation of the coefficients of their significance and percentage match of the FSDD of the second document in relation to the first one. Figure 1 shows the general diagram of the process of FSDD generation from the text of the English-language document.

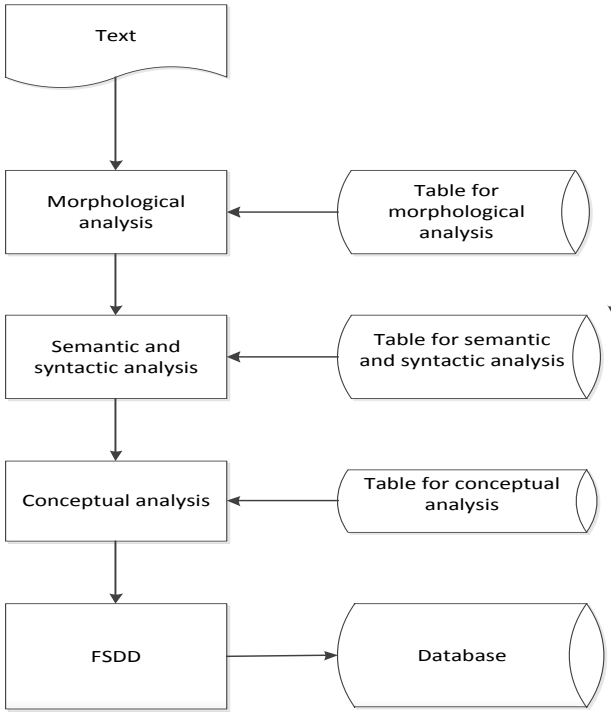


Figure 1 - General diagram of the process of FSDD generation from the text of the English-language document

The main objective of this study - the automatic determination of the degree of semantic similarity of English-language scientific and technical documents on the base of analysis of their semantic content can be accomplished by automatic comparison of all documents of the collection. Implementation of this objective within the file system would require large work efforts (over 3 million comparing operations for the document content). Thus, we decided to restrict ourselves to comparison of several dozens of randomly chosen documents with all documents of the collection. Since the FSDD generation is required for comparison of any document of the collection with all the rest of the documents of the collection, the FSDDs for all documents of the collection were generated at first. Table 3 gives the statistics on the sample of documents used in the experiment on determination of the thematic similarity of documents. This table shows the number of the document, its size (in symbols), amount of word combinations, and the sum of the coefficients of their significance.

Table 3 Statistics on the sample of processed documents of the collection

Number of document	Size (in symbols)	Amount of word combinations	"Weight" of documents
00168	84694	476	1604
00300	89613	500	1342
00419	82082	506	1778
00596	53845	434	2118
00835	107986	715	2258
00928	112248	506	1678
00942	633102	3207	9307
01354	14238	3398	97443

In the next stage of the experiment the pair-wise comparison of the FSDD of each document in the sample with FSDDs of all documents of the collection was conducted. The results of comparison of each document were analyzed and the only one document having the best matching of the sums of the significance coefficients was chosen for each document. Table 4 provides the results of the pair-wise comparison of the FSDDs of documents in the collection. This table shows the source document number (text №1), best-matching document number (text №2), the number of matched word combinations, weight of matched word combinations (the sum of their significance coefficients) and the matching percentage.

Table 4 Results of the pair-wise comparison of the FSDDs of documents in the collection

Text №1	Text №2	Number of matched word combinations	Weight of matched word combinations	Matching percentage
00168	00835	45	183	11%
00300	01354	64	189	14%
00419	00942	39	337	18%
00596	00942	62	299	14%
00835	00168	45	183	8,1%
00928	01354	37	124	7,3%

6 Conclusions

The methods proposed in this work were implemented in the form of the program complex and their efficiency was tested with the use of large samples of texts with different degrees of thematic similarity, both in English and Russian. The conducted experiments confirmed the efficiency of the presented methods.

The program complex developed on the base of these methods provides the efficient retrieval of similar documents and it can be used e. g. for classification and rubrication of documents. The further implementation of the complete complex of document semantic analysis procedures for different languages will allow to develop the tools for retrieval of similar documents in multi-language knowledge base and determination of relations between multilingual documents.

References

[1] Kozerenko E. B. Cognitive Approach to Language Structure Segmentation for Machine Translation Algorithms // Proceedings of the International Conference on Machine Learning, Models, Technologies and Applications, Las Vegas, USA, 2003. — CSREA Press, 2003. P. 49–55.

[2] Belonogov, G. G., Kalinin, Yu. P., Khoroshilov, A. A. Computational linguistics and Advanced Information Technologies. Theory and Practice of Constructing of Automatic Text Processing Systems. - Moscow: Russkiy mir, 2004. – 264 p.

[3] Belonogov, G. G., Bystrov, I. I., Kozachuk, M V., Novoselov, A. P., Khoroshilov A.A. Automated Conceptual Text Analysis. Col. :" Scientific Technical Information", Series 2. - 2002. № 10.

[4] Vasilyev V.G., Kryvenko Methods of automated text processing. Moscow: IPI RAN, 2008.– 301p.

[5] Zelenkov Yu.G., Segalovich I.V., Comparative analysis of near-duplicate detection methods of Web documents. Proceedings of the 9th Scientific Conference "Digital Libraries: Advanced Methods and Technologies, Digital Collections"- RCDL'2007, Pereslavl-Zalesskij, Russia, 2007. Vol. 1, P. 166-174.

[6] U. Manber. Finding Similar Files in a Large File System. Winter USENIX Technical Conference, 1994.

[7] A. Broder, S. Glassman, M. Manasse and G. Zweig. Syntactic clustering of the Web. Proc. of the 6th International World Wide Web Conference, April 1997.

[8] S.-T. Park, D. Pennock, C. Lee Giles, R. Krovetz, Analysis of Lexical Signatures for Finding Lost or Related Documents, SIGIR'02, August 11-15, 2002, Tampere, Finland.

[9] V. N. Zakharov, A. A. Khoroshilov / Automatic assessment of the similarity of the texts' thematic content on the base of their formalized semantic descriptions comparison // Proceedings of the 14th Scientific Conference "Digital Libraries: Advanced Methods and Technologies, Digital Collections"- RCDL'2012, Pereslavl-Zalesskij, Russia, 2012. Vol. 1, P. 145-153.

[10] Borzyk H A. I., Bragin G. A.. Khoroshilov A.A. Methods of automatic clustering of documents in storages of scientific and technological information for solving of search problem of plagiarism in the document texts. Informatization and connection, Vol. 8, 2012.

[11] V. N. Zakharov, A. A. Khoroshilov / Semantic methods for solving a problem of automatic detection of plagiarism in structured scientific and technical documents. Proceedings of the 15th Scientific Conference "Digital Libraries: Advanced Methods and Technologies, Digital Collections"- RCDL'2013, Yaroslavl, Russia, 2013. Vol. 1, P. 145-153.

[12] A. A. Khoroshilov / Methods for automatic determination of semantic proximity of documents on the basis of their conceptual analysis. Proceedings of the 15th Scientific Conference "Digital Libraries: Advanced Methods and Technologies, Digital Collections"- RCDL'2013, Yaroslavl, Russia, 2013. Vol. 1, P. 145-153.

[13] Hartrumpf, Sven; Tim vor der Brück; and Christian Eichhorn (2010a). Detecting duplicates with shallow and parser-based methods. In Proceedings of the 6th International Conference on Natural Language Processing and Knowledge Engineering (NLPKE), pp. 142-149. Beijing, China.

[14] Tim vor der Brück and Sven Hartrumpf (2009). A readability checker based on deep semantic indicators. In Human Language Technology. Challenges of the Information Society (edited by Vetulani, Zygmunt and Hans Uszkoreit), volume 5603 of Lecture Notes in Computer Science (LNCS), pp. 232-244. Berlin, Germany: Springer.

[15] Sven Hartrumpf, Tim vor der Brück, Christian Eichhorn: Semantic Duplicate Identification with Parsing and Machine Learning. TSD 2010: 84-92.

[16] Carmen Banea, Samer Hassan, Michael Mohler, and Rada Mihalcea, UNT: A Supervised Synergistic Approach to Semantic Text Similarity, Proceedings of the Sixth International Workshop on Semantic Evaluation SemEval 2012.

[17] S. Hassan and R. Mihalcea. Measuring semantic relatedness using salient encyclopedic concepts. Artificial Intelligence, Special Issue, 2011.

[18] M. Mohler and R. Mihalcea. 2009. Text-to-text semantic similarity for automatic short answer grading. In Proceedings of the European Association for Computational Linguistics (EACL 2009), Athens, Greece.

Cognitive Machine Translation Model: POVs and Musketeer Classes as Algorithms of Shared Reality

Boris Gorbis
Beverly Hills, CA, USA

Abstract

A multimodular Cognitive Translation Model (CTM) offers algorithms of recognition, identification and equivalency in a model of translation replicating human cognitive-linguistic competency. Translation adequacy is achieved by triangulating unique alpha-numeric addresses of CMT elements.

This protocol (1) relates a word-form to a referent; (2) identifies one referent's POV spectrum –. an inventory of all Points of View of any aspect of the referent. (3) identifies members of the class of referents created by a POV; ((4) assigns each POV a unique alpha-numeric address. (5) connects POV to its four proprietary sets of Musketeer classes of operants serving all referents grouped by the POV. (6) identifies that each referent word-form pair claims unique coordinates of (a) all POV addresses in its spectrum and (b) all addresses of operants bundled with each POV. (7)Textual equivalency between coda is established and verified by finding referents with identical coordinates.

Key-words: Machine Translation,Cognitive Translation, Equivalency, Translation Model, Artificial Intelligence, Cognitive modeling;

1. Universal Platform of *Shared Reality*

The paradox of machine translation (MT) is this -- to achieve competent results in processing fruits of cognition by machine, we must replicate the tree of human thinking. However, the heuristic-cognitive nature of translation is not yet a friend of MT developers. Our logical-statistical means are not adequate to formalize and integrate mental processes into an MT paradigm. To resolve this paradox we need metalanguage tools.

Back in the dawn of MT era, I had a dream of a *'dictionary'* that, "given the rules of use, anyone can generate semantically, lexically and grammatically correct sentences in another language'. (Gorbis, 1976). This is one possible interpretation of this vision..

The assumption of most MT efforts is that correlating means of expressing our *thoughts* in one coda[1] as strings of words or signs in another is a lexical-grammatical task. This assumption is incorrect. Just because the end product of translation is a *text*; identifying *correlations* and establishing *equivalency* are not linguistic procedures.

The basis for translating *'texts'* of any length is *competence* in *universally shared reality* previously defined as "*a sum total of all concepts and their instrumental and qualitative characteristics shared (potentially) by all humans regardless of when and where they happen to communicate."* (Gorbis 2006 (a)). Possession of databases of 'word-forms', and rules of organization is a condition necessary, but insufficient for translation purposes.

Fundamentally, *language is an instrument of mediating between individual representations (of referents) in shared reality*. (ibid). Our starting point is that translation protocols are a part of Universal Cognitive Domain (UCD) - a meta-language, meta-culture and meta-time abstraction of reality regardless of the many coda through which mankind thinks and communicates. Any referent in UCD is an aggregate of all particular (and peculiar) *forms of its individual cognitive presence*. This is what makes all acts of communication (even coughing cautiously) possible and recognizable.

Communication (translation) is *teleological*; it expresses a finite network of human needs (tasks) resulting in some specific *rearrangement* of or *change* in the structure of cognitive representation of any referent for a (potentially) unlimited audience. Such needs and tasks are serviced by (metalinguistic) *cognitive algorithms*. Because these cognitive tools are the same for all, it is irrelevant that recipients have an incomplete or a different representation of any referent. Individual competence differs, but human cognitive machinery is additive and elastic. It enables comprehension (internal translation) of the technical, the emotional, the metaphysical, the surreal and the fabulous.

Any communication within and about our *shared reality* requires recipients to possess a *minimally uniform map of its structure.* One may have never heard of a "Zerasian God" or "Kenetian deity'[2] but we comprehend these narratives by virtue of *cognitive representation* of 'deity' in our Universal cognitive domain's 'knowledge library' such as patterns *of what a referent 'deity' can do, what can be done to it and what attributes 'deity' may display*. Sci-fi writers took advantage of this phenomena to lift a thought-image of a space-probe in search of its "Creator", to a level where universal *algorithms* establish our comprehension.

[1] "*Coda*" refers to all natural languages, including sign language and includes all verbal and non-verbal formats.
[2] Both names invented to make this point.

Because 'shared reality' representation in the cognitive domain is 'all-inclusive', it is different from 'objective reality' defined by logic and facts.

The obstacles of formalizing human competence tempt MT workers to dig deeper into data-mining of natural languages; with statistical-probabilistic excavators. A few entrenched perceptions hamper MT movement from linguistic mining into mapping of our cognitive realm.

The belief that most speech is produced through *ad-hock stringing of words* (with some use of cliches) means that *creative* process of text production is impossible to formalize. In fact, the opposite is correct. Because all languages reflect *structure of shared reality*, communication (speech) is the most *re-patterned* activity. Text generation employs cognitive algorithms as *shortcut paths* from concrete to abstract and back. Unnoticeable to speaker (but relied on by recipients) these heuristic paths are paved with *prefabricated blocks* - linguistic correlates of a "*units of thought'* Recognition, correlation, establishment of equivalency of expressions employs powerful short-cuts st*rategies* of choosing one specific *aspect* of a referent and preset means to express that *aspect*.

The threat of *word-form polysemy i*n its most poetic (and in our task misleading) view of general semanticists, warns translators that every time a *word-form* is used, it acquires a new m*eaning different from all prior uses.* Even if this 'threat' is ignored, MT still faces semantic complexity as *volumes of meaning* of word-forms point to a variety of referents and thus rarely coincide in any language-pair.
We offer the 'Universal Cognitive Domain' (UCD) as a modular platform of our Cognitive Translation Model (CTM) as a way to avoid these concerns by replicating cognitive algorithms connecting our many internal knowledge libraries.

2. 'Perfect Translator' in Universal Cognitive Domain

If we imagine a *perfect translator,* (it) possesses a *comprehensive matrix of human competence of any referent in shared reality* . To approximate a *perfect translator*, we enable text analysis and conversion within a scalable model that mimics cognitive protocols of establishing textual equivalency.

Cognitive translation is a result of arriving at textual equivalency (E) established between *text elements* in the *"*from-coda*"(A)* and *linguistic means in the "*to-coda" (B) if algorithms *of identification, categorization, and differentiation* verifiably correlated *cognitive representation of referents* in (A), and (B).

Our working proposition is that UCD contains meta-lingual protocols necessary to convert text in coda (A) to coda (B).While we recognize, following Sapir-Whorf hypothesis, that languages may parse reality in somewhat different ways, programs of parsing 'shared reality' are *universal*, that is meta-lingual and meta-cultural.

Any *object*[3] in any culture is an element of a universal cognitive map. As such, every speaker would *'know'* only some parts of what can be done with any *object*, what *it* is capable of and what criteria differentiate *it* from and which criteria group *it* with other points on the map. This operational knowledge differs from and compliments 'encyclopedic' knowledge libraries.

The UCD '*knows*' what other *objects* are capable of the same actions, treatment, assessment, valuation and use and which are not. Within this domain one may use a cell phone as a hammer, a door-stop, a weapon, a paperweight, and so on. To teach a child which objects cannot be used as "hammers", we easily connect "hammer" to cognitive universals of "purpose", "possession", "value" "ownership", "permission", or "authority"

In children, the analogue of *Universal* C*ognitive Domain* develops in stages, but what features make it functional from the get go? The answer is in the congruence of 'need' and 'circumstances'. While the computing device is incapable of having either, it can recognize when and how they are expressed through three integrated cognitive modules: *Representation*, *Selection* and *Identification.* The first one enables parsing of Cognitive Domain (UCD) into elements. The second provides strategies (algorithms) for selection of these elements by choice of their cognitive features (referent aspects). The last one defines their *cognitive coordinates* and thus provides parametric means of verifying adequate conversion of classes similar to cell phones into multilingual forms.

3. Structure and elements of the Universal Cognitive Domain

To discuss *cognitive representation of a referent* within *UCD*, we identify it as a *concepimage* – a p*otentiality of* all actions, features, capabilities, uses, conditions, values, states and impressions.

A complete Cognitive Domain model is not merely a representation of all UCD *concepimages* with all the *links*

[3] We use the word-form *'object'* as a common tag for any referent.

between them accessible through a name-tag of a word-form. As text-speech recipients we are able to recognize that in a speaker's mind each element correlates with what that speaker knows and what all recipient(s) must discern about *referent*: its features, qualities, images, associations, tactile properties and even smells. How do we handle this task?

Let us use two real examples: In the first, a 'diamond bracelet' elicits a statement of its "uselessness" because of "propensity"' to catch on delicate fabric. In simple words, the owner cannot wear it. The second event unfolds as a guest pushes away his soup plate and declares that "this chicken broth smells funny".

Both examples involve *rejection*, not as a speech act, but as an illustration that a *concepimage* of either object (the bracelet and the broth) includes "capacity to be rejected". I cannot think of many objects that do not possess this aspect, but please come up with your own examples. It seems that almost anything can be *'rejected*.' 'liked' or 'ignored'. The human reasons are many but not relevant. Recognizing that a 'perfect translator' is a machine that without knowing the reasons models a process we just used to find referents without *'rejectability'* leads to conclusion that in parsing our shared reality we can cognitively yet illogically, group different objects as those we reject, accept, like, ignore and use.

4. Point of View (POV)-Mechanism of Parsing Shared Reality

The mental experiment above underscores the role of *categorization* - a cognitive process of attributing an *aspect* to referents that identifies its membership in some cognitive class. Of course, not every object in the Universe has a feature "*to be rejected*" but from where did we obtain this criterion of belonging? This is not a physical, or encyclopedic feature of anything, but note how effortlessly we imagined and tested name-tags as belonging to the cognitive class defined by *'rejectability'*. So how do we do it?

Our understanding of 'shared reality' (as opposed to 'objective reality') includes presence of an abstract human figure whose parsing of the Universe is metalingual and all-encompassing due to universality of human needs. Parsing occurs along a fairly defined web left by mental scalpels. The same "dollar bill" or "apple" is *not 'the same'* when it belongs to different spheres of human relations and in this sense, general semanticists were correct in raising the red flag, even though the threat was misidentified and exaggerated.

We assert that: 1. There is a *finite set* of relations that any speaker can potentially enter into with any object or feature of 'shared reality', and 2. That every language provides specific means servicing that speaker's choice , and 3. That each *relational sphere* provides cognitive tools of parsing reality to focus on, express, or satisfy a certain need, purpose or task by offering specific keys to unlock cognitive classes.

Ability to identify (or choose and apply) a *criterion by which an concepimage* of a referent *is recognized as belonging to a certain relational sphere and its cognitive classes* is the fundamental feature of our model. By itself, a word-form has no cognitive existence -- *concepimages* of 'duty', 'bracelet', 'broth' , 'love', or 'rejection' are only a potentiality of *all relations* that its corresponding referents can enter into in our shared reality. To *'recognize'* a word-form we have to identify into which relational sphere the speaker placed the referent by spotlighting one of its features to differentiate between strength and value of a cell phone. Only then a word-form actualizes a *cognitive 'meaning'*.

We accomplish this task by using several programs. Cognitive *categorization* begins with the choice of criteria of selection and inclusion of an element in a class and results in identification *of the speaker's point of view*. Any Point of View (POV) is metalingual and thus any text can be considered as a sequence of POVs of all of its referents.[4] The Point of View is the major universal algorithm by which the word-form potentiality is reduced to one choice, and while it is not the only cognitive program we use, this is a key feature of the CMT model. The above can be stated in an axiomatic format:

A.1. Each referent can be grouped with a number of other referents as members of a cognitive class through choice of a common POV aspect.
A.2. This common aspect (feature) is a criterion of referent's belonging to a class.
A.3. Any referent can be defined by a number of choices and belong to more than one cognitive class.
A.4. In every coda there is at least one form (linguistic means) that actualizes and expresses the common cognitive aspect of any class.[5]

[4] All choices of any past, present or future speaker in any code are different but cognitive competence including ability to decide by which criterion to group/separate objects of the shared reality is universal.
[5] We casually create a *class* and draw from such cognitive groupings as: *things that fly, things that hurt, ways we see, ways we speak, people we hate*, *things to buy*, and so on.

5. POV as an Algorithm of Categorization, Recognition and Identification

This seemingly simple mechanism underlies our capacity to talk about a referent in *shared reality* from a great number *of vantage points.* It allows us to *group* anything with anything -- physical objects, states, images, emotions, observations, phenomena, living beings and, of course apples and oranges. But it also *differentiates*, by splitting a word-form into different concepimages of referents while referring to an *apple* as a body part, a coveted prize or a trademark.

Speech recognition, just as speech generation, begins with spotlighting one point from which to view any referent. However, this step may be insufficient for "identification" of the actual referent. For example, the *cognitive algorithm* of POV "commercial object" does not yet enable us to disambiguate "*an apple*" in "bought a fruit, a computer, or shares".

The 'meaning' of "apple" in any of its appearances can be discerned only if we recognize specific POVs that the speaker provides in a given text. They include such POVs of *apple* as a *physical object, complex object, biological object, growing object, edible object, man-made object, computing object, communication object, desired object, defective object, owned object, etc.* All POVs of one referent form its *cognitive spectrum*. To solve the 'same form-different referent' problem, each word-form in our database has to be identified by all of its POVs. A CMT module with a complete inventory of all word-form POVs still identifies cognitive classes that belong to more than one *concepimages*. While each referent is defined by its unique *spectrum,* how to differentiate them in natural speech and in the CMT model? The human solution is to analyze "context" from which we heuristically differentiate *food* from *computer.* In CMT model each POV is provided with an alpha-numeric code, it's unique address. Text analysis identifies referents by all addresses in their spectrum. If the sequence of POVs shows presence of at least two and preferably more identical codes, this points to the referent with a verified lexical meaning. But what to do, if we do not find POVs with the same address?

6. *Musketeer Classes* as Algorithms of Cognitive Identification

A working answer to these questions is found in the next step of text analysis, best explained in the axiomatic format continued from above:

A.5. A word-form in any coda identifies a number of potential referents in shared reality.
A.6. A referent *corresponds to a concepimage* in the Universal Cognitive Domain (UCD).
A.7. A *concepimage* referent can be viewed from different POVs each corresponding to a word-form in some coda and provided with a unique address in our model.
A.8. All referents of a class share the same aspect with all other referents in the class and thus the POV code identifies coordinates of each member of the class.
A.9. All POVs available to identify aspects of the referent constitute that referent's *spectrum*.[6] A spectrum of a referent can be represented as a sequence of codes of all POVs.
A.10. A spectrum of a referent is metalingual and identical regardless of the word-form by which a given referent is identified in any coda.
A.11. If word-forms in different coda have the same POV spectrum they identify the *same referent* and are potentially *equivalent* to each other.
A.12. Any referent can be *formally identified* by its criteria of belonging to cognitive classes, represented as a (mostly) *unique sequence of addresses* of all POVs of its spectrum.
Our key premise is this: if a word-form of any referent (concepimage) represented as a sum-total of all of its POVs, it is likely to be equivalent to a word-form in any coda that has the same spectrum. Addresses of all POVs in the spectrum of a referent are basic coordinates of the referent on our map of 'shared reality'.

B.1. One word-form in a coda can identify a number of referents, all POVs of these referents form a coda specific inventory of all POVs unique for the coda.
B.2. An unknown word-form can be identified by POV Inventory of its referents.
B.3. An unknown coda can be identified through unique Inventory coordinates of their word-forms.

C.1.Each POV address identifies a *cognitive class* of referents that share the same POV aspect and address.
C.2. A POV aspect identifies a complex of cognitive operations over each referent member of its class.
C.3. Reversely, each referent from a cognitive class is subject to any cognitive operation from the complex.
C.4. A cognitive operation is a response to a task-specific query addressed to the identified (or identifiable) aspect of the POV.
C.5 Each POV is serviced by a finite set of cognitive operations.

[6] Any word-form referent is a cognitive potentiality of its POV 'spectrum' available to and chosen from a individual and finite (hierarchical) representation of UCD in the cognitive domain of a given speaker ' UCD however represents an abstraction of all cognitive domains. See Gorbis, 2006(a), and Gorbis, 2006 (b)).

C.6. Cognitive operations identify concepimages of referents, differentiate concepimages of referents and link each concepimage of a referent to other concepimages.
C.7.Referents from the same POV class are serviced by the same set o*f operations identifying what a referent can or cannot do? (2) What can be done to or with a referent? (3) What kind of a referent is it?, and (4) What are the quantitative parameters of a referent?*
C.8. Any *POV aspect* defines a *hierarchical complex of cognitive operations* over *and* by *the class of referents that this POV creates*.

D.1.Each *cognitive operation is actualized through at least one verbal (or non-verbal) operant*.
D.2. An operant is a ready-made, or an ad-hock created verbal structure of a natural language.
D.3. As a correlate of a cognitive operation, each *operant* actualizes all referents of the cognitive class created by a given POV.
D.4. In any coda each POV is bundled with a set of *operants that service that POV*.
D.5. Each *operant* from the set identifies a POV aspect it services.
D.6 Each referent of a class created by a POV will have equal access to use operants from the set.
D.7. If an operant identifies a POV aspect, it identifies referents as members of the cognitive class of the POV.
D.8. Because an operant can describe (express) a complex of (permissible) cognitive operations upon concepimages of more than one class this *operant* belongs to *each member of these classes.*
D.9. When a *cognitive class is identified*, any *operant* from its shared set is (potentially) available to identify other referents from that class.
D.10. Each operant is identified by an alpha-numeric address consisting of the addresses of all POV codes it services.
D.11. *Word-forms* of a *referent* in natural languages may differ, but *the inventory of cognitive operations over any POV aspect of a referent is universal* and *so is the set of operants over or by the referent, however it may be expressed.*

Any *POV aspect* thus defines what we call *a musketeer class,* where the *'one for all and all for one'* banner means that all (or most) *operants* are *shared* by all (or most) members of that class of referents. This interconnection of a set of language means (*operants*) with the finite complex of cognitive operations available to a speaker is what allows us to adequately differentiate any word-form such as "*duty*" or "*apple"* into different *concepimages* that might have more than one corresponding word-form in other languages.
All POVs being universal - the result of speaker's need or task purposeful attribution of a *quality, feature, activity, capacity, status, ability, value, etc* to the *concepimage of a referent*, all languages have a nearly identical rray of *operants* linked to a specific referent (concepimage) by a specific POV.

In Indo-European languages *operants* are stable verbal structures: N+V, V+N and A+N. Because of their correlation with *cognitive operations*, *operants* correspond to the basic *subject-predicate* structure of human thought. To complete most utterances, these *operants* are available in their prepositional formats, e.g. 'to take off from (where)', 'to take off with (what)', ' to place (what?) into (whose?) (what?) etc.

While discussion of this subject is outside the scope of this paper, we shall point out that these ready-made dual-nature blocks represent linguistic equivalents of what we earlier called a *'unit of thought'*. They offer themselves to a language speaker as *verbal frames* - correlates of cognitive operations *with, upon or by a referent* thus communicating a thought complete with variables of 'who', when', 'how' 'where', 'how many', 'by whom', and other elements of universal grammar.

Natural speakers can (and do) introduce new POVs of any referent, thus placing it in a new cognitive class with readily available operants. Through operant coding, we can triangulate any element in the CMT model. Through the parametric nature of POV, spectrum, and operant coding we can identify the *common* POV *aspect* of the class,our model, we can identify a specific referent, the class of referents, identify any member of the class *communicate* about any aspect of the referent, by expressing any action upon or by the referent in any language through the defined sets of operants servicing such aspect.

We refer to this capacity as *cognitive competence*. A CMT model may never fully approximate human cognitive competence, but it can be constructed to replicate it as we shall show in what follows.

7. Thought Trees in Generating and Translating Texts

Let us compare text production and text translation in humans. When a *speaker* chooses a referent it appears as a word-form but because the speaker "knows" the POV aspect that corresponds to and guides his choice of operations and thus a set of operants from which he will select his ready-made units of thought. From this starting point a speaker is operating with a certain *concepimage*. His choice of aspect from which he communicates connects the referent to the cognitive class it "belongs to" and the cognitive operations complex from which the speaker selects one or several actions *over, with or by the referent*

he needs to express, which he does by using ready-made verbal correlates in a natural language.

Once class membership is chosen, a specific set of operants - cognitive-linguistic data-base of 'shared', 'ready-made' verbal frames - becomes available and offers opportunities to fill in specific details of place, time, agent, object, and conditions, of action. Knowing (or establishing) referent membership in a *cognitive class* provides a key benefit. Once referent is validated as a member, it gets instant access to sets of linguistic databases with verbal correlates of cognitive operations over the subject-referent available to move the communication forward.

The process of translation is not simply a reversal of these steps. First, these processes work in parallel with each outcome informing and thus changing others. Secondly, the processes involved in translation rely more on *verification*, which means that translator is confronted with options that offer a number of choices, some erroneous and some *better* than the others. For CMT purposes, obtaining verification of the 'best' choice of correlation at each step is one of the major tasks. That notwithstanding, here is an outline of a step-by-step decision-making tree within our theoretical framework.

When a translator encounters a text in a "from-language", he has to discern one, some or all *operants* for a *given word-form* in the text and (hypothetically) correlate them with the cognitive operations complex. To identify *concepimage*, and the proper referent, translator must correlate the operand-operation *pair* with every POV which uses this pair to express its particular aspect.

In other words, translator has to ID all POV aspects that own all (or as many) cognitive operations established through correlation with operants in the original text. The remaining POV then identify one or several cognitive classes of the *concepimage*, and opens access to its (their) sets of shared operants. If operants from the original text belong to one POV set, the *concepimage* of the referent is clear. Otherwise, the referents must be identified through verification of 'belonging' on a class-by class basis. The translator now has to determine which word-form in the "to-language" best corresponds to which *concepimage* and which operants in the "to-language" correlate to earlier identified cognitive in the "from-language".

If CMT 'knows' the selected POV the model can identify the set. If our model can *recognize an operant* as belonging to a set, the model can identify the POV.
Thus, a separate module is needed to mimic a cognitive algorithm that enables *recognition of an operant as belonging to a unique set that services a given cognitive class*. If something is being *lifted*, *weighed, seen, observed, destroyed*, etc, our model recognizes a set that services a 'physical object" POV. Thus, this module is a program that firmly ties a set of operants to the POV. To simplify, with an appearance of *any operant* this module recognizes a POV and the cognitive class it services. The more operants appear in the text, the more means of identification and correlation are available for the purposes of translation.

Let us review this in reverse. When a speaker chooses a *POV* and applies it to a *concepimage*, a cognitive connection of 'belonging' to a certain *class* is actualized together with the cognitive-linguistic set of *operants*. Membership in any class provides a major benefit; once validated, the concepimage-member gets instant access to a preset linguistic database that actualizes a given cognitive class. Like a Russian doll, a POV of a CD element may contain *nested POVs***.** An example of nesting: "*physical object*" may nest "*living object* ", that nests "*moving object*"," "*breathing object*", "*growing object*", "*procreating object*", "*communicating object*" etc. The last class includes class of "*sound emitting object*", that nests such classes as "*speaking object*", "*singing object*" etc.

Textual ambiguity becomes a resolvable issue with establishment of a hierarchical structure of POVs with each cognitive class offering its own operant set to confirm or modify the original choice of equivalency. A differentiation program, provides verification of equivalency between general and more particular POVs (and their operant sets) within the spectrum.

8. CMT Model Propositions

In constructing our Cognitive Translation Model (CTM), we proceed from the following interdependent propositions:

Equivalency between texts in different codes can be established in a language-independent multimodular format, called Universal Cognitive Domain (UCD). Every word-form is found in Module 1, it contains a dictionaries in coda A, coda B, coda C, and so on. There is no limit on coda-dictionaries in Module 1. Each word-form in Module 1 (regardless of coda) belongs to a number of cognitive classes. Each class is constituted by choice of a specific POV and will include a number of members referents. Each word-form in Module1 (regardless of coda) is correlated with elements of Module 2. Module 2 contains a POV inventory - a list of all POVs from which any referents can be viewed by any speaker, that is they are *shared* by all speakers in every coda.[7]Each POV is assigned a unique

[7] Plural word-forms are subject to different cognitive operations over referents than the corresponding word-form

alpha-numerical code. Each referent in each coda is identified by all codes from its *POV spectrum*, Any word-form of a referent in each coda is identified by a string of all alpha-numeric codes of each POV in the spectrum of referents it spectrum. Module 1 is therefore a parametric system where a word-form in every coda is identified by unique cognitive coordinates.

Textual equivalency is a range of values from zero (no common codes) to (a value equal to) the number of POVs in a given word-form spectrum. Word-forms that share at least one POV (aspect) are considered *potentially equivalent*. The equivalency rate increases with the number of shared POVs. Word-forms that share all addresses in their spectrums will have the same cognitive coordinates and are considered *cognitively identical*. Technically, the greater the number of shared codes is found, the higher the rate of word-form equivalency. A word-form that has a lesser number of POVs is equivalent to a word-form in another coda that has a larger number of POVs but the reverse is not true.
In Module 3. each POV from Module 2 is linked to *a finite set* of proprietary linguistic *operants* it owns in *every coda* and each such set is identified by its alpha-numeric address.

Each *set of operants* actualizes (services) a cognitive *aspect* revealed by a given POV and consists of two main and two auxiliary databases in each coda. Main subset 1 contains all linguistic forms found in a given coda that express all operations (actions) a referent can perform when it is viewed from a particular POV.[8] Main subset 2 contains all linguistic forms found in a given coda that express all operations (actions) that can be performed over a referent when it is viewed from a particular POV[9]. Auxiliary subset *1* contains all linguistic forms found in a given coda that express all qualitative features of a referent when viewed from a particular POV[10]. Auxiliary subset *2* contains all linguistic forms found in a given coda that express all quantitative features of a referent when viewed from a particular POV[11].The alpha-numeric address of each POV is the common search and linkage criteria of the CTM. Each address links its POV aspect to a class of referents and to its proprietary set of operants.

Let's now review this in reverse;

Any element from any *set of operants* is a *cognitive operation*. n *operant* can service a number of POVs. An *operant* thus can service a number of cognitive classes defined by (an aspect of) each POV. Each *operant* is assigned a unique identifier code expressed as a sequence of addresses of all POVs using the operant. An operant in one coda is equivalent to an operant in another coda if they have the same addresses or identifier codes. An operant that has a smaller identifier code is equivalent to an operant that has a larger sequence but the reverse is not always true.

An operation identifies all referents in a class and thus can identify any referent. An operant potentially identifies all referents in all classes it services. To differentiate these classes, a referent in a given text that is serviced by more than one operant from the same set defines the class and its POV. Each UCD link of 'concepimage - referents - word-forms' have (potential) access to all operants of all of its POVs. Identification of an operant as belonging to a particular POV -- identifies a class of cognitive members of that class. Identification of an operant as belonging to a particular POV identifies all referent members of the class. A word-form of a referent from the class identified by a common POV can potentially 'access' and use operants of all classes in the POV spectrum of that referent. All members of the class created by the shared POV are linked to all operants proprietary to the POV.

9. CMT Module Connections and Some Basic Search Protocols

CMT provides the following protocols of search identification and recognition:
1. CTM Modules are interconnected by links searchable by (from) any element of a link: 'word-form (in *'from-coda'*)-- *concepimage* -- *referent* -- POV spectrum -- POV aspect cognitive class -- set of operants (in *'from-coda'*) -- operant (*in 'from-coda'*) -- operant (in *'to-coda'*) - set of operants (in *'to-coda'*)
2. CMT elements that are linked to (share) the same POV spectrum are *translational correlates* .
3. CMT elements that are linked to another element by at least one operant are *potential equivalents*.
4. CMT elements identified as *translational correlates* that are linked to (share) all or most operants of a given POV set are *cognitive-linguistic equivalents*.

11. Concluding Remarks. The structure of the CMT model offered here is modular. The module-building tasks must be accomplished for each language we wish to place in the CMT. This is work in progress. Presently we are compiling the English language data base of referents and their operants and identifying the POV they serve. This

in the singular and plurals and thus have different POV spectrums than a singular word-form.
[8] Corresponds to Indo-European form of N+V
[9] Corresponds to Indo-European form of V+N
[10] Corresponds to Indo-European form of A+N or N+N expressing state, condition or attribute of the referent.
[11] Quantity expressed in a numerical or non-specific measure format.

process is incremental and time consuming. If we are to enjoy the benefits of translation without human mediation but with formalized cognitive features, there is no substitute for
significant human effort which this paper hopes to generate.

References

Gorbis, Boris "*Psycholinguistic Volume of Meaning*" in "Methods of Foreign Language Acquisition: Issues and Developments" Kiev, (1972) (original in Ukrainian)

Gorbis, Boris "*Psycholinguistics and Generative Lexicography: Preliminary Description of*" in "Translator's Journal" "Tetradi Perevodchika" issue 14, "Foreign Relations Publishing House Moscow, (1977) (original in Russian)

Gorbis, Boris "*A Primitive Model of Metalanguage for Universal Grammar*". Proceedings of MLMTA Conference, *MLMTA,* page 39-45. CSREA Press, (2005)

Gorbis, Boris (a) "*The COG: Making Sentences from Concepts*" In 'Proceedings of the 2006 International Conference on Machine Learning; Models, Technologies & Applications, MLMTA 2006, Las Vegas, Nevada, USA, June 26-29, (2006)

Gorbis, Boris (b) "*Borrowing With Interest: Aspect Semantics View of Language Extension and Expansion"* in Proceedings of the 2006 International Conference on Intelligent Linguistic Technologies, Hamid R. Arabnia, Elena B. Kozerenko, Sebastian Shaumyan, editors, Las Vegas, June 26-29, (2006).

Gorbis, Boris *"Cognitive Dictionary: a Representation of Shared Reality"* in the Proceedings of the 50th Annual Meeting of the ISSS, pages 50-56, CSREA Press, 2006.

Paul Kay, Willett Kempton (1984): What is the Sapir-Whorf hypothesis?" in American Anthropologist 86, pp. 65-79.

Conceptual Business Process Structuring by Extracting Knowledge from Natural Language Texts

O. Zolotarev[1], **M. Charnine**[2], **and A. Matskevich**[2]
[1]Russian New University, Moscow, Russia
[2]Institute for Informatics Problems, Russian Academy of Sciences, Moscow

Abstract - *This article discusses methods of constructing a formalized structure of a subject domain based on analysis of natural language texts, including discovering objects, their properties and related actions, followed by discovering business processes specific to the subject domain and the formation of thesaurus and business processes of the subject domain. At the same time the thesaurus can be changed based on the results of text analysis. Implementation of this approach involves automatic identification of objects mentioned in different documents, determining their properties and relationships, as well as the construction of formalized structure of subject domain by automatic extraction information from natural language texts. The article describes methods of primary text classification in association with their particular subject domain, allowing substantially reduce the number of irrelevant documents for consideration*.

Keywords: Business models, semantic networks, fragments of knowledge, objects, processes, thesaurus, Big Data

1 Introduction

This article describes the extraction of objects and business processes from the natural language texts. To develop methods for constructing models of business processes in the field of business prose assumed a detailed study of the subject domain, researching of modeling standards used as in the implementation projects of enterprise resources planning (ERP) systems as well as in projects of improving the efficiency of enterprises. During the investigation of the company dynamic and static domain models are build. This paper describes the methods of selection of objects and processes from natural language texts based on the analysis of a large volume of documents, so-called Big Data. This approach is based on development and maintenance of thesaurus of objects and processes.

One of the most labor-intensive and time-consuming phases during the implementation of enterprise resources planning systems is the initial inspection of the enterprise. Data collection methods are quite diverse. The phase of analyzing documents and subsequent construction of formalized structures of business processes are very time-consuming. The most characteristic/specific properties of the subject domain are extracted from the text documents. Automated processing of texts allows substantially reduce the costs of labor resources during the discovering information from texts as well as in formation of generalized knowledge structures inherent to the specific subject domain. Currently, various methods of natural texts processing are widely used for formalization of domain knowledge and for creation of business process models.

2 Modern investigations in the field

Syntactical analysis of the business model often reveals unused, useless functions and relationships. A more careful analysis reveals redundant and inefficient functions. One indication of the inefficiently used connections or functions is the absence of feedback loops. There is research project by Chapparro et al. (2012), which offers a selection of structured templates for extracting business rules. Business knowledge are extracted from a variety of sources - databases, structured documents, definitions, etc.

Putrycz and Kark (2007, 2008) describe discovering of business rules from the analysis of natural language texts in the form of <condition> <Action>.

Kalsing (2010) and Nascimento (2012) propose a technique of extraction of business processes from legacy information systems by identifying business rules.

In the Rocket AeroText system the information is extracted from natural language texts based on analysis of large volumes of texts in order to discover business processes. Currently, the model of semantic vector space based on distributive hypothesis [7, 8, 4] is widely used for the analysis and processing of large volumes of natural language texts including for the automated construction of thesauri and classification systems.

Sebastyan Pado from University of Saarland (Germany), and Mirella Lapata (UK) consider construction of semantic spaces based on traditional vector models with regard to syntactic relations. Semantic properties of the words are presented in the frequency matrix, each row of

the matrix corresponds to a unique target word and each column corresponds to a linguistic context. Semantic information is extracted from a large volume of texts based on context analysis of the word. The word is considered as a point in a multi-dimensional semantic space. Semantic similarity between words is calculated based on the proximity between points of semantic space using metrics. Semantic similarity analysis is performed based on statistical methods with the calculation of the frequency of close points in the text of the semantic space. The authors use different metrics for similarity analysis: Euclidean, Jaccard's, Kullback-Leibler and others. The study concludes that context of a word is very important for discovering of lexical and semantic relations of the word.

3 Creating a model of enterprise management using analysis of domain texts

Currently there is no single definition of a subject domain. A subject domain is considered primarily from the point of view of solved tasks. Creation of a mathematical model for a subject domain with a large number of parameters is a very difficult task. The domain model should adequately reflect all the processes taking place in it. To create the domain structure it is necessary not only detailed information about the objects used in it, but also the description of relationships between objects, as well as the incorporation of external relations and restrictions. The domain model can be described as a collection of objects of the real world with the whole set of attributes and relationships, taking into account the dynamics of changes of the domain. The domain model should adequately reflect reality.

Competitiveness of any enterprise in modern society depends on choosing the right development strategy, also depends on improving the structure of enterprise management and optimization of business processes of the enterprise. Implementation of ERP (Enterprise Resource Planning) systems today is impossible without a thorough analysis of the company. Otherwise, the money spent on the implementation can be wasted. It is known that mistakes made early in the design of information systems, cost quite expensive in the later stages of implementation. One of the most time-consuming steps in the analysis of the company is the process of separation, formalization and optimization the business processes of the enterprise.

In this situation it is important to have some software tools to extract information from the documents in order to facilitate the formation of functional and data structure of the enterprise. Information required for the formation of functional structures and data structures can be extracted from documents - reports, documents, letters, business process descriptions, instructions, records, etc. These documents can be specially prepared and formalized, or can be presented as informal descriptions in natural language.

4 Statistical and classification methods used for processing of natural language texts

The phase of the company survey during the project implementation and subsequent analysis of the data collected, their structuring and formalization requires great investment of time and resources. Huge volumes of information collected has to be processed manually by a qualified technicians and expensive consultants. The model is built. It reflects current business processes of the enterprise, the so-called model "as is". The next stage is to perform analysis of this model. This model is based on the examination of documents (reports, instructions for working with legacy systems, the provisions of the enterprise, orders and so on), results of questionnaires and interviews with employees of the enterprise, and other sources.

Model parsing reveals unused, useless functions and relationships. A more careful analysis reveals redundant and inefficient functions. One the indication of inefficiently used functions is the absence of feedback management. Model-building process "as is" is very long, with a lots of iterations.

As part of the process of extracting information from natural language texts are formation of concepts and processes thesauruses. These concepts and processes thesauruses are of two types: general - to describe the common elements that are not tied to a particular domain and domain-specific taking into account the features of objects and processes specific to a given subject region. The thesauruses are built taking into account the dominance hierarchy.

During the work on the project, it is expected to use the results of other projects led by E. Kozerenko in analytic word processing and retrieval of information objects and their relationships.

This team developed a series of linguistic processors based on the unit extended semantic networks [11], including the fastest Semantix linguistic processor with built-in tone analysis texts [3].

For optimum performance, the language processor requires a fairly complete linguistic knowledge base containing linguistic and subject knowledge, which can be represented as thesauri and dictionaries. For creating linguistic knowledge base are used the results of the core team led by M. Charnine that described below, in according with the Grant RFBR ("The methods for automatic creation of associative portraits of subject domains on the basis of big natural language texts for knowledge extraction systems") in the field of searching, classifying, statistical and analytical processing of natural language texts (NL-texts) of very large volumes (Big Data) [4].

M. Charnine and N. Somin in the "Conceptual text generation based on key phrases" [5] described the improved method of NL-texts semantic search from the Internet with the aim of directional extracting the encyclopedic information from texts. They used not only their own achievements such as concepts key encyclopedia Keywen but well-known search engines like Google, Yandex, Yahoo , etc. and library catalogs and electronic stores as well. The collected texts handled by the statistical analysis module for the formation of the subject domain association portrait (SDAP) containing associations between specific term of the subject domain, keywords and relevant phrase. Formation algorithm prototype version of SDAP [4] tested on the material of some subject domains including business process modeling. Relevant texts founded by the algorithm SDAP were processed with the linguistic processor to build subject dictionaries and thesauri.

SDAP formation algorithm [4] supplemented with the classification methods described below, allowing more accurately select relevant texts of the subject domain.

In the article "Identification of interests of Internet users based on associative approach " [6] , the author M. Sharnin, are discussed questions of definition and keyword extraction from natural language texts to conform these words to a certain category. Each keyword is assigned a weight, which is calculated taking into account the frequency of occurrence of keywords in the texts of some category. Dedicated keywords serve as the foundation for building classifiers. Category in classifiers are represented as a tuple:

< Primary keyword , category , weight >

There are many different measures of the degree to which two facts co-occur. We use in our work Pointwise Mutual Information (PMI) [9, 10], presented by formula(1) as follows:

weight(keyword) = weight(category) =

Log(p(keyword&category)/(p(keyword)p(category)))

Formula (1)

Here, p(keyword&category) is the probability of co-occurrence of two facts: keyword exists in the document and document belongs to category. If keyword and category-i are statistically independent, then the probability that they co-occur is given by the product p(keyword)p(category). If they are not independent, and they have a tendency to co-occur, then p(keyword&category) will be greater than p(keyword)p(category). Therefore the ratio between p(keyword&category) and p(keyword)p(category) is a measure of the degree of statistical dependence between keyword and category. The Log of this ratio is the amount of information that we acquire about the presence of keyword when we observe document of category. Since the equation is symmetrical, it is also the amount of information that we acquire about belonging the document to the category when we observe keyword, which explains the term mutual information.

The formula (1) allows us to calculate weight of keyword associated with a given category if we know the probability of the presence of this keyword in the documents of the category.

Using of associative links between the terms for the classification purposes [6] and associative term relations with categories, calculated by the formula (1), has led to the following results:

• the complex of texts classification methods was developed; this complex uses the unit of associative links, allowing to classify the texts do not contain terms of training samples;

• the developed software implements the proposed methods of texts classification used for the selection of relevant texts on the topic of the subject domain;

• the volume of relevant NL-texts was substantially increased with the help of texts automated partitioning on the subject domain;

• the important phrases dictionaries and domain dictionaries were extended, accumulated by processing large amounts of data from the Internet.

As a result, with the help of noted above methods and programs the domain-specific knowledge bases are formed, which include thesauri (classifiers) objects and

processes, associative portraits with the specified domain knowledge processing methods for creating of domain models, fragments of semantic networks describing the relationships between processes.

5 Building a domain model based on the "process" approach

As a part of the description of models the concept of "process" and "sub-process" are regarded as interchangeable. These concepts differ only by considering different levels of processes (process - a top level or parent sub-process - subordinate level - a descendant).

At the stage of morphological analysis of texts formalized structure of sentences is built. Objects, their properties, links between objects and actions they take part in are distinguished. At the next phase phrases are formed, statistical analysis of occurrence of terms, phrases, actions in the text is spent. At the next step, the identification of the selected items is done for the entire document; associative portrait of domain-based elements extracted from the document is constructed. As a result of analysis of a representative body of texts the degree of semantic proximity of different documents is determined. So semantically similar documents relate to a particular subject area. On a given subject domain formed document library. Based on statistical analysis of documents portraits the semantically similar processes and objects are defined. Objects and processes are represented as fragments of semantic network. Consider for example the formation of fragments of a semantic network based on the analysis of natural language text with the selection of objects and processes.

The fragments of semantic network are built on the base of the following below text. The selected part of the text is sufficient for the formation of presented fragments of semantic network describing the process of repairing the vehicle:

"The car is repaired based on the order in accordance with the regulations. Manager on order examines vehicle. Because of examination is built a list of faults. According to the list of faults mechanic replaces parts. Repaired car is transferred to the owner."

To process this text with the help of the linguistic processor are used special domain vocabularies of terms included in the description of business processes, and for each term are specified its possible role (input, management, participant/mechanism, exit). For example, let's take a number of terms of subject dictionaries representing different elements of business processes:

• name of business processes (repair, inspection, parts replacement, manufacturing of details, holding invoice, delivery);

• input (order, the list of failures, procurement, invoice, consignment note);

• management (regulations, drawing, regulation, invoice);

• participant/mechanism (master mechanic, manufactures Turner, accountant, driver);

• output (repared auto, fault list, item, invoice, goods).

After the processing of text with the help of the linguistic processor using a thesaurus and domain-specific dictionaries the following fragments of semantic network are constructed:

Repair (process _) (1)

Order (Object, Input Repair) (2)

Repared_car (Object, Output, Repair) (3)

Regulation (Object, Control, Repair) (4)

Manager (Object, Member, Repair) (5)

Manager (Object, Member, Repair) (6)

Consider the fragment (1).

Dedicated process "... repaired ..." is transformed into a verbal noun "Repair."

Zero place in this fragment of a semantic network is facing the bracket element "Repair", it is extracted from the text process. The first argument indicates the type of element - the object or process. On the second argument place in the fragment, describing the process, placed the top-level process. Empty 2nd argument place in the process means that the process is a process of top-level, and he has no parent node.

On the second argument place in the fragment describing the object (2), fixed object type. If the 2nd argument

place signified (not empty), it indicates the process that owns this object.

Process models can be spitted (decomposed) to sub-processes. The process of "Repare" is split into two sub-processes "Inspection" and "Change_Parts." As a result will be constructed the following fragments of the semantic network:

Inspection (Process, Repair) (7)

Change_Parts (Process, Repair) (8)

Order (Object, Inspection) (9)

List of faults (Object. Input, Inspection) (10)

List of faults (Object, Input, Change_Parts) (11)

Repaired auto (Object, Output, Change_Parts) (12)

To be simple, we consider only the objects related to the inputs and outputs of sub-processes.

We can see that sub-processes (7) - (8) are executed sequentially because the inputs and outputs of the processes coincide. Fragments (9) - (12) are constructed by analogy with the already described above.

There are also some fragments that define connections between processes of the same level (in our case - one fragment):

Link (Inspection, Change Parts) (13)

This fragment defines a sequence of sub-processes (7) and (8). This information is not redundant, because the process of determining the sequence of operations based on inputs and outputs is quite complicated and can be ambiguous.

Formation of a knowledge base (semantic network) is based on the coincidence of the parent process input and output information - on the upper level with the input of the first sub-process and output information for the last sub-process of decomposition. As a result of the analysis of natural language texts and automatic construction of a semantic network formed thesauri of objects and processes, as well as the knowledge base for a particular domain.

6 Conclusions

This paper presents a conceptual model of a system for extracting objects and business processes from natural language texts, based on the formation of thesauri and knowledge base of objects and business processes. The processing of natural language texts that describes a certain subject domain, allows us to build sets of fragments of semantic network, which form the knowledge base of the domain. Subsequent processing of incoming documents will not only allow recognize already selected objects, but also to determine their properties, relationships to other objects of the domain. A large amount of dedicated and verified knowledge fragments is able to minimize the number of errors in the development of functional structure and data structure for a given subject domain. In addition, the analysis of constructed knowledge base gives us the opportunity to create meta-knowledge that will hold general knowledge, i.e. knowledge about the features of the structure of the domain.

The described approach significantly reduces the development costs and allows optimize both the functional structure and the data structure of the enterprise. As a result it is possible to significantly reduce the costs and duration of the preparatory phase of the development of company models.

By using this method, we can greatly speed up the analysis of the domain and creating models of business processes, that is one of the first steps to modernizing the organization, improving the manageability and efficiency of its operations. This can significantly shorten the process of optimization of business processes and further restructuring of its operations.

7 Acknowledgements

This work is supported by the Russian Foundation for Basic Research, grant #13-07-00272 “The methods for automatic creation of associative portraits of subject domains on the basis of big natural language texts for knowledge extraction systems”.

8 References

[1] O. Zolotarev. “Control in implementation projects of distributed Enterprise Resource Planning Systems”. Bulletin of the Russian New University, Moscow, RosNOU, 2012.

[2] O. Zolotarev. “Innovative solutions in the formation of functional structure of the subject domain”. Bulletin of the Russian New University, Moscow, RosNOU, 2013.

[3] I. Kuznetsov, E. Kozerenko, M. Charnine. “Technological peculiarity of knowledge extraction for logical-analytical systems”. Proceedings of ICAI’12, WORLDCOMP’12, 2012, Las Vegas, Nevada, USA, CRSEA Press, USA.

[4] M. Charnine, N. Somin, I. Kuznetsov, U. Morozova. "Statistical mechanisms of the subject domains associative portraits formation on the basis of big natural language texts for the systems of knowledge extraction". Informatics and Applications scientific journal, 2013, v.7, №2, IPI RAN, Moscow, pp.92–99.

[5] M. Charnine, N. Somin, V. Nikolaev. "Conceptual text generation based on key phrases". Proceedings of the 2014 International Conference on Artificial Intelligence (ICAI 2014), WORLDCOMP'14, July 21-24, 2014, Las Vegas, Nevada, USA, CSREA Press.

[6] M. Charnine, A. Petrov, I. Kuznezov. "Association-based identification of Internet user interests". Proceedings of the 2013 International Conference on Artificial Intelligence (ICAI 2013), v.I, WORLDCOMP'13, July 22-25, 2013, Las Vegas, Nevada, USA, CSREA Press, pp.77-81.

[7] A. Lenci. "Distributional semantics in linguistic and cognitive research". Rivista di Linguistica, 1, 2008, pp.1-30.

[8] M.Baroni, A.Lenci. "Distributional Memory: A General Framework for Corpus-Based Semantics". Computational Linguistics. V.36, Issue 4, 2010, pp. 673-721.

[9] K.W. Church, P. Hanks. "Word association norms, mutual information and lexicography". Proceedings of the 27th Annual Conference of the Association of Computational Linguistics, 1989, pp. 76-83.

[10] K.W. Church, W. Gale, P. Hanks, D. Hindle. "Using statistics in lexical analysis". In: Uri Zernik (ed.), "Lexical acquisition: Exploiting on-line resources to build a lexicon", New Jersey, Lawrence Erlbaum, 1991, 115-164.

[11] Web site "Knowledge extraction for Analytical Systems": http://ipiranlogos.com/english/.

[12] Michael Charnine. "Keywen Automated Writing Tools". Booktango, USA, 2013, ISBN 978-1-46892-205-9.

SESSION

NATURAL LANGUAGE PROCESSING, TOPIC DETECTION, LINGUISTIC BASED METHODS, HUMAN-MACHINE INTERACTION

Chair(s)

TBA

Representing Syntactic-Semantic Knowledge from English Texts

Richard Guidry Jr. and Jianhua Chen
Division of Computer Science and Engineering
School of EECS, Louisiana State University
Baton Rouge, LA 70803, USA

Abstract - *We present a unified representational framework LSeN (Language to Semantic Network) for representing syntactic-semantic knowledge extracted from natural language (e.g. English) texts. Our representation is a variant of the standard semantic networks with some important distinctive features that facilitate bridging the interface between syntactic and semantic knowledge using automatic computational tools. We show the basic constructs in our representation and briefly describe the developed tool for automatically building the knowledge structure in LSeN from parsed English sentences. Our representational framework and the associated tool establish a good platform for further studies in automatic knowledge extraction from texts.*

Keywords: Semantic Network, Syntactic and Semantic Knowledge, Natural Language Processing

1 Introduction

Natural Language Processing (NLP) has achieved great progress in the last two decades. With the rapid development of research in this area, nowadays we have various computational tools for extracting knowledge from natural language texts represented in one form or the other. Developing a suitable representation formalism for syntactic/semantic knowledge from texts and subsequently designing an efficient computational tool for actually extracting knowledge into the desired form (structure) are important research problems tackled by numerous researchers in the NLP field.

In natural language processing tasks (such as language understanding, query-answering and text summarization), we need first perform syntactical processing of the natural language texts through parts-of-speech tagging and syntactical parsing. Then semantic processing is performed on the syntax parse trees to obtain semantic knowledge. Clearly there is a close connection between the syntactic knowledge and the subsequent semantic knowledge, as both could be represented as graphs/trees (parse trees) of some sort, with nodes representing words and edges labelled by dependencies/relationships between nodes, as remarked by some researcher in the "research gate" social media website [12].

We believe that a representational framework facilitating the incorporation of both syntactical and semantic knowledge is beneficial to the natural language understanding task. Syntactic knowledge often provides very good clues to the semantic aspects of sentences. Many shallow semantic processing techniques such as semantic role-labeling, conceptual dependency analysis, etc., make extensive use of the close connection between syntactic and semantic knowledge. The availability of various syntactic parsers such as the Link grammar parser [13], the Stanford Parser [14], and the Malt Parser[15] made it much easier to exploit the results of syntactic parsing in building shallow semantic models.

In this paper we present a framework for representing syntactic-semantic knowledge from English text with the objective to make it easier to utilize the close connection between syntactic and semantic knowledge in NLP tasks. Based on this model, we have constructed a computational tool for automatically building a semantic-network-like structure from English sentences. This forms a good basis for performing query-answering with the constructed semantic structures.

What we present is a generic language knowledge interface which is domain independent. In the arena of computer systems handling natural human language, we often meet challenges that require extensive domain-specific knowledge in order to achieve high level of performance. The current systems for NLP seem to fall into two categories. The first category is domain specific. These cater specifically to a small domain and work well in that domain only. The second category is systems that require priming with hand fashioned databases of knowledge. These systems often require such an amount of perquisite data that it is difficult to produce a sufficient base from which to get the desired results. We propose a system which does not deal in the restrictiveness of these categories. The system could operate without hand fed data, and operate in any domain. The proposed system representation is outlined (Section 3) after the discussions in Section 2 on relevant works and existing technologies.

2 Background and Relevant Works

Natural Language Processing (NLP) is the field which lies at the junction of Linguistics and Computer Science, also commonly known as computational linguistics. The field had its beginnings in the 1930's with the creation of mathematical and logical formalisms, which were the predecessors to

current linguistics theories, and also modern computer languages. NLP is concerned with the interactions between computers and human (natural) languages. As such, NLP is related to the area of human–computer interaction. Many challenges in NLP involve natural language understanding, that is, enabling computers to derive meaning from human or natural language input, and others involve natural language generation. Other major tasks in NLP include automatic text summarization, discourse analysis, machine translation, question-answering, etc.

2.1 Stages of Language Processing

The processing of natural language can be divided into stages as seen here [5].

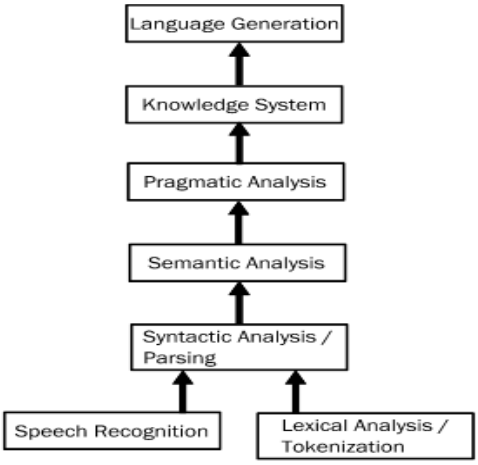


Fig. 1: Stages in NLP

Lexical analysis and tokenization encapsulate linguistic elements into tokens, usually words. These tokens are provided in a stream to the next stage. For audio input, a speech recognizer would provide the token stream. The syntactic analyzer or parser takes the token stream, and derives a hierarchical syntax tree. Parsers may be based on a grammar specification, which generates a parse table; or they may be based on statistical methods as the Maximum Entropy Markov Model (MEMM) parser [11].

Semantic analysis is a broad step, where information is extracted from the syntax tree. The extracted information may then be analyzed pragmatically for application of context. Between semantic and pragmatic analyses, tasks need to be performed such as resolving references, disambiguating words, and resolving the scope of modifiers. Through interaction with the knowledge system, the result of semantic and pragmatic analyses is encoded into some body of information that can be consulted at later times. If an inquiry is made, it will be necessary to move to the language generation stage.

2.2 Graph/Semantic network-based Approach for Knowledge Representation in NLP

The focus of this paper is on knowledge representation for syntactic and semantic knowledge which are essential for natural language understanding and question-answering. The approach taken in the representation framework is graph-based, but it has close connections with the symbolic/logic-based approach to natural language knowledge representation. We can see three distinct approaches to NLP tasks: The Symbolic/logic-based approach, The Graph/Semantic network-based approach, and The Empirical approach. Symbolic NLP deals with the more traditional logic-based methodologies, where semantic knowledge is represented by a set of logic formulas, and algorithms operate on the symbols to perform inference and answer queries. Empirical NLP uses corpora of language data which are processed by a system to extract syntactic/semantic knowledge which would then be used for various NLP tasks such as Parts of Speech Tagging and semantic similarity computation. The graph/semantic network-based approach represents the syntactic/semantic knowledge in natural languages by various graphs or semantic networks. Algorithms for graph processing (traversal, path-finding, homomorphism, etc.) are utilized to perform inference and query-answering. Our proposed representational framework is a graph-based approach and so we discuss below in some detail the relevant existing works in graph/semantic network-based knowledge representation for NLP.

John Sowa [19] in 1976 proposed the Conceptual Graph (CG) as a common knowledge representational framework. Conceptual graphs are closely related to Pierce's Existential Graphs (EGs) [25] which were developed by Pierce in 1896 as a simple, readable, expressive graphic notation for First Order Logic (FOL). Conceptual Graphs combine a logical foundation based on EGs with the structural features of Semantic Networks [21] in providing an expressive and formally grounded formalism for representing knowledge [24]. In Conceptual graphs, concepts are represented by rectangles and relations are represented by ellipses (ovals). Links (directed or undirected) indicate the connection between concept nodes and relation nodes. The Conceptual Graph Interchange Format (CGIF) is a linearization of CGs specified by the ISO/IEC standard 24707 for Common Logic (CL). CGIF essentially specifies the equivalent common logic expression corresponding to a conceptual graph.

As an example, the sentence "If a farmer owns a donkey, then he beats it" is represented in CG as in the following Figure 2. In Figure 2, the concepts "Farmer" and "Donkey" are drawn as rectangles. The "Owns" relationship between "Farmer" and "Donkey" is shown by an ellipse with links, indicating that "Farmer" is the subject of the "Owns" relationship and "Donkey" is the object in that relationship. Similarly for the "Beats" relationship.

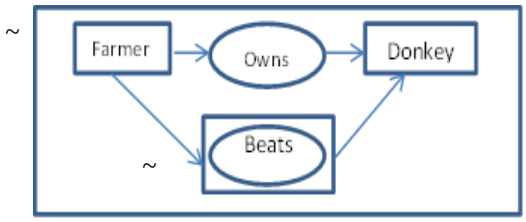


Fig. 2 The Conceptual Graph for the "farmer-donkey" sentence from [24]. Here "~" denotes negation.

Conceptual graphs are closely related to semantic networks [21] that have been widely used in Artificial Intelligence and NLP research. According to Sowa [20], "A semantic network or net is a graph structure for representing knowledge in patterns of interconnected nodes and arcs. Computer implementations of semantic networks were first developed for artificial intelligence and machine translation, but earlier versions have long been used in philosophy, psychology, and linguistics." Semantic networks are intended for a graphical and declarative representation of knowledge, and should support automated reasoning systems to perform inference on that knowledge. Some semantic networks are defined quite informally for the sake of human cognitive purposed whereas some do have formally specified semantics - and intended for computer automatic inference. For example, the system KL-One (Knowledge Language One) [27, 28] specifies a subset of the classical First Order Logic (FOL), in which nodes represents concepts (types or instances of types) and labeled arcs represents roles (attributes) and relations. Generalization ("is-a") hierarchy can also be represented in KL-One.

The representational system we propose in this paper is also a semantic network. Our proposal is different from Sowa's conceptual graphs in several ways. Our system is designed with the intension to work with NLP tasks and not necessarily as a generic knowledge representation tool, a category CG would be classified into. CG has a formal semantic grounding in FOL (with the flexibility to be extended to handle natural languages and meta-languages) whereas we are yet to work out the semantic aspects of our system (which we believe should have a logical underpinning). We have introduced various types of nodes (Class Restriction and Derived Class, etc.) in our semantic network which seem to be quite similar to some of the graph operations (e.g., "restrict"/"un-restrict") in CG.

2.3 Approaches Addressing the Connection between Syntactic-Semantic Knowledge

Keep in mind that the key motivation of the current work is to develop a representation formalism that would facilitate the use of connections between syntactic and semantic knowledge in NLP. Similar efforts can be seen from the works in [16, 18]. In [16], the long-term project is the use of Transparent Intentional Logic (TIL) as a semantic representation of knowledge and subsequently as a transfer language in automatic machine translation. In [18], researchers focused on the mapping between syntactic and semantic knowledge, applied hyper-graph homomorphism for language comprehension and constraint satisfaction for language generation. One can also see the various works in computational linguistics and NLP using, for example, dependency grammars and conceptual dependency graphs (Schank [26]). Clearly the conceptual graphs approach could be seen as addressing this issue.

3 The Proposed Representation: the LSeN Interface

LSeN stands for Language to Semantic Network. It is our automatic conversion tool that takes an English sentence as the input and encodes it into a semantic network. It currently works with generic sentences in the present tense. The figures shown in this section are automatically generated by our system.

3.1 LSeN Encoding Structure

The LSeN encoding structure strives to represent general English language with as basic a tool set as possible. This is done in the form of a semantic network with node types and connections as follows. Classes represent a set or predicates. This includes adjectives, verbs, and nouns that are types. An adjective as a class represents all things to which that adjective applies. For verbs, any subject to which that verb applies generically is in the class. For nouns that are types, its class just represents anything that is considered that type. These classes may be related by subset links, denoted as is-a. Classes that we have words for are named as such.

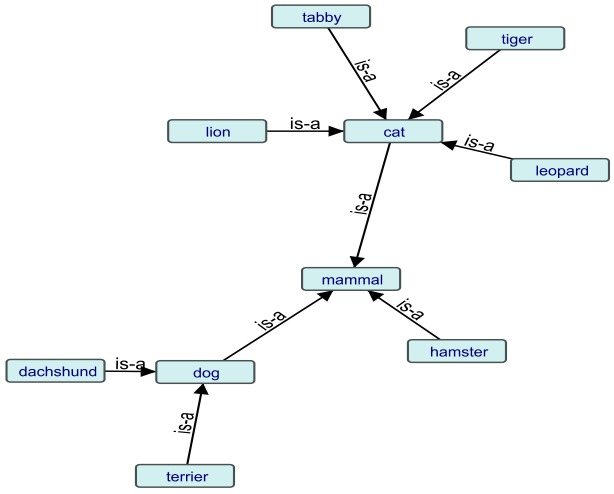


Figure 3. Basic Class Relationships

However, classes may be composed of other classes. A class which is the intersection of two other classes is said to be a restricted class, and is formed with a class restrictor (CR) node. CR nodes derive their meaning from their two out-links, labeled is-a.

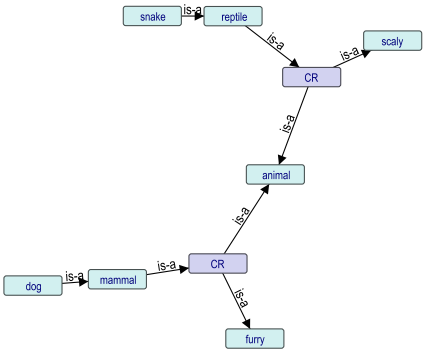


Fig. 4. Class Restriction (CR)

The above Figure 4 is obtained after processing sentences "Mammals are furry animals." as well as a few others (such as "dogs are mammals"). CR nodes are static; they are created with their two links that do not change throughout its lifetime. This does not enable us to say something new about a class represented by a CR (we call it an unqualified class because of this), so another node type is used. The derived class (DC) node is pinned to a CR, linked with an equal sign. The DC is qualified and may be assigned a superclass (an is-a link) just as named classes, which is what allows us to express new information about a CR represented class. Instances are determined objects that are members of some class. Each instance has one out-link labeled inst-of, which is linked to the most specific class node that this object is known to be in.

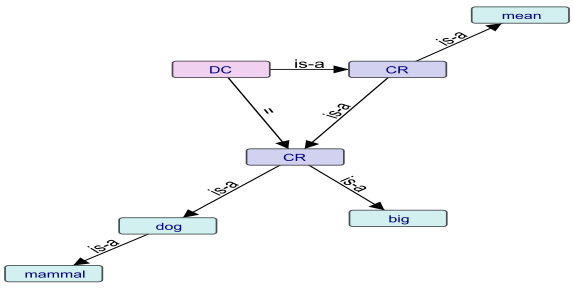


Fig. 5. Derived Classes (DC) – "big dogs are mean".

The DC Verbs are represented as classes; however, a class node that represents an intransitive verb may have instances linked to it while a class node that represents a transitive verb may not. Transitive verb classes must first have a direct object argument bound via a BIND node. BIND nodes have two out-links: pred links to the transitive verb class, and arg links to the class or instance being bound as the verb's direct object. BIND nodes represent unqualified classes, similar to CR nodes.

The information structure heretofore discussed is intended to facilitate being built from a syntactical parse. In that process, we obtain intermediate stages of representation (fragments) which start close to the parse. The fragments are then integrated together, up to a top fragment which can be merged into the network.

After a sentence has been processed into a subject and predicate fragment, we represent the notion that these concepts have been unified with a unified concept (UC) node, bearing a link to each concept; specifically the uc-a link is to the subject, and the uc-b link is to the predicate. The order is not reversible. The unified concept represents three different notions depending on the nature of subject and predicate links, and it does so in the same manner as to be. In the case of unifying two instances, it represents equality; for two classes, a subset relationship; and for instance and a class it represents membership.

We further illustrate the various constructs in the proposed semantic network by some examples. Consider the sentences "Roy is a dog. Jim is a cat. Roy likes Jim. Jim ate a mouse." Figures 6, 7 and 8 show the fragments of the semantic network constructed in processing the above sentences.

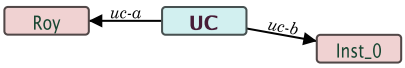


Fig. 6. "Roy is a dog" – Fragment "Roy is a ".

Fig. 7. "Roy likes Jim" – Fragment "likes Jim".

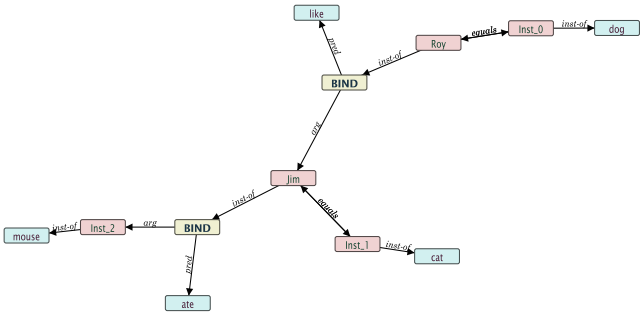


Fig. 8: Network1 obtained after the sentences.

Consider the sentence "Roy saw the mouse Jim ate" which was presented to the LSeN system after the network1 has been constructed. We see the following fragment (Fig. 9):

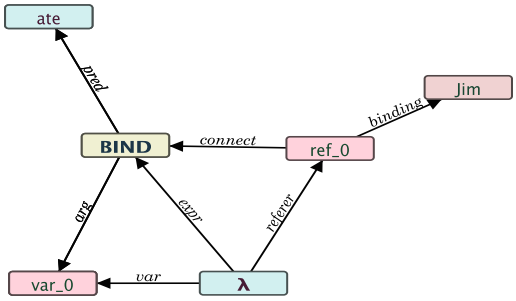


Fig. 9. Fragment "hypothetical-mouse_whom_Jim_ate"

Note that here hypothetical is used to match the right mouse instance that is seen by Roy that Jim ate. Hypothetical (inst2_mouse...) is filled in, and can be tested to see if inst_2 is that mouse. Network2 is the result semantic network after processing the last sentence "Roy saw the mouse Jim ate".

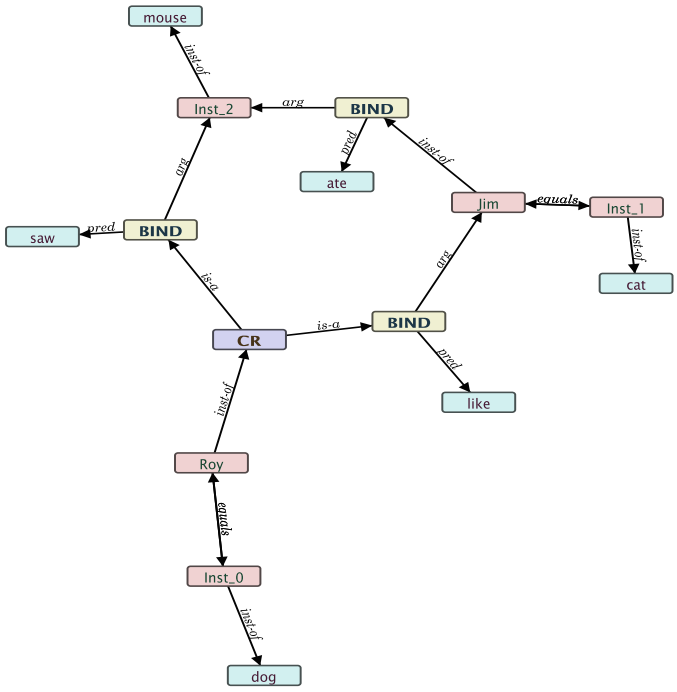


Fig. 10. Network2 obtained after "Roy saw the mouse Jim ate"

The other types of nodes thus far included are lambda (λ) nodes, eta (η) nodes, and their associated variable and reference nodes. The λ node is used to create a class from a subject and an incomplete predicate. Suppose someone says, "I know what you ate." In order to begin to represent this, we need a way of representing the class of that which you ate. The λ node can represent this by leaving a variable stand in as the object of "ate" under a bind node.

The last type of node, η, represents a hypothetical, which can be matched against the network and evaluated for truth. These can be used to match definite noun phrases to an object already in the tree if there is one. Note that the previously discussed UC node could be represented as an η , though the nodes under the η would be different in the each of the three cases discussed.

3.2 Semantic Analysis in LSeN

LSeN's chief function is to perform semantic analysis on user input in order to generate a semantic network. Before semantic analysis can begin, however, it is required that the input be processed into a syntax tree. An MEMM parser from OpenNLP [17] is used for this. Once LSeN has the syntax tree, it determines additions to the semantic network. Handling functions are recursively called on the syntax tree. Handling functions produce a result on each node in the syntax tree, and that result is a reference to a node in the semantic network. Each type of node in the syntax tree has a separate handing function.

For example, let's take the noun phrase "the dog". It is composed of two child nodes: the determiner "the", and the noun "dog". The noun phrase handler processes its children, and incorporates them together. It requests the result of handling dog. The handler for nouns processes "dog", and produces a result which is the class node for dogs. The noun phrase handler now holds a reference to the class of dogs. The determiner tells the noun phrase handler to find the unique thing that is an instance of the class denoted by the rest of the noun phrase. This has the effect of finding the unique instance of a dog in the semantic network.

The previous example demonstrates the workings of noun phrases in the system. Verb phrases are handled similarly. Transitive verbs take the result of processing the noun phrase direct object, and bind it as an argument to the verb to yield the result. Intransitive verbs are simply returned as a class. Once at the top of the sentence, the resultant semantic network nodes of handling the subject and predicate are connected. This usually involves creating a subclass relationship, except in the case of equating two instance objects.

4 Conclusions and Future Work

4.1 Contributions

In this paper we present a generic framework for representing syntactic and semantic knowledge for NLP, along with a computational system for automatically constructing the knowledge structure (semantic network) from syntactic parse trees. The proposed framework is generic (domain-independent), and facilitates the mapping between syntactic and semantic knowledge and the tool provides a good starting point for further experimental studies in various tasks of NLP such as co-reference resolution and question-answering. Although the system is in its primitive stage, building such a system is an important step towards further development.

4.2 Issues

The system is currently in its very early stages, and does have problems as it is. Currently, it would be impossible to handle relative clauses with a direct object binding, but with an unbound subject, as there is no way to represent this in the semantic network. Rather than adding another structure,

We hope to modify the BIND node concept to be more flexible, or perhaps even replace the BIND nodes. It is a goal of the system to maintain minimum complexity in the semantic network, which is why we would prefer not to add multiple types of binders. The system can also answer textually posed questions regarding the semantic net, though it has not been tested or updated in the last wave of expansion. With this being the target user interface, rather than visually inspecting the graph, it is imperative to update it to be current with the rest of the system. The MEMM parser is sometimes a bit off in generating the syntax tree. Means need to be developed to handle this gracefully. Ambiguities and failed determiner references also need graceful handling.

4.3 Future Work

Considerable future work can be done, but here are some specific and realizable tasks. It needs to be resolved how to incorporate knowledge represented by DC nodes into determiner resolution and similar tasks. For instance if Rex is a mean dog, and Mean dogs are big, then does the big dog reference Rex? Prepositional phrases need to be added. It is not a problem to represent them in the semantic network, as they would work like transitive verbs; however, they are often ambiguous in scope. This leads to a whole new problem of scope resolution, which is not completely handled in the MEMM parser. Adverbs could also be added, but with all the problems of prepositional phrases in addition to needing a semantic network representation. A whole system of handing spatial-temporal information, verb tense, modals, and consequently state vs. individual-level predicates can be added, but not before the current system is refined.

5 Reference

[1] James Allen, George Ferguson, Bradford W. Miller, Eric K. Ringger, and Teresa Sikorski Zollo. Dialogue systems: From theory to practice in TRAINS96. In Robert Dale, Hermann Moisl, and Harold Somers, editors, *A Handbook of Natural Language Processing.* Marcel Dekker, 2000.

[2] Jim Barnett, Kevin Knight, Inderjeet Mani, and Elaine Rich. Knowledge and Natural Language Processing. *Communications of ACM*, 33(8):50–71, 1990.

[3] Adam L. Berger, Vincent J. Della Pietra, and Stephen A. Della Pietra. A Maximum Entropy Approach to Natural Language Processing. *Computational Linguist*., 22(1):39–71, 1996.

[4] John A. Carroll. Statistical parsing. In Robert Dale, Hermann Moisl, and Harold Somers, editors, *A Handbook of Natural Language Processing*. Marcel Dekker, 2000.

[5] Robert Dale. Symbolic approaches to natural language processing. In Robert Dale, Hermann Moisl, and Harold Somers, editors, *A Handbook of Natural Language Processing*. Marcel Dekker, 2000.

[6] Robert Dale, Hermann Moisl, and Harold Somers, editors. *Symbolic Approaches to Natural Language Processing*. Marcel Dekker, 2000.

[7] Douglas B. Lenat, R. V. Guha, Karen Pittman, Dexter Pratt, and Mary Shepherd. Cyc: Toward Programs with Common Sense. *Communications of ACM*, 33(8):30– 49, 1990.

[8] Risto Miikkulainen. Text and discourse understanding: The discern system. In Robert Dale, Hermann Moisl, and Harold Somers, editors, *A Handbook of Natural Language Processing*. Marcel Dekker, 2000.

[9] Hermann Moisl. NLP based on Artificial Neural Networks: Introduction. In Robert Dale, Hermann Moisl, and Harold Somers, editors, *A Handbook of Natural Language Processing*. Marcel Dekker, 2000.

[10] Harold Somers. Empirical Approaches to Natural Language Processing. In Robert Dale, Hermann Moisl, and Harold Somers, editors, *A Handbook of Natural Language Processing*. Marcel Dekker, 2000.

[11] Jinghui Xiao, Xiaolong Wang, and Bingquan Liu. The Study of a Non-stationary Maximum Entropy Markov Model and Its Application on the PoS-Tagging Task. *ACM Transactions on Asian Language Information Processing* (TALIP), 6(2):7, 2007.

[12]http://www.researchgate.net/post/What_is_the_border_line_between_syntax_and_semantic_analysis_in_dependency_parsing_technique_stack

[13] http://www.link.cs.cmu.edu/link/

[14] http://nlp.stanford.edu/software/lex-parser.shtml

[15] http://www.maltparser.org/intro.html

[16] http://www.fi.muni.cz/research/nlp/analysis.xhtml.en

[17] https://opennlp.apache.org/index.html

[18] R. Lian et. Al. Syntax-Semantic Mapping for General Intelligence: Language Comprehension as Hypergraph Homomorphism, Language Generation as Constraint Satisfaction. *Proceedings of 5th International Conference on Artificial General Intelligence,* Oxford, UK, Dec. 2012.

[19] John F. Sowa (1976). Conceptual Graphs for a Data Base Interface. *IBM Journal of Research and Developmen*t 20:4, pp. 336-357, 1976

[20] John F. Sowa (1984). Conceptual Structures: Information Processing in Mind and Machine, Reading, MA: Addison-Wesley.

[21] John F. Sowa (1992). Semantic networks, in S. C. Shapiro (ed), *Encyclopedia of Artificial Intelligence*, Second Edition, New York: Wiley.

[22] John F. Sowa (2000). Knowledge Representation: Logical, Philosophical, and Computational Foundations, Brooks/Cole Publishing Co., Pacific Grove, CA.

[23] John F. Sowa (2010). The Role of Logic and Ontology in Language and Reasoning, in R. Poli & J. Seibt, eds., *Theory and Applications of Ontology: Philosophical Perspectives*, Berlin: Springer, pp. 231-263.

[24] John F. Sowa (2013). From Existential Graphs to Conceptual Graphs, *International Journal of Conceptual Structures* 1:1, 2013, pp. 39-72.

[25] Charles Sanders Peirce (1906). Manuscripts on Existential Graphs, *Collected Papers of Charles Sanders Peirce*, vol. 4, Harvard University Press, Cambridge, MA, pp. 320-410.

[26] Roger C. Schank, ed. (1975) Conceptual Information Processing, North-Holland Publishing Co., Amsterdam.

[27] R.J. Brachman and J.G. Schmolze (1985). "An Overview of the KL-ONE Knowledge Representation System". *Cognitive Science* 9 (2): 171.

[28] William A. Woods & James G. Schmolze (1992). The KL-ONE Family, in *Computers and Mathematics with Applications*, **23** (2–5):133. doi:10.1016/0898-1221(92)90139-9

Performance of a Natural Language Framework

Waleed Faris and Kam-Hoi Cheng
Computer Science Department, University of Houston, Houston, Texas, USA

Abstract *– An ideal intelligent program is one that can communicate with humans using a natural language. Three key operations of communication are to read, understand, and write sentences. We have developed a natural language processing framework that allows an incremental learning of a grammar, such as the English grammar. The framework contains algorithms for the three operations and also allows grammar terms to be taught using multiple components that define the purpose, structures, and rules of the terms. These components are used not only for parsing a sentence, but also for understanding its intent and for producing a sentence to reflect a thought. The grammar is defined in a manner that allows more complex terms to be introduced incrementally using a combination of simpler terms and an expansion of existing terms. The performance of this framework on all three operations improves as more grammar is introduced.*

Key Words - Human-Machine Interaction; Natural Language Processing; Parsing; Semantics; Grammar

1. Introduction

It is desirable for an artificial intelligence program to communicate with humans using a natural language. The program should be able to read texts after acquiring the grammar of a natural language. Reading texts is normally done by parsing, which translates the given input string into multiple parts-of-speech (POS). However, this is not enough for communication since it is important to understand the semantics of what has been communicated. In addition, communication also implies the ability to write correctly when given a thought (semantic representation), presumably without the need to learn a different grammar.

Recently, we have developed a program called A Learning Program System (ALPS) [1, 2, 3] to acquire knowledge learned by humans. In this paper, we present a flexible framework to allow a grammar of a natural language to be taught dynamically and incrementally to ALPS. Specifically, instead of being built into the code, the grammatical details are provided by a teacher as needed. This is achieved by defining grammar terms (POS) with multiple components that specify various aspects of the term such as the purpose, grammatical structure, and any required rules. This framework supports not only the reading of a sentence [4] but also contribute to the understanding of that sentence [5, 6] as well. In addition, it supports the capability to write grammatically correct sentences, which we accomplish by creating a sentence generating component based on the taught grammar [7], resulting in a bidirectional grammar. The strength of the framework is its ability to expand the types and complexity of texts and thoughts that it can handle via a growing grammar. Our solution completely separates the knowledge domain for the grammar of the natural language from those of the subject-area, allowing both the development of the solution and the learning of these knowledge bases to be done separately. Consequently, any subject-area may be acquired and updated without modification to our grammatical solution and vice versa.

We demonstrate the communication ability of this framework by measuring the performance of three key operations: read (parse), understand, and write. A subset of the English grammar is divided into multiple segments. After the program acquires each segment, we test these operations by using representative sentences. Although the program only knows about the taught portion of the English grammar, the correctness of these operations are classified based on the entire English grammar, and we assume the program has the necessary subject-area knowledge. Based on these classifications, we compute four performance metrics: accuracy, precision, sensitivity, and specificity [8]. By comparing these metrics among these incremental grammar segments, our results demonstrate the success of the grammatical framework: the system's capability to read, understand, and write improves as its grammatical knowledge increases incrementally. The eight segments of the English grammar are organized in such a way that the first segment introduces the basic structure and terms for composing a simple sentence. The grammar is then expanded in later segments by both introducing new terms and adding more alternatives to existing terms.

Our language framework is derived from four well-known paradigms in the fields of computational linguistics and natural language processing: feature structures, dependency grammars, frame semantics, and bidirectional grammars. With feature structures, the ability to incorporate the grammatical constraints allows for a more generalized definition of the grammar, such as Head-driven Phrase-structure grammar [9]. Dependency grammars emphasize the dependency relation between two words, identifying the semantic roles that determine the behavior of the pair [10]. Frame semantics provide an internal semantic structure relative to the context of the sentence [11]. Bidirectional grammars have the benefit of a single source for generating both a semantic representation from text and vice versa. In [12], three formulations for defining a bidirectional grammar are proposed. Our solution is based on the second formulation and creates a bidirectional grammar by generating a grammar for writing based on the grammar provided for parsing.

The rest of the paper is organized as follows. Section 2 provides a brief overview of ALPS natural language framework; specifically, on how a grammar is structured so that it can be taught incrementally to ALPS and how it is used to read, understand, and write sentences. Section 3 explains how we test the framework, Section 4 analyzes the results of each operation, and Section 5 concludes the paper.

2. The Natural Language Framework

Learning the English grammar involves learning various English grammar terms, such as sentence and verb, as well as learning how and why these grammar terms are used. A grammar term is composed of several components: structure, kind, role, control and semantic rule. The structure of a grammar term defines the grammatical format of the term using either a sequence or an alternative set of terms. A term may have multiple kinds, which are subsets that may share the same structure but must have different roles. The role of a grammar term defines its purpose. Using concepts similar to frame semantics, roles form a bridge between the grammar (syntax) and the internal knowledge base (semantics) by associating the different components in the role to a grammatical usage in the phrase or sentence. The idea of the control stems from the notion that phrases contain a head and potential dependencies, frequently seen in dependency grammars. The control indicates which grammar term's role from within a sequence of terms is responsible for organizing the various identified knowledge during the parsing process. Finally, just as feature structures can define lexical rules between structures, the rule component of our grammar term specifies a condition to be satisfied by either the grammar term or its structure. Grammar rules are used to either guide the parser towards the correct grammar structure or restrict certain terms from being accepted. The use of rules in a grammar can enhance the scope of the grammar while maintaining a manageable size for parsing. A grammar term may contain some combination of these components. For example, a sentence is defined using all five components while a noun only has a structure. Flexibility in teaching the grammar is achieved by allowing these components to be learned independently and extended later to include more details and alternatives. The details of these components and how they are used in parsing can be found in [4].

The role component is the part of the grammar that shows how and when to use a grammar term, and provides a bridge between the grammar and the knowledge base. This component is critical in transforming a sentence into a semantic representation (or thought). The system uses the produced to understand the given sentence. We have defined four categories of role classes to represent the intention of a grammar term. The category of Sentence role represents a thought that either the user or the system is trying to express. The Verb role category is attached to verbs and is used to carry out the function of the main verb. The third category, known as the Usage role, has the task of locating knowledge objects based on how they are related to other objects in a noun phrase. Finally, the Reference role category is designed to identify knowledge referenced by a pronoun. Each of these categories contains several sub-categories, where each has its own semantic meaning. A grammar term can only be associated with one role, but because the English language allows a role to be expressed several ways, a role can be applied to multiple terms. For instance, the Aspect-of-an-Object role, a sub-category of the Usage role category, can be associated with both the preposition *of* and possessive adjective. The role associated with a grammar term has three important aspects: the grammatical role, the semantic role, and its role structure. The grammatical role (in italics), is the label given to the role according to the grammatical purpose of the term, e.g, *subject* and *predicate*. The semantic role (a logical object) is the label given to the role in relation to its semantic purpose within a higher-level role. The role structure defines how the grammatical roles map to the semantic roles. For example, a declarative sentence using a *Be* verb defines an object, so the semantic roles of the Define role are object-of-interest (the object being defined) and definition. This role structure specifies that the grammatical role *subject* reflects the object-of-interest, while the grammatical *predicate* maps to the definition. The details on how roles are used to understand grammar terms can be found in [5, 6, 13]. Note that these bridges linking grammatical roles to semantic roles work in both directions. As a result, roles are used not only for creating a thought from parsing a sentence, but also for generating a sentence from an existing thought.

At the time our program learns the grammar; additional knowledge is generated automatically based on learned roles to create the bidirectional grammar. Specifically the knowledge generated includes role sets, role sequences, and sequence collections. A role-set represents the alternative representations for a role, corresponding to the alternative structure of a grammar term. When a new role becomes associated to one of the grammar term's alternatives, the role is added to the role-set for the grammar term's role. For example, the verb grammar term has a role called the Verb role. When a new alternative is added to verb, such as action verb, its role – called Action role – is added as an alternative to the Verb role. A role-sequence defines a precise way to express a role by indicating the specific order roles should be structured, corresponding to the sequence structure of a grammar term. The principles behind creating role-sequences ensure the correct grammatical relationships among the terms in the sequence. When a grammar term with a sequence grammatical structure creates a role-sequence, it contains the grammatical roles used for expressing the various parts of the grammar term along with the proper order and occurrence of each. For instance, a noun phrase may be defined by the sequence {noun + prepositional phrase}, where noun is compulsory and has the *simple subject* label, and prepositional phrase is optional and defined in turn by the sequence {preposition + nominal} with this nominal having the *object-of-the-preposition* label. The corresponding role-sequence created for noun phrase is [*simple subject* + *prepositional* + *object-of-the-preposition*]. While roles and role sets are attached to grammar terms, a role sequence is attached to each role structure in the role set

of the control. Once a new role sequence is attached, the originating grammatical sequence may then be used to express that structure. A collection of these role sequences is called the sequence collection for that role structure. Any role sequence found in its sequence collection may be used to express the content of the role structure. By using these roles, role sets, role sequences, sequence collections, and bridges, it removes the dependency on the grammatical terms when determining the structure of a sentence. These additional role-based structures allow our program to generate a sentence from roles within a thought. The details on how this is done may be found in [7]. By generating a writing grammar automatically from the grammar taught for parsing, there is no need to manually create a separate grammar or maintain multiple grammars.

The parsing algorithm in ALPS is completed in two steps: syntax and semantic. The task of the syntax stage is to separate an input string of characters into individual words with its associated knowledge and properties identified. This is handled by an algorithm that stems words into their root forms and identifies words as either known by ALPS or an unknown term. Each term in a sentence falls under one of three categories. Either it is immediately understood by the system in its current form, or it can be understood once it is transformed into its base form, or it is not understood by the system even after transforming to its base form. Since some words need to be transformed before they can be understood, the syntax stage processes the input sentence in two phases. In the first phase, the parser traverses the sentence, transforming any words it does not immediately understand into its base form using learned morphology rules. A new sentence is then compiled using these new base forms. During the second phase, this new sentence is traversed in an attempt to identify single and multiple word terms that the system recognizes. Any words that are still not recognized (i.e., represent knowledge unknown to our program) are marked as unknown words. In such a case, a "dummy" knowledge object, which refers to knowledge unknown to the system, is created to represent the word. At the end of the syntax step, each term is represented by a unit that contains the original word, its base, properties of the original word such as tense and number, and all the identified knowledge for that term. This sequence of word units is then passed to the semantic step. The semantic step deals with recognizing the various POS and relationships among them in order to produce an appropriate thought interpreting the meaning of the sentence. It is based on a depth-first top-down template parser, and its execution consists of processing in two major manners: top-down followed by bottom-up. The top-down processing traverses the grammar structures and calls upon each grammar term's unique internal parser to determine if the current grammar term matches the next word unit. The top-down processing also checks the grammar rules, a concept similar to that of the lexical constraints seen in feature structures, to validate the matched results. After the grammar term has been identified, the role associated to that grammar term needs to be satisfied such that it reflects the semantics of the grammar term. This is accomplished by organizing the detailed content of the role. Much like frame semantics map the actors in a scene, the logical objects within the role are mapped to semantic roles collected in lower levels. The processing in the bottom-up manner continues upward satisfying each higher level role by organizing all the satisfied sub-roles appropriately. By the time the semantic step finishes, the result is a specific Thought role that reflects what the sentence wants to express, and all identified knowledge is stored as the appropriate parts of this thought. More details on how both the syntax and semantic steps work can be found in [4, 14].

The produced thought provides a high-level intention of the given sentence such as it may declare a fact, inform an action performed by an object, or pose a question. However, the exact detailed meaning of the sentence is based on the relationships among the various knowledge objects referred to within the given sentence. For instance, a declaration using a *Be* verb may declare the hierarchical relationship between two categories or the state/value of an aspect of an object [6]. Similarly, the declaration of an action may need to be disambiguated if that action word has multiple meanings. It may also imply the occurrence of an event or activity, and may also imply several sub-actions that are intuitive but not explicit in the sentence [5, 14]. For example, the act of buying includes at least two transactions: the transfer of money and the transfer of the product. This is accomplished in ALPS by the understand operation, which interprets the thought produced from parsing. In addition, this operation uses knowledge learned about the objects identified by the sentence to determine the sensibility of the given sentence. A declarative sentence is determined to be not sensible if it contradicts with what the system knows. For example, stating that 3 is a factor of 7 will be considered as not sensible by ALPS.

The writing of a sentence begins with a logical thought, i.e., its content is identified by logical objects instead of grammatical terms. Given that a thought is a collection of roles, each representing a specific semantic element to be expressed, the task of producing an English sentence to reflect that thought involves expressing the intent of each role. The writing algorithm works from the highest level roles downward and compiles words and phrases until a completed sentence is produced. Specifically, this process for generating a sentence consists of three steps. It begins by identifying a sequence of roles from the sequence collection for a sentence. Each role within the chosen sequence is then called sequentially to produce a textual representation of that role. Finally, the right form of each word is created based on the properties of the knowledge that it represents. The first step of selecting the role sequence is akin to sentence planning [15]. It selects the role sequence from the sequence collection that best fits the given thought or role. The sequence that best fits is the one that expresses exactly all the information contained within the role. For example, suppose the program is given a thought that expresses the act of a boy selling lemonade to his neighbors and the declaration role has the following sequences in its collection: [*subject*, *verb*, *predicate*],

[*subject*, *verb*, *direct object*], and [*subject*, *verb*, *indirect object*, *direct object*]. Although, the first sequence may be used to express a declaration to define a subject, it is not suitable to express a declaration about an action. The second sequence is to express an action, but does not express all the information stored in the thought; specifically, the indirect object representing the neighbors cannot be expressed. The last sequence is the best fit for this thought and should be the one selected to generate the sentence. The second step of writing is to identify the correct words to be used based on the knowledge associated with each logical object. It is accomplished by assembling a list of word-units according to the chosen sequence of roles. A word unit is created by extracting the content of the role including the name of the knowledge and any properties associated with that knowledge. For example, a role representing the person John would produce a word-unit containing 'John' and the properties singular, third-person, male, and unique. The object-oriented nature of the system allows for each classification of role to have its own unique function to determine what properties are needed and where to locate them. For example, a Verb role will retrieve its properties from the properties stored in the Thought role, such as the tense of the sentence, and from the properties of the subject, such as person and number properties, initiated from the subject-verb agreement rule. It should be noted that when assembling a list of word units, if a role in the sequence is composed of multiple sub-roles, then the process of selecting the best sequence repeats itself at this level. The final step of writing selects the correct form of the word based on the properties stored in each word unit, and applies any syntactic rules needed to generate a complete sentence. The syntactic rules here are concerned with word morphology and are different from the grammatical rules associated with grammar terms which determine whether the terms are semantically correct.

3. The Experiment

To measure the performance of our incremental language framework, we divide the learning of a subset of the English grammar into eight segments. The first segment introduces the basic structure for composing a sentence and the simplest definition of some basic grammar terms. Specifically, the basic structure for composing a sentence is a complete subject followed by a complete predicate and ended by a punctuation mark. It also includes a simple definition for noun and the *Be* verb allowing the formation of simple sentences like *"Mary is a human."* Each new segment expands the kinds and complexity of the sentences that the program can handle, such as *"Which number is the next prime number after 7?"*, by introducing new terms and/or alternatives to existing terms. By the time the eighth grammar segment is introduced, some of the more complex grammar terms taught to the program include infinitive phrases, comparative conjunctions, possessive adjectives, and interrogative adjectives. To test the various segments of the grammar, we use a total of 107 representative sentences. Each of these sentences represents a unique combination of grammatical terms, some of which may contain grammatical errors. For example, the sentences "*John bought his bike from Craig's List"* and "*Sally buys 4 shares of Microsoft from her broker"* both have the same grammatical structure. With the exception of their individual meanings, the processing by the system for both sentences is the same: identifying the effects of a transaction and dereferencing possessive adjectives (*his*/*her*). Therefore, either example will suffice in representing all the endless sentences that can be formed by using similar nouns and/or verbs. Out of the 107 sentences, only 88 are considered valid English sentences, while the other 19 sentences are used to measure how the program handles grammatically incorrect input. After learning a new segment of the English grammar, a new group of sentences are expected to be handled correctly, and after all the eight segments of the English grammar, all 88 valid English sentences are expected to be handled correctly. We measure the performance on three operations of the program: parse (read), understand, and write. Note that since no thought is created if a given sentence fails to be parsed by ALPS, it cannot be understood nor written correctly. Consequently, the performance of understanding and writing is, as expected, much lower in the earlier segments than the actual capability since a large number of sentences fail to parse successfully as a result of missing grammar. Finally, batch scripts are provided to create a consistent knowledge base before we test sentences, so that the only factor affecting the performance is the additional grammar segment.

After each segment of the grammar is taught, all 107 representative sentences are tested. For each operation, each of the 107 test sentences is classified as either a true positive, true negative, false positive, or false negative. When classifying parse results, the expectation is that the parsing of a sentence should succeed if the sentence is a valid sentence according to the complete English grammar; otherwise it would fail. Consequently, a sentence that is considered a grammatically correct sentence and has been parsed is classified as a true positive, while an invalid sentence that is parsed is classified as a false positive. A false positive may occur when certain rules in the grammar are not taught. For example, assume the subject-verb agreement rule has not been taught and so cannot be enforced. If a sentence violating this rule is parsed, the result is classified as false positive because the expectation is that it should not be a valid sentence in the complete grammar. For sentences that are not parsed, if the sentence is not considered valid, such as *"Primates gorillas are."*, then the sentence is a true negative. On the other hand, valid sentences that failed to parse are considered false negatives. The most common reason for failing to parse them is that the necessary grammar has not been taught yet, and once the appropriate grammar component was taught at a later learning stage, those sentences were seen to be parsed.

If a sentence is parsed successfully, a thought is created that is used to test both understand and write operations. On the other hand, since no thought is produced when parsing fails, the results are classified as negative automatically;

whether it is classified as a false negative or a true negative depends on whether the original sentence is a valid English sentence or not. For understanding a thought, we based our classification of a sentence by comparing ALPS interpretation of the sentence to how a human would interpret it. For instance, a true positive is a valid sentence that was understood correctly, while a false positive is when a correct understanding is made of an invalid sentence, typically those with grammatical mistakes that are overlooked. The analysis of the writing stage looks at the sentence generated by ALPS, which reflects the thought produced by the parse operation. A written sentence is classified as true positive when it is grammatically correct. Note that it is possible to have an incorrect thought because the sentence may be parsed incorrectly. Understanding and writing using such an incorrect thought will result in negative classifications for both.

The classification results of each operation are used to compute four common performance metrics: accuracy, precision, sensitivity, and specificity, shown respectively in equations (1)-(4). Rather than focusing on the exact value of each metric or the amount that it changes between grammar segments, we show that there is a trend of improvement after each new grammar segment is introduced. The steady improvement in performance and the increase in the range of handled sentences confirm that the communication capability of ALPS improves as more grammar is learned.

$$Accuracy = \frac{number\ of\ true\ positives + true\ negatives}{total\ number\ of\ sentences} \quad (1)$$

$$Precision = \frac{number\ of\ true\ positives}{number\ of\ true\ positives + false\ positives} \quad (2)$$

$$Sensitivity = \frac{number\ of\ true\ positives}{number\ of\ true\ positives + false\ negatives} \quad (3)$$

$$Specificity = \frac{number\ of\ true\ negatives}{number\ of\ true\ negatives + false\ positives} \quad (4)$$

4. Results

For parsing, the classification and metric results for all eight iterations are shown respectively in the two charts of Fig. 1. Only 17 sentences are identified as true positives after learning the first grammar segment, but it improved to 84 out of the expected 88 after the last segment. Of the remaining four valid English sentences, three were classified as false negatives and one as false positive. The false negative sentences, which dropped from 70 after the first segment, are *"Apple fritters are food."*, *"David rides a bicycle."*, and *"Camels can be found outside Houston."* The first sentence shows how a sentence can fail with compound word knowledge unknown by ALPS; in this case, the unknown term apple fritter results in the parsing of two consecutive nouns apple and fritter. As for the second and third sentences, the actions ride and found were left unknown to ALPS deliberately. This was intended to show that parsing of a sentence is based on more than just the sentence having a grammatically correct structure, but also on a sufficient knowledge base. Finally, the number of false positive classifications dropped from eight initially to only one at the end. This false positive sentence is *"John receives 3 apple."* Although this sentence is grammatically incorrect with the singular form of the word apple, the sentence is still parsed because the parsing algorithm tags the number 3 as the indirect object. Finally, the 19 invalid English sentences, included to measure how the system handles invalid input, were all correctly identified as true negative after all eight segments were taught. Note that after grammatical rules to enforce proper punctuation, subject-verb agreement, subject-predicate agreement, and number agreement are introduced in the third grammar segment, a drastic shift is seen in the number of true negative and false positive classifications, resulting in notable improvements in the precision and specificity. Before these rules were taught, invalid sentences such as *"You is a robot."* were parsed and considered false positives. The introduction of these rules prevents invalid sentences from being parsed by an overly lax grammar.

In terms of performance metrics, both the accuracy and sensitivity improved from the mid 30% range to over 90% because of the large drop in false negatives as well as the corresponding increase in true positives. The precision and specificity improved as well, but quickly reached their plateau after grammar rules were introduced in the 3rd segment. Overall, the results show that the performance of the parser approaches expectations as more and more complex grammar structures, roles, and rules are introduced.

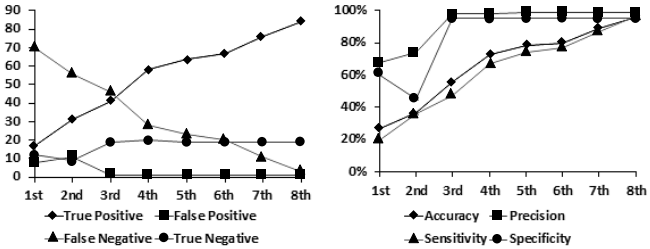


Figure 1. Results for Parsing

The results for the understand operation are shown in Fig. 2, from which we see trends of improvement similar to that of parsing. After the initial segment, only eight sentences were classified as true positive and 78 were false negative. Although the number of true positives improved to 80 after the final segment was taught, 8 false negatives remained. Both classifications showed consistent improvements on an average of 10 sentences per new segment. This results in a similarly consistent upward trend in both the accuracy and sensitivity metrics, which climbed from 21% and 10% to 93% and 91%, respectively. On the other hand, the number of false positives and true negatives settled at their final numbers, 0 and 19 respectively, after grammatical rules were taught in the 3rd grammar segment. This corresponds to the number of false positive parsing results dropping to only one and when precision and specificity spike to 100%.

Finally, the performance results for the write operation also show a similar trend of improvement, as seen in Fig. 3,

although not at the same level as the other operations. Specifically, the number of true positive classifications increased from 15 to 55, while the number of false negatives decreased from 72 to 32 sentences. Analyzing these results

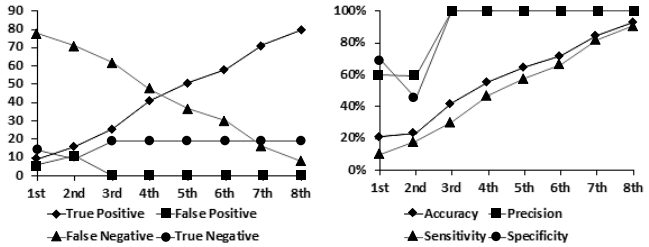


Figure 2. Results for Understanding

further, 29 of the 32 false negative classifications occur because of undeveloped writing functionality for certain roles: for example, future work is needed to create the proper role structure to express possessive pronouns. The remaining three false negatives are the same as those found during parsing. In addition, the final grammar segment produced 20 true negatives. The additional true negative, which is not part of the 19 intentionally invalid sentences, comes from the sentence *"Jump is a racehorse."* Although the parser reads this sentence as illegally beginning with an action verb, resulting in a failure in write, this particular sentence would be parsed correctly if a racehorse named Jump is taught. Finally, after grammar rules were introduced, the write operation did not produce any false positive results.

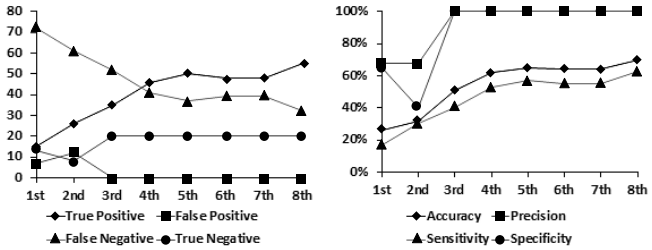


Figure 3. Results for Writing

Note that the performance for the understand operation is artificially depressed in the earlier segments. The reason is that if a valid sentence is not parsed, it is automatically classified as false negative. For instance, after the initial grammar segment is taught, out of the 107 test sentences, 82 are classified as false negative only because they failed to parse. This large number of initial false negatives does not indicate a failure in the program's ability to understand. It is merely indicating a lack of grammatical knowledge. A better indicator on the program's capability to understand would be the correlation between the understand operation with parsing. Our results show that there is a [0.982 – 0.992] positive correlation between parsing and understanding for all four categories of classifications. This can be attributed to the fact that both parsing and understanding depend on the knowledge that the program knows, which is confirmed by the results of another experiment described later in this section. If the proper knowledge and grammar are available at the time of parsing a sentence, the likelihood of successfully understanding that sentence is very high if the sentence is successfully parsed. Similarly, even though writing experiences the same artificial depression of performance initially, a strong correlation can also be found between parsing and writing, ranging between 0.97 and 0.996 for all four categories. These correlation results tell us that a large number of sentences can both be understood and written once they are parsed. Although the correlations are high, they are all less than one because of incorrectly parsed sentences, which will lead to incorrect understandings and writings. Finally, although we saw a slightly lower correlation between writing and understanding for true positive as well as false negative, the correlation is still high at 0.916 and 0.912, respectively. Since it is not perfect, it indicates that not all sentences that are understood result in properly written sentences. This is due to portions of the write capability that need to be implemented for newer roles, and should correctly write these sentences once completed.

Besides the basic experiment, we also conduct two additional experiments. The first is to see the effect of changing the grammar to a grammatical error tolerant grammar; rules for tolerating grammatical errors are introduced in [16]. The second is to investigate the importance of the system knowing the knowledge that these sentences are expressing. In other words, can the system still parse a sentence if the semantic knowledge of each word is absent? The first experiment uses grammatically tolerant rules in place of standard restriction rules. We see an increase in sentences parsed; specifically, 16 of the 19 invalid sentences can be parsed. This result is a 15% increase in the number of sentences that are parsed. The same increase was recorded in the percentage of correctly understood sentences. This validates that the system is able to properly understand a sentence even when it contains a grammatical error. However, since these newly parsed sentences are considered false positive results, the accuracy is naturally impacted negatively. The remaining three sentences that could not be parsed contain structurally invalid orderings, such as with the sentence *"Primates gorillas are."* Although restriction rules were necessary to ensure accuracy in measuring the parser's ability to recognize grammatically correct sentences, communication with a human user benefits from a tolerance for grammatically incorrect sentences, if the correct intended meaning can still be determined.

The second experiment aims to show that the ability to handle a sentence depends on the amount of knowledge known by ALPS. This experiment was conducted without the subject-area knowledge provided to the program by the knowledge-base batch files. The results show a large increase in the number of sentences that failed to parse, from 2.8% to 52.3%. However, the number of incorrectly parsed sentences increased by less than 2%. This can be attributed to the framework's ability to parse, what are now considered as *unknown* words. Ultimately, whether due to its failure to parse, or parsing of unknown words, the lack of knowledge

correlates to a 73.8% decrease in the number of sentences that could be understood. In other words, just as the extent of a human's ability to communicate depends on our general knowledge, the ability of the framework to parse, understand, and write sentences depends significantly on the amount of knowledge known by the program. An in-depth explanation of the results of these experiments, a summary of the grammar subset used, and a list of the representative sentences can be found in [17].

5. Conclusion

We presented a framework of communication whose performances in reading, understanding, and writing sentences improve by learning more grammar. The framework unifies several existing independent paradigms into one combined solution. This framework includes the introduction of a component based grammar that allows for the introduction of complex grammar terms as well as extensions to existing grammar terms. The components are used to read and understand sentences. In addition, our solution also uses the components to compile a bidirectional grammar to generate sentences based on a thought. The three key operations of communication are tested using a subset of the English grammar, divided into eight segments, beginning with simple definitions for some basic grammar terms to form the basic structure of a sentence. The grammar is expanded in later segments and the results show a steady improvement in all four performance metrics. This trend verifies our postulation that the range and complexity of sentences that ALPS can read, understand, and write will expand as additional grammar and knowledge is introduced to the system. Finally, since both subject-area knowledge and grammatical knowledge are not hard-coded into the program and can be introduced to the program dynamically, incrementally, and independent of one another, our framework has the advantage of learning from multiple domain experts over the course of several teaching sessions.

6. References

[1] K. Cheng. "An Object-Oriented Approach to Machine Learning"; International Conference on Artificial Intelligence, 487-492, 2000.

[2] K. Cheng. "The Representation and Inferences of Hierarchies"; Proceedings of IASTED International Conference on Advances in Computer Science and Technology, 269-273, 2006.

[3] K. Cheng. "Representing Definitions and Its Associated Knowledge in a Learning Program"; Proceedings of the International Conference on Artificial Intelligence, 71-77, 2007.

[4] W. Faris & K. Cheng. "An Object-Oriented Approach in Representing the English Grammar and Parsing"; Proceedings of the International Conference on Artificial Intelligence, 325-331, 2008.

[5] W. Faris & K. Cheng. "Understanding and Executing a Declarative Sentence involving a forms-of-be Verb"; Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics, 1695-1700, 2009.

[6] E. Ahn, W. Faris, & K. Cheng. "Recognizing the Effects caused by an Action in a Declarative Sentence"; Proceedings of the International Conference on Artificial Intelligence, 149-155, 2009.

[7] W. Faris & K. Cheng. "Generating a Sentence from a Thought"; Proceedings of the International Conference on Artificial Intelligence, 554-560, 2011.

[8] T. Fawcett. *An Introduction to ROC Analysis;* Pattern Recognition Letters, vol 27, issue 8, pg 861-874, June 2006.

[9] R. Levine & W.D. Meurers. Head-Driven Phrase Structure Grammar: Linguistic Approach, Formal Foundations, and Computational Realization, In: Keith Brown (Ed.): *Encyclopedia of Language and Linguistics*, Second Edition. Oxford: Elsevier. 2006

[10] J. Nivre. Dependency grammar and dependency parsing; Växjö University, 2005

[11] C.J. Fillmore & C.F. Baker. "Frame semantics for text understandin"; Proceedings of WordNet and Other Lexical Resources Workshop, NAACL, June 2001.

[12] T. Strzalkowski. *Reversible logic grammars for natural language parsing and generation*. Computational Intelligence, vol 6, number 3, pg 145-171, 1990

[13] W. Faris & K. Cheng. "Understanding Pronoun"; Proceedings of the International Conference on Artificial Intelligence, 850-856, 2010.

[14] W. Faris & K. Cheng. "A Knowledge-Based Approach to Word Sense Disambiguation"; Proceedings of the International Conference on Artificial Intelligence, 111-117, 2013

[15] E. Reiter. "Has a consensus NL generation architecture appeared, and is it psycholinguistically plausible?"; In Proceedings of the Seventh International Workshop on Natural Language Generation, Association for Computational Linguistics, Stroudsburg, PA, USA, 163-170, 1994.

[16] W. Faris & K. Cheng. "A Grammatical-Error Tolerant Parser"; Proceedings of the International Conference on Artificial Intelligence, 711-717, 2012.

[17] W. Faris "Building a Natural Language Interface for ALPS"; Doctoral Dissertation, Computer Science Department, University of Houston, May 2014.

Assigning stress to out-of-vocabulary words: three approaches

Manex Agirrezabal*, Jeffrey Heinz†, Mans Hulden§ and Bertol Arrieta*
*University of the Basque Country (UPV/EHU)
IXA NLP Group, Department of Computer Science
Donostia, Basque Country 20018
†University of Delaware
Department of Linguistics & Cognitive Science
Newark, Delaware 19716 (USA)
§University of Colorado Boulder
Department of Linguistics
Boulder, Colorado (USA)
manex.aguirrezabal@ehu.es, heinz@udel.edu, mans.hulden@colorado.edu, bertol.arrieta@ehu.es

***Abstract*—In this paper we address the task of automatically assigning primary stress to out-of-vocabulary words in English. This work forms a necessary component in a scansion system for English poetry. We propose three different approaches based on (1) word similarity, (2) hand-written linguistic rules and (3) machine learning. The first and last approach require stress-annotated corpora to train a model for stress assignment, while the linguistic approach relies on grapheme to phoneme conversion, a syllabification procedure and hand-written stress assignment rules. An implementation of each approach is provided and the precision of the systems is compared. The linguistic approach proves to be the most effective, but the machine learning approach is not far behind. We anticipate that including part of speech information will improve the accuracy of each system. The source code of the approaches is released[1] under the GNU GPL license.**

I. Introduction

Automatic scansion of English Poetry has attracted different scholars from all around the world. *Scandroid* [1], *AnalysePoems* [2] and *Calliope* [3] are some examples of computational systems created to analyze English verse. [4] is another recent attempt to scan English poetry using unsupervised learning algorithms and weighted finite-state machines.

The authors would like to acknowledge the University of Delaware, the Department of Linguistics and Cognitive Science in particular, for receiving the first author as a visiting student. Also we would like to thank the University of the Basque Country, whose grant has made possible this stay in Newark, Delaware.

[1]Code available at http://athenarhythm.googlecode.com

The purpose of a scansion system is to determine the rhythmic nature of lines of verse. It has long been recognized that knowing the location of primary stress in English words is an important component of successful scansion of English poetry. Typically, this information can be found in dictionaries. However, many poems make use of words which are *not* found in *any* dictionary. This is because these words have atypical spellings, are derived through a series of complex morphological processes, or because they are nonce words completely made up (cf. Lewis Carroll's 'the vorpal blade'). The problem of out-of-vocabulary words is greater for older English poetry. Scansion systems which cannot determine the location of primary stress in such out-of-dictionary words introduce a source of error wherever they occur. The scansion of a poetry line gives a representation where each syllable is marked with a level of stress and these syllables are then grouped into feet. Here you can see an example from Henry Wadsworth Longfellow's *The song of Hiawatha* and it's scansion, where stressed syllables are marked with the symbol ′:

```
′     - |  ′   - |  ′  -| ′  -
And the | mighty | Mudje|keewis,
```

This paper examines three systems which aim to reduce this sort of error by correctly predicting the location of stress in out-of-dictionary words. The first approach is based on *analogy*: a word *similar* to the target word is searched in a dictionary. The *linguistic* approach follows from the observation by linguists that, in many

of the world's languages, the location of primary stress in words is largely predictable [5], [6], and may often be encoded as phonological generalizations, or rules. These systems of rules or constraints can be used to predict the location of primary stress on words not found in dictionaries. The third approach is based on *machine learning*. A dictionary can be used to train a system to predict the location of primary stress on words not found in dictionaries. This is not unlike the linguistic approach. Both the linguist and the machine are inferring patterns from data. The difference is in the kinds of models the two approaches entertain and whether the inference is made mentally or automatically.

We emphasize that there are several ways each of the above approaches can be instantiated. For analogy-based systems, for example, there are many different solutions to the problem of defining 'word similarity.' Also, linguists disagree to some extent about the nature of the linguistic generalizations regarding stress assignment in English (see for example [7]). Likewise, there are many machine learning techniques that could in principle be applied to the task of predicting the location of stress in words [8]. It is of course beyond the scope of this paper to investigate all the possibilities. However, the implementation choices made here straightforwardly represent each of these approaches.

Our results show that both the linguistic and machine learning approaches significantly outperform the analogical approach. Additionally, the linguistic approach slightly outperforms the machine learning approach.

II. Background

The background for this work is found in ZeuScansion [9], a tool for scansion of English poetry based on finite-state technology. This system was designed around several finite-state machines that analyzed verse by performing tokenization, part of speech tagging, stress placement, and—for unknown words—stress pattern guessing. The analogy method examined here is the one used originally in the ZeuScansion tool.

There exist various linguistic approaches for stress allocation, such as the one proposed in [5]. The works [6] and [7] also address the issue of stress location from a linguistic point of view. This paper employs a slightly simplified version of the rules mentioned in these references as generalization rules.

Only a few studies have investigated machine learning of stress systems from corpora.[2] These include [12], where Russian stress location in predicted using Maximum Entropy Ranking and [13] which uses a Support Vector Machine (SVM) ranker with linguistically-inspired features. Our machine-learning method is most similar to Dou et al. [13], though we employ different features. They operate with more linguistically motivated features, while our system works with more standard ones. Based on their reported results, we would expect their methods to outperform our machine learning method. However, we do not have a direct comparison and leave such a comparison for future research.

III. Corpus

We have used the NETtalk pronunciation dictionary [14] for training and testing our models. The dictionary contains various pieces of information about English words such as: pronunciation, number of syllables, possible part of speech and primary and subsidiary stress location. The version we have employed uses an ASCII encoding for the phonemic representations of words.

IV. Task

As mentioned before, the main task to be resolved in this paper is the main stress assignment of unknown English words. In this section, we describe three approaches that perform it.

A. Similarity approach

The similarity approach was first proposed in [9]. The system, represented as a finite-state machine [15], attempts to make small changes to the unknown word to yield a similar known word. We suppose that the scansion of the 'most similar' word found in the dictionary will be the same as that of the input word. For example, in Lewis Carroll's *Phantasmagoria and Other Poems* the following phrase can be found, whose scansion cannot be inferred by simply using dictionaries:

I prophesy there'll be a row

The word "prophesy" is not found in our dictionary, so the similarity approach can find out that the most similar word is "prophecy", whose first syllable will receive main stress.

[2] A few studies with a cross-linguistic emphasis examine learning from artificial corpora [10], [11].

The framework could be described as a series of transducers. The idea is that each transducer is programmed to make specific changes in specific parts of a word. The allowed changes are substitution, deletion and addition of one or two letters (a consonant or a vowel) and we divide the word in two different parts: The last part roughly corresponds to the final syllable and the first part to the rest of the word. For example, the words "deletion" and "smartphone" would be divided as "dele|tion" and "smartp|hone".

These are the changes that can be done using the transducers of the machine, which can be applied either in the first or the second part of the word.

- Change one vowel
- Change one consonant
- Change two vowels
- Change one vowel and a consonant
- Change two consonants

In total, we will have ten transducers for performing the above operations[3]. Each of them will produce an output if a word in the dictionary is found by making the allowed changes to the input word. These blocks will be installed as an ordered set of transducers, and if the first transducer gives an output, this will be returned by the entire system. If the first transducer can't give an output, the same input will be given to the second transducer, and so forth. That is, the order of these transducers will reflect a priority that certain changes have over other changes. The output will be preferentially the one given by the top transducers.

It is noteworthy that no matter the order of the transducers is, the recall of the entire system will not vary. However, the precision will change, so in order to find the best ordering we have tried to optimize the precision. To this end, we calculated the precision of each of the ten transducers and decided to place first the one that produced the highest precision, etc.

B. Linguistic approach

In this approach we have programmed a linguistic chain that performs grapheme to phoneme conversion, syllabification and stress assignment.

[3]It's important to mention that when a character is replaced, an element of the same category will be inserted. So, for example, if we change a vowel, we can't put a consonant in it's place, only a vowel.

As mentioned in [16], we assume that English words first have to be syllabified after which word stress rules are applied, which in turn depend on the syllable type. Before syllabification we perform a grapheme-to-phoneme conversion, based on [17]. The `g2p` program uses a model trained from the NETtalk dictionary for the task. After this, we syllabify the word with the finite-state syllabification system presented in [18]. The main concern for the stress assignment is the weight of the syllables, which might be light or heavy:

Heavy syllable: The syllable has a coda or ends in a tense vowel.

Light syllable: The other set of syllables will be considered light.

Once we apply all these processes to the input word, we execute some main stress assignment rules, encoded as a set of finite-state transductions. The main active rule is the so-called Latin stress rule ([6], P.50), which also applies in many English polysyllabic words. The rule in question codifies the generalization that heavy syllables tend to attract stress. Below is a description of this generalization, divided into four subrules:

- If the penultimate syllable is light, the antepenultimate syllable is stressed
- If the penultimate syllable is heavy, it is stressed
- In the case of disyllabic words, the first syllable is stressed
- Monosyllabic words are stressed

In figure 1 you can see the workflow of the linguistic chain with an example. Anyway, there are words that don't follow these generalization rules, such as, "an.té.nna", "A.la.bá.ma" or "po.líce". Hayes [6] estimates that in the 87% of English vocabulary, main stress can be predicted using linguistic rules. Using the aforementioned rules, it can be seen in the results section that we have not reached that precision.

C. Machine Learning approach

For the third approach we have trained Support Vector Machines [20] [21] using a stress-annotated dictionary. We have treated the stress assignment task as a multi-class classification problem. In the task, the class to be assigned is the stress pattern that each word follows, taking into account only the main stress. We extracted 25 different stress patterns from our corpus.

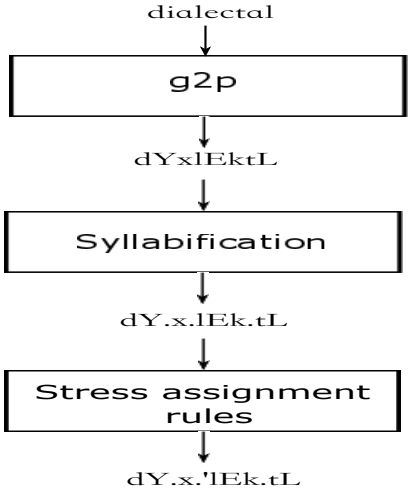


Fig. 1. Workflow of the linguistic approach.

We extracted two different sets of features for the purpose of training the SVMs. In the first set, *FS1*, we used character bigrams as features, including word boundaries. In the second feature set, *FS2*, we used character trigram frequencies, also known as Wickelfeatures [22]. For example, given the word "abate", in *FS1* we would train the SVM with the information that the {#a}, {ab}, {ba}, {at}, {te}, {e#} appeared once and the other possible bigrams zero times. These, together with the length of the word, are the training features for the first set. In the second model, we include the frequencies of trigrams {#ab}, {aba}, {bat}, {ate}, {te#}. For the example word "abate", the correct class would be $< \_' >$, indicating that the second syllable carries primary stress. Naturally, the features need to be encoded as numbers, so we used a dictionary and a mapping function[4] for the first feature set:

$$D(ch) = \{\#\} \rightarrow 0, \{a\} \rightarrow 1, \{b\} \rightarrow 2, ...$$
$$map(ch0, ch1) = ch0 * L^0 + ch1 * L^1$$

In the case of the second feature set the mapping function is similar but with some improvements to reduce the number of features.

After converting the data, we produce a corpus with 19,528 instances, one instance per word in the original corpus. In FS1, each instance has 899 attributes. On the other hand, in FS2, each instance has 5,495 attributes.

We trained different Support Vector Machines and evaluated each of them using 10-fold cross-validation.

[4] L will be the length of the dictionary D

In the next table, you can see the accuracy that we can get using different linear solvers[5] with each feature set. In some cases we have not included the accuracy because the computer ran out of memory.

Linear SVMs		
Solver type	Accuracy	
	FS1	FS2
1	0.48177	-
2	0.30065	-
3	0.41991	0.44659
4	0.50195	0.53899
5	**0.56580**	-

We also used nonlinear kernels[6] in our experiments. In the next table you can see the different results we got for each feature set.

Non-linear SVMs		
Kernel type	Accuracy	
	FS1	FS2
P	0.28052	0.278626
RBF	**0.461594**	0.278626
S	0.431176	0.278626

After training different SVMs and kernels, we performed a grid-search as it is recommended in [23] trying different C and γ parameters for the best SVMs, kernels and feature sets. Finally, the highest accuracy of 70.9820% was found by using the C-Support Vector Classifier with a RBF kernel and $C = 1024$ and $\gamma = 0.0078125$.

V. Experiment and results

In this section, we present the results and evaluation of the previously mentioned systems. In the following table, you can see where each of the approaches assign primary stress to these out-of-vocabulary words:

[5] 1: L2-regularized L2-loss SVC (dual)
2: L2-regularized L2-loss SVC (primal)
3: L2-regularized L1-loss SVC (dual)
4: SVC by Crammer and Singer
5: L1-regularized L2-loss SVC

[6] P: Polynomial
RBF: Radial Basis Function
S: Sigmoid

Input word	SIM-OO	LING	SVM
mudjekeewis	X	__'_	'__
pantroductory	__'__	__'__	__'__
joology	_'__	_'__	'_
amtenna	_'_	'__	'__
cate	'_	'	'

As some of the above methods are data-driven and others not, we evaluated each one slightly differently. Both the similarity approach and the machine learning approach were evaluated using 10-fold cross-validation. The linguistic approach, however, was evaluated with the whole corpus without any splitting, as it does not rely on any training data and is essentially an expert system.

In the following table[7] the obtained results can be observed:

	Accuracy
SIM	0.5843
SIM-OO	0.6777
LING	**0.7362**
SVM	0.7098

At this point, we can tentatively establish that the best result is obtained using the linguistic generalizations. However, the accuracies of both SVMs and hand-encoded generalizations are sufficiently close to warrant further research in improvement of both linguist-devised rules on the one hand, and better use of features for SVM modeling on the other.

VI. Discussion & Future Work

In this work, we have presented three different approaches to assigning a stress pattern to English out-of-vocabulary words. Accurate stress assignment is an important component of many larger systems, such as a scansion system discussed above.

The three approaches that we have proposed and evaluated are based on (1) word similarity, (2) linguistic rules and (3) machine learning. The results show that the best performance can be got using the linguistic generalizations, with a 73.62% precision, or the Support Vector Machine, with 70.98%.

There are several ways we believe the system's performance can be improved. One way is to include part-of-speech information. Other method, for the machine learning approach, is to experiment with different feature sets. In the case of the linguistic approach, we think that the rules can still be polished.

We are aware that trying to guess English word stress without having the information about the part of speech, can result in several errors. For example, in words like "present" primary stress can be placed both in the first or the second syllable, depending on the part of speech[8]. Having access to part of speech information in the linguistic and machine learning paradigm will without doubt improve the results of each one, perhaps significantly.

Furthermore, we hope that adding more features to the SVM and applying a feature selection algorithm will improve its results. In this regard, the work by Dou et al. is likely to be instructive. The features employed in that particular work are substrings that are slightly different from the n-grams employed here. Their feature-substrings consist of all the vowels in a word, with maximally one surrounding consonant include on both sides (if present), mimicking syllables in a way.

Additionally, in the SVM approach, we are considering to improve the feature mapping function, trying to map the proximity between letters[9], based on the IPA table.

For the linguistic approach, we are considering improving the rules, especially the one referring to the disyllables, in which stress is always located in the first syllable. Our insight is that primary stress can be assigned taking into account the weight difference between the two syllables.

Another future extension of these algorithms is to develop them for other languages. In this case, the machine learning approach is especially interesting because no linguistic knowledge is necessary, only a training corpus of words marked with stress.

References

[1] C. O. Hartman, *Virtual muse: experiments in computer poetry*. Wesleyan University Press, 1996.

[2] M. R. Plamondon, "Virtual verse analysis: Analysing patterns in poetry," *Literary and Linguistic Computing*, vol. 21, no. suppl 1, pp. 127–141, 2006.

[7] SIM: Similarity approach
SIM-OO Similarity approach w/ optimal ordering
LING: Linguistic approach
SVM: Support Vector Machine (with parameter C=30)

[8] If the word is a noun, the stress goes on the first syllable (présent). If it is a verb, the second syllable is stressed (presént).

[9] The distance between 'p' and 'b' would be smaller than the one between 'b' and 'k'

[3] G. McAleese, "Improving scansion with syntax: an investigation into the effectiveness of a syntactic analysis of poetry by computer using phonological scansion theory," -, 2007.

[4] E. Greene, T. Bodrumlu, and K. Knight, "Automatic analysis of rhythmic poetry with applications to generation and translation," in *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2010, pp. 524–533.

[5] M. Halle and J.-R. Vergnaud, *An essay on stress*. Mit Press Cambridge, 1987.

[6] B. Hayes, *Metrical stress theory: Principles and case studies*. University of Chicago Press, 1995.

[7] L. Burzio, *Principles of English stress*. Cambridge University Press, 1994, no. 72.

[8] K. P. Murphy, *Machine learning: a probabilistic perspective*. MIT Press, 2012.

[9] M. Agirrezabal, B. Arrieta, A. Astigarraga, and M. Hulden, "ZeuScansion: a tool for scansion of English poetry," *Finite State Methods and Natural Language Processing*, p. 18, 2013.

[10] E. Dresher and J. Kaye, "A computational learning model for metrical phonology," *Cognition*, vol. 34, pp. 137–195, 1990.

[11] J. Heinz, "On the role of locality in learning stress patterns," *Phonology*, vol. 26, no. 2, pp. 303–351, 2009.

[12] K. Hall and R. Sproat, "Russian stress prediction using maximum entropy ranking," *Empirical Methods in Natural Language Processing*, 2013.

[13] Q. Dou, S. Bergsma, S. Jiampojamarn, and G. Kondrak, "A ranking approach to stress prediction for letter-to-phoneme conversion," in *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, ser. ACL '09. Stroudsburg, PA, USA: Association for Computational Linguistics, 2009, pp. 118–126.

[14] T. J. Sejnowski and C. R. Rosenberg, "Parallel networks that learn to pronounce English text," *Complex systems*, vol. 1, no. 1, pp. 145–168, 1987.

[15] M. Hulden, "Foma: a finite-state compiler and library," in *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics: Demonstrations Session*. Association for Computational Linguistics, 2009, pp. 29–32.

[16] S. Duanmu, H.-Y. Kim, and N. Stiennon, "Stress and syllable structure in English: Approaches to phonological variations," *University of Michigan*, 2005.

[17] J. Novak, N. Minematsu, and K. Hirose, "Wfst-based Grapheme-to-Phoneme conversion: Open source tools for alignment, model-building and decoding," *Finite-State Methods and Natural Language Processing*, 2012.

[18] M. Hulden, "Finite-state syllabification," *Finite-State Methods and Natural Language Processing*, pp. 86–96, 2006.

[19] R. A. Mester, "The quantitative trochee in latin," *Natural Language & Linguistic Theory*, vol. 12, no. 1, pp. 1–61, 1994.

[20] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 27:1–27:27, 2011, software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.

[21] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "LIBLINEAR: A library for large linear classification," *The Journal of Machine Learning Research*, vol. 9, pp. 1871–1874, 2008.

[22] D. E. Rumelhart and J. L. McClelland, *On learning the past tenses of English verbs*. Institute for Cognitive Science, University of California, San Diego, 1985.

[23] C.-W. Hsu, C.-C. Chang, C.-J. Lin *et al.*, "A practical guide to support vector classification," 2003.

[24] L. Carroll, *Alice's adventures in wonderland and through the looking glass*. Penguin, 2003.

[25] H. W. Longfellow and H. R. Schoolcraft, *The song of Hiawatha*. TY Crowell & company, 1899.

A System for Answering Scheduling Questions based on Concept Association

Eriko Yoshimura[1], **Misako Imono**[2], **Seiji Tsuchiya**[1] **and Hirokazu Watabe**[1]
[1] Dept. of Intelligent Information Engineering & Sciences, Faculty of Science and Engineering
Doshisha University, Kyo-Tanabe, Kyoto, Japan
[2] Dept. of Knowledge Engineering & Computer Sciences, Graduate School of Engineering,
Doshisha University, Kyo-Tanabe, Kyoto, Japan

Abstract - *In this paper, the authors propose a method for realizing one element of this goal: an intelligent system for answering questions. The questions a user asks may not necessarily incorporate words that* precisely *match the words used in the schedule table. Existing systems for answering scheduling questions depend on the words written into the schedule table; for this reason they are not robust against variations in word choice and have a difficult time constructing appropriate responses based on an understanding of the purpose of the question. To address these difficulties, in this paper we describe the construction of a more autonomous system for responding to scheduling questions; the system we propose is built atop a concept association system based on human associative abilities and exhibits excellent applicability.*

Keywords: Computer conversation, Knowledge engineering, Knowledge representation, Natural languages

1 Introduction

In recent years, machines have become intimately involved in society and our daily lives, and the role they play in our day-to-day existence is becoming essential and irreplaceable. For this reason, the profile that we should strive to achieve for our machines is that of "machines that coexist with humans"—namely, *robots.* This goal is already en route to being partially realized thanks to the development of large numbers of robots that exhibit strong physical capabilities—including robots capable of walking on two legs, running, and even dancing. In the future, if robots are to achieve true coexistence with humans, it will be important to equip them not only with superior physical capabilities but also with *intelligence.* At present, researchers are trying a variety of disparate approaches to realize intelligence [1]. In this paper, we propose a method for realizing one element of this goal: an intelligent system for answering questions.

Secretarial systems constitute one important area of functionality desired of robots that coexist with humans. With the recent development of software tools such as Apple's "Siri" personal-assistant application [2] or NTT DoCoMo's "iConcierge" [3], secretarial systems have begun to penetrate our personal environment. These systems incorporate schedule-management systems, which work with a schedule table—input by a user—and then assist the user by delivering reminders. In this paper we propose a system for answering questions regarding these schedules.

Of course, given a schedule table, it is not difficult to understand someone's plans. Indeed, the software applications mentioned above are capable of outputting appropriate appointments based on questions asked by a user and on the words contained in the schedule table. This output is inferred by pattern-matching the notations used for the words in the schedule table against structural elements in the user's questions. However, the questions a user asks may not necessarily incorporate words that *precisely* match the words used in the schedule table. For example, given a schedule description that features the word "ocean," a user may ask "When am I going swimming?" Existing systems for answering scheduling questions depend on the words written into the schedule table; for this reason they are not robust against variations in word choice and have a difficult time constructing appropriate responses based on an understanding of the purpose of the question.

To address these difficulties, in this paper we describe the construction of a more autonomous system for responding to scheduling questions; the system we propose is built atop a concept association system based on human associative abilities [4] and exhibits excellent applicability.

2 The Concept Association System

Our Concept Association System[5] consists, in essence, of a system for defining basic relationships between words. Indeed, for machines to exhibit human-like conceptual association abilities, we must first develop a quantitative characterization of semantic similarities between words. In addition to the familiar relationships that exist between words—such as exact synonyms, near synonyms, and inclusion relations—we also have co-occurrence relations, partial relations, and other relations that are intuitively clear but not systematically formalized. Thus, to achieve concept association close to the intuitive, common sense capacity of

humans, it is necessary first to construct a quantitative representation of the semantic similarities between words.

Spurred by this challenge, Okamoto [6] and Watabe [4] studied the relationships between concepts. What these authors proposed was a *concept-base*. A concept-base takes each entry from a source such as multiple electronic dictionaries (or some similar resource) to be a *concept*, and takes each independent word in the description of that entry to be an attribute associated with that concept. This results in a large database that is machine-constructed automatically.

In a concept-base, a given concept A is represented by a pair $\{a_i, w_i\}$, in which a_i is an attribute representing a feature of the semantic properties of concept A and w_i is a weight quantifying the importance of attribute a_i in representing concept A. If concept A has N attributes, then it may be expressed in the form of an equation as follows. Here attribute a_i is termed a *first-order* attribute of concept A.

$$A = \{(a_1,w_1), (a_2,w_2), \cdots, (a_N,w_N)\}$$

Because each first-order attribute a_i associated to concept A is itself a concept defined in the concept base, we can also derive attributes from a_i in the same way. The attributes a_{ij} of a_i are termed *second-order* attributes of concept A. Each "concept" in the concept base is defined by a chained set of attributes ranging up to order n. To the descriptions of each concept we assign an average of 30 attributes, with weights assigned to the attributes indicating the degree of importance of that attribute for that concept; the sum of all attribute weights for a given concept is normalized to 1.0.

The *degree of association* gives a quantitative assessment of the strength of the relationship between two concepts. More specifically, the degree of association between two concepts is computed by using concept chains to expand the concepts, noting the first-order attributes between which the strongest correspondence is exhibited, then computing the degree to which these match. In this work we use a dynamic computational algorithm for computing the degree of association [4] [7].

The degree of association is a value between 0.0 and 1.0, with higher values indicating more closely related words.

Table 1 presents some examples of calculated degrees of association between pairs of concepts *A, B*.

Table 1 Example of calculated degrees of association

Concept *A*	Concept *B*	Value for degree of association
car	automobile	0.488
car	flower	0.008

3 System for answering scheduling questions

In this paper we use an association system based on intuition and associativity to construct a system for answering questions regarding a user's schedule. This question-answering system first prepares a table containing information on a user's schedule (referred to below as a "schedule table"), then responds to question sentences input by the user regarding appointments (referred to below as "appointment-question sentences") by referencing the appointments contained in the schedule table and constructing appropriate responses.

3.1 The schedule table we will use

The schedule table we use in this work includes seven data parameters: Start Date, End Date, Start Time, End Time, Place, Participants, and Event Description. Table 2 is an example of the schedule table format used in this work.

Table 2: Example of a schedule table

Start date	End date	Start time	End time	Place	Participants	Event Description
2012/7/25	2012/7/27	9:00	21:00	Tokyo	Friends	Trip
2012/12/12	2012/12/12	9:00	15:00	School	(None)	Lecture
2012/12/15	2012/12/15	13:00	19:00	Kyoto	Friends	Karaoke

3.2 Analyzing input sentences to identify question sentences

It is possible to extract the meaning of sentences from certain characteristic patterns they contain. For example, if the word "who" is present at the beginning of a sentence, we understand that the sentence is a *query* asking about a person. Thus we identify patterns that utilize case particles and grammar rules, register these patterns to a database as a form of intelligence, and use these to analyze sentences.

First consider analyzing the purpose of input sentences. We classify input sentences into four categories: greetings and salutations, commands or requests, queries, and information. For queries, we further identify whether the query is a 5W1H query, a YES/NO query, or an N-option multiple choice query. These determinations are made using grammar rules stored in a database.

In analyzing the purpose of input sentences, sentences determined to be queries may be identified as question sentences. For the purposes of the procedure discussed below for identifying appointment-question sentences, we classify question sentences into three categories: "when" queries, "non-when" 5W1H queries, and non-5W1H queries.

If the analysis of an input sentence's purpose deems that sentence to be a command, it is considered a candidate for a question sentence. Indeed, a sentence that looks grammatically like a command may in fact, depending on its content, have the purpose of a question; an example is the sentence "Tell me my appointments for today." Thus, for command sentences taken to be question-sentence candidates we focus on the lexical verbs in the sentence to determine whether or not they incorporate the purpose of a question. To this end, we construct a database of lexical verbs that are used—possibly with different declensions such as past tenses or participles—with high frequency in questions regarding schedules; we refer to such words as "scheduling-question words." This data base is shown in Table 3.

Table 3: Database of scheduling-question words

Scheduling-question words
Teach, tell, instruct, inform, notify, say, speak, respond

Given a command sentence considered to be a candidate for a question sentence, we ask if any of the lexical verbs in the sentence are contained in the above database; if so, we consider the sentence to be a question sentence. However, because the entries in this database are not selected from *all* possible word declensions or conjugations, the database cannot be said to be an exhaustive tabulation of all possible words. Indeed, even expressions with very similar meanings may include lexical verbs that are not contained in this database of scheduling-question words. To address this difficulty, we use a concept-association system: given a command sentence considered as a question-sentence candidate, we deem the sentence indeed to be a question sentence if any of the lexical verbs it contains bears a sufficiently strong association to any of the entries in the database of scheduling-question words.

More specifically, we compute the degrees of association between lexical verbs in input sentences and words in the database of scheduling-question words, and we deem the input sentence to be a question sentence if any of these degrees of association meet or exceed a certain threshold value.

To establish an appropriate number for this threshold value, we use the results of *X-A, B, C* tests conducted using test data of the form shown in Table 4. These data have the following structure: Denoting an arbitrary reference concept by *X*, we let *A* denote a concept bearing an extremely strong association with *X*, such as an exact synonym or a near synonym; we let *B* denote a concept that appears to have some association with *X* but not as strong an association as that of concept *A*; and we let *C* denote a concept that has absolutely no association with *X*. Treating this set of 4 concepts (*X-A, B, C*) as a single 4-tuple, we considered a set of 500 such 4-tuples—determined, by hand, to comport with common-sense human notions of associativity between concepts—and computed their degrees of association, then averaged these values over the entire 500-element data set. The average degree of association between *X* and *A* in our test data was 0.302, while the average degree of association between *X* and *B* was 0.0653; we take the median value between these two numbers, 0.183, as our threshold.

Table 4: Examples of typical 4-tuples used for *(X-A,B,C)* tests

X	*A*	*B*	*C*
Restaurant	Cafeteria	Customer	Talent
Beverage	Drink	Liquid	Selection
Sick person	Patient	Treatment	Magnet

3.3 Identification of appointment-question sentences

We next consider whether or not input sentences identified as questions by the methods discussed in Section 3.2 are in fact questions regarding appointments. In this paper, questions regarding appointments are termed "appointment-question sentences." First, "when" queries ask about timing, so we unconditionally identify such queries as appointment-question sentences. We also identify other 5W1H queries that are not "when" queries as appointment-question sentences, if they do not include phrases that refer to times in the past. More specifically, for each independent word in the input sentence we use a time-identification database to determine whether or not the word refers to time. Our "time-identification database" contains words related to time (time words) and the corresponding times they signify; a portion of this database is presented in Table 5.

Table 5: A portion of our time-identification database

Time word	Time
Tomorrow	*[Today]* + 1 day
End of year	*[This year]* 12/20 – 12/31
Summer	6/1 – 8/31
[A] / [B]	*[A] / [B]*

If a sentence does not contain any time-related words, then we deem it to be an appointment-question sentence. If a sentence does contain time-related words, an appropriate time specification is output for each independent word identified as time-related. For example, in a "where" query such as "Where will tomorrow's meeting take place?" the noun "tomorrow" will generate output for the corresponding date. This output is compared with the present date and time; if it does not refer to a date or time in the past, the sentence in question is identified as an appointment-question sentence. In addition, the sentence is identified as an appointment-question sentence if its lexical verbs are not in past tense form.

For queries other than 5W1H questions, and for command sentences, we carry out the above procedure, then check to see whether the sentence contains any words related to appointments (we refer to these as "appointment words"); if so, we identify the sentence as an appointment-question

sentence. For this purpose, we constructed a database of appointment words. This database is presented in Table 6.

Table 6: Database of appointment words

Appointment words
Appointment, free time, agenda, schedule, plan, errand, busy, full, free, agreement, reservation

3.4 Retrieving appointments relevant to scheduling questions

We next consider how to determine whether or not the schedule table includes an appointment containing the information required to answer a given appointment-question sentence.

First, if the question contains any time-related words, we compare the time to which the appointment-question sentence refers against time-related parameters contained in the schedule table (namely, the start time/date and the end time/date); appointments that match any are identified as candidates. To determine whether or not the input sentence contains any time-related words, we inspect all independent words in the sentence to ask whether or not they are contained in the time-identification database. For example, if on 2/10 we receive the input question "What are my appointments for tomorrow?" we note that the sentence contains the time-related word "tomorrow" and we determine that this word refers to the date 2/11. Then we search the parameters in the schedule table for all start and end dates on 2/11, identifying all appointments stored for that date as candidates. If the input sentence does not contain any time-related words, we take all scheduled appointments as candidates.

Next, from among the candidate appointments we identify those appointments for which the location, participants, and event description parameters match the appointment-question sentence. However, here we must be cognizant of the fact that the schedule table may not contain phrases that are *exact* matches for the phrases in the appointment-question sentence. For example, suppose the appointment-question sentence is "Where am I going swimming next week?" while a candidate appointment contains the event description "Public pool." In this case, there are no matching phrases, but any human would quickly recognize the associations and would correctly reference this appointment in responding to the question. Thus in computing the associativity between an appointment-question sentence and a candidate appointment, we utilize computations of the degree of association. More specifically, from the full list of independent words in an appointment-question sentence we eliminate time-related words, question words, and words on the elimination list in Table 7; we call the list of remaining words "question keywords," and we compute the degree of association between the question keywords and the full list of independent words contained in the "Location," "Participants," and "Event Description" fields of the candidate appointment, which we call "appointment keywords."

Table 7: Elimination list

Words on the elimination list
Do, go, meet, go to do, is, end, go to see, go to meet, go out, appointment, doing, conduct, see, come, get to know

At this time, we obtain a list of each of the question keywords and the corresponding appointment keywords for which the degree of association is highest. If this degree of association exceeds the threshold (the same value discussed in Section 3.2), we note the appointment keyword, and we select the scheduled appointments containing all such selected keywords. This yields a list of scheduled appointments that are related to the appointment-question sentence. If no such appointments are obtained, we refer to the database of question keywords in Table 7. If the appointment-question sentence contains only time-related words, question words, and entries in this database, then we select all scheduled appointments that were identified as candidates. For example, if the input sentence were "Tell me my appointments for next week," we would fetch all scheduled appointments for next week, even though none of the events in the schedule table for next week is directly related.

If no scheduled appointments are obtained, we output "No relevant appointments found."

3.5 Generating responses

Next, we construct responses based on the scheduled appointments identified by the procedure discussed in Section 3.4. We infer what is being asked about by considering the pattern of the appointment-question sentence, and then fill in the appropriate entries of a response template depending on the content of the appointment parameters.

Table 8: Templates for responding to questions

Appointment-question sentence	Response template
When queries	The event is from *[start time]* on *[start date]* to *[end time]* on *[end date]*.
What, Why, How queries	The event consists of *[event description]*.
Who queries	You will be *[input verb]*ing with *[participants]*.
Where queries	The event will take place at *[place]*.
Other appointment-question sentences	From *[start time]* on *[start date]* to *[end time]* on *[end date]*, you are scheduled to participate in *[event description]* at *[location]* with *[participants]*.

These response templates are shown in Figure 8. Portions of response templates written in *[italics]* describe schedule information that must be filled in to construct the response. If any pieces of schedule information needed to complete the response are not included in the schedule table entry, those items are excluded from use in the response. If *none* of the pieces of schedule information needed to complete the response are contained in the schedule entry, we output "I don't know, because that information is not in the schedule."

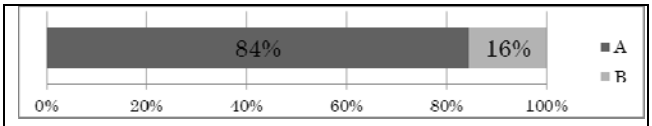
4 Test

We prepared a schedule table containing schedule information covering a two-month span (from 7/29 to 9/29), then showed this table to 15 test subjects and asked them to construct questions. Our set of evaluation sentences contains a total of 120 sentences, including 30 "when" queries, 30 "where" queries, 30 "who" queries, and 30 other question sentences including "what" queries. The answers to all 120 of these appointment-question sentences may be found within the appointment information contained in the schedule table.

In addition, for use in testing our system's ability to identify appointment-question sentences, we prepared an additional set of 45 sentences. These 45 sentences consist of 15 appointment-question sentences inquiring about matters not in the schedule table, 15 sentences that are questions but not appointment-question sentences (unrelated to scheduling), and 15 sentences that are not questions at all.

4.1 Testing the proper identification of appointment-question sentences

We first consider the 45 sentences that are not appointment-question sentences relating to appointments in the schedule, and we test whether or not the stage of our algorithm that serves to identify appointment-question sentences succeeded in correctly characterizing these sentences. The results of this test are shown in Figure 1.



A: Sentence was determined to be unrelated to schedules
B: A relevant appointment was determined to exist in the schedule table

Figure 1: Identification of appointment-question sentences

The following table presents some examples of successful and unsuccessful identifications of appointment-question sentences.

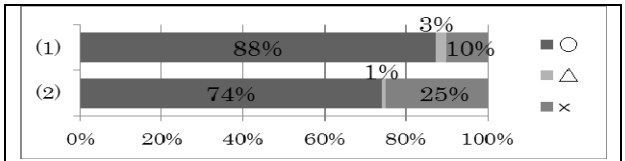
Table 9: Successful and unsuccessful examples of identification of appointment-question sentences

Successes	--Don't tell anyone. → Not a question sentence. --Would you like to go mountain climbing? → This is a question sentence, but it is not an appointment-question sentence. --What are my Christmas plans? → This is an appointment-question sentence, but it is not one for which a relevant appointment exists in the schedule table.
Failures	--When is payday? → The correct answer is "This is an appointment-question sentence, but not one for which a relevant appointment exists in the schedule table." However, the question has a high associativity with the word "part-time job" that exists in the schedule table, whereupon the system erroneously concluded that the schedule table contained an appointment relevant to this question.

4.2 Testing the proper retrieval of scheduled appointments

We next consider appointment-question sentences for which relevant appointments do exist in the schedule table, and we test whether our system is able to retrieve from the table the correct appointments needed to respond to these questions. As discussed above, our tests used a total of 120 sentences—including 30 "when" queries, 30 "where" queries," 30 "who" queries, and 30 other question sentences including "what" queries—and a two-month schedule table. It is possible to retrieve multiple appointments. In the figure below, the symbol ○ denotes cases in which all appropriate appointments were retrieved; Δ denotes cases in which not *all* appropriate appointments were retrieved, but at least *one* correct appointment was retrieved; and × denotes cases in which not even one appointment was retrieved, or in which all retrieved appointments were irrelevant for the question at hand.

Figure 2 compares results obtained using the proposed method against results obtained using a simple notation-matching method to retrieve appointments. The notation-matching method simply retrieves all appointments in the schedule table that contain all words whose notations match the question keywords.



(1) Proposed method
(2) Notation-matching method

Figure 2: Results of appointment-retrieval tests

Table 11 presents some cases of successes and failures obtained with the proposed method. Table 10 displays a portion of the content of the schedule table used in this example.

Table 10: Partial example of sample schedule table

ID	Start date	End date	Start time	End time
1	8/4		20:00	
2	8/7		10:00	18:00
3	8/12		12:00	
4	8/14		10:00	18:00
5	8/16		12:00	
6	9/13		18:00	

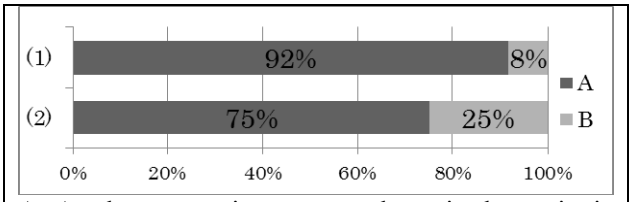
ID	Location	Participants	Event description
1	Sports park	Girlfriend	Star gazing
2	Convenience store		Part-time job
3	Ohta Dental Clinic		Dentist
4	Convenience store		Part-time job
5	Osaka		Haircut
6	Shimizu's house	Shimizu, Nakayama	Mahjong

Table 11: Success cases and failure cases obtained with the proposed method

	Question	ID of appointment retrieved	Appointment keyword for which association was identified
Successes	When am I going to see the night sky?	1	Star
	Where am I getting treatment for my tooth?	3	Dentist / dentist
Failures	When am I getting my hair cut?	None	Haircut / Mahjong
	Which days do I not go to my part-time job?	2,4	Part-time job

5 Discussion

The results of Section 4.1 indicate that our proposed method achieved an 84% success rate in accurately identifying sentences that are *not* appointment-question sentences for which a relevant appointment exists in the schedule table. The method failed in cases where some association existed between the appointment-question sentence and an appointment in the table, as shown in the example in Table 9. In contrast, the notation-matching method—which does not consider such associations—achieved a success rate of 100% accuracy at this stage. We next considered a set of 120 appointment-question sentences for which a relevant appointment exists in the schedule table, and we tested the accuracy with which these sentences are identified by **(1)** our proposed method, and **(2)** the simple notation-matching method. The notation-matching method simply retrieved all appointments in the schedule table that contain all words whose notations match the question keywords. The results are shown in Figure 3.



A: A relevant appointment was determined to exist in the schedule table
B: A relevant appointment was determined not to exist in the schedule table

Figure 3: Identification of appointment-question sentences for which relevant appointments exist in the schedule table.

We see that our proposed method exhibits excellent associative capabilities, is robust with respect to variations in word choice, and is able to consider associations between appointment-question sentences and scheduled appointments in the process of identifying those questions. In contrast, the simple notation-matching method is incapable of considering associativity; for this reason, it is unable to identify appointment-question sentences that exhibit associations with scheduled appointments, but on the other hand it succeeds in entirely eliminating all question sentences that bear no relation to scheduled appointments.

The results of Section 4.2 indicate that our proposed method performs much better than simple notation-matching method in tests of retrieving scheduled appointments. As noted above, this may be attributed to the ability of our proposed method to consider associations between appointment-question sentences and scheduled events. For example, as indicated in Table 11, in response to the appointment-question sentence "When am I going to see the night sky?" our system was capable of retrieving an appointment described as "star gazing," which is not considered by the notation-matching method. On the other hand, as examples of situations in which our system fails we cite cases where words could be identified which have strong associations—such as the words "cut" and "Mahjong"—as well as sentences that include negating words, which could not be handled, such as "Which days do I not go to my part-time job?" To correct such deficiencies it may be necessary to go beyond calculating degrees of association for question keywords individually, instead regarding all question

keywords in a sentence as a single set and then calculating the degree of association with due consideration to any negating words.

6 Conclusion

In this paper we introduced the notion of concept association for systems designed to answer scheduling questions, and we proposed a specific question-answering system based on relationships between sentences and schedules. In tests, our proposed method achieved a schedule-retrieval accuracy of 88%, an improvement of 14 percentage points over the 74% rate achieved by a simple notation-matching method. Based on these results, we conclude that our proposed method exhibits excellent associative capabilities and is capable of retrieving scheduled appointments by considering their association with a given appointment-question sentence, thus offering the promise of question-answering systems with broader applicability than previously possible.

Acknowledgment

This research has been partially supported by the Ministry of Education, Science, Sports and Culture, Grant-in-Aid for Scientific Research (Young Scientists (B), 24700215).

References

[1] E.Yoshimura, M.Imono, S.Tsuchiya, H.Watabe, :Association Reply Method for Intelligent Conversation Processing, *Journal of JSAI*, Vol.28, No.2, pp.100-111, 2013.

[2] Siri - Apple, http://www.apple.com/jp/iphone/features/siri.html, 2014.02 access.

[3] i-concier – NTT docomo, https://www.nttdocomo.co.jp/service/information/shabette_concier/, 2014.02 access

[4] H.Watabe, N.Okumura, T.Kawaoka,: The Method of Measuring the Degree of Association between Concepts using Attributes of the Concepts and Coincidence Information, *Journal of natural language processing*, Vol.13,No.1, pp.53-74, 2006.

[5] H.Watabe and T.Lawapla: The Degree of Association between Concepts using the Chain of Concepts", Proc. Of SMC2001, pp.877-881, 2001.

[6] J.Okamoto, S.Ishizaki,: Construction of Associative Concept Dictionary with Distance Information, and Comparison with Electronic Concept Dictionary, *Journal of natural language processing*, Vol.8, No.4,pp.37-54, 2001.

[7] T.Araki, N.Okumura, H.Watabe, T.Kawaoka: Dynamic Calculation Method of Degree of Association Considering the Common Attributes of Target Concepts, *Science and Engineering Research Institute of Doshisha University* 48(3), pp.140-150, 2007.

Graph-based Model for Topic Detection

Ang Zhao[1], Xin Lin[2], Jing Yang[2]
Department of Computer Science and Technology
East China Normal University, Shanghai, P.R. China
[1]angzh@ica.stc.sh.cn, [2]xlin@cs.ecnu.edu.cn, [2]jyang@cs.ecnu.edu.cn

***Abstract*—In this paper, a novel graph-based model (GBM) is proposed for topic detecting. Different from existing statistical methods, our proposed model considers more semantic factors which combines named entity and dependency relation between words derived from a dependency parse tree. In our model, a graph is constructed for representing words and their association. By utilizing spectral clustering algorithm, we get clusters of words, each cluster represents a topic respectively. Our contribution includes as follows: modeling the topic detection problem as a graph-partitioning problem; proposing a new method of ranking the words association, and based on that, the document collection is represented as an undirected weighted graph. The performance of experiment task for dimensionality reduction and text classification indicates the feasibility and potentiality of our method.**

***Index Terms*—topic model; graph partitioning; words association; dimensionality reduction; text classification;**

I. INTRODUCTION

The rapid developing of Internet brings vast amounts of unstructured texts in the form of news articles, Web pages, social network sites and etc. It's a particularly challenging task with a simple search to find a user's interest, especially the hidden thematic-level information in the returned results related by keywords. In this process, an important part is to learn the semantic information from the document texts and capture the links between them within a topic. Thematic information discovered on the archives can vary from different level. An all-around business website provides broader-level themes, which might be corresponding to a list of sections in the homepage. A broad theme can be zoomed out to various aspects, for example, sports section has subsections such as the Badminton World Cup and Australian Open sometime; world politics can be subdivided into the Korean Peninsula nuclear issue, the Middle East situation and so on. It's quite reasonable to track the specific theme varying with time. The thematic structure would offer a new form to explore and digest the collection. While more and more texts of various are available online, it's not convenient for human to read and study them to provide the hidden knowledge of that kind described above for a user's browsing preference. Topic models, such as Latent Dirichlet Allocation (LDA)[1], can uncover the specific or broader hidden thematic structure that pervades a large, unstructured document collection and capture the links or topical features. It has emerged as a popular solution to understand the disordered information. However, in a semantic aspect, some named entities and the information co-occur with them are actually expressing varying degrees of significance comparing to general words. Particularly, in texts of news broadcast, the differential may be ignored or treated indiscriminately during topic discovery with standard statistical topic model aforementioned. In this case, it would be reasonable to take some explicit semantic information such as named entities into account. We refer to the task of topic modeling considering linguistic notions, such as "semantic" and "syntactic", other than absolutely "statistical" .

Semantic information, especially entity information, is potential for our work. The entity is referred to specific noun such as person name, place name, organization name and so on. Some words may have syntactic relation with each other, such as dependency relation. The presence of relation could be directly or indirectly passing by other words. For example, a person entity "Tiger Woods" is very likely to co-occur with sport topic words such as "golfer", "Championship" or sportswear maker "Nike", in the meantime, they may be also in a dependency relation within the sentence. And the word "Championship" could be also connected with a person entity like "Rory McIlroy" by a dependency relation. Here we assume that these words are topically related to varying degrees, and the experiment results prove it as well. The hypothesis "One Sense Per Discourse"[2], which focuses on the word sense disambiguation problem, confirmed that a polysemous word appears several times in a discourse is extremely likely that they will all share the same sense. Inspired by it, we extend the scope of "Discourse" to "Topic" and suppose that "a topic of certain type is likely to contain entities and words of topical-related around them". Then we build a series of correlation evaluation rules to rank the association between words. Based on the correlation evaluation rules among terms in a sentence level, an entity-centroid graph is constructed for coarse-grained topic, and then clustering method is applied to aggregate the words. Each cluster shares the similar topic which could be represented by a group of words of different importance.

The rest of the paper is organized as follows. Section II gives the review of related research. In section III we give detailed definition of graph based model. The evaluate methods and comparative experiments are presented in section IV.

Conclusion and future work are described in section V.

II. RELATED WORK

Topic models have emerged as an increasing useful tool to find short descriptions and preserve the essential statistical relationships of the collections of discrete data. The effort of mining the hidden structure in large collections can be dated from Latent Semantic Analysis (LSA)[3]. In LSA, a mathematical technique called the singular value decomposition of the document-term matrix is used to identify a linear subspace in the space of TF-IDF features and preserve the similarity structure among rows to reveal the major associative words' patterns. A significant step forward in this field was made by Hofmann[4], who presented the probabilistic latent semantic analysis (pLSA), also known as the aspect model. The pLSA approach models each document as a list of mixing proportions for the mixture components, where the mixture components are multinomial random variables that can be viewed as representations of "topics". Each word in a document is generated from a single topic, and different words in a document may be generated from different topics. Thus, each document is reduced to a probability distribution on a fixed set of topics. This distribution is a "short description" associated with the document. However, in pLSA, there is no probability distribution over the numbers of topic, which leads to mainly two problems: first, the number of parameters in the model grows linearly with the size of the corpus, which results in overfitting; second, it is unclear how to assign probability to a document in the testing set. To extend pLSA, Latent Dirichlet Allocation (LDA) [1], a three-level hierarchical Bayesian model, adds Dirichlet priors on topic distributions, resulting in a more complete generative model. Due to its nice generalization ability and extensibility, LDA achieves great success in text mining and related domain of information retrieval. In recent years, many customized and complicated variants and extensions of LDA and pLSA have been proposed, such as the author-topic model [5], a Bayesian nonparametric topic model [6] inferred the unknown topic number automatically using the clustering property of the Dirichlet process, and some improving work with labeled data[7], regularized topic model [8]. [9] developed a continuous time dynamic topic mode which using Brownian motion to model the latent topics through a sequential collection of documents.

Topic modeling could be viewed as a kind of clustering problem[10]. A topic (i.e., a cluster) is a distribution across words while documents are viewed as distributions across topic. [11] proposed a multi-grain clustering topic model (MGCTM) which integrates document clustering and topic modeling into a unified framework and jointly performs the two tasks to achieve the overall best performance. An important aspect of clustering is computing the similarity or association between data. The study on words similarity can be classified in primarily three groups: rule based methods, such as [12] , used the distance to assess the conceptual distance between sets of concepts when used on a semantic net of hierarchical relations; knowledge based methods, such as static knowledge base HowNet, dynamic online knowledge base Wikipedia and so on, which are mainly used to measure the relatedness of concepts, but expensive manual work is needed to maintain and can't handle with unknown words; statistical methods, such as Mutual Information and Google similarity distance[13] mainly based on the frequency of pairwise words to evaluate the relatedness.

III. GRAPH-BASED MODEL FOR TOPIC DETECTING

We build a weighted graph to present the words' relation. The relation mainly includes two aspects: entity-centroid relation and dependency relation. We consider topic detection problem as a graph partition issue solved by spectral clustering algorithm. In this section, first of all, we introduce the motivation of constructing the graph-based model. Then we specify the algorithm of modeling the documents as a weighted graph. Finally, a short description for graph partition using spectral clustering algorithm[14] is presented.

A. Motivation Of "One Sense Per Topic"

Bayesian conditional probability model provides a unifying framework for capturing the relationships among the random variables, and it is also the theoretical basis of the LDA topic model. Generally, a theme existing in a discourse of a text or utterance could be summarized as topic words or key words. These words express the intuition of semantical level rather than the lexical level of "what actually has been said or written". Do these words have the property to reveal the same or similar topic in other similar discourse? And how to evaluate the association between these words? We hypothesize that not only some term-pairs co-occur frequently, which agrees with the pLSA model, but also co-occur at short range or have dependency relation within a sentence. The line of analysis makes a link to the key point of our work for constructing a graph.

B. Document Representation

We use a rich document representation that employs a graph structure obtained by augmenting the syntactic dependency analysis of document's sentence to deal with complex relationships between data objects. The documents are represented as a graph $G_D = (V_D, E_D)$, where vertex set V_D in this graph represents the word set. Two vertices are connected if the similarity s_{ij} between the corresponding data vertices v_i and v_j are positive or larger than a certain threshold. The problem of clustering can be transformed into a graph partition or community detection problem that edges between different groups have low weights and edges within the same group have high weights. Finally, points (i.e. words) in the same group are similar to each other

while points in different groups are dissimilar to each other. We use the weighted adjacency matrix of the graph to formalize this intuition, which is a symmetric adjacency matrix $W_D = (w_{ij})_{(i,j=1,\cdots,n)}$, each edge between two vertices v_i and v_j carries a non-negative weight $w_{ij} \geq 0$. The vertices can't be connected with themselves. As G_D is undirected, it is obvious that $w_{ij} = w_{ji}$. The degree of a vertex v_i is defined as $d_i = \sum_{j=1}^{n} w_{ij} (0 < i \leq n)$ and the degree matrix D is defined as the diagonal matrix with $d_1 \cdots d_n$ on the diagonal. The method of evaluating the weight is introduced in detail as follows.

We evaluate the weight of association value between two words covering syntactic dependency relation and position distance in a sentence level. A weighted summation score of pairwise words within all sentences of document collection is the final result. Important words, especially words associating with named entities play a role of centroid of a graph. We divide the weight w_{ij} between words v_i, v_j (i.e. vertex of the graph) into two parts:

- If two words (v_i, v_j) have dependency relation, we use the equation

$$Er_{(v_i,v_j)} = \frac{\cos\left(\pi \cdot \frac{1+dis}{maxLen}\right)}{\ln(e+dis)} + \lambda \tag{1}$$

to denote the association weight between any words which have a dependency relation. At least one of v_i and v_j is a named entity, while $maxLen$ is a constant of the maximum length of a sentence and dis refers to the natural position distance within a sentence. λ is a regularization parameter $\lambda \in (0, 1)$, where λ=λ_1 if either v_i or v_j is named entity, or λ=λ_2 if neither v_i nor v_j are named entity. The parameter λ_1, λ_2 meet the condition: $0 < \lambda_2 < \lambda_1 < 1$, here they are tuned to 0.25 and 0.4 respectively.

$$Nr_{(v_i,v_j)} = \frac{1}{1+dis} \tag{2}$$

equation (2) denotes the natural association relation weight of natural path distance within a sentence;

- The result association weight between two words v_i,v_j is defined as the weighted summation of $Er_{(v_i,v_j)}$ and $Nr_{(v_i,v_j)}$

$$W_{ij} = \sum_{1}^{N} w_{ij} \cdot \ln \frac{N}{1 + cr_{(v_i,v_j)}} \tag{3}$$

where

$$w_{ij} = \theta \cdot Er_{(v_i,v_j)} + (1-\theta) \cdot Nr_{pairs(v_i,v_j)} \tag{4}$$

In the above equation, θ denotes a regularization parameter, here it is set as 0.5, N is the sentence number of all documents in corpus, $cr(v_i, v_j)$ is the co-occur number of two specific words (v_i, v_j). Er equals to $Er1$ or $Er2$ corresponding to λ=λ_1 or λ=λ_2 respectively. The values of the two types of associate relations are evaluated in a tendency of $Er > Nr$, and the value gradually decreases with the increase of position distance.

As described above, we weight two words' association in a sentence level, and mainly focus on both the entities and the information surrounding. The algorithm for building any two nodes' association weight is depicted as following pseudo code shown in Algorithm 1. In Figure 1, different colors represent different entity-centroid based topic discourse, edges represent the association relation among words.

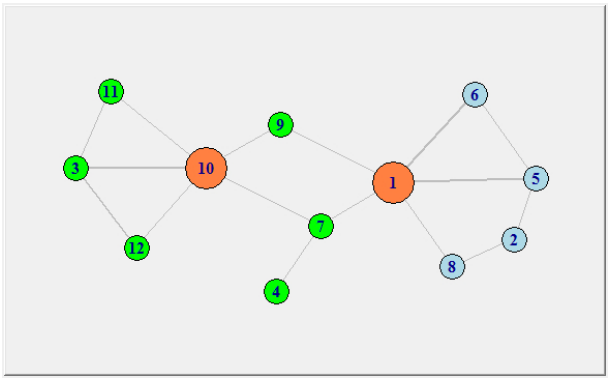


Figure 1: *the presentation of graph based words association model, brown dark vertices represents entities, others are raw terms.*

For clustering points in R^n, the intuitive goal of clustering is to divide the data points into meaningful groups. One standard method is based on generative models to generate k clusters then determine the corresponding centers or determine the centers directly, in which algorithms such as EM are used to learn a mixture density. These methods suffer from several drawbacks: first, harsh simplifying assumptions to parametric density estimators usually need to be made (e.g., the density of each cluster is Gaussian); second, the result is sensitive to initial condition, which often leads to different results, and it needs multiple restarts to establish the best solution while using iterative algorithms; third, the likelihood can have many local minima and trap in local optimum. Algorithms such as K-means have similar problems. A proper alternative for our work is to use spectral methods for clustering.

C. Spectral Clustering Algorithm

Spectral clustering is a kind of clustering method based on graph theory, which makes use of the eigenvalues and their corresponding eigenvectors of the normalized graph Laplacian matrix of the data to perform dimensionality reduction before clustering in fewer dimensions. Given a set of points $S = \{x_1, \cdots, x_n\}$ in R^n, applied the pairwise similarity function $S_{ij} = similar(S_i, S_j)$ which is symmetric and non-negative, and the corresponding similarity matrix is denoted by $S = (s_{ij})_{i,j=1\cdots n}$.

Algorithm 1 pseudo-code for constructing association algorithm

Input: D: documents collection, $D \in R^{(n\times d)}$
Output: V_D: documents' all word features vector space set, $V_D \in R^{(d\times d)}$
Output: any two words' association score w_{ij} included in set W_D, $w_{ij} \in R^+$

Set $w_{ij} \leftarrow 0$
Set $cr_{(v_i,v_j)} \leftarrow 0$
for all sentence $s_k \in D$ **do**
  for all pairwise $v_i, v_j \in s_k$ **do**
    V_D add v_i, v_j
    $cr_{(v_i,v_j)} \leftarrow cr_{(v_i,v_j)} + 1$
    if pairwise v_i, v_j has dependency relation **then**
      if v_i *or* v_j is named entity **then**
        $Er_{ij} \leftarrow Er1(v_i, v_j)$
      else
        $Er_{ij} \leftarrow Er2(v_i, v_j)$
      end if
    else
      $En_{ij} \leftarrow Nr(v_i, v_j)$
    end if
    $w_{ij} \leftarrow w_{ij} + Er_{ij} \cdot \theta + En_{ij} \cdot (1 - \theta)$
  end for
end for
for all pairwise $v_i, v_j \in V_D$ **do**
  $W_{ij} \leftarrow w_{ij} \cdot \frac{N}{1+cr_{(v_i,v_j)}}$
end for
Return W_{ij}

Input: Similarity matrix S, number k of clusters to construct.

1) Construct a similarity graph. Let W be its weighted adjacency matrix and D the degree matrix.
2) Computer the graph Laplacian $L = D - W$.
3) Computer the first k smallest eigenvectors $u_1, \cdots, u_n$ of L (i.e. the top k biggest eigenvalues).
4) Let $U \in R^{n\times k}$ be the matrix containing the vectors $u_1, \cdots, u_n$ as columns.
5) For $i = 1, \cdots, n$, let $y_i \in R^k$ be the vector corresponding to the i-th row of U.
6) Cluster the points $(y_i)_{i=1\cdots n}$ in R^k with K-means algorithm into clusters $C_1, \cdots, C_n$

Output: Clusters $A_1, \cdots, A_n$ with $A_i = \{j \mid y_i \in C_i\}$

IV. EXPERIMENT AND RESULT

In this section, we first describe the task and the experimental settings, then evaluate our graph based model comparing with baseline system.

A. Experimental Settings and Task Description

We utilize the SogouC.Reduced.20061102 corpus[1] as our test bed, which contains 9 categories (sports, culture, job, education, military, IT, healthy, finance and travel), each category contains 1990 documents. Topic detection result is evaluated as a feature dimensionality reduction problem. 300 documents per category are randomly selected and build a relatively coarse-grained topic model, the topic number k is set to 9 by default. The preparative work on the text collection, includes word segmentation, named entity recognition and the dependency tree parsing under the LTP-Cloud platform [2]. Similar to standard latent semantic analysis, our job focuses on extracting and representing the contextual-usage meaning of words for per topic.

B. Evaluation

We build a Gibbs sampling LDA model(gLDA)[15] as baseline, and extract the top N important words for the first task on both methods. In the second task, a SVM classifier is applied to binary classification after dimensionality reduction.

1) Empirical contrast: We rank the words and obtain the following sets of words for per topic. The topic names are manually annotated with being aware of the corpus' categories. We obtain the topic word sets of different proportion of data on both models.

Table I shows one of the comparative results by manual labeling the words' topic tags. If the word list is informative enough to predict their topic tag, then labeling them with a tag or labeling them with a "else" for an unknown one. Notice that, LDA's keywords of each topic in a probabilistic order, and a word may belong to different topics. In our method, a keyword belong to specific topic only and can be weighted in order. Figure 2 shows the labeling results of different proportion of data, if the number of topics manual labeled is closer to corpus' category number 9, it suggests the "recall" is higher. From Figure 2 we can note that our method achieves an improvement relatively. Next, we introduce a classifier to calculate "precision", the performance of a classification model is determined by feature selection, which can judge the performance of the words extracted from per topic.

2) Document classification: In text classification problem, a challenging aspect is the choice of features, and it's our focus in this part. We utilize these two models for dimensionality reduction, by using the Gibbs sampling LDA model and the graph-based model for each class, then a generative model for classification was obtained. We conducted two binary classification experiments by training a support vector machine (SVM) on the low-dimensional word features(65 words per topic) induced by a Gibbs sampling

[1] http://www.sogou.com/labs/dl/c.html
[2] http://www.ltp-cloud.com/

Table I: list of words by topics using the Graph-based Model and the Gibbs Sampling LDA with data proportion 50 documents out of 300 per category

Model	Manual Labeling	Lists of Words
the Graph based model	体育(sports)	投篮 詹姆斯 防守 篮板 得到 接到 犯规 分 取得 攻 传球 戈登 休斯 右翼 拉希德·华莱士
	其他(else)	分子 风格 提 眼镜 恐怖 浪漫 中央 球迷 大西洋 教练 小姐 护理 能否 穿越 隐形
	旅游(travel)	旅游 游客 增长 收入 共 元 接待 期间 五一 黄金周 黄金 周 今年 人次 旅客
	财经(finance)	中国 最 市场 公司 没有 目前 里 好 已经 时间 做 这种 出 美元 生活
	文化(culture)	搭配 春装 供应 土地 笔者 本报 依旧 公园 中学 逛 总经理 黑色 朱熹 网上 卡拉罗
	军事(military)	训练 直播员 搜狐 特战 部队 武器 装备 情况 队员 部分 解放军 展示 优秀 必须 逼近
	其他(else)	本地 分钟 通话 政策 含 通话费 超出 电子 力度航空 人员 邮件 控制 预防 套餐
	健康(health)	药品 管理 部门 监督 经营 医院 药柜 企业 进行艾滋病 市中心 食品 国家 相关 安康
	招聘(job)	型 职业 男性 男 女性 敬业 职业女 职业男 精心 策划 女 指数 心理 朋友 测验
the Gibbs Sampling LDA	健康(health)	作用 药物 吃 治疗 妈妈 使 大量 名 患者 使用 眼睛 发现 食用 黑色 最
	文化(culture)	中国 社会 朱熹 历史 没有 文化 却 当时 先生 西方 里 同性恋 关系 认为 正
	旅游(travel)	旅游 游客 北京 元 增长 五一 黄金 消费 期间 接待 黄金周 收入 记者 人次 城市
	体育(sports)	搜狐 直播员 分 节 考生 篮板 詹姆斯 投篮 得分 高考 犯规 第三 防守 第一 考试
	军事(military)	训练 美国 部队 导弹 伊朗 装备 系统 战争 武器 进行 军事 特战 战斗 支 俄罗斯
	招聘(job)	敬业 员工 指数 工作 发展 重要 媒体 企业 学生 斯大林 不同 精神 人才 教育 西沙
	其他(else)	去 做 工作 没有 好 却 看 里 想 页 问题 没 可能 最 天
	财经(finance)	公司 市场 中国 企业 元 产品 问题 家 目前 药品 投资 进行 记者 美元 价格
	体育(sports)	印度 比赛 国家队 没有 球员 赛季 长江 球队 俱乐部 中国 已经 轮 场 出 最

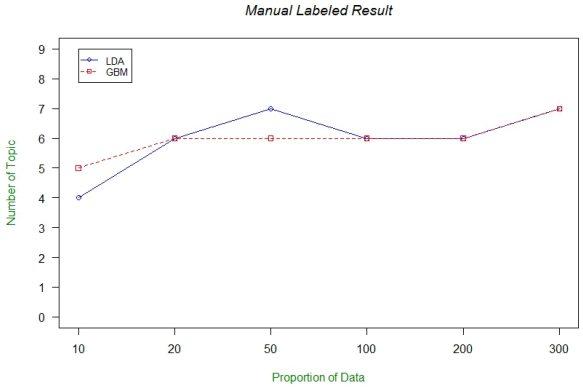


Figure 2: *the discovered number of topics by human annotated in different proportion of data.*

LDA and a graph-based model respectively and compared this SVM compared to an SVM trained on all the word features. Notice that the feature space is reduced by about 99 percent in this case.

Table II: classification results on two binary classification problems for different proportion of training data. We choose topics: Finance Vs. Sports and Finance Vs. Travel.

precision (SVM binary classification)						
	Fiance Vs. Sports			Finance Vs. Travel		
Prop.	allword	gLDA	GBM	allword	gLDA	GBM
5%	58.75%	63.50%	67.00%	52.90%	51.10%	51.10%
10%	55.85%	87.90%	91.42%	54.70%	61.20%	61.50%
15%	71.20%	90.90%	87.27%	73.10%	68.60%	74.90%
20%	81.28%	88.36%	88.60%	83.50%	83.40%	84.80%
25%	85.68%	85.50%	83.23%	85.90%	94.80%	91.63%

The table II above shows the classification results. Both models achieve an overall improvement over all-word features model in classification performance, though the feature space is obviously reduced. With the increase of the data proportion for training, the performance of all words features improves steadily, while both the Gibbs sampling LDA and the GBM show some fluctuations, which may be caused by the quality of the training texts' potential dissimilarity. The performance between the Gibbs Sampling LDA and the GBM has very subtle difference with the change of topic. We can conclude that the topic-based representation provided by the GBM is appropriate as a filtering algorithm for feature selection in text classification.

V. CONCLUSION AND FUTURE WORK

In this paper, we proposed a graph-based model for topic detecting, and treat it as a feature dimensionality reduction issue. Our contributions mainly include two parts: first, we proposed a new rule for evaluating the association between words from an entity-centroid perspective; second, we build a graph-based topic model based on the words-association, and it's viewed as a clustering issue solved by spectral clustering algorithm. We demonstrate the effectiveness of our proposed model by comparing it with the Gibbs sampling LDA. Manual evaluation on Chinese Sogou corpus illustrated the effectiveness of the proposed topic detection model.

One can improve in several directions for our work. One basic aspect of our work is words association evaluating, the position association and dependency association need further substantiation. Another sophisticated direction is converting a broad topic into some specific topics. Different topics can range in different scope, such as the culture topic, which is more likely to have a relatively wider range than the sports topic. Thus a necessary preprocessing is needed, for instance, utilizing hierarchical clustering algorithm to transform a broad topic into narrow ones.

VI. ACKNOWLEDGEMENT

This research is supported by Shanghai Postdoctoral Scientific Program (No.13R21412900) and China Postdoctoral Science Foundation (No. 2013M540348).

REFERENCES

[1] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *the Journal of machine Learning research*, vol. 3, pp. 993–1022, 2003.

[2] W. A. Gale, K. W. Church, and D. Yarowsky, "One sense per discourse," in *Proceedings of the workshop on Speech and Natural Language*. Association for Computational Linguistics, 1992, pp. 233–237.

[3] S. C. Deerwester, S. T. Dumais, T. K. Landauer, G. W. Furnas, and R. A. Harshman, "Indexing by latent semantic analysis," *JASIS*, vol. 41, no. 6, pp. 391–407, 1990.

[4] T. Hofmann, "Probabilistic latent semantic indexing," in *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 1999, pp. 50–57.

[5] M. Rosen-Zvi, T. Griffiths, M. Steyvers, and P. Smyth, "The author-topic model for authors and documents," in *Proceedings of the 20th conference on Uncertainty in artificial intelligence*. AUAI Press, 2004, pp. 487–494.

[6] Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei, "Hierarchical dirichlet processes," *Journal of the american statistical association*, vol. 101, no. 476, 2006.

[7] D. M. Blei and J. D. McAuliffe, "Supervised topic models," *arXiv preprint arXiv:1003.0783*, 2010.

[8] D. Newman, E. V. Bonilla, and W. L. Buntine, "Improving topic coherence with regularized topic models." in *NIPS*, 2011, pp. 496–504.

[9] C. Wang, D. Blei, and D. Heckerman, "Continuous time dynamic topic models," *arXiv preprint arXiv:1206.3298*, 2012.

[10] Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei, "Sharing clusters among related groups: Hierarchical dirichlet processes." in *NIPS*, 2004.

[11] P. Xie and E. P. Xing, "Integrating document clustering and topic modeling," *arXiv preprint arXiv:1309.6874*, 2013.

[12] R. Rada, H. Mili, E. Bicknell, and M. Blettner, "Development and application of a metric on semantic nets," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 19, no. 1, pp. 17–30, 1989.

[13] R. L. Cilibrasi and P. M. Vitanyi, "The google similarity distance," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 19, no. 3, pp. 370–383, 2007.

[14] A. Y. Ng, M. I. Jordan, Y. Weiss *et al.*, "On spectral clustering: Analysis and an algorithm," *Advances in neural information processing systems*, vol. 2, pp. 849–856, 2002.

[15] T. Griffiths, "Gibbs sampling in the generative model of latent dirichlet allocation," *Standford University*, vol. 518, no. 11, pp. 1–3, 2002.

SESSION

ROBOTICS AND APPLICATIONS + BCI + PATTERN RECOGNITION AND DIMENSIONALITY REDUCTION METHODS

Chair(s)

TBA

Cloud Infrastructure for Mind-Machine Interface

Chris S. Crawford, Marvin Andujar, Sekou Remy, Juan E. Gilbert
Human-Centered Computing Division
Clemson University
Clemson, SC, USA
cscrawf, manduja, sremy, juan@ {clemson.edu}

***Abstract*— There are several implementations of Brain-Computer Interfaces (BCI) systems to control robots. Although there are a wide variety of implementations, there is minimal work on researching infrastructures that work independent of hardware. Most systems presented in previous work are built to work with specific hardware components. The infrastructure presented in this paper serves as an alternative to these approaches and has the ability to function independent of hardware. This paper gives a thorough infrastructure design and presents results collected while using the infrastructure to control a robot with a non-invasive EEG device.**

Keywords—Brain-Computer Interface; BCI; Brain-Robot Interaction; BRI; ROS; Robots

I. INTRODUCTION

Brain-Computer Interface (BCI) is known as the measurement of the central nervous system (CNS) activity, which translate into an artificial output CNS that replaces, restores, enhances, supplements or improves the natural CNS output [1]. These functions serve as the five applications that affect the CNS. There has been a growth in BCI publications from the year 2001 in different areas of the field [2]. One of the areas is Brain-Robot Interaction (BRI), which is based on the five applications that affect the CNS. The area of BRI has also had a significant rise in number of publications recently [13].

BRI consists of two main input modalities: goal selection and process control. With goal selection the user through the BCI communicate their intent to the application. On the other hand, during process control the BCI handles the complete process of mapping user intent to control. Process control is often considered difficult because there is no stimulus aiding the user.

Many of the implementations in BCI publicaitons are based on P300 BCIs based on intermittent flashes, Steady State Visual Evoked Potential (SSVEP) based on ocillating lights, and Event Related Desynchronization (ERD) based on imagined movement [17]. These applications includes both local and web implementations.

While there have been many more papers published, there is no common implementation framework that researchers can use to quickly evaluate or compare scholarly contributions.

This paper presents an alternative implementation that leverages web and modern robot technology to provide an infrastructure that functions independent of hardware components to address this issue. The infrastructure this paper presents is concentrated in the supplement application [1], which is the use of BCI as an input to facilitate user's needs. This infrastructure is built as a goal selection which is a common method implemented for mind-machine control. Specifically, this implementation uses the Hypertext Transfer Protocol (HTTP) to send cognitive intent? to a robot which allows users to control it remotely. Data from the robot is presented to the user via HTML and other web standards. The use of these standards provides a flexible and robust framework that can be used to support BCI experimental work as well as enable the research community to readily validate published results.

II. RELATED WORK

BRI related work is based in several areas such as medical, gaming, and others. This paper focuses on previous local and web implementations and how they differ from the infrastructure this paper is proposing.

A. Local Implementation

Much of the related work has involved controlling a machine with a BCI device locally. Carlon and Millan implemented a brain-controlled wheelchair robotic architecture for those with severe motor disabilities. In their work they investigated how participants performed while using an asynchronous motor-imagery based BCI protocol [3]. Though this work showed improvement over other similar implementations, the authors reported that inexperince users had issues utilizing it. Bell et. al. presented a non-invasive BCI for controlling a semi-autonomous humanoid robot. In their work a locally controlled humanoid allowed users to navigate, manipulate and transport objects [4]. Though the users of this system were able to complete complex tasks, their performance depended greatly on the autonomy of the robot and sitmulus presentation. Another work related to robot assistance is Galan et. al.. They used a non-invasive EEG-based BCI to control a simulated wheelchair in a 3D environment for continuous mental control. Their work demonstrated the feasibility of continuously controlling complex simulated and physical robots using a non-invasive BCI [5].

B. Web Implementation

There has also been work that leverages the web to provide users the ability to control machines remotely. Nicolelis and Chapin demonstrated and early example of how a primate could control machines over the web using an invasive BCI device [6]. In their work they were able to train the primate to

control a machine 600 miles away using electrical signals from the brain. Even though this work shows positive results, invasive BCI is often not a feasible option for humans. Escolano et. al. also implemented a web based BCI system that allowed users to control robots remotely. The system they implemented allows users to perform navigation and exploration tasks[7]. This system performed well, but included an autonomous driving system that could conflict with desired behavior from the user. Vourvopoulos and Liarokapis used two off-the-shelf BCI wireless devices (NeuroSky and Emotiv) to control a Lego Mindstorm NXT robot. Their project was based on two experiments, which were to test each BCI device while controlling the Lego Mindstorm robot [8]. Afterwards, the authors conducted a user study. The users controlled a 3D simulated robot, where the data recorded was their ability to manipulate the robot, the application responsiveness, naturalness mechanism, and the user interactions with the robot [9]. Chella et. al. work involved a paralyzed patient that used the P300 brain signal to control a physical robot called PeopleBot. The authors provided a two-dimensional map that showed the robot's current location information that was viewable by the user [10]. Thobbi et. al. used video goggles as the projection of the robot movement while controlling the NAO robot with the Emotiv BCI device [11].

C. *Infrastructure Approach*

Although there has been previous work done on controlling robots through various types of BCI devices (invasive, non-wireless non-invasive, off-the shelf wireless wearable non-invasive BCI devices), there has been minimal investigation on the optimal way to connect these two types of hardware components. Research on this topic could help move the control aspect of BCI beyond the tradional lab setting. Exploring posibilities of integrating BCI devices, web technologies (i.e. HTTP, WebSockets, WebRTC), and display modalities (i.e. tablet, phone, computer) is key to enhancing user experince during telepresence secnarios. This paper presents an infractructure that allows a user to control a remote robot over the web using a BCI device. By leveraging recent advances in robotics and web technology the possibility of getting visual feedback from a remote robot is demonstrated. This paper also reports data collected while using this infrastructure.

III. INFRASTRUCTURE DESIGN

This section presents a detailed procedure of how the infrastructure functions. Furthermore, this design section is entirely based on figure 3. The process starts with signals being converted into cognitive commands. After a command is properly processed the simulated robot responds by performing the corresponding action. The labels in figure 3 reflect the components that are involved in this sequence of events.

A. *Non-Invasive Emotiv BCI Apparatus Integration*

The Emotiv EPOC non-invasive device (figure 1) provides wireless EEG data acquisition and processing. It connects via Bluetooth to a computer or tablet. This device consists of 14 electrodes (AF3, AF4, F7, F3, FC5, T7, P7, O1, O2, P8, T8, FC8, F4, F8) and 2 references (P3/P4 locations) to obtain the EEG signals. These channels are based on the international 10-20 locations, which is the standard naming and positioning for EEG devices [12]. The sampling method of the device is based on the sequential sampling with a sampling rate of 2048 Hz. This device was selected for the infrastructure as it has been widely used by other researchers as an input and to study the user's state, which shows its adaptability and accuracy among different task assignments [14-16].

Figure 1. EEG Emotiv EPOC Device.

The Emotiv device interacts with the Emotiv Control Panel Software (figure 2). The purpose of the Control Panel is to display the quality of the contact for the neuroheadset's EEG sensors. Furthermore, it provides guidance to the user in how to fit the neuroheadset correctly. The panel consists of three suites: expressiv suite, affectiv suite, and the cognitiv suite. The cognitiv suite was used to train the users to perform the given commands and to use the train actions to move the simulated robot. The training procedure with the Emotiv software is done as follows:

Step 1: The Emotiv device is mounted on the user's head.

Step 2: The channels are adjusted to obtain good signal (figure 2), which is obtained once the channels are colored green. If the signals are fair, then they are colored yellow. If the signals are poor, then they are colored orange. If the signals are very poor, then they are colored red. Lastly, when the signals are black there is no signal.

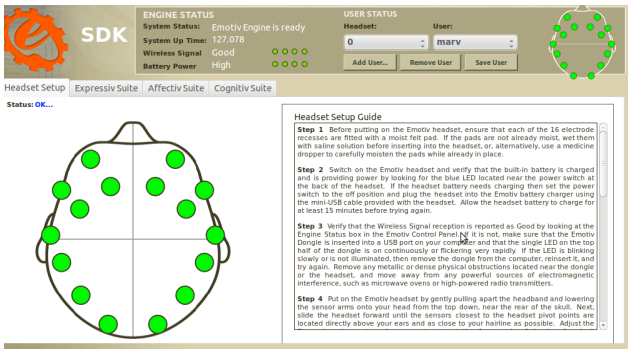


Figure 2. Good Signals Emotiv is mounted correctly

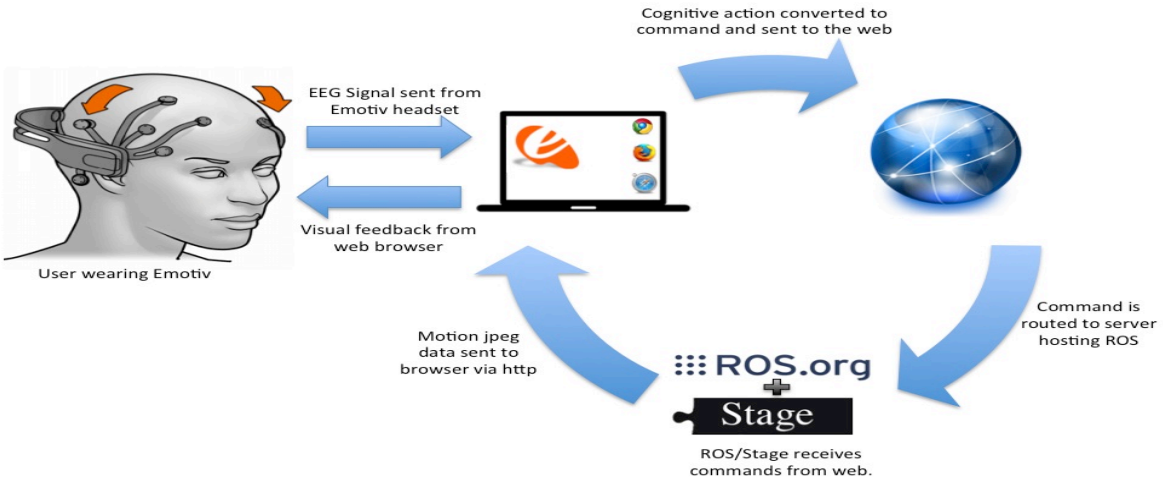


Figure 3. Infrastructure Design

Step 3: A profile is created in the control panel to save cognitive commands data after training.

Step 4: The user trains by moving a simulated cube (figure 4) integrated with the control panel. During the training period the user visualizes how to move the cube according to the action. The training for each action last eight seconds. The actions trained for this particular setup were neutral and push (meaning forward). Neutral represents the baseline where the user just looks at the cube in a relaxed status.

Step 5: Once training is completed, the user intends to move the cube via cognitive commands sent via web protocols.

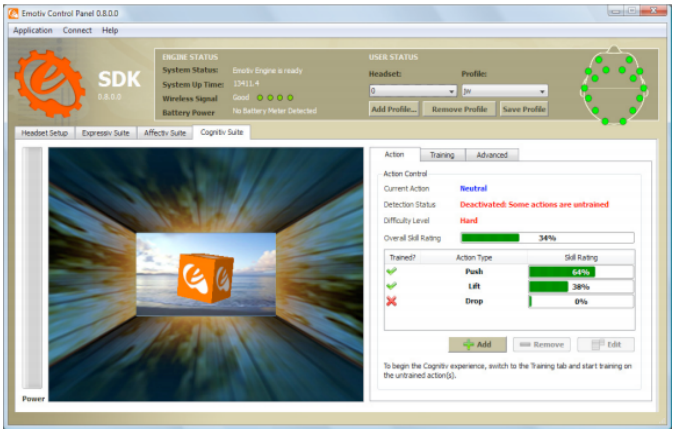


Figure 4. Simulated Cube in Cognitive Suite in Control Panel

EmoKey, another component from the Emotiv software suite, was used in this work. The EmoKey translates Emotiv detection results to sequences of keystrokes. By default the EmoKey will attempt to connect to the Control Panel when the panel launches. The EmoKey automates the key presses based on the received cognitive command. The keys are captured with a program that communicates with the webpage.

B. *Robotics Operating System (ROS)*

ROS is an open source robot operating system platform that provides standardized access to robotics capabilities or functionalities. It provides a structured communication level for multiple processes/nodes. In this case, the robot is the node. The robot controller node handles the messages coming from the web going to the robot. This communication level serves as the backbone of how the system is able to send commands to the robot. This program communicates by publishing and subscribing to messages from each other. For example, the robot controller node receives information from the web, then alerts ROS of the recently received command. Afterwards, the robot receives this alert and the robot position is updated according to the received cognitive command. The decision to use ROS in this implementation was highly affected by its ability to convert actions that takes place in simulation to real world events. Also ROS's modular struture makes it easier to add support for new machines on exsisting infrastrutures. Instead of mapping standard controls such as key presses or mouse events to robot commands, the cognitive commands from the Emotiv were sent to ROS. The browser was used to display visual feedback retrieved from ROS so that users are able to see the robot actions as their cognitive commands are processed.

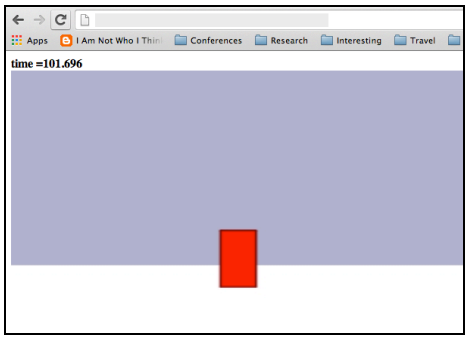


Figure 5. Simulation Environment in a web-browser.

C. HTML & HTTP

This framework leverages Hypertext Markup Language (HTML) and HTTP, well established standards, to enable user interaction. There are few networked devices that do not also have web browsers or other tools that use these standards. As an example, visual feedback from a robot's onboard camera is provided as an video stream and rendered in browser to show users the robot's environment (figure 5). HTML provides the ability to incorporate other communication modalities (sound, dynamic text, etc.) but those are not used in this work.
Browsers are not the only applications that can process these standards however. Every major programming language provides HTTP and HTML resources that can be used to integrate relevent components.

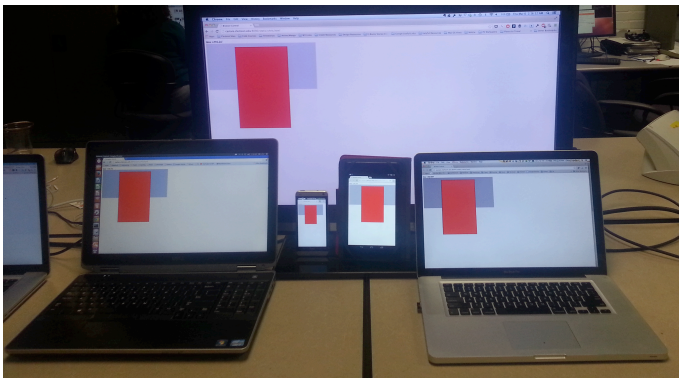


Figure 6. Various display modalities showing the robot simulation

As brain-machine interaction moves beyond medical applications, hardware advancements will play a key role in improving this area [2]. This not only applies to BCI hardware, but also to the machines being controlled. Using HTML and HTTP in an infrastructure instead of custom communication protocols enables systems to connect to any machine with the ability to communicate over a network. There are many scenarios that could benefit from this capability. For example, when users needs to complete tasks that require hands-free control cognitive commands could serve as a useful input modality. There has been work that suggests this could be useful in cases such as space operation [12]. Although there has been research on various aspects of brain-machine interaction, not much focus has been on identifying the optimal set-up for each scenario. The infrastructure presented in this paper demonstrates that utilizing HTTP and HTML could give users more flexibility when choosing what hardware best fits there needs. As shown in figure 6 feedback can be displayed on various different devices.

D. Emotiv and Message Client Integration

This section of the paper explains how a client program running on a computer with the Emotiv SDK communicates cognitive commands to the robot via HTTP. The program begins the process by capturing keyboard input triggered by the Emotiv SDK. Afterwards, it sends the information to ROS which passes it to the robot. The string sent to the server in this implementation corresponds to the forward command as described in the results. When commands reach ROS, they are sent to the robot. Once the robot's position is updated the changes are reflected back to the webpage. As a user performs cognitive commands, the user can see the robot's movement on this webpage. Figure 4 shows an example of how this environment appears in a web browser. An advantage presented by this infrastructure is that it is dynamic and not static to one particular setup. The setup presented in this paper can be substituted with any machine and program that have the ability to communicate with the web.

IV. RESULTS

This works demonstrates data of a robot's movement over time using cognitive commands based on the infrastructure presented in this paper. This section shows the variation of the target's distance and how completion time and error are effected each time. The robot performed a docking task where it moved one meter towards the red target each time it received a push (move forward) cognitive command. After completing a received command, the robot stopped and waited for further instruction. Task completion time was measured as the time it took a user to move the robot from the start position to the target. To make sure the user purposely stopped the robot after reaching the target, there was a 10 seconds grace period before the task ended. During these 10 seconds the user was instructed to not send a push command to the robot. This alleviated the issue of determining whether or not the actually meant to stop the robot. Error was measured as the difference between the robot's finals position and the position of the target. This data includes a total of four trials that vary in target distance. Only the target distance varied each trial.

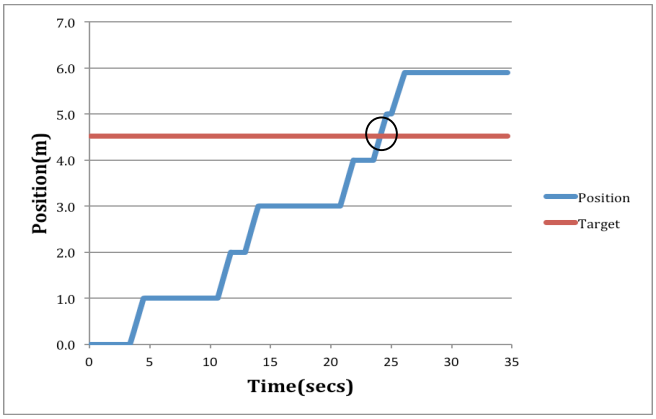


Figure 7. Robot movement through cognitive command throughtout time (4.5 m)

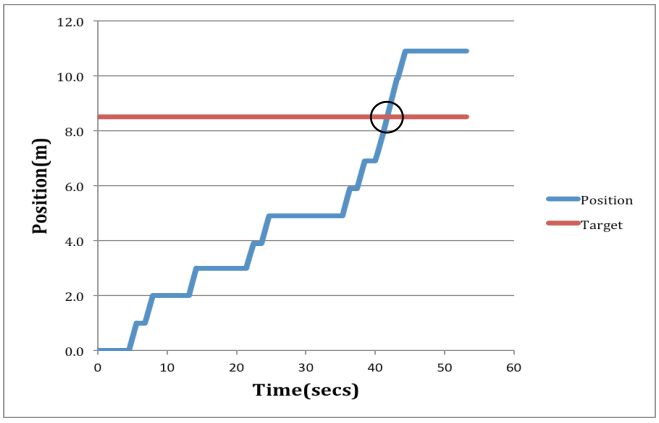


Figure 8. Robot movement through cognitive command throughtout time (8.5 m)

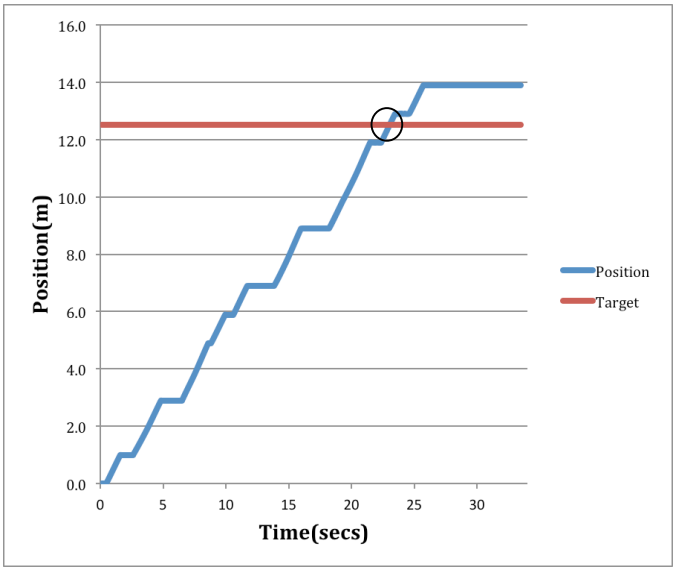


Figure 9. Robot movement through cognitive command throughtout time (12.5 m)

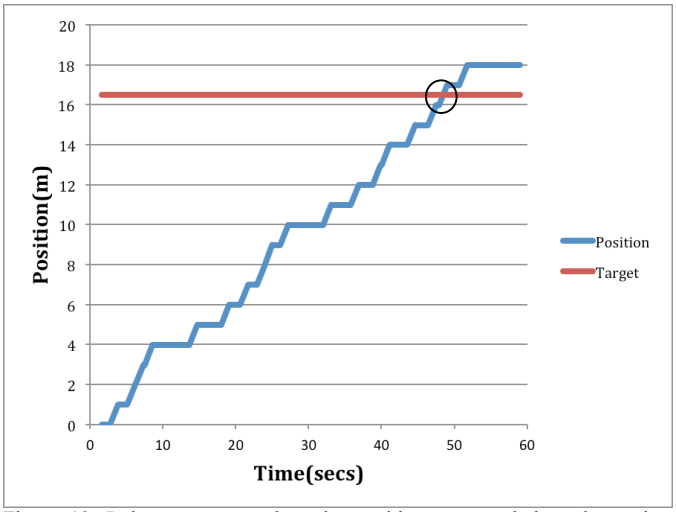


Figure 10. Robot movement through cognitive command throughtout time (16.5 m)

Figures 7 through 10 show data collected during four trials with varying target distances. Trials were completed in ascending order in reference to the target distances (4.5m,8.5m,12.5m,16.5m). Areas in the graph where the position data slope is 0 indicate that the robot stopped. When the slope is positive the robot is in motion. The position of the target is presented as a linear line in each of the movement graphs. The intersection of the target and position lines represents (shown in a black circle) when the robot reached the target

Figure 7 shows data recorded during the trial with a target distance of 4.5 meters. The completion time and error for this task was 34.63s and 1.38m respectively. Though this task had the lowest target distance it did not have the lowest completion time. As shown in the graph the robot stops for approximately eight seconds about 12 seconds into the task. This seems to directly affect the completion time. Figure 8 represents data collected during the second task with a target distance of 8.5 meters. The user finished this task in 53.12s and with an error of 2.38m. This task had the largest error. Many assumptions can be made from this increase in error. One assumption is that it became easier for the user to send commands more frequently after completing the initial task. This in result caused the user to overshoot the target after approaching it. Figure 9 shows data from the third task that had a target distance of 12.5 meters. It was completed in 33.5s with and had an error of 1.38m. The user reached the target fastest during this task. This shows that the user was able to get to the target faster after completing prior tasks even when the target was further away. The final task had a target distance of 16.5 meters. The completion time for this task was 59.05s with an error of 1.48m. Even though the target was twice the distance away in comparison to the second task, it was only six seconds behind the eight-meter task. The data reported in this section shows that target distance did not affect the time it took to reach the target while only using cognitive commands over the web. Completion time however did seem to be affected by the order of the tasks.

A. Frequency of Commands Through Time

The task completion time results varied not only by target distance, but also by the frequency of cognitive commands received. Based on the data presented in this section, task completion is not just dependent on the target distance, but also on the amount of commands sent consequently to the robot. Figure 11 through Figure 14 presents the frequency of commands per trial. Their differences in frequency of commands are described as standard deviation (STD). STD in this case demonstrates the different between the sent commands from time to time. Four meters (figure 11) = 8.47 STD, eight meters (figure 12) = 14.73 STD, twelve meters (figure 13) = 7.80 STD, sixteen meters (figure 14) = 15.84 STD. It is noted that the highest STD reflect the highest amount of time taken to complete the task. One commonality between trials was the low STD at the end of each trial.

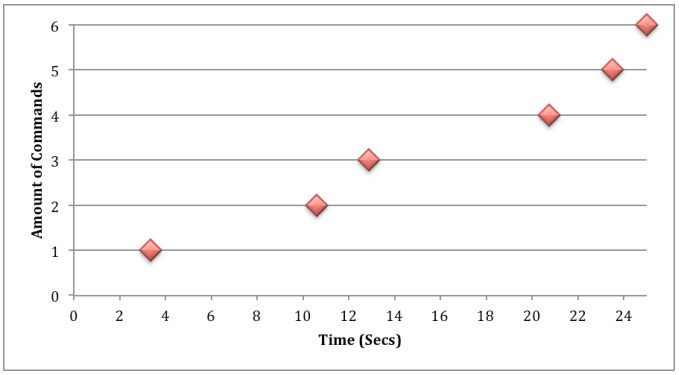


Figure 11. Frequency of command through time (4 m)

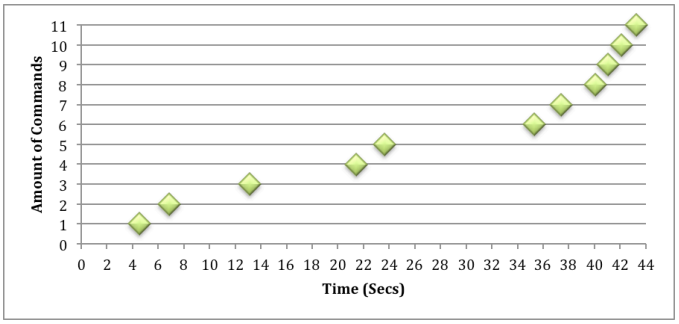


Figure 12. Frequency of command through time (8 m)

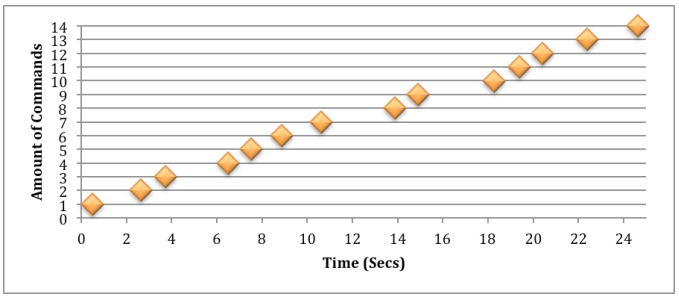


Figure 13. Frequency of command through time (12 m)

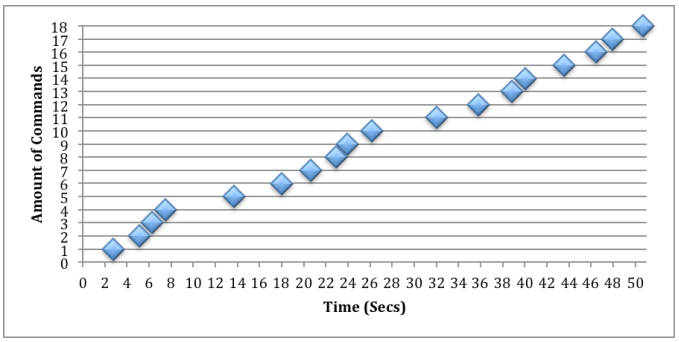


Figure 14. Frequency of command through time (16 m)

V. SUMMARY

This paper presents an infrastructure for controlling machines using cognitive commands. The setup of the infrastructure shows how BRI systems could be designed to support multiple types of hardware. This approach serves as an alternative to many custom built BRI systems presented in previous research. By using HTML and HTTP the user is able to get visual feedback from a robot on various types of display modalities. As the field of BCI advances beyond medical medical applications, it is important to research how leveraging advancments in other areas of technolgy can help this area move forward. Presented in this paper is an example of how this concept can be applied. The data reported in the results section show how a robot moves in an environment using only cogntive commands. The frequency of commands are also reported and analyzed in the results section.

The next step is to analyze user's performance while performing tasks using various types of hardware. Also work will be done studying user experince while controlling robots over the web with a BCI device. Additonal work will also consist of comparing performance and experience between web and local implementations. These works will give more insights on factors to consider when implementing web based BRI systems.

ACKNOWLEDGMENT

This material is based in part upon work supported by the National Science Foundation under Grant Number DUE-1060545. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

REFERENCES

[1] Wolpaw, J.R. and Wolpaw, E.W, "Brain-Computer Interfaces: Something New Under the Sun," in *Brain-Computer Interfaces Principles and Practice*, New York, New York: Oxford University Press, 2012, pp. 3-12.

[2] J. B. F. Van Erp, F. Lotte, and M. Tangermann, "Brain computer interfaces: Beyond medical applications," Computer, vol. 45, no. 4, pp. 26–34, Apr. 2012.

[3] T. E. Carlson and J. d. R. Millan, "Brain-actuated wheelchairs: A robotic architecture," IEEE Robot. Autom. Mag., vol. 20, no. 1, pp. 65–73, 2013.

[4] C. J. Bell, P. Shenoy, R. Chalodhorn, and R. P. N. Rao, "Control of a humanoid robot by a noninvasive brain-computer interface in humans," J. Neural Eng., vol. 5, pp. 214–220, 2008

[5] F. Gal´an, M. Nuttin, E. Lew, P.W. Ferrez, G. Vanacker, J. Philips, and J. del R. Mill´an, "A brain-actuated wheelchair: Asynchronous and noninvasive brain-computer interfaces for continuous control of robots," Clin. Neurophysiol., vol. 119, pp. 2159–2169, 2008.

[6] M. Nicolelis and J. Chapin, "Controlling robots with the mind," Sci. Amer., vol. 287, no. 4, pp. 46–53, 2002.

[7] C. Escolano, J. Antelis, and J. Minguez, "Human brain-teleoperated robot between remote places," in Proc. IEEE Int. Conf. Robot. Autom., 2009.

[8] A. Vourvopoulos and F. Liarokapis, "Brain-controlled NXT robot: Tele-operating a robot through brain electrical activity," in Proc. 3rd Int. Conf. Games Virtual Worlds Serious Appl., 2011, pp. 140–143.

[9] A. Vourvopoulos and F. Liarokapis, "Robot Navigation Using Brain-Computer Interfaces,". In Proc. of 11th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), 2012, pp. 1785-1792

[10] A. Chella, E. Pagello, E. Menegatti, R. Sorbello, S. M. Anzalone, F. Cinquegrani, L. Tonin, F. Piccione, K. Prifitis, C. Blanda, E. Buttita, and E. Tranchina, "A BCI teleoperated museum robotic guide," in Proc.

Int. Conf. CISIS, 2009, pp. 783–788.

[11] A. Thobbi, R. Kadam, and W. Sheng, "Achieving Remote Presence using a Humanoid Robot Controlled by a Non-Invasive BCI Device," International Journal on Automation, Robotics and Autonomous Systems, 10(1), 41–45, 2010.

[12] E. Coffey, A.-M. Brouwer, E. Wilschut and J. B. F. Van Erp, "Brain–machine interfaces in space: Using spontaneous rather than intentionally generated brain signals. Acta Astronautica., 2010

[13] L. Bi, X. Fan, and Y. Liu, "EEG-based brain-controlled mobile robots: A survey," IEEE Trans. Hum. Mach. Syst., vol. 43, no. 2, pp. 161–176, Mar. 2013.

[14] Campbell, A.T., Choudhury, T., Hu, S., Lu, H., Mukerjee, M.k., Rabbi, M., & Raizada, R. D. D. (2010). NeuroPhone: Brain-Mobile Phone Interface using a Wireless EEG Headset, MobiHeld 2010, August 30, 2010, New Delhi, India, 3-8.

[15] Vi, T. C., and Subramanian, S. 2012. Detecting Error-Related Negativity for Interaction Design. Proceedings of the 2012 ACM Annual Conference on Human Factors in Computing Systems (Austin, Texas, USA, May 5-10, 2012). CHI'12. New York, NY, 493-502.

[16] Andujar, M., & Gilbert, J. E. (2013). Let's learn! In *CHI '13 Extended Abstracts on Human Factors in Computing Systems on - CHI EA '13* (p. 703). New York, New York, USA: ACM Press.

[17] Allison, B.Z., and Neuper, C, "Could Anyone Use a BCI?" in Brain-Computer Interfaces: Applying our Minds to Human-Computer Interaction, Tan, D.S. and Nijholt, A. (Eds) Human-Computer Interaction Series, Springer Verlag, London, 35-54.

Bipedal Robot Walking and Balancing using a Neuronal Network Model

Robert Hercus[1], **Litt-Pang Hiew**[1], **Nur'Ain Saaidon**[1], **Kit-Yee Wong**[1], **and Kim-Fong Ho**[1]
[1]Neuramatix Sdn. Bhd., Mid-Valley City, Kuala Lumpur, Malaysia

Abstract - *This paper presents an alternative approach for controlling the walking and balancing of a bipedal robot. The proposed method uses a neuronal network to learn the sensor events obtained via the force sensors and accelerometer and to control the motor events of Bioloid's Dynamixel motors, to walk and balance the bipedal robot. A neuron layer called the controller network links the sensor neuron events to the motor neurons. The proposed neuronal network model (NNM) has demonstrated its ability to successfully control the walking and balancing of a bipedal robot, in the absence of a dynamic model and theoretical control methods.*

Keywords: Neuronal network, bipedal robot, balancing, control

1 Introduction

Research in the field of humanoid robotics has received great attention during recent years. Apart from their potential use in the development of prosthetics and rehabilitation devices, humanoid robotics is also being studied with the intention of creating humanoid robots which are able to interact with humans and assist them in everyday tasks. One of the advantages of humanoid robots lie in their human-like structures which allow them to move in areas that are normally inaccessible to wheeled robots, such as stairs, making them suitable for assisting the sick and elderly, as well as aiding humans in dangerous tasks and exploration missions. As such, much focus has been placed on the study of bipedal walking robots.

The motion of a bipedal walking robot can be categorized into the single support phase (with one foot on the ground), double support phase (with two feet on the ground) and the transition phase. In ordinary human gait, the length of the double support phase lasts for approximately only 20% of the step cycle [1], hence there exists a major challenge in generating a stable bipedal gait to prevent the humanoid robot from falling. In general, there are two types of bipedal gaits: static and dynamic walking. Static walking assumes that all dynamic forces produced by the motion of the robot limbs are negligible compared to the gravitational forces on the robot, therefore although it is easier to implement, the resulting gait can be unacceptably slow, with individual steps taking several seconds [2]. In dynamic walking, posture control based on dynamic generalizations of the concept of center of mass (CoM), such as the zero moment point (ZMP) [3], center of pressure (CoP) and the FRI [4], are used for generating stable bipedal gaits. The ZMP, originally introduced in published literature [5], is defined as the point on the ground where the total moment generated due to gravity and inertia equals zero. The ZMP is calculated and manipulated so that it remains in the support polygon in order for the robot to be dynamically stable and not fall. For stable walking, the ZMP of the robot must follow the desired ZMP trajectory estimated based on the desired configuration of the robot. A similar concept, the CoP, defines the point on the ground where the resultant ground-reaction forces act. Likewise, it is calculated and manipulated so that it does not reach the edge of the support polygon (or foot, in the case of the single support phase) to prevent the bipedal robot from falling. ZMP and CoP points have been shown to coincide despite the difference in their core definitions [4, 6], as long as all the contact points occur on a single plane.

The ZMP is prominently used for gait planning of bipedal humanoid robots, and it is the stability control used in ASIMO [3], a 26-DOF humanoid robot developed by Honda Motor Company in 2000. In addition to [3], there are many other researchers such as Vukobratovic et al. [7], Shih et al. [8], and Dasgupta and Nakamura [9] who proposed methods for robot walking pattern synthesis based on the ZMP, and they have all successfully demonstrated walking motions either using real robots or simulations. The importance of coordination between the hips and ankles for balancing have also been highlighted in [10,11], where the hip-ankle strategy was demonstrated to exhibit better balancing performance compared to the algorithms employing only the ankle strategy (rotating the ankles while locking the knee and hip joints) for push recovery [12,13].

In terms of intelligent control methods, many methods have been proposed and published. In [14], researchers have proposed a genetic-fuzzy controller for biped robots in which the dynamic stability of two-legged robots climbing staircases is simulated. Researchers [15] have also employed fuzzy logic to determine effective walking control of biped robots where two different fuzzy controllers for the support leg and the swing leg are described. The other noteworthy published works involve the application of a neuro-fuzzy algorithm [16], adaptive neuro-fuzzy algorithm [17], supervised learning using neural network [18] and a back-propagation artificial neural network as the learning scheme in [19]. The neural network with back propagation [19], neuro-fuzzy [20], and SVM [21] methods are popular in this research discipline because they can perform regression between the input and the

output (error between measured ZMP and ZMP trajectory) in the supervised learning manner. The RNN [22] and CMAC [23] methods have showed that unsupervised learning can be applied in a bipedal walking robot, but these methods are typically used as part of a hybrid controller together with a compensated torque controller (PID type), and are aimed at modeling and then compensating the system's faults, uncertainties or environmental disturbances.

2 Neuronal Network Model

The Neuronal Network Model (NNM) is a multi-node, multi-level, and multi-network neuronal network which is based on the concept of how the brain is thought to work [24]. The basic unit in a network is an elementary neuron, which represents a sensor or motor event. These neurons can be associated spatially or temporally to represent sequences of sensor or motor events. The NNM concept has been applied in the balancing control of an inverted pendulum [25,26], trajectory control of a muscle-actuated manipulator [27] as well as the navigation control of a UAV [28]. A software implementation of the NNM, the NeuraBASE toolbox, is available for download at [29].

The proposed NNM controller for the bipedal walking robot consists of three distinct and fundamental network types, namely a) sensor neurons and events - inputs to the system (CoP values for the feet, bipedal tilt direction, observed joint angles); b) motor neurons - outputs from the system (joint rotations); c) interneurons - association between two sensor events (to form a linked sensor neuron), or between a sensor event and a motor action (to form a controller neuron). Each type of event builds up an association of events in their respective network. The sensor network and motor network store sensor neurons and events, motor neurons and events, and interneurons associations respectively. A simplified data structure of the neurons used in the NNM is described in Table 1. More detailed descriptions of the sensor, motor and controller neurons are provided in Section 3.

Table 1: Data Structure of a Neuron (basic)

Field	Data Type
Head	unsigned int
Tail	unsigned int
Successor	unsigned int
Frequency/ Weight*	signed int
Next	unsigned int
Overshoot/Undershoot Flags*	unsigned short

* Denotes Fields only applicable to the Controller Neuron

3 Experimental Setup

The bipedal walking robot system hardware is made up of multiple joint structures, which imitate the motion of human legs whereby each joint is controlled by a Dynamixel servomotor. The current angular position and speed of the Dynamixel motor are accessible via the CM501 controller that comes with the Bioloid Robot Kit. Figure 1 depicts the partially assembled Bioloid robot. Each foot of the robot is attached to a force pad sensor, which measures the CoP (center-of-pressure) point. An accelerometer is also attached to the torso of the Bioloid to detect the falling state.

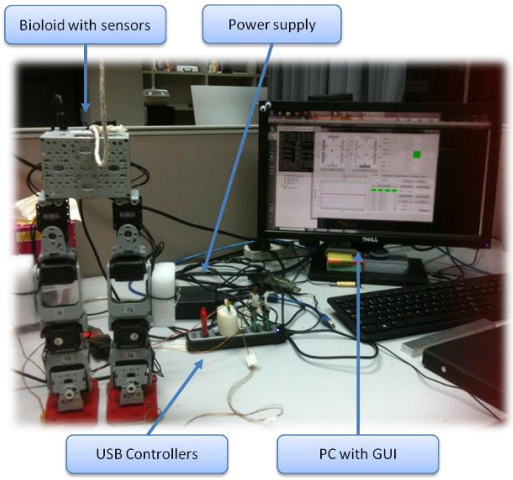


Figure 1: The partially assembled Bioloid Robot and its mechatronic components.

The walking gait cycle is a sequence of postures that describes how the bipedal robot should walk (see Table 2). The ideal posture for a stable walking gait of the bipedal robot has been developed as a reference for training. In this work, the NNM controller only controls selected motors throughout the gait cycle of the bipedal robot. The joint assignments for the Bioloid are shown in Table 3.

Table 2: Walking gait of the Bioloid

Dyna	Sub-Step						
	Starting Position	1/7	2	3	4	5	6
Front view							
Side view							
Process		weight shifting (dual support)	weight shifting (single support)	linear extension	weight shifting (dual support)	weight shifting (single support)	linear extension
Target		CoP LX	CoP LX	right leg fully extends	CoP RX	CoP RX	left leg fully extends

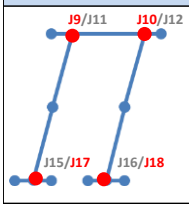
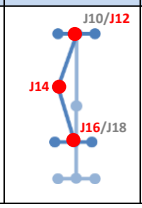
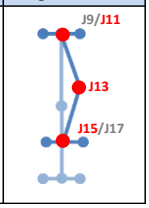
● NB controlled joint ● Joint with pre-set angular position

Walking is a repetitive motion consisting of the following basic phases that alternate on each leg:

- *Dual Support (Weight Shifting):* In this phase, both feet must be on the ground while the ankle and hip joints will move in a parallelogram configuration to shift the weight of the biped sideways until the body weight is concentrated on one foot.
- *Single support (Balancing):* Only one foot is in full contact with the ground and this foot will support the full weight of the bipedal robot. While the other leg is lifted off the ground, the supporting leg needs to ensure the stability of the structure.

- *Linear Extension:* Once the biped has successfully lifted its foot and tilted forward, its knee needs to be extended a certain amount to return the structure to a parallelogram state, thus preparing it for the start of a new walking cycle.

Table 3: Joint assignments for the Bioloid

Front view	Side view (left) - left foot lifted	Side view (right) - right foot lifted
		

There are six main sub-steps/postures involved in the balancing process, and they are weight shifting processes (sub-steps 1, 2, 4 and 5) on single planes and linear extend process (sub-steps 3 and 6). During each step, the NNM controller will control specific sets of motors, and the other motors will be set with ideal pre-set angular positions.

3.1 Training Controller

The controller program for Bioloid training acquires the input value from three sensors: readings from the two force pad sensors and readings from the accelerometer. The output of the controller program is the angular increment of the Dynamixel motor, which can be retrieved, via the NNM controller's prediction or by using a pre-set value. The program also includes the graphical user interface (GUI) to monitor the input signal from the sensors throughout the training session. The readings from the force pad sensors are used to compute the CoP values and the readings from the accelerometer will be used to detect the falling state of the bipedal robot. While the controller program is executing the gait cycle, the NNM controller will save the sequence of CoP values and return the angular increment to the respective motors as the executed output. The controller program was designed using a multithreaded programming method as timing is very crucial for the control system.

Figure 2 shows the data flow in the controller program code. Each sensor has its own thread where each thread will request six sets of 2-byte data from its respective sensor device at every ~5ms interval and then segment it accordingly for future use in the program. Then, the sensor sequence refresh thread will update all segmented sensor values from the sensor thread and rearrange the sensor sequence for the NNM controller every 50ms. The main thread will also take the sensor values from the sensor thread for data monitoring and plotting purposes. The training thread will activate the NNM controller thread which will update its own sensor sequence at its chosen time interval, evaluate the sensor sequence and then give predictions in the form of serial commands to specific Dynamixel motors controlled by its network.

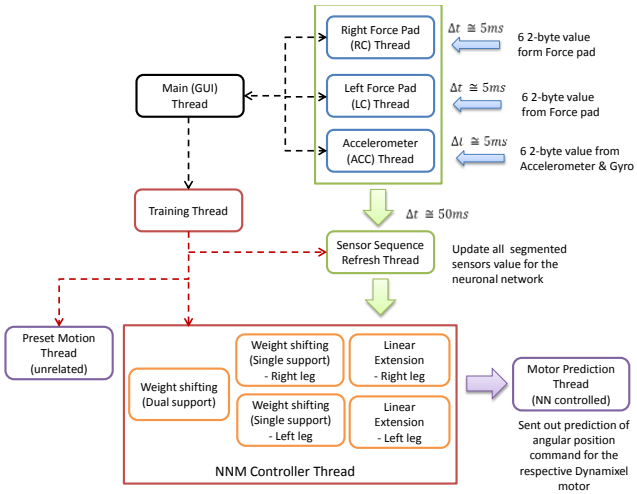


Figure 2: System data flow chart

3.2 NNM Controller

The neuronal network architecture of the robot's weight-shifting phase is depicted in Figure 3. There are three sensor networks: two for building/storing the sequence of CoP values and the bipedal robot's tilt direction, and a third network for the CoP target. The events of these sensor networks are connected in the interneuron networks, which act as the head of the controller neurons. The CoP sensor networks A and B keep the sensor events, which are the observable variables while the target CoP network keeps the sensor events that represent the target positions of the CoP value. The network that connects these sensor networks together and provides the sensor neuron to the controller network is called the CoP-state Interneuron network.

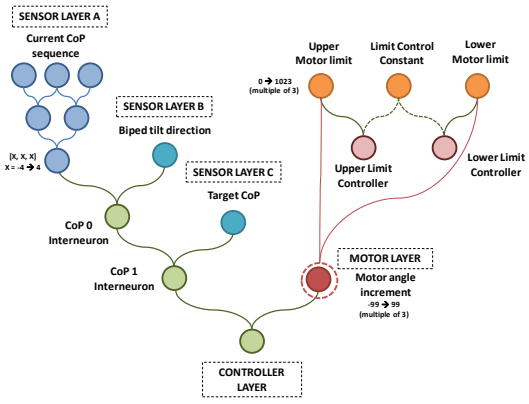


Figure 3: The neuronal network architecture for the weight shifting phase

In the motor layer, the angle increment network will keep the motor events that respond to the observed sensor events and required target positions. The motor neurons consist of angular

increments ranging from -99 units to 99 degree units, and the frequency/strength of controller neurons is capped at 10 and -10. However, an additional network of angle boundary controls the selection of the motor neuron. The angle boundary network keeps the maximum and minimum angle limit of the respective group of motors that will be updated only when the biped falls. The falling event of the biped may result from the extreme angle issued to the motor that is unsuitable for the balancing process. Thus, elimination of the extreme angle in the movement's history buffer will help to keep the biped within an acceptable angle range instead of executing excessive turns. The controller layer maintains the controller events connecting the sensor events to the motor events.

Table 4: NNM Controller I/O for the Weight Shifting Controller

Input(s)	Output
CoP sequence (historical)	Angle increment for joint group 1 (joints 9,10,17,18)
Current parallelogram tilt direction	
Target CoP	

For the linear extension phase as referred to in Figure 4, the network architecture is much simpler, given that the target is to merely extend the leg fully. The controller event will connect the sensor event of current joint angle and the motor event of angle increment needed in order to extend the lifted leg. Eventually, the biped will know how much extension is required to straighten the leg given the initial position of the leg in order to touch the ground before the start of the next phase, which is the weight shifting phase.

All NNM controllers have near-identical algorithms, and the only difference between them is the choice of sensors, the complexity of the NNM network (see Tables 4 and 5), and the activation duration which solely depends on the phase of the walking cycle. Each phase will activate its own NNM controller(s) which will process their respective sensor inputs and issue their angle increment commands as outputs of the controller.

After the initialization of the neuronal networks that will be used in that particular controller thread, the program will execute a loop of prediction and feedback processes (see Figure 5). However, the loop will only proceed to the prediction process if the controller status is active, meaning it is in the correct walking phase and the prediction status is idle. When both conditions are satisfied, it will refresh its target sensor value, and get the current sensor sequence and current angular position and polarity of its respective Dynamixel motor. The current CoP sequence will then be evaluated with respect to the target CoP position to determine if the motor requires a prediction value to move. If the average of the CoP sequence is not within the target position, the prediction process will be executed, in which it searches for the longest match sensor neuron and its respective controller neuron, in order to get the controller neuron with the highest frequency and retrieve the motor neuron for the predicted angle increment value. However, the increment value will undergo a process of verification within the angle boundary network.

Increments resulting in an angle exceeding the angle limits will not be issued, in which case the angle increment will be saturated. If there is no suitable controller neuron above the threshold, a new random value will be generated and issued to the respective motors.

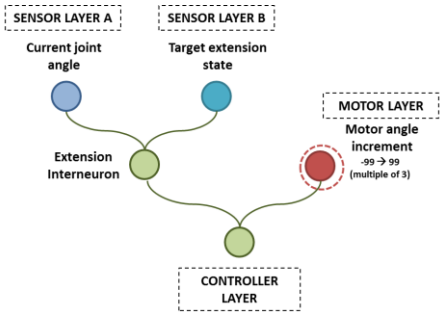


Figure 4: The neuronal network architecture for the linear extension phase

Table 5: NNM Controller I/O for the Linear Extension Controller

Input(s)	Output
Current hip joint angle	Angle increment for joint group 2 (joints 11,13,15) / joint group 3 (joints 12,14,16)
Target extension state (0 or 1)	

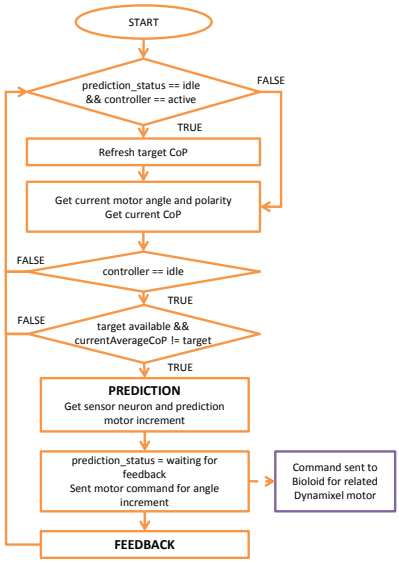


Figure 5: The general flow chart for each NNM Controller

Once the angle increment has been issued, the controller thread will awaken at an interval of 50ms to check the status of the execution. The feedback process will only be executed if the biped has stopped moving or the sensor values have stopped oscillating due to mechanical vibrations. No feedback is given if the target has not been reached but the biped does not fall. A positive feedback will be given if it reaches the target CoP region, whereas a negative feedback will be given when the biped falls.

During the feedback process, the actual motor neuron from the prediction process may not be rewarded/penalized; instead a motor neuron representing *the difference between the current motor angle and the motor angle when the prediction was first executed* will be linked to the sensor neuron. However, it is noted that since the motor's speeds will be adjusted according to its angle increment bracket, therefore by learning these motor events, the NNM controllers will eventually begin to issue smaller motor commands when the sensor values are close to the target, leading to less oscillations in the biped's movements.

Referring to Figure 6, a reward or penalty will only be given when the average CoP value has reached the target region or when the biped falls. All of the sensor neurons in the storage buffer will be connected to the motor neuron that represents the angle difference between the current angle and its angle before the prediction.

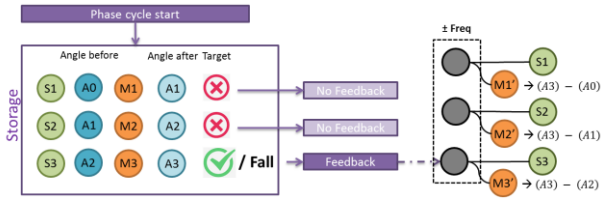


Figure 6: NNM Controller feedback reward/penalty concept

4 Experimental Results

4.1 Experimental Setup

The biped is trained to maintain its balance while walking by targeting different Center-of-Pressure points (CoPs) at different sub-steps. The forward motion (along the y-axis) is preset, whereas the process of balancing and weight shifting along the x-axis are controlled using the NNM controller. The reward/penalty system for all controller neurons is designed as shown in Table 6. The different NNM controllers activated during different points of a walking cycle (Figure 7) are shown in Table 7. Note that sub-steps 1 and 7 have identical targets, and differ only in the biped's initial pose, where the biped stands upright at the beginning of sub-step 1 (at the start of the training program) whereas it is greatly tilted towards its right side at the beginning of sub-step 7 (at the end of sub-step 6).

As all the controllers in Table 7 give predictions in the form of angle increments, this could lead to continuous increment recommendations in case of improper foot placements which may possibly affect the force sensor readings, resulting in final joint angles that may cause the structure to topple over. Therefore, additional NNM controllers have also been added for each of the three support phase controllers (Single Support (Left), Single Support (Right), and Double Support Controllers) to control the absolute joint limits, and any angle increment recommendations commanding the joints to rotate beyond these limits will then be capped accordingly. Since the biped's joint angles are acquired every ~50ms, this allows the current motor prediction to be overridden in case the target CoP has been overshot. However, as the joint servos have been programmed to slow down when the observed joint angle is close to the commanded joint angle, a mere change in direction may result in heavy oscillations or failure, thereby resulting in longer training times. Therefore, the role of the NNM controller would be to recommend smaller angle increments in order to prevent these overshoots. In order to speed up training, biased random predictions will now be given, whereby a random angle increment will be recommended according to the target CoP relative to the current CoP, as opposed to giving random predictions in either direction. Training was conducted for approximately 200 walking trials (~7 hours) and was terminated after the biped managed to take 20 steps (10 walking cycles) without falling.

Table 6: The reward and penalty system for all Controller Neurons

Type	Reward/Penalty
Success (on target)	+3
Partial success (within tolerance limit)	+1
Failure (biped falls)	-2

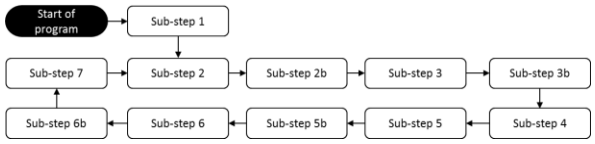


Figure 7: Flowchart showing the sub-step transitions of the training program.

4.2 Results and Discussion

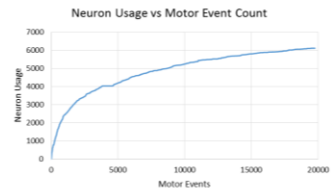
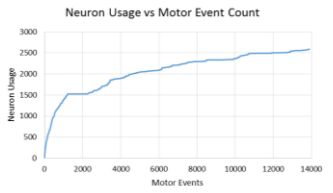
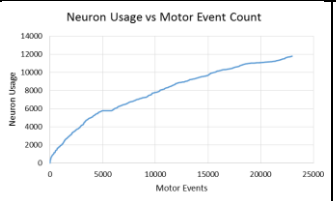
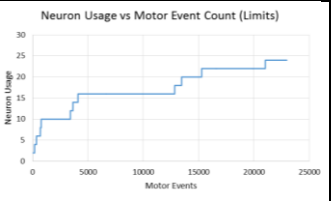
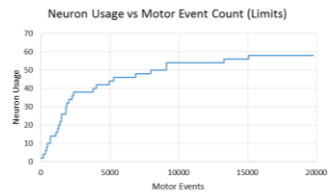
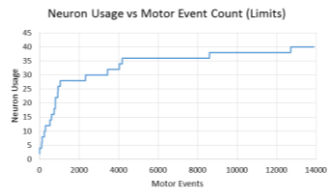
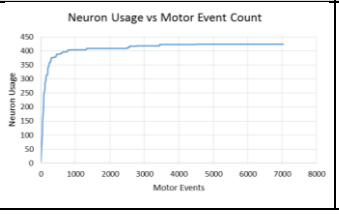
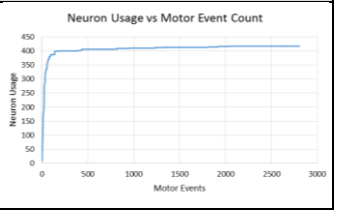
Performance of the controllers is gauged by the NNM controllers' ability to learn the correct $\Delta\theta$ increments to achieve the desired CoP while learning the joint limits to prevent the biped from falling down during the process. Consequently, the success of training is determined as follows: the biped should be able to maintain its balance while moving forward for longer durations. Due to table size limitations, the maximum number of steps allowed was capped at 20 steps. Each step covers the time from when the biped needs to shift its weight in the double support phase up to the point when it puts its foot back down on the surface, e.g. sub-steps 1/7-3b and sub-steps 4-6b.

Referring to Figure 8, we observed that the average number of completed steps taken by the bipedal robot increased with training; a video is available online [30]. However, there are still some instances where the biped is unable to reach the targeted 20 forward steps, and occasionally fell during the first step. While the biped is expected to balance itself indefinitely, this does not appear to be the case.

Table 7: Table detailing the active NNM controllers, preset motions and their respective controlled joints during each sub-step in the bipedal's walking cycle

Sub-Step	**1 and 7**	**2**	**2b**	**3**	**3b**
Description	Shift weight to the *left*	Lift *right* foot while maintaining balance	Maintain balance while keeping *right* foot up	Tilt body forward	Put *right* foot down
NB Controller (1)	Double Support Controller	Single Support Controller (Left)	Single Support Controller (Left)	Single Support Controller (Left)	Linear Extension Controller (Left)
NB Controller (1) Target	CoP within range of [+1 +2]	*Left* foot CoP >= 0	*Left* foot CoP within range of [-1 0]	*Left* foot CoP within range of [-1 0]	Full extension of *right* leg. Does not use any CoP values for prediction or feedback.
Joints involved	9,10,17,18	9,10,17,18	9,10,17,18	9,10,17,18	11,13,15
NB Controller (2)					Single Support Controller (Left)
NB Controller (2) Target					*Left* foot CoP within range of [-1 1]
Joints involved					9,10,17,18
Preset motion		Foot lifting		Forward tilt	
Joints involved		11,13,15		11,12,15,16	
Sub-Step	**4**	**5**	**5b**	**6**	**6b**
Description	Shift weight to the *right*	Lift *left* foot while maintaining balance	Maintain balance while keeping *left* foot up	Tilt body forward	Put *left* foot down
NB Controller (1)	Double Support Controller	Single Support Controller (Right)	Single Support Controller (Right)	Single Support Controller (Right)	Linear Extension Controller (Right)
NB Controller (1) Target	CoP within range of [-2 -1]	*Right* foot CoP <= 0	*Right* foot CoP within range of [0 +1]	*Right* foot CoP within range of [0 +1]	Full extension of *left* leg. Does not use any CoP values for prediction or feedback.
Joints involved	9,10,17,18	9,10,17,18	9,10,17,18	9,10,17,18	12,14,16
NB Controller (2)					Single Support Controller (Right)
NB Controller (2) Target					*Right* foot CoP within range of [-1 1]
Joints involved					9,10,17,18
Preset motion		Foot lifting		Forward tilt	
Joints involved		12,14,16		11,12,15,16	

Table 8: Total Neuron Consumption vs. Number of Motor Events for Each NNM Controller in the Bipedal Walking System

Neuron Usage			
Single Support (Left)	Single Support (Right)	Double Support	Joint Limits (Double Support)
			
Joint Limits (Single Support Left)	Joint Limits (Single Support Right)	Linear Extension (Left)	Linear Extension (Right)
			

This is attributed to new CoP sequence combinations caused by either new CoP transitions arising from previous motor actions, or sensor noise from the force pads due to mechanical vibrations during the walking process. Apart from those, this could also be caused by the imperfect walking gait programmed into the system.

As shown in Table 8, the neuron usage curves for each controller is observed to have begun plateauing (with the exception of the double support phase controller), with a total neuron count of ~21,500 at the end of training. As expected, the single and double support controllers recorded the highest neuron consumption due to there being many CoP sequence permutations for training.

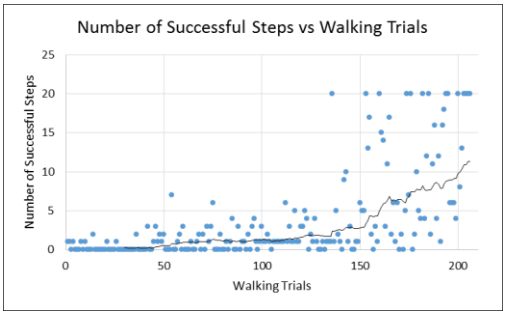


Figure 8: Number of completed steps vs. number of walking trials

5 Conclusion

The NNM controller model presented in this paper is a proof of concept that the NNM can be adapted to control the walking and balancing of a bipedal robot in the absence of dynamical models, where learning is evidenced by the increase in the number of steps successfully taken with training. However, it is also noted that this controller model is still in its early stages, and the walking gait presented in this paper is a slow and relatively stable one. Future work involves enhancing the controller model to also handle velocity-based stability control in order to make faster walking possible.

6 References

[1] Huang, Q., Yokoi, K., Kajita, S., Kaneko, K., Arai, H., Koyachi, N., Tanie, K., (2001) Planning walking patterns for a biped robot, *IEEE Transactions on Robotics and Automation*, Vol. 17, No. 3, pp. 280–289.

[2] Cheng, M.-Y., Lin, C.-S., (1995) Genetic algorithm for control design of biped locomotion, *Proc. of the IEEE International Conference on Robotics and Automation*, pp. 1315–1320.

[3] Hirai, K., Hirose, M., Haikawa, Y., Takenaka, T., (1998) The development of the honda humanoid robot, *Proc. of the IEEE Int. Conf. on Robotics and Automation*, 16-20 May, pp. 1321-1326.

[4] Goswami, A., (1999) Postural stability of biped robots and the foot rotation indicator (FRI) point, *International Journal of Robotics Research*, Vol, 18, pp. 523–533.

[5] Vukobratovic, M., Juricic, D., (1969) Contribution to the synthesis of biped gait, IEEE. Trans. Bio-*Med. Eng. BME*, Vol. 16, No. 1, pp. 1–6.

[6] Sardain, P., & Bessonnet, G. (2004). Forces acting on a biped robot. Center of pressure-zero moment point, *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, 34(5), 630-637.

[7] Vukobratovic, M., Brovac, B., Surla, D., Stokic, S., (1990) Biped locomotion. Springer, New York.

[8] Shih, C.L., Li, Y.Z., Churng, S., Lee, T.T., Cruver, W.A., (1990) Trajectory synthesis and physical admissibility for a biped robot during the single support phase, *Proc. IEEE Int. Conf. Robotics and Automation*, pp. 1646–1652.

[9] Dasgupta, A., Nakamura, Y., (1999) Making feasible walking motion of humanoid robots from human motion capture data, *Proc. IEEE Int. Conf. Robotics and Automation*, pp. 1044–1049.

[10] Stephens, B., (2007) Humanoid Push Recovery, *Proc. 7th IEEE-RAS Int Humanoid Robots Conf*, pp. 589–595

[11] Pratt, J., Carff, J., Drakunov, S., Goswami, A., Capture point: A Step Toward Humanoid Push Recovery, *Proc. 6th IEEE-RAS Int Humanoid Robots Conf*, pp. 200–207

[12] Kajita, S., Kanehiro, F., Kaneko, K., Yokoi, K., Hirukawa, H., (2001) The 3D Linear Inverted Pendulum Mode: a Simple Modeling for a Biped Walking Pattern Generation," *Proc. IEEE/RSJ Int Intelligent Robots and Systems Conf*, Vol. 1, pp. 239–246

[13] Ott, C., Roa, M., Hirzinger, G., (2011) Posture and Balance Control for Biped Robots Based on Contact Force Optimization, *Humanoid Robots (Humanoids), 2011 11th IEEE-RAS International Conference* on, pp. 26 –33, Oct

[14] Jha, R.K., Singh, B., Pratihar, D.K., (2005) On-line stable gait generation of a two-legged robot using a genetic–fuzzy system, *Robot. Auton. Syst.*, Vol. 53, pp. 15–35.

[15] Murakami, S., Yamamoto, E., Fujimoto, K., (1995) Fuzzy control of dynamic biped walking robot, *Proc. of the 4th IEEE International Joint Conference on Fuzzy Systems*, Vol. 1, pp. 77–82.

[16] Wongsuwarn, H., Laowattana, D., (2007) Neuro-fuzzy algorithm for a biped robotic system, *Int. J. Appl.Math. Comput. Sci.*, Vol. 3, No. 4, pp. 195–201.

[17] Shieh, M.Y., Chang, K.H., Lia, Y.S., (2008) Design of a biped locomotion controller based on adaptive neuro-fuzzy inference systems, *J. Phys. Conf. Ser*. Vol. 96, pp. 1–8.

[18] Miller, W.T., (1994) Real-time neural network control of a biped walking robot, *IEEE Control Syst. Mag*., Vol. 14, pp. 41–48.

[19] Shieh, M.Y., Chang, K.H., Chuang, C.Y., Lia, Y.S., (2007) Development and implementation of an artificial neural network based controller for gait balance of a biped robot, *Proc. 33rd Annual Conf. the IEEE Industrial Electronics Society (IECON)*, pp. 2778–2782.

[20] Kim, D., Seo, S.-J., Park, G.-T., (2005) Zero-moment point trajectory modelling of a biped walking robot using an adaptive neuro-fuzzy system, *IEE Proceedings - Control Theory and Applications*, Volume 152, issue4, July 2005, p. 411 – 426.

[21] Ferreira, J.P., Crisóstomo, M.M, Coimbra, A.P., (2009) Control of a Biped Robot With Support Vector Regression in Sagittal Plane, *IEEE Transactions of Instrumentation and Measurement*, Vol. 58, No. 9, pp. 3167-3176.

[22] Wu, Y., Song Q., Yang. X., (2007) Robust Recurrent Neural Network Control of Biped Robot, *Journal of Intelligent and Robotic Systems*, Vol. 49, No. 2, pp 151-169.

[23] Lin, C.-M., Fan, W.-C., Chen, C.-H., Hou, Y.-L., (2006) Robust Control for Biped Robot Using Cerebellar Model Articulation Controller, *International Joint Conference on Neural Networks*, July 16-21, pp. 2485-2490.

[24] R. G. Hercus, "Neural networks with learning and expression capability," U.S. Patent 7412426 B2, 2008.

[25] R. Hercus, K.-Y. Wong and K.-F. Ho, "Balancing of a Simulated Inverted Pendulum using the NeuraBase Network Model," *LNCS*, Vol. 8131, 2013, pp. 527-536.

[26] R. Hercus, K.-Y. Wong, S.-K. Shee and K.-F. Ho, "Control of an Inverted Pendulum Using the NeuraBase Network Model," ICONIP 2013, Part II, *LNCS*, Vol. 8227, 2013, pp. 605–615.

[27] R. Hercus, K.-Y. Wong and K.-F. Ho, "Control of a Muscle Actuated Manipulator using the NeuraBASE Network Model," *JOACE*, Vol. 2(3), 2014, pp. 302-309.

[28] R. Hercus, H.-S. Kong and K.-F. Ho, "Control of an Unmanned Aerial Vehicle Using a Neuronal Network," *IEEE Symposium on Computational Intelligence, Cognitive Algorithms, Mind, and Brain (CCMB)*, Singapore, 2013, pp.73-79.

[29] NeuraBase Generic Toolbox, http://www.neuramatix.com

[30] R. Hercus, L.-P. Hiew, N.-A. Saaidon, K.-Y. Wong, and K.-F. Ho. NeuraBase – Learning of a Bipedal Robot in Walking and Balancing II: http://www.youtube.com/watch?v=Ru8Uzl8mvgc

Fault Detection in Industrial Plant Using κ-Nearest Neighbors with Random Subspace Method

Fellipe do Prado Arruda,
Valniria da Silva Bandeira,
Kleiton Vinícius Braga,
Clarimar José Coelho
Computer Science Departament
Pontifical University Catholic of Goiás
Goiânia, GO, Brazil

Anderson da Silva Soares,
Telma Woerle de Lima, and
Gustavo Teodoro Laureano
Intitute of Informatics
Federal University of Goiás
Goiânia, GO, Brazil

***Resumo*—In this paper we propose a ensemble approach using κ-nearest neighbors (κ-NN) combined with random subspace method (RSM) to achieve an improved classification performance to fault detection problem. Fault detection and isolation is a subfield of control engineering which concerns itself with monitoring a system, identifying when a fault has occurred, and pinpointing the type of fault and its location. Fault detection is utilized to determine that a problem has occurred within in a certain channel or area of operation. In other words, the software application may recognize that the system is operating successfully, but performing at a level that is sub-optimal to predetermined target. In our study we showed that the proposed methodology is more efficiently than classical artificial neural network.**

Keywords: Fault detection, Pattern recognition, Random subspace method, κ-nearest neighbors.

I. Introdução

Faults in industrial plants are deviations from the normal behavior of the plant or its instrumentation. Fault occur due to sensor fault, actuator fault or fault in the plant. The diagnosis of the fault determines whether a fault has occurred, the location of the fault, the fault size and the time it occurred [1].

The rapid fault detection is critical to preventing its spread and cause further deterioration of the loss system equipment and human lives [2], [3]. If a failure is detected, the diagnosis consists to locate the component/sub-system involved and, if possible, the cause of the problem source. The correct and detailed diagnosis is of great value to guide procedures for accommodation of failure, timely, corrective maintenance actions, reducing the time spent on this process [4], [5].

Algorithms traditionally used in pattern classification problems in different fields can be adapted to detect and diagnose faults. In this case, the problem of fault detection and diagnosis is formulated as a classification problem. Are defined two classes, one for normal operation and one for the fault operation based on sensor data [6].

Reddy [7] developed a detection scheme of simultaneous failures at a functional level applied to a particular class of problems.

Chiang [8], [9], [10] developed an information criterion that automatically determines the order of the dimensionality reduction for Fisher's discriminant analysis (FDA) and discriminant partial least square (DPLS). The Tennessee Eastman process (TEP) simulator was used to generate overlapping datasets to evaluate the classification performance.

Ling [11] proposed a novel optimization algorithm, based on a modified binary particle swarm optimization with mutation combined with support vector machine (SVM) to select the fault feature variables for fault diagnosis. The simulations on Tennessee eastman process show the proposed methodology can effectively escape from local optima to find the global optimal value comparing with standard particle swarm optimization.

Samanta [12] compare the performance of three types of Artificial Neural Networks (ANN), namely, multi layer perceptron (MLP), radial basis function (RBF) network and probabilistic neural network (PNN), for bearing fault detection. ANNs have potential applications in automated detection and dignosis of machine conditions. Multi layer perceptrons and radial basis function networks are most commonly used ANNs.

Peter and Jin [13] propose a fault detection method using the kNN is developed to explicitly account for some unique characteristics of most semiconductor processes. The traditional kNN algorithm is adapted such that only normal operation data is needed to build a process model. The results are illustrated by simulated examples. The kNN method performs better than PCA in a real industrial example with limited data preprocessing.

Traditional univariate statistical process control charts have long been used for fault detection. Recently, multivariate statistical fault detection methods such as principal component analysis (PCA)-based methods have drawn increasing interest in the fault detection in manufacturing industry. However, the unique characteristics of the semiconductor processes, such as nonlinearity in most batch processes, multimodal batch trajectories due to product mix, and process steps with variable durations, have posed some difficulties to the PCA-based methods. To explicitly account for these unique characteristics, a fault detection method using the k-nearest neighbor rule (FD-kNN) is developed.

Baraldi [14] presents an ensemble-based scheme for nuclear

transient identification. The approach adopted to construct the ensemble of classifiers is bagging. The performance of the proposed classification scheme has been verified by comparison with a single supervised, evolutionary-optimized fuzzy C-means classifier with respect of the task of classifying artificial datasets. The results obtained indicate that in the cases of datasets of large or very small sizes and/or complex decision boundaries, the bagging ensembles can improve classification accuracy.

In this work, classification techniques are investigated to fault detection, in particular the algorithm κ-NN. Ensemble techniques are also considered to enhance the performance of the classifier. The main idea about ensemble is learn a set of classifiers and combine the predictions them. The combination of classifiers is based on the hypothesis that the decision to combine a set of classifiers can increase the rate of correct detection, surpassing the performance of individual classifiers. Once created, set members must have their opinions combined into a single decision. The rule of the majority vote is the function most popular to combination of the set classifiers [15], [16].

In particular, we explore the RSM proposed by Ho [17]. The RSM algorithm creates different training sets with resampling without repetition of variables available. Each set is used to generate a classifier according to a predetermined learning rule, por example, a ensemble where each classifier vote with equal weight. The ultimate classifier is obtained as a combination of individual classifiers based, for example, by majority voting [18].

Simple majority voting is a decision rule that selects one of many alternatives, based on the predicted classes with the most votes. It is the decision rule used most often in ensemble methods. Weighted majority voting can be made if the decision of each classifier is multiplied by a weight to reflect the individual confidence of these decisions. Simple majority voting is a special case of weighted majority voting, assigning an equal weight of $1/k$ to each classifier where k is the number of classifiers in an ensemble [19].

II. Methodology and Algorithms

In this work, fault detection is formulated as classification problem where two classes are defined

$$w_0: \text{ Normal operation}$$
$$w_1: \text{ Fault}$$

The process of building the classifiers model involves the separation of data into a training set and testing. In the classifier training stage, is delimited the decision frontiers in the space P of features in order to separate it into two regions S_0 and S_1, corresponding to classes w_0 and w_1, respectively. Such features should be conveniently extracted from the signals monitored in order to emphasize the differences between normal operating conditions and faults [20].

During system operation the decision rule makes a set of features extracted from signals at a given instant is labeled as a standard class w_0 (normal) or w_1 (fault). For this purpose, several classification techniques such as linear discriminant analysis (LDA), ANN, SVM and based on clustering of patterns can be used [21], [22].

A. κ-Nearest Neighbors Algorithm

The κ-NN algorithm identifies the category of unknown data point on the basis of its nearest neighbor whose class is already known. The training set consists of vectors of n-dimensional and each element of this set is a point in space n-dimensional [23].

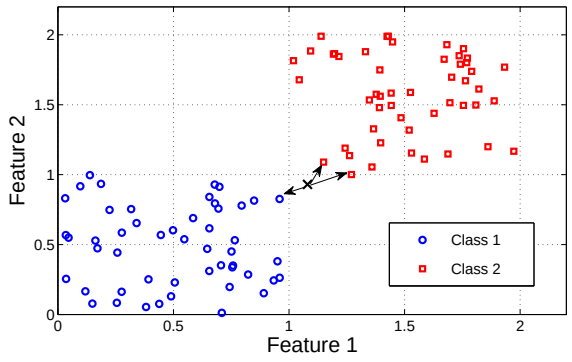


Figura 1. Illustration of the classification of a new pattern using κ-NN algorithm.

The class of an element that does not belong to the training set is determined by the classifier κ-NN searching for the κ elements of the training set which are the smallest distance of the unknown element, that is, nearest the unknown element [24].

In this work the distance is calculated by euclidian distance. Let $X = [x_1, \ldots, x_\kappa]^T$ the elements of a known class and $Y = [y_1, \ldots, y_\kappa]^T$ the elements of a unknown class

$$D(X,Y) = \sqrt{(x_1 - y_1)^2 + \ldots + (x_\kappa - y_\kappa)^2}.$$

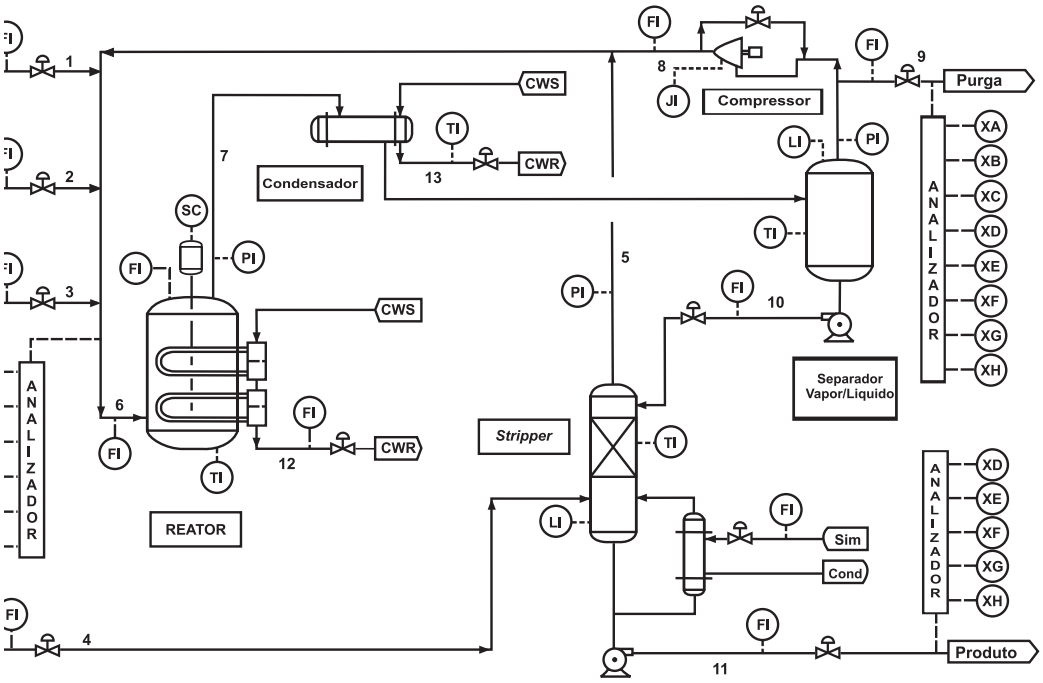
The κ elements are called κ-nearest neighbors. The algorithm checks which are the classes of κ nearest neighbors and the most frequent class is assigned to the element class unknown [25].

The Figure 1 illustrates the classification of a new pattern described by two attributes according to the rule of algorithm κ-NN for $\kappa = 3$. You can verify that the vicinity of the pattern considered comprises an element belonging to the class 1 and two elements belonging to class 2 which implies assigning Class 2 to the new standard.

The κ-NN is a nonparametric method does not require the choice of a model of data distribution and does not present problems of ill-conditioning in relation to parameter estimation. However, its performance can be very sensitive to the influence of outliers and the choice of the number of neighbors to be considered [26].

B. Random Subspace Method

The RSM, also named attribute bagging, is an ensemble classifier that consists of several classifiers and outputs the



FI: *Flow Indicator*;
SC: *Speed Controller*;
TI: *Temperature Indicator*;
CSW: *Cold Water Source*;
PI: *Pressure Indicador*;
CWR: *Cold Water Return*;
LI: *Level Indicador*;
JI: *J Indicator*

Figura 2. Schematic representation of Tennessee Eastman Process.

class based on the outputs of these individual classifiers. RSM is a generalization of the random forest algorithm [27].

The ensemble classifier is constructed using the following algorithm:

- Let n the objects of training set and the D features in the training data;
- Let L the individual classifiers in the ensemble;
- For each individual classifier, l, choose d_l ($d_l < D$) input variables for l. It is common to have only one value of d_l for all the individual classifiers;
- For each individual classifier, l, create a training set by choosing d_l features from D without replacement and train the classifier;
- For classifying a new object, combine the outputs of the L individual classifiers by majority voting or by combining the posterior probabilities.

The RSM may benefit from using both random subspaces for constructing the classifiers and aggregating it. When the dataset has many redundant features, one may obtain better classifiers in random subspaces than in the original feature space [19].

The combined decision of such classifiers may be superior to a single classifier constructed on the original training set in the complete feature space. To perform the classification, all subspaces are applied to classify a new pattern, counting the final results obtained by single classifiers. The class with the most votes is assigned [18].

III. Case Study: Tennessee Eastman Process

The process simulator for the Tennessee Eastman Process (TEP) industrial challenge problem was created by the Eastman Chemical Company to provide a realistic industrial process for evaluating process control and monitoring methods. The TEP simulator has been widely used by the process monitoring community as a source of data for comparing various approaches. TEP is recognized like as benchmark in fault detection studies.

It was proposed and implemented by [28] in FORTRAN language for a real need for realistic problems for the application and discussion of different techniques of multivariate process control. Posteriorly, [29] proposed some improvements

and provided the SIMULINK TEP application. The TEP can be described as an arrangement of recycle reactor-separator-schematic representation showed in Figure2.

The TEP produces two products (G and H) from four reactants (A, C, D and E). A further inert trace component (B) and one byproduct (F) are present. The reactions are:

$$\begin{aligned} A_{(gas)} + C_{(gas)} + D_{(gas)} &\longrightarrow G_{(liq)} \\ A_{(gas)} + C_{(gas)} + E_{(gas)} &\longrightarrow H_{(liq)} \\ A_{(gas)} + E_{(gas)} &\longrightarrow F_{(liq)} \\ 3D_{(gas)} &\longrightarrow 2F_{(liq)} \end{aligned}$$

These reactions are irreversible and exothermic with rates that depend on temperature and on the reactor gas phase concentration of the reactants. In addition to the reactants inserts through streams 1, 2, 3 e 4, a purge is necessary to prevent the inert component B from building up in the recycle stream. The reactions occur in the reactor with cooling coil to control the temperature of the mixture.

The recycle is then fed back to the mixing zone, using a compressor to compensate for pressure losses in the reactor, condenser and flash. The liquid bottom stream of the flash is pumped into a stripper in order to obtain the desired product purity [30].

The historical database for the TEP was generated by simulating the plant for a period of 1020 sampling periods for each of the 52 variables (41 measured variables + 11 manipulated variables) [31].

The data was sampled every 3 min, and the random seed (used to specify the stochastic measurement noise and disturbances) was changed before the computation of the data set for each fault. Ten testing sets were generated using the preprogrammed faults (fault 1-10) described in Table I. In additional, one testing set (Normal operation) was generated with no faults [30].

Tabela I
SIMULATED FAULTS.

Fault	Fault Description
1	Step in A/C feed ratio, B composition constant
2	Step in B composition, A/C ratio constant
3	Step in D feed
4	Step in reactor cooling water inlet temperature
5	Step in condenser cooling water inlet temperature
6	A feed loss.
7	C header pressure loss-reduced avalability
8	Random variation in A-C feed composition
9	Random variation in D feed temperature
10	Random variation in C feed temperature

A. Proposed Algorithm and Implementation Details

All algorithms were implemented in Matlab 2013b. The proposed algorithm is named κ-NN-RSM and the number of classifiers for ensemble was defined in 30. Artificial Neural Networks (ANN) using the multilayer perceptron architecture were implemented for comparisons objective. Each ANN contains 20 neurons in hidden layer. In booth, κ-NN-RSM and ANN, one classifier is build to distinguish between normal operation class and fault condition, that is, for 10 types of fault, 10 classifiers are obtained.

To measure the classifier overall accuracy we used hit rate defined according to

$$\text{Accuracy} = \left(1 - \frac{\text{Number of errors}}{\text{Number of objects}}\right). \quad (1)$$

IV. RESULTS

First, we implemented a experiment to determine the number of classifiers in ensemble. We vary the number of classifiers from 5 to 100 and measure the overall accuracy (average of all faults). The result can be seen in Figure 4. Since 30 classifiers, the accuracy has a significant gain. Thus, the number of classifiers was set at 30.

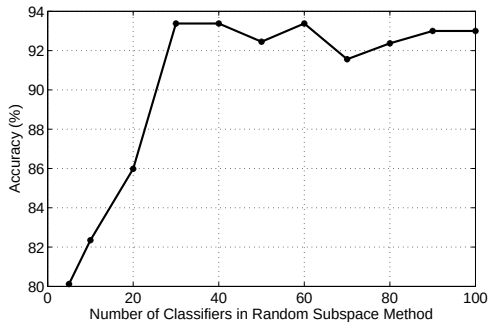


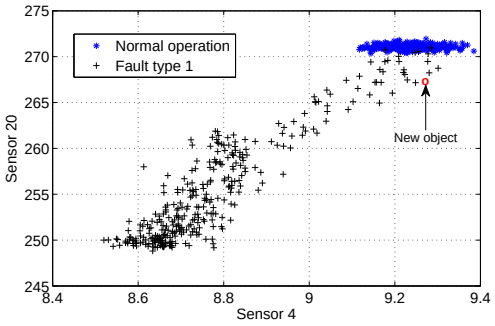
Figura 4. Accuracy versus number of classifiers in ensemble.

The Table II presents the results obtained by proposed method (κ-NN-RSM) and ANN. κ-NN-RSM and ANN obtained 93.38% and 81.70%, respectively. The κ-NN-RSM has better accuracy in 7 of the 10 faults considered. In average, the κ-NN-RSM is 15% better than ANN. Only for faults types 1, 2, 6 and 10, the ANN has better accuracy than κ-NN-RSM, however, the κ-NN-RSM has good classification performance for this faults.

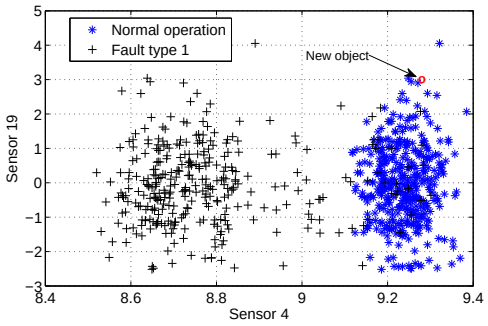
Tabela II
FAULT DETECTION RESULTS FOR PROPOSED METHOD (κ-NN-RSM) AND COMPARISON METHOD (ANN).

Fault	κ-NN-RSM	ANN
Type 1	96.67	98.36
Type 2	96.67	98.33
Type 3	97.67	82.23
Type 4	98.33	62.96
Type 5	82.73	50.93
Type 6	97.67	99.60
Type 7	96.34	93.56
Type 8	90.68	89.59
Type 9	97.01	55.83
Type 10	80.06	85.60
Average	93.38	81.70

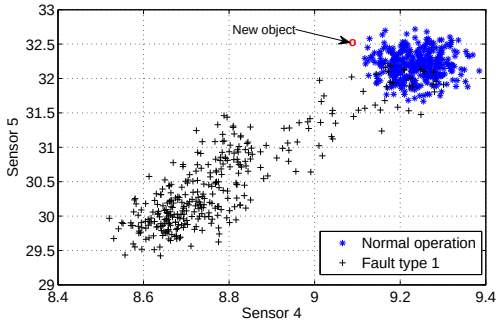
The κ-NN-RSM presents good results because of the use the classifier ensemble while ANN is just a single classifier. The Figures 3(a), 3(b) and 3(c) presents the scatter plot of training



(a) The new object is assigned for the class 1 (normal operation)



(b) The new object is assigned for the class 2 (fault type 1)



(c) The new object is assigned for the class 2 (fault type 1)

Figura 3. Scatter plot of training objects and recognition of a new object using three classifiers with different subspaces.

objects. We choose three classifiers from the all classification ensemble to show what happened in classification of a new object. As can be seen, in the first classifier, using the data from sensors 4 and 20, the new object is wrongly assigned to class 2 (fault type 1). However, the others two classifiers, using the data from sensors 4, 19, 4 and 5, assigned correctly to class 1 (normal operation).

V. Conclusion

In this work we proposed a ensemble based on κ-nearest neighbors for fault detection in industrial plant. The ensemble was implemented using the random subspace method. The proficiency of these technique for fault detection is evaluated by application to data collected from the Tennessee Eastman chemical plant simulator. The combination of κ-nearest neighbors with random subspace method (κ-NN-RSM) were shown to be better accuracy than classical implementation of artificial neural network. In average, the κ-NN-RSM was 15% better than ANN.

Acknowledgements

The authors thank the research agencies CAPES, FAPEG, FAPESP and CNPq for the support provided to this research. This is also a contribution of the National Institute of Advanced Analytical Science and Technology (INCTAA) (CNPq - proc. no. 573894/2008-6 and FAPESP proc. no. 2008/57808-1).

Referências

[1] J. J. Gertler, *Fault Diagnosis: Models, Artificial Intelligence, Applications*. Marcel Dekker, 1998.

[2] P. Mhaskar, J. Liu, and P. D. Christofides, *Fault-Tolerant Process Control: Methods and Applications*. Bristol, UK, UK: Springer, 2013.

[3] A. Zolghadri, D. Henry, J. Cieslak, D. Efimov, and P. Goupil, *Fault Diagnosis and Fault-Tolerant Control and Guidance for Aerospace Vehicles: From Theory to Application*, 2nd ed. London: Springer, 2013.

[4] J. M. Candelaria, *Fault detection and isoloation in low-voltage DC-bus microgrid systems*. ProQuest, 2012.

[5] R. Casimir, E. Boutleux, and G. Clerc, *Fault diagnosis in an induction motor by pattern recognition methods*, 2003, no. August, pp. 294–299.

[6] S. Wang and F. Xiao, "Ahu sensor fault diagnosis using principal component analysis method," *Energy and Buildings*, vol. 36, no. 2, pp. 147–160, 2004.

[7] A. Reddy and P. Banerjee, "Algorithm-based fault detection for signal processing applications," *IEEE Transactions on Computers*, vol. 39, pp. 1304–1308, 1990.

[8] L. H. Chiang, E. L. Russell, and R. D. Braatz, "Fault diagnosis in chemical processes using fisher discriminant analysis, discriminant partial least squares, and principal component analysis," *Chemometrics and Intelligent Laboratory Systems*, vol. 50, pp. 243–252, 2000.

[9] L. H. Chiang and R. J. Pell, "Genetic algorithms combined with discrimant analysis for key variable identification," *Journal of Process Control*, vol. 14, pp. 143–155, 2004.

[10] L. H. Chiang, M. E. Kotanchek, and A. K. Kordon, "Fault diagnosis based on fisher discriminant analysis and support vector machines," *Computers and Chemical Engineering*, vol. 28, pp. 1389–1401, 2004.

[11] L. Wang and J. Yu, "Fault feature selection based on modified binary PSO with mutation and its application in chemical process fault diagnosis," *Lecture Notes in Computer Science*, vol. 3612, pp. 832–840, July 2005.

[12] B. Samanta, K. R. Al-Balushi, and S. Al-Araimi, "Artificial neural networks and genetic algorithm for bearing fault detection," *Soft Computing - A Fusion of Foundations Methodologies and Applications*, vol. 10, no. 3, pp. 1433–7479, February 2006.

[13] Q. P. He and J. Wang, "Fault detection using the k-nearest neighbor rule for semiconductor manufacturing processes," *Semiconductor Manufacturing, IEEE Transactions on*, vol. 20, no. 4, pp. 345–354, 2007.
[14] P. Baraldi, R. Razavi-Far, and E. Zio, "Bagged ensemble of fuzzy c-means classifiers for nuclear transient identification," *Annals of Nuclear Energy*, vol. 38, no. 5, pp. 1161–1171, 2011.
[15] F. P. G. Marquez and M. Papaelias, *Fault Detection: Classification, Techniques and Role in Industrial Systems*, 2nd ed. New York: Nova Science Pub Inc, 2013.
[16] C. Zhang and Y. Ma, *Ensemble Machine Learning: Methods and Applications*. New Work: Springer, 2012.
[17] T. K. Ho, "The random subspace method for constructing decision forests," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 20, no. 8, pp. 832–844, 1998.
[18] Y. M. Cha Zhang, *Ensemble Machine Learning: Methods and Applications*. New York: CRC Press, 2012.
[19] Z.-H. Zhou, *Ensemble Methods: Foundations and Algorithms*. Springer, 2012.
[20] S. Theodoridis and K. Koutroumbas, *Pattern Recognition*, 3rd ed. Academic Press, February 2006.
[21] S. K. Pal, *Pattern Recognition: From Classical to Modern Approaches*. Singapore: World Scientific, 2002.
[22] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed. New York: John Wiley, 2001.
[23] N. Bhatia and V. S. S. C. S., "Survey of nearest neighbor techniques," *International Journal of Computer Science and Information Security*, vol. 8, no. 2, 2010.
[24] G. Shakhnarovich, T. Darrell, and P. Indyk, *Nearest-Neighbor Methods in Learning and Vision: Theory and Practice*, ser. Neural Information Processing Series. The MIT Press, 2005.
[25] B. V. Dasarathy, *Nearest Neighbor: Pattern Classification Techniques*, third edition ed., ser. Nn Pattern Classification Techniques. IEEE Computer Society, 1990.
[26] S. Cost and S. Salzberg, "A weighted nearest neighbor algorithm for learning with symbolic features," *Machine Learning*, vol. 10, no. 1, pp. 57–78, 2001.
[27] T. K. Ho, "Nearest neighbors in random subspaces," in *Advances in Pattern Recognition*. Springer, 1998, pp. 640–648.
[28] J. J. Downs and E. F. Vogel, "A plant-wide industrial process control problem," *Computer and Chemical Engineering*, vol. 17, no. 3, pp. 245–255, 1993.
[29] N. L. Ricker, "Decentralized control of the tennessee eastman challenge process," *Journal of Process Control*, vol. 6, no. 4, p. 205, 221 1996.
[30] J. Biegler and Wachter, "Tennessee eastman plant-wide industrial process challenge problem. complete model," *Comp. Chem. Eng.*, vol. 27, 2003.
[31] P. N. Lodal, "Case history: A steam line rupture at tennessee eastman division," *Process Safety Progress*, vol. 19, no. 3, p. 154–159, 2003.

A Pattern Recognition Model based on Invariant Representations of Space Time Data

Ollantay Medina[1], Vidya Manian[2], and Roxana Aparicio[3]
[1]Computing and Information Sciences and Engineering, University of Puerto Rico at Mayaguez, Mayaguez, Puerto Rico
[2]Department of Electrical & Computer Engineering, University of Puerto Rico at Mayaguez, Mayaguez, Puerto Rico
[3]Institute of Statistics and Computer Information Systems, University of Puerto Rico at Rio Piedras, San Juan, Puerto Rico

Abstract - *This paper proposes a generic pattern recognition model based on invariant representations of space time data, this model is mainly based on known operation aspects of the neocortex and it is oriented to deal with sequences of patterns. The theoretical model also establishes a more general framework for treatment of space time data through a dimensionality reduction process. For a given instance of space time data, the process characterizes a space time region that might be called an invariant representation. The model exhibits desirable properties for a pattern recognition system, such as spatial and temporal autoassociativity, spatial and temporal noise tolerance, recognition under sequence contextualization, and input prediction*.

Keywords: Autoassociativity, dimensionality reduction, invariant representations, pattern recognition, space time data

1 Introduction

Humans are well known for their remarkable abilities to recognize patterns; the generalization and invariance capabilities of the human brain enable us to recognize successfully new patterns just after exposure to few new samples. The human brain has a lot in common with the rest of the mammalian brains, which in turn have an almost unique feature compared to other species, the outermost layer known as the neocortex; this part is believed to be the source of the superior intelligence of mammals endowing them the ability to deal with complex patterns of information; and, to do so in a hierarchical fashion [1].

The accepted view among evolutionary biologists and cognitive neuroscientist is that evolution molded the human brain to solve problems related to surviving in an unstable outdoor environment and to do so in nearly constant motion [2]. Solving survival problems requires building and storing a model of the real world, task mainly done by the neocortex. The brain uses this memory-based model to make continuous predictions of future events; which would define intelligence as the ability to successfully predict the future, far enough to be of real use to the survival of an organism [3]. The real world model built by the brain, at first, is just a suitable representation for survival purposes. The stored representation is not a comprehensive representation of the real world, but a sample from a limited domain, dictated by the resolution of our natural sensors, the senses [4].

The progress in neurosciences has revealed a lot of facts about the structure and operation of the neocortex [5] and [6]; the current knowledge is good enough to formulate computational models, at different levels of detail that might capture the abilities to recognize patterns successfully [7], [3] and [8].

The framework proposed in this study is an attempt to make a generalization of neocortex operation, and to capture its functional equivalence in aspects such as sparse coding, hierarchical architecture and prediction.

2 Theoretical Background

The relevant concepts to formulate the model are introduced in this section.

2.1 Local, dense and sparse codes

A neural system like the brain, with a set of N binary neurons, could use one of the following coding schemes to represent information [9]. A local code represents items using separate neurons. This scheme is simple and easy to decode. However, it has a very low representational capacity (N items at most). A dense code represents items using a combination of activities of all neurons. This scheme has a very high representational capacity (2^N items). However, coding and decoding is difficult since all neurons are involved in the process. A sparse code represents an item using a small set of neurons. This scheme combines advantages of local and dense codes while avoiding most of their drawbacks. It is still easy to decode and has a sufficiently high representational capacity.

2.2 Sparse Coding in the Brain

The spatial receptive fields of simple cells in mammalian cortex can be characterized as being localized, oriented, and bandpass, comparable with the basis functions of wavelet transforms. These receptive field properties seem to produce, in terms of a strategy, a sparse distribution of output activity in response to natural images [10]. Field [11] applied this idea to simple cells in primary visual cortex suggesting that basis functions limited both in space and frequency maximize sparseness when applied to natural images. Földiak [12] proposed an algorithm to find sparse encodings in neural network models based on anti-Hebbian

lateral connections within a layer of nonlinear artificial neurons and Hebbian forward connections between these layers, combined with a local threshold control mechanism to learn a sparse code with low information loss, representing the inputs by the combination of uncorrelated components.

2.3 Hierarchical Temporal Memory

Hierarchical temporal memory (HTM) is a model that attempts to implement the memory-prediction framework. The memory-prediction framework is a theory of brain function based mainly on the role of the neocortex in matching sensory inputs to stored memory patterns and how this process leads to predictions of what will happen in the near future. The theory postulates that the remarkably uniform morphology of cortical tissue reflects a common single algorithm which underlies all cortical information processing [3].

Since its conception, HTM has evolved in several aspects, as can be seen in [13], [14] and [15]. The current known version of HTM has important influence in the model proposed in this study. HTM is fundamentally a memory-based system with a hierarchical organization, trained on lots of time-varying data. The main characteristics are:

1) Hierarchy: An HTM network consists of regions arranged in a hierarchy. The region is the main unit of memory and prediction in an HTM. A hierarchy significantly reduces training time and memory usage because patterns learned at each level are reused when combined in novel ways at higher levels.

2) Regions: The notion of regions wired in a hierarchy comes from biology. Biologists divide the neocortex into different "regions" primarily based on how the regions connect to each other. It is the region-to-region connectivity that defines the hierarchy.

3) Sparse distributed representations: Although neurons in the neocortex are highly interconnected, inhibitory neurons guarantee that only a small percentage of the neurons are active at one time. Thus, information in the brain is always represented by a small percentage of active neurons within a large population of neurons.

4) Learning: An HTM region looks for spatial patterns, combinations of input bits that occur together often. It then looks for temporal patterns or sequences, how these spatial patterns appear in sequence over time.

5) Prediction: Every region of an HTM stores sequences of patterns. By matching stored sequences with current input, a region forms a prediction about what inputs will likely arrive next.

2.4 Dynamical Systems

A dynamical system is a tuple (T, S, R), with T a set of non-negative times, S a state space, and R an evolution function, $R: T \times S \to S$. The coordinates in the state space give a complete description of the system. Given a current state of the system, the evolution function predicts the next state or states. These predictions implicate the concept of time, which may be discrete or continuous [16].

2.5 Deterministic Finite Automata

A deterministic finite automaton (DFA) is a computational model defined as a 5-tuple [17]

$$(Q, \Sigma, \delta, q_0, F)$$

where:
Q is a finite set called the states,
Σ is a nonempty finite set called the alphabet,
$\delta: Q \times \Sigma \to Q$ is the transition function,
$q_0 \in Q$ is the start state, and
$F \subseteq Q$ is the set of accept states.
The members of the alphabet are regarded as symbols. A string over an alphabet is an ordered finite sequence of symbols taken from the alphabet. A DFA either accepts or rejects a string produced over the alphabet of the DFA. The language of a DFA M is the set of all strings that M accepts, denoted as $lang(M)$.

3 A Pattern Recognition Model

The theoretical model uses the idea of a semantic representation by means of a local code. It introduces the concept of codebooks to model archetypes that represent a (sparse) group of characteristics. Also, it aims to perform a dimensionality reduction of space time data using hierarchies and grouping sequences of patterns. And finally, it uses DFA to handle the prediction of incoming patterns. The input data is considered as sequences of states generated by a dynamical system.

The problem formulation is stated regarding the brain as a specialized mechanism in pattern recognition tasks. This formulation considers a version of the formal supervised pattern recognition problem, suitable to handle sequences of patterns.

3.1 Supervised Pattern Recognition Problem

Given:

1) A set of instances X, where $x \in X$ is a finite sequence of states generated by an unknown dynamical system $D = (T, S, R)$, with T a non-negative time interval, S the state space, a Banach space over the vector space $\mathbb{R}^n$, and R the evolution function defined as $R: T \times S \to S$.

2) A set of labels (classes) Y.

3) An unknown function $g: X \to Y$ that maps input instances $x \in X$ to output labels $y \in Y$.

4) A training set $E = \{(x_1, y_1), \ldots, (x_n, y_n)\}$, assumed to represent accurate examples of mapping g.

Produce:

A function $\bar{g}: X \to Y$ that approximates as closely as possible the mapping g.

3.2 Theoretical Model

The model solves the supervised pattern recognition problem by constructing an automaton A capable of computing $\bar{g}$. The following assumptions are made:

1) $\bar{g}$ is a computable function.

2) For the set of instances X, the principle of locality, in the spatial and temporal sense, holds. Instances that exhibit spatial proximity or temporal proximity can be considered *similar* under some metric.

3.3 The Automaton *A*

Let $U = \{D_i : D_i = (T_i, S_i, R_i)\}$ be a set of unknown discrete dynamical systems that generates X, thus $X \subseteq lang(U)$.

Let A be an automaton defined as:

$$A = (H, M)$$

where $H = (h_1, h_2, \cdots, h_{n+1})$ is a tuple of mappings and $M = (M_1, M_2, \cdots, M_n)$ is a tuple of deterministic finite automata. The automaton A can be thought as a hierarchy of deterministic finite automata that uses mappings in every level to preprocess and to transform its input sequence. An input sequence x is said to be accepted by A if every automaton in the hierarchy accepts the respective transformation of x. $lang(A)$ is the set of all sequences that A accepts. Automaton A is used to recognize X in the sense that $X \subseteq lang(A)$. The construction of A is as follows:

1) h_1 is defined as $h_1 : S \to \Sigma_1$ such that $h_1(s) = a$ where $s \in S$ and $a \in \Sigma_1$. h_1 internally performs four mappings using a bounded domain $\Omega \subset S$ such that $dom_S(X) \subseteq \Omega$, where $dom_S(X) = \{s \in S : ((\exists x \in X) x = (s_i) \wedge s \in x)\}$. The mappings are as follows:

i. $h_{1,1} : \Omega \to \Lambda$, where Ω is a bounded set such that $dom_S(X) \subseteq \Omega \subset S$, and Λ is a lattice.

ii. $h_{1,2} : \Lambda \to \{0,1\}^m$, where Λ is a lattice, $m = |\Lambda|$, and $\{0,1\}^m$ is a m-dimensional Boolean space.

iii. $h_{1,3} : \{0,1\}^m \to \Gamma_1$, where $\{0,1\}^m$ is a m-dimensional Boolean space and $\Gamma_1 \in \{0,1\}^m$ is a codebook obtained by a Vector Quantization process in the Boolean space under Hamming distance.

iv. $h_{1,4} : \Gamma_1 \to \Sigma_1$, where $\Gamma_1 \in \{0,1\}^m$ is a codebook and Σ_1 is a finite set of symbols (an alphabet), this mapping is bijective.

v. $h_1(s) = h_{1,4}\left(h_{1,3}\left(h_{1,2}\left(h_{1,1}(s)\right)\right)\right) = a$, where $s \in S$ and $a \in \Sigma_1$.

The notation h^i is a compact way to represent a transformation of a sequence over Ω into a string over Σ_i. $h^i : \Omega^* \to \Sigma_i^*$ computed as $h^i(s) = h_i\left(h_{i-1}\left(\ldots h_2\left(h_1(s)\right)\ldots\right)\right) = r$, where $s \in \Omega^*$ and $r \in \Sigma_i^*$.

2) $h_i, 1 < i \leq n$ is defined as $h_i : \Sigma_i^t \to \Sigma_{i+1}$ such that $h_i(s) = b$, where $s \in \Sigma_i^t$ and $b \in \Sigma_{i+1}$. h_i computation requires choosing a fixed parameter t to group subsequences of length t in an unordered way. A goal of h_i is to reduce the dimensionality of the problem by imposing the constraint $|\Sigma_i| > |\Sigma_{i+1}|$. The computation is as follows:

i. $h_{i,1} : \Sigma_i \to \{0,1\}^m$, where $|\Sigma_i| = m$, such that $h_{i,1}(a_j) = b_j$ with $\Sigma_i = \{a_1, a_2, \cdots, a_m\}$ and $b_j \in \{0,1\}^m$ a canonical Boolean vector with a 1 in the jth position.

ii. $h_{i,2} : \Sigma_i^t \to \{0,1\}^m$, where Σ_i^t represents the strings of length t over Σ_i and $\{0,1\}^m$ is a m-dimensional Boolean space, such that given a $s \in \Sigma_i^t$, $s = s_1 s_2 \ldots s_t$:

$$h_{i,2}(s) = h_{i,2}(s_1 s_2 \ldots s_t) = h_{i,1}(s_1) \vee h_{i,1}(s_2) \vee \ldots \vee h_{i,1}(s_t) = r$$

with $r \in \{0,1\}^m$.

iii. $h_{i,3} : \{0,1\}^m \to \Gamma_i$, where $\{0,1\}^m$ is a m-dimensional Boolean space and $\Gamma_i \in \{0,1\}^m$ is a codebook obtained by a Vector Quantization process in the Boolean space under Hamming distance.

iv. $h_{i,4} : \Gamma_i \to \Sigma_{i+1}$, where $\Gamma_i \in \{0,1\}^m$ is a codebook and Σ_{i+1} is a finite set of symbols (an alphabet), this mapping is bijective.

v. $h_i(s) = h_{i,4}\left(h_{i,3}\left(h_{i,2}(s)\right)\right) = b$ where $s \in \Sigma_i^t$ and $b \in \Sigma_{i+1}$.

3) M_i is defined as $M_i = (Q_i, \Sigma_i, \delta_i, q_0, F_i)$, where if $p = |\Sigma_i|$ then:

i. $Q_i = \{q_0, q_1, \cdots, q_p\}$

ii. $\Sigma_i = \{a_1, a_2, \cdots, a_p\}$

iii. $\delta_i(q_j, a_k) = q_k$,

if $((\exists x \in X, s \in \Sigma_i^*) h^i(x) = s \wedge a_j a_k$ is a substring of s)

iv. q_0 is the initial state

v. $F_i = Q_i - \{q_0\}$

4) Choose n and t in such a way that $h^n(x) = a$ with $x \in X$ and $a \in \Sigma_n$.

5) h_{n+1} is defined as $h_{n+1} : \Sigma_n \to Y$ such that:

$$h_{n+1}(a) = y$$

if $((\exists x \in X, a \in \Sigma_n, y \in Y)\ h^n(x) = a \wedge (x, y) \in E)$

By construction, automaton A accepts, across its hierarchy, the sequences of X, thus $X \subseteq lang(A)$.

The complete definition of H produces the mapping $h^{n+1}(x) = y$ built using the information of the training set E, hence h^{n+1} can be considered as a mapping that approximates the unknown function g. Therefore $\bar{g} = h^{n+1}$ is a solution of the pattern recognition problem.

3.4 Properties of Automaton *A*

3.4.1 Semantic Representation

The data is converted into a semantic representation using mappings h_i, $i < n + 1$. This conversion is costly in terms of storage, but necessary to achieve the property. Once the representation is semantic, measuring similarity is straightforward using Hamming distance.

3.4.2 Autoassociativity

Autoassociativity is defined as the ability to associate a pattern with a part of itself. The model obtains this property from its semantic representation and by using the set of codebooks Γ_i, partial inputs have semantic similarity with already stored codewords, enabling the recovery of a complete pattern.

3.4.3 Noise Tolerance:

The model obtains this property for the same reason that it obtains autoassociativity, the use of semantic representations and codebooks across the hierarchy. Semantic similarity and codewords enable the restoration of the pattern to a version that is useful to the model.

3.4.4 Invariance

Autoassociativity and noise tolerance are important steps toward invariant representations. Missing or distorted parts in an input sequence, in lower levels of the hierarchy, can be completed or restored by autoassociativity and noise tolerance in higher levels of the hierarchy, avoiding interruption or failure of the recognition process. An invariant representation seems to be the characterization of a region of space time for a given input; after that, new inputs that fall within this region produce the same invariant representation. If the model is still learning, this region of space time is dynamic, adjusting its shape to the more frequent inputs.
Since all these four properties are consistent across the hierarchy, their influence extend to space and time.

3.4.5 Input prediction

The model has one deterministic finite automaton for every level of the hierarchy; these automata are in charge of prediction duties. They learn from the training data their transition functions that later dictates what to expect. Using finite automata in such a natural way is only possible due to the invariance property that enables the use of alphabets based on invariant representations.

3.4.6 Subsequence contextualization

This aspect is not considered in the theoretical model; but, it can be achieved in the implementation considering the technique of splitting states [18] as follows, every time a context is detected, new states are created for the subsequence in the new context, with the condition that these states should respond to the same corresponding input in the original subsequence. Transitions between sequence elements are learned independently for every context. This is equivalent to transform sequences s=abcd and r=ebcf into s=ab$_1$c$_1$d and r=eb$_2$c$_2$f. All b$_i$ respond to b but they only activate depending on the previous sequence element, once a context is activated, it is clear to the model what sequence element should be next.

3.5 A simple example

The following example tries to show how automaton *A* processes a given sequence of patterns in order to accept or reject the sequence as part of its language. Working with a low-dimensional example is an oversimplification of the actual high-dimensional input; the process may seem counter-intuitive or flawed, but this example only aims to show the procedure that automaton *A* follows.

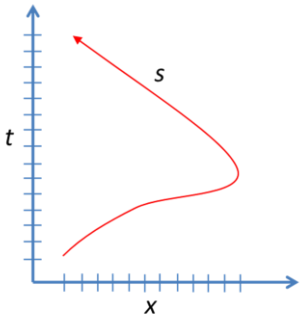


Fig. 1. A simple sequence *s* over space time

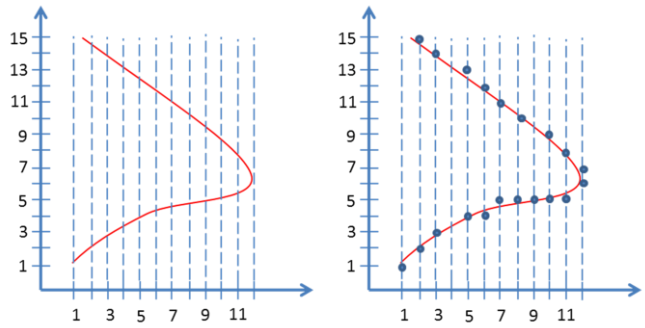


Fig. 2. Discretization of space and time that transforms *s* into a finite sequence

The example presented is a path over a space time plane, where space is the positive reals and time is also the positive reals (Fig. 1). The first step is to choose a domain $\Omega = [1,12]$ that contains all the states used by *s*. Time is mapped to the positive integers transforming *s* into a finite sequence. Then a lattice Λ transforms Ω into a finite set, leaving *s* also as a finite sequence (Fig. 2).
The next step creates a semantic representation of space by mapping the discrete space into a Boolean space; 12 elements in space imply a 12-dimensional Boolean space. The representation is semantic in the sense that if two vectors have an active bit in common, those vectors are using the same portion of space. More bits in common indicate more similarity between vectors. Table I shows the resulting set of Boolean vectors for sequence *s*, where zeroes are omitted for clarity. A vector quantization procedure is applied to the obtained semantic representation of *s*; this procedure uses Hamming distance and a threshold to group vectors. As a result, the numbers of different vectors to represent *s* is reduced. The resulting codebook is mapped to an alphabet which in turn serves to rewrite the sequence from Table I as s=abcdeffeeeeddcb. For the sake of the example, the codebook of Table II is obtained with a very loose threshold of 1 bit: two vectors are considered the same if they have at least 1 bit in common. At this point a DFA can learn to accept the current representation of *s*; for every new representation of *s* in higher levels of the hierarchy more DFAs are used.

TABLE I
BOOLEAN VECTORS REPRESENTING SEQUENCE *S*

Time step	Space 1	2	3	4	5	6	7	8	9	10	11	12	Alphabet
1	1												a
2		1											b
3			1										c
4					1	1							d
5							1	1	1	1	1		e
6												1	f
7												1	f
8											1		e
9										1			e
10								1					e
11							1						e
12						1							d
13					1								d
14			1										c
15		1											b

TABLE II
FIRST CODEBOOK AND FIRST ALPHABET OBTAINED FROM THE BOOLEAN SET OF VECTORS THAT REPRESENT SEQUENCE *S* IN TABLE I

Space 1	2	3	4	5	6	7	8	9	10	11	12	Alphabet
1												a
	1											b
		1										c
				1	1							d
						1	1	1	1	1		e
											1	f

The previous codebook reduced the space state by grouping semantically-similar vectors (Fig. 3).

The procedure that follows is an attempt to compress time by taking close-in-time subsequences of an arbitrary length. This process maps the input into a more abstract time-dependent space where the set of states are fragments that represent change across time. To produce this mapping, time is considered in an unordered way; the temporal grouping parameter should not be too small, the purpose is to reduce the probability that two sequences, that should be considered different, are one the permutation of the other. For the example, the temporal parameter is set to 3. The number of vectors in the first codebook is 6, which creates a new 6-dimensional Boolean space, where sequence *s* is represented by 5 vectors built upon subsequences of length 3 (Table III). The new codebook obtained in Table IV comes after applying vector quantization on Table III vectors; this procedure returns the alphabet seen in Table IV and sequence *s* now is represented as *s*=ghhhg.

After applying the temporal grouping procedure once more, the final representation of sequence *s* is just simply *s*=k, a single point in the last space time domain. At this moment, the construction of automaton *A* is complete; now, *A* should be able to accept sequences similar to *s* and reject those that don't resemble it.

4 Experimental Results

The goal of this section is to show and test specific aspects of the theoretical model under an implementation.

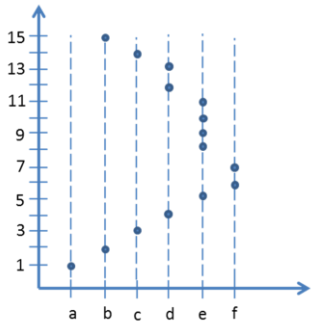


Fig. 3. Space time representation of *s* using the first alphabet

TABLE III
BOOLEAN VECTORS REPRESENTING SEQUENCE *S* AFTER APPLYING A TEMPORAL GROUPING PARAMETER OF 3 TIME STEPS

Time step	Space a	b	c	d	e	f	Alphabet
1	1	1	1				g
2				1	1	1	h
3					1	1	h
4				1	1		h
5		1	1	1			g

TABLE IV
BOOLEAN VECTORS REPRESENTING SEQUENCE *S* AFTER APPLYING A TEMPORAL GROUPING PARAMETER OF 3 TIME STEPS

Space a	b	c	d	e	f	Alphabet
1	1	1				g
			1	1	1	h

The implementation will construct an instance of automaton *A* by learning from the training set *E* defined in Section 3.1. This implementation has similarities to the HTM implementation [15].

The hierarchy is represented by a set of layers; every layer has a set of columns with a specific feed forward input, a random subset of the whole input. A column has a group of cells that respond to the input of the column. A cell has a set of segments of dendrites for prediction purposes; a segment remembers what columns where active just before the cell became active. The sparse code is achieved by selecting a small set of columns that best match the current input, all the rest of columns are inhibited. The set of winner columns is rewarded to improve its response to the respective input.

All experiments have these specifications in common: Every column has 6 cells. Dendrites of a column randomly cover 75% of the input, 60% of these dendrites have connected synapses. Every cell has 12 segments. Sparsity factor is 2% for all layers. The input is always a 50×50 binary image per time step.

4.1 Experiment I

Given a simple sequence S as a training data set, the model is tested using different versions of sequence R, where individual patterns have different levels of noise or have been cropped. The reconstructed pattern is shown using the learned sparse code.

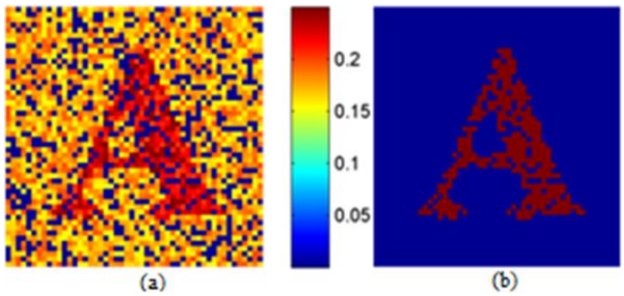


Fig. 4. Dendrites of a column after two iterations. (a) Values of synapses. (b) Connected synapses

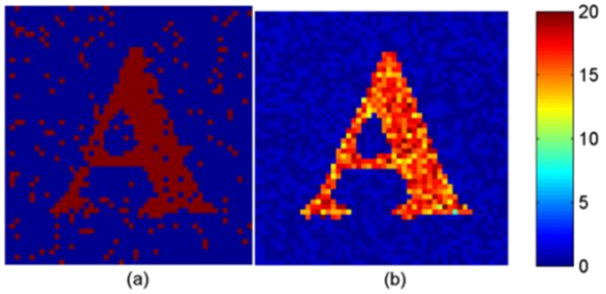


Fig. 5. Testing spatial noise. (a) Input with 50% noise. (b) Voting under the resultant sparse code.

Data and parameters used

- Model: L1=32×32 columns. Sparsity factor returns 21 columns for every pattern. Learning is done with 4 iterations.
- Training: S=AFM. Testing: R=AF
- Noise levels: A random set N of values switched in the testing sequence, where |N| is 10%, 30%, 50% and 70% of nonzero values in the original binary image.
- Cropping: 50% of nonzero values, lower or right halves in the testing sequence.
- Pattern reconstruction: Majority vote. A pixel is accepted if more than half of the sparse code accepts it.

Results

Fig. 4 shows the dendrites over the input of one winning column for pattern "A". A synapse is considered connected if its value is equal or greater than 0.2. The sparse code for pattern "A", represented by a set of 21 winning columns, can be used to reconstruct the learned pattern under a majority vote criterion. To test tolerance to noise, different levels of random noise are introduced in the input. Noise is in terms of percentage of nonzero values in the pattern. Inputs with spatial noise of 10% and 30% respectively are successfully reconstructed with an error of only 1 pixel. At 50% level of noise, Fig. 5 (b) shows how the sparse code obtained for the input pattern is including untrained columns that are actually matching the noise; still, majority vote leads to a successful reconstruction, the error is 1 pixel. At 70% level of noise, the sparse code obtained for the input pattern no longer resembles pattern "A", most columns in the sparse code are untrained columns producing a random pattern in the voting. The binary reconstruction fails to recover pattern "A". To test autoassociativity, three incomplete instances of pattern "F" are submitted as input. Fig. 6 shows how the model is capable of recover the original learned pattern from an incomplete

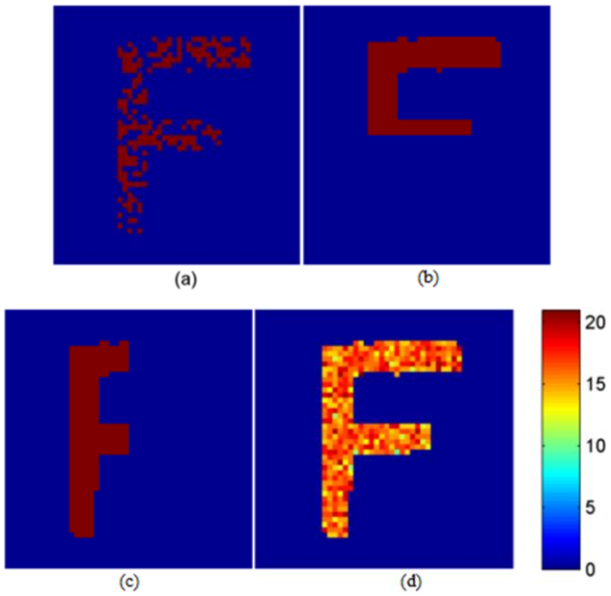


Fig. 6. Testing autoassociativity. (a) Input with 50% pixels omitted. (b) and (c) Input with cropped halves. (d) Voting according the resultant sparse code.

version of itself. The reconstruction error in all cases is 1 pixel.

Remarks

This experiment demonstrates two desirable properties of the pattern recognition model, spatial noise tolerance and spatial autoassociativity; it also shows the importance of noise free training data and well characterized patterns at the beginning of the learning process.

4.2 Experiment II

Given sequences S and R as a training data set, the model is trained to show that both sequences are consistently recognized across the hierarchy. These sequences have common subsequences that the model should be able to handle in proper context making the right predictions.

Data and parameters used

- Model: L1=32×32, L2=16×16 and L3=10×10 columns. Temporal parameters are 5, 4 and 1 for layers L1, L2 and L3 respectively. Learning is done with 2 iterations.
- Training sequences:

S=ABGMLNRST_VXYZ_HMRQK
R=EBGWLNRST_MXYH_HMQRK

- Testing sequences: S and R
- Common subsequences {BG, LNRST_, XY, _HM, K}

Results

The sparse codes are used to present these results. Sparse codes for predictions in L1 show consistent single pattern prediction in all cases. Consistent in the sense that last elements in common subsequences, patterns {G, _, Y, M, K}, are represented by the same sparse code (columns) in sequences S and R. Table V shows consistent single pattern predictions in L2, at this level there are two common subsequences {NRST_, HMRQK}. L3 recognizes correctly sequences S and R in the testing phase.

TABLE V
SPARSE CODE IN L2 AND L3 FOR SEQUENCE *S*

Time step	L2 Activations	L2 Predictions	L3 Activations
5	[59,89,144,149,150,162]	[21,41,59,131,154,216]	
10	[21,41,59,131,154,216]	[15,27,53,85,210,219]	
15	[15,27,53,85,210,219]	[2,5,9,36,48,117]	
20	[2,5,9,36,48,117]		[19,71]

Remarks
Recognizing proper context is very important for the model to make accurate predictions of what will happen next, without context one pattern can trigger several predictions with many of them unacceptable for the current input.

4.3 Experiment III

Given sequences S and R as a training data set, the testing set {S1, S2, R1, R2} is used to evaluate recognition across the hierarchy. The testing data set introduces temporal perturbations as missing elements, extra elements, changes in order and different values.
Data and parameters used

- Model: L1=32×32, L2=16×16 and L3=12×12 columns. Temporal parameters are 10, 5 and 1 for layers L1, L2 and L3 respectively. Learning is done with 4 iterations.
- Training sequences:

S=ABCDEFGHIJKLMNOPQRSTUVWXYZ_ABCDHIJKLMNOPQRSTUVWXYZ
R=GHTSRPQNOPQRSTUABCDEFABCD_HIJKLMNOPZXYWVUABCSOPQGE

- Testing sequences:

S1=ABCEFGHIJKLMNOQRSTUVWXYZ_ABCDHI_JKLMNO_PQRST_UVXYZ
S2=ABDCEFGHIJKAMNOPQTRSUVAXYZ_ABCDHLAJKMNOPQURATVWXYZ
R1=GHTSRPNOLQR_STUABDEFABCD_HIJKBMNOPZBYWVUABCSOBQGE_
R2=GHTSRPCQNPOQRSTUABCDFEABC_HIJKLXMNOPZXVYWUABCSPOQG

Results
Since the input for L1 only represents 1 time step, temporal analysis is meaningful only for higher levels. Due to particular temporal parameters for this experiment, the input to L2 represents sequences of 10 time steps and the input to L3 represents sequences of 50 time steps mapped to just 5 L2-patterns. Table VI compares sparse codes of training data and testing data. Some L2-patterns are not identical; still, using majority vote in L3, puts R, R1 and R2 in the same class.

TABLE VI
R, R1 AND *R2* SPARSE CODES

Time step	Sequences	L2 Activations	L3 Activations
10	R	[113,182,188,193,202,222]	
	R1	[113,188,190,193,202,222]	
	R2	[113,188,202,222,227,255]	
20	R	[31,53,89,133,192,234]	
	R1	[31,57,79,93,117,178]	
	R2	[31,34,89,112,133,138]	
30	R	[9,41,105,118,139,154]	
	R1	[9,41,105,118,139,154]	
	R2	[9,33,41,105,118,139]	
40	R	[35,146,159,207,219,233]	
	R1	[35,146,159,207,219,233]	
	R2	[35,146,159,207,219,233]	
50	R	[34,53,89,133,192,234]	[2,72,112]
	R1	[34,53,89,133,192,218]	[72,85,112]
	R2	[3,34,111,112,124,133]	[72,112,127]

Remarks
In some cases the sequences are considerably distorted, but due to sparse coding over a semantic representation, the similarity between patterns degrades gradually (in time and space), allowing higher levels to still use the representation.

5 Conclusions

In this paper, a pattern recognition model, based on invariant representations has been presented. According to the model, an invariant representation is a characterization of a space time region for a given input. The theoretical model also establishes a more general framework for treatment of space time data through a dimensionality reduction process; this task is mainly achieved by using semantic representations, codebooks and hierarchies. Experiments verified important properties of the model such as spatial and temporal autoassociativity, spatial and temporal noise tolerance, recognition under sequence contextualization, and input prediction.

6 References

[1] Daniel J Felleman and David C Van Essen, "Distributed hierarchical processing in the primate cerebral cortex," Cerebral cortex, vol. 1, no. 1, pp. 1-47, 1991.
[2] John Medina, Brain rules: 12 principles for surviving and thriving at work, home, and school.: Pear Press, 2010.
[3] Jeff Hawkins, On intelligence.: Macmillan, 2004.
[4] HB Barlow, "Cerebral cortex as model builder," Models of the visual cortex, pp. 37-46, 1985.
[5] Henry Markram and Rodrigo Perin, "Innate neural assemblies for lego memory," Frontiers in neural circuits, vol. 5, 2011.
[6] Van J Wedeen, Douglas L Rosene, Ruopeng Wang, Guangping Dai, Farzad Mortazavi, Patric Hagmann, Jon H Kaas, and Wen-Yih I Tseng, "The geometric structure of the brain fiber pathways," Science, vol. 335, no. 6076, pp. 1628-1634, 2012.
[7] Stephen Grossberg, Adaptive resonance theory.: Wiley Online Library, 2003.
[8] Ray Kurzweil, How to Create a Mind: The Secret of Human Thought Revealed.: Penguin. com, 2012.
[9] Peter Földiak and Malcom P Young, "Sparse coding in the primate cortex," The handbook of brain theory and neural networks, vol. 1, 1995.
[10] Bruno A Olshausen and David J Field, "Sparse coding with an overcomplete basis set: A strategy employed by V1?," Vision research, vol. 37, no. 23, pp. 3311-3325, 1997.
[11] David J Field, "What is the goal of sensory coding?," Neural computation, vol. 6, no. 4, pp. 559-601, 1994.
[12] Peter Földiak, "Forming sparse representations by local anti-Hebbian learning," Biological cybernetics, vol. 64, no. 2, pp. 165-170, 1990.
[13] Dileep George, "How the brain might work: A hierarchical and temporal model for learning and recognition," Stanford University, Ph.D. dissertation 2008.
[14] Dileep George and Jeff Hawkins, "Towards a mathematical theory of cortical micro-circuits," PLoS computational biology, vol. 5, no. 10, 2009.
[15] Jeff Hawkins, S Ahmad, and D Dubinsky, "Hierarchical temporal memory including HTM cortical learning algorithms," Techical report, Numenta, Inc, Palto Alto http://www.numenta.com/htm-overview/education/HTM_CorticalLearningAlgorithms.pdf, 2011.
[16] Morris W Hirsch, Robert L Devaney, and Stephen Smale, Differential equations, dynamical systems, and linear algebra.: Access Online via Elsevier, 1974, vol. 60.
[17] Michael Sipser, Introduction to the Theory of Computation.: Thomson Course Technology Boston, 2006, vol. 2.
[18] Jeff Hawkins, Dileep George, and Jamie Niemasik, "Sequence memory for prediction, inference and behaviour," Philosophical Transactions of the Royal Society B: Biological Sciences, vol. 364, no. 1521, 2009.

Analogy and pattern recognition

Vitor Manuel Dinis Pereira[1]
[1]LanCog (Language, Mind and Cognition Research Group), Philosophy Centre, University of Lisbon
Lisbon, Portugal

Abstract - *I'll show in an initial section (1.) that the kind of analogy between life and information (argue for by authors such as [1], [2], [3] [4], [5], [6]) is like the design argument and that if the design argument is invalid, the argument to the effect that artificial mind may represents an expected advance in the life evolution in Universe is also unfounded and invalid. However, if we are prepared to admit (though we should not do) this method of reasoning as valid, I'll show in an second section (2.), that the analogy between life and information seems suggest some type of reductionism of life to information, but biology respectively chemistry or physics are not reductionist, contrary to what seems to be suggested by the analogy between life and information.*

Keywords: analogy, pattern, recognition, reductionism, life, information.

1 Introduction

The analogy between life and information - for example, pattern recognition, with hierarchical structure and suitable weightings for constituent features [5] - seems to be central to the effect that artificial mind may represents an expected advance in the life evolution in Universe, since information (namely, pattern recognition) is supposed to be the essence of mind and all information (namely, pattern recognition) is implemented by the same basic neural mechanisms. And since we can replicated these mechanisms in a machine, there is nothing to prevent us from set up an artificial mind—we just need to install the right pattern recognizers. (To create a mind, as argue by [5], we need to create a machine that recognizes patterns, such as letters and words. Consider: translate a paper. In despite the best efforts to develop artificial universal translators, we are still very far from being able to dispense the human correction of what we write in another language.)

However, this analogy is of the same kind of analogy involved in, for example, the argument from design.

1.1 The classic watchmaker analogy

The design argument presented and criticized, for example, by Hume in his Dialogues concerning natural religion (1779) [7], can be formulated as the classic watchmaker analogy as follows.

1. The clock for its complexity and the way is ordered, is a machine that has to have an intelligent author and builder, with proportional capacities to his work - the human watchmaker.
2. The world, for its complexity and the way is ordered, it is like a clock.
3. So, the world also has to have a smart author and builder, with proportional capacities to his work - the divine watchmaker (God).

Basically, this argument holds that, just as a clock before being built, we can assume the existence of an intelligent being who built it in a certain order, also for the world, we can assume the existence of an intelligent being who built it according with a specific purpose, given the similarities between a watch and the world. While in the first case the most plausible hypothesis for the builder of the clock would be a human watchmaker, in the second case the most plausible hypothesis for the builder of the world would be a "divine watchmaker" because only this could be capable of such a work.

This argument is an analogy, but, as we shall see below, raises several problems.

Consider: it is obvious that the world is complex, has an order and natural events have a regularity, yet the analogy with the watch is fragile, remote and reductive.

1.2 The classic watchmaker analogy is fragile, remote and reductive

Firstly, it is fragile, because while the clock is a perfect machine, the world is a "machine" full of imperfections and irregularities that go beyond their usual order or regularity.

Secondly, it is remote, because any similarities between the watch and the world can only be regarded as very distant similarities, only in some aspects - one can not say with certainty that the world order is similar to the order of the clock, because while we are sure, by experience, that the clock and their order were created according to a end, we have no certainty, for not having had any experience of this, that the world and its order were even created, much less that there are also in accordance with a end (that would be divine) and not just the natural accident (the latter explanation is, moreover, the scientific explanation).

Thirdly, it is a reductive analogy, because while the clock is a machine with a limited complexity to its small dimensions, the world is a "machine" not comparable to the dimensions of the watch, so that its complexity can not also be compared with the clock.

Now, an analogy can be established from an example that is similar in a relevant aspect - in the case of the watchmaker analogy, the example would be the clock and the relevant aspect would be the complexity of the clock comparable to the complexity of the world - and we have seen that the watchmaker analogy does not fulfil these conditions, we conclude that the analogy is not valid, so the argument is invalid and should not be considered as a good proof of the existence of God.

The analogy between mental life and information is of the same kind of analogy involved in the argument from design.

From the fact that there are mental operations as thought and intention in some parts of nature, particularly in humans and other animals, it does not follow that this may be the rule of the whole that is the nature that far exceeds parts as humans and other animals.

The analogy between life and information takes a part (information) by the whole (life).

The idea that a natural biological function of the brain is processing information has not been established empirically by cognitive neuroscience, is a metaphor. The concepts of "processing" and "information" are concepts of folk psychology that seems scientifically rigorous, but are not scientifically rigorous. Concepts as "pattern recognition" does not exhaust all mental activity: if any mental activity falls under the concept of "pattern recognition", is only part of the activity of the mind.

In what way does thinking co-occur with a stimulus and categorizing it? When I am thinking about Las Vegas while in Lisbon I am not recognizing any presented stimulus as Las Vegas —since I am not perceiving Las Vegas with my senses. There is no perceptual recognition going on at all in thinking about an absent object. So concepts as "pattern recognition", although some part of what there is to say about the nature of thought – as, when I am perceiving Las Vegas with my senses - is far from all there is to say about the nature of thought[1].

Reach to the explanation of the whole (nature, as in the discussion of the argument from design by Hume; life, as in the discussion of the analogy between life and information by authors such as [1], [2], [3] [4], [5], [6]) starting with just one part (humans and other animals, as in the discussion of the argument from design by Hume; information, as in the discussion of the analogy between life and information), without more, makes any of the arguments very weak:

to the effect of the existence of God (criticized by Hume); to the effect of the analogy between life and information (argue for by authors such as [1], [2], [3] [4], [5], [6]).

At the same time, as Hume says, if we are prepared to admit (though we should not do) this method of reasoning as valid, why then choose the part of nature that says more about us, and not another?

Or, as I says, why then choose the part of mental life that says more about perceptual cases and not emotion, imagination, reasoning, willing, intending, calculating, silently talking to oneself, feeling pain and pleasure, itches, and moods—the full life of the mind? Certainly they are nothing like the perceptual cases on which the analogy between life and information rest.

According to science, were natural events that, in a succession of chances (without any special or divine plan), although according to the "laws of nature", led to the creation of the world and existence as we know it.

Thus, even before being able to dream even with Darwinian theories and how they revolutionized scientific knowledge, Hume, through his character Philo, already had an objection to the argument from design that he could not imagine be one of scientific basis of the most devastating effects against such an argument from design - namely the watchmaker analogy.

Indeed, the hypothesis of Hume of a succession of chances, besides being more logical and plausible than the theistic hypothesis, is the that most closely matches Darwinian theories of evolution by natural selection, which would arise a century later (century XIX), as well as approaches all subsequent scientific discoveries, not only of biology, but also of chemistry, and physics, regarding the possible certainties we can have about the creation of the Universe.

2 The analogy between life and information seems suggest some type of reductionism

The analogy between life and information, if we are prepared to admit (suppose you do not agree with the previous section 1.) this method of reasoning as valid (though we should not do), seems suggest some type of reductionism of life to information.

However, biology respectively chemistry or physics is not reductionist, contrary to what seems to be suggested by the analogy between life and information.

2.1 Biological level

On the biological level, for example, molecular genetics cannot provide a derivation base for evolutionary biology ([8]; [9]) or even for classical genetics ([10]). Particularly, Kitcher ([10]) writes: ''the molecular derivation forfeits something important. [...] The molecular account objectively fails to explain because it cannot bring

[1] qz.com/2660/toyota-is-becoming-more-efficient-by-replacing-robots-with-humans/

out that feature of the situation which is highlighted in the [biological] cytological story''. Richard Lewontin (quoted in [11]), in its turn, claim: "Any textbook or popular lecture on genetics will say: 'The gene is a self-reproducing unit that determines a particular trait in an organism'. That description of genes as self-reproducing units which determine the organism contains two fundamental biological untruths: The gene is not self-replicating and it does not determine anything. I heard an eminent biologist at an important meeting of evolutionists say that if he had a large enough computer and could put the DNA sequence of an organism into the computer, the computer could 'compute' the organism. Now that simply is not true. Organisms don't even compute themselves from their own DNA. The organism is the consequence of the unique interaction between what it has inherited and the environment in which it is developing (cf. [12]; [13], [14]), which is even more complex because the environment is itself changed in the consequence of the development of the organism". So, as exemplified by these two quotes from people working in the field, biology are not reductionist.

2.2 Chemical level

Neither chemistry nor physics is reductionist. On the chemical level, for example, the reduction of chemistry to quantum mechanics ([15]; [16]) is a case of failed or incomplete reduction.

2.3 Physical level

And the presumed reductionism in physics is also not more successful than biology or chemistry, on physical level, for example, it is not always possible to combine models of gravitation and electromagnetic forces in a coherent way: they generate inconsistent or incoherent results when applied, for example, to dense matter. This is the main problem currently driving people working in search for a unified field theory.

3 Conclusions

Things out there are not representational, intentional mental states about them is that they are representational, but phenomenological, physical and functional characteristics of mental states (certain type of nerve cell activation co-occurring with our looking at the world) also are not representational, are sensations and experiences.

Cognitive mental states represent, but sensations not represent anything: if certain things out there stimulate nerve cells, are not these cells that representing things out there to being of such and such a manner.

Semantics is out there, things out there stimulate nerve cells, but the co-occurring configuration of these nerve cells with that stimulation, if claim to be "representational or informational or coding", is just a misuse and overuse of terms like "representation": neurons, their synapses, neurotransmitters, receptors molecular et al. are cellular organisms more than we can access because there is no information or representation to explain what in fact we felt and experienced.

The idea that neurons (their chemistry and physics) "encode" or represent "information" is wrong (cf. [17]). If neurons encode or represent, is starting to take for granted what is intended to show: there is no difference between saying that certain BOLD[2] (fMRI) or electroencephalogram (EEG) signal correlates with certain information and saying that certain BOLD (fMRI) or EEG signal is correlated with certain conscious mental states (phenomenal or access). What's there here is question-begging. A fallacy, because they assume "information", they study "consciousness": but someone already showed that neurons encode or represent?

The metaphor of information or of representation is a fallacy of the same kind: neurons neither encode nor represent anything or nothing: what the human voice is encoding or represent? Certain sound waves.

Expressions such as "neural code" are not neurons, are us talking about them. They are to be things out there, they are being represented by us, but they themselves are not representations. Expressions like "information" and "representation" can be eliminated, that what the relevant discipline says about neurons (and related) remains informative. And if "information" is a certain kind of frequency, the frequency is enough! We telephoned, the listener understands us. But we do not say that the signal between these devices, represent or encode or is information!

A book about oceans is not an ocean: we can bathe ourselves in parts of the ocean without have any concept of "ocean" or of "part", we can see red things without seeing that are red (not having the concept of "red ").

Having information about living organisms does not make this information living organisms – they can be "automata" (Descartes in 2nd of his Meditations on the First Philosophy, 1641, [19]). By definition an artificial, for example, plant (information about the way plants look like) is not a living organism, is not a plant. In the same vein, artificial mind is not mind and can not represent an expected advance in the life evolution in Universe in a way suggest by the analogy between life and information: but as a tool - héla! pattern recognition - can help us to have more information about humans and other animals perceptual cases.

4 Acknowledgments

My mother, Maria Dulce. LanCog research projects FCT Project PTDC/FIL-FIL/121209/2010.

[2] Blood Oxygenation Level Dependent (for example, [18]).

5 References

[1] Davies, P. *The Fifth Miracle: the search for the origin and meaning of life.* Simon & Schuster, 2000.

[2] Walker, S.; Davies, P. The algorithmic origins of life. *Journal of the Royal Society Interface*, 10(79), 1-9, 2013.

[3] Dyson, F.J. Time without end: physics and biology in an open universe. *Reviews of Modern Physics*, 11, 443–460, 1979.

[4] Gleick, J. *The information: a history, a theory, a flood.* Vintage, 2011.

[5] Kurzweil, R. *How to Create a Mind: The Secret of Human Thought Revealed*. Viking Adult, 2012.

[6] Ward, P. *The Medea Hypothesis: is life on Earth ultimately self-destructive?* Princeton University Press, 2009.

[7] Hume, D. *Principal writings on Religion including Dialogues concerning natural religion and The Natural history of religion*. Oxford University Press, edited with an introduction and notes by J. C.A. Gaskin, 1998.

[8] Lewontin, R.C. Biological Determinism. *Tanner Lectures on Human Values*. Salt Lake City: University of Utah Press, 1983.

[9] Levins, R. *Evolution in Changing Environments.* Princeton, NJ: Princeton University Press, 1968.

[10] Kitcher, P. 1953 and All That: A Tale of Two Sciences. *Philosophical Review*, 93, 331–33, 1984.

[11] Callebaut, Werner. *Taking the Naturalistic Turn, or, How Real Philosophy of Science Is Done.* Chicago: University of Chicago Press, 1993.

[12] Changeux, J.-P. *Neuronal Man: The Biology of Mind.* Oxford: Oxford University Press, 1985.

[13] Edelman, G. *The Remembered Present: A Biological Theory of Consciousness*. New York: Basic Books, 1988.

[14] Edelman, G. *Topobiology: An Introduction to Molecular Embryology.* New York: Basic Books, 1988.

[15] Cartwright, Nancy. Why Physics? in Penrose, R. ; Shimony, A. ; Cartwright, N. & Hawking, S.(eds.). *The Large, the Small and the Human Mind*. Cambridge: Cambridge University Press, 1997.

[16] Primas, H. *Chemistry, Quantum Mechanics, and Reductionism*. Berlin: Springer-Verlag, 1983.

[17] Burock, Marc. *Evidence for Information Processing in the Brain*, 2010. (Preprint.) URL: http://philsci-archive.pitt.edu/8845/ (7 January 2014).

[18] Ogawa, S., Tank, D. W., Menon, R. Ellermannn, J. M., Kim, S-G., Merkle, H. & Ugurbil, K. Intrinsic signal changes accompanying sensory stimulation: Functional brain mapping with magnetic resonance imaging. *Proceedings of the National Academy of Sciences of the United States of America*, 49, 211-211, 1992.

[19] Descartes, R. *The Philosophical Writings of Descartes*, 3 vols., translated by John Cottingham, Robert Stoothoff, and Dugald Murdoch (Volume 3 including Anthony Kenny), Cambridge: Cambridge University Press, 1988.

Using Multilayer Perceptron Networks in Early Detection of Late Blight Disease in Tomato Leaves

G. K. Vianna[1] and S.M.S. Cruz[1,2,3]
[1] UFRRJ - Universidade Federal Rural do Rio de Janeiro, Seropédica, RJ, Brazil
[2] PPGMMC/UFRRJ – Programa de Pós Graduação em Modelagem Matemática e Computacional
[3] PET-SI – Programa de Educação Tutorial – Sistemas de Informação

Abstract – *In this work we applied simple computer graphics techniques and artificial neural networks technology to identify the tomatoes that had foliage diseases. We propose the use of low cost computational apparatus to support early detection of diseases and pests in Brazilian tomato crops. A description of methods and results are presented in this work. Our proposal consorts field experiments with inexpensive computer-aided experiments based on pattern recognition. The recognition was aimed to identify diseases, more specifically, the late blight, which is characterized by the occurrence of brown spots in tomato leaves.*

Keywords: Pattern Recognition, Digital Image Processing, Neural Networks, Multilayer Perceptron, Late Blight

1 Introduction

In Brazil, most traditional tomato (*Solanum lycopersicon*) quality tests still remain at the preliminary stage in which identification is determined by human sense organs. These subjective evaluations are affected by personal abilities, emotions, fatigue and observational appreciations whose objectivity and accuracy are inadequate to face the increase of production. Nowadays, the tomato has an important role in the Brazilian agricultural production, being one of the most important vegetable crops grown in the state of Rio de Janeiro, which is one of the largest Brazilian producers [1]. Thus, the development of new technological solutions that reduce operating costs on small properties to produce vegetables with higher quality is essential.

The aim of this work is to present a set of computer-based techniques that may be used to increase the productivity of organic crops of tomatoes on small properties in the state of Rio de Janeiro. In this study, the approaches that have been adopted are based on pattern recognition using multilayer perceptron (MLP) neural networks. The adoption of such technology on small properties are very challenging endeavor [2], especially if we consider the aspects involved in the digital data manipulation by workers and its further treatment, in order to enhance the early detection of late blight that is caused by the *Phytophthora infestans*.

Late blight is the most destructive disease of tomatoes and potatoes in Brazil. The total crop losses due to the disease are commonly reported up to 20% of the total production [3]. The disease is visually recognized by the appearance of dark spots on tomato leaves, whose blotches vary from brown or gray to pale green, often located at the edges of the tomato leaves, which can develop into large brown necrotic areas [4]. The traditional model of tomato crop plant in Brazil requires large investments by farmers, as sprays of pesticides every three days and manpower to keep the crop healthy. These factors increase the production costs and generate losses on local fauna, bringing the local ecosystem imbalanced, and even diseases on farmers and workers [5]. We advocate that the use of computer-based systems to alert, detect or predict the occurrence of tomato diseases may aid farmers to reduce the chemical management of them. It may also provide a reduction in the number of sprayings and the total amount of pesticide residues in fresh and processed fruits and also diminish production costs and environment contamination [6,7].

Nowadays, the small Brazilian farmers do not have many options of computational tools that could help them in early detection of late blight. The commonest approach involves naked eye observations, as the key manual classification of the disease, defined by Correa [8]. This key is based on stylized images of tomatoes leaves that quantify the degree of infestation by *P. infestans*, allowing visual assess of the contamination degree. However, early detection in this case must be performed by experts, which is neither always an economically viable option, nor suited to large-scale production. An alternative would be to automate the process of cultures evaluation, to accelerate the identification of strains by separating them into two sets: one containing samples of plants in good condition, and the other the samples that must undergo further examination, to be made by specialists.

This paper is organized as follows: Section 2 presents the related work and compares them with our approach; Section 3 characterizes the samples images used in the pattern recognition procedures, and how they were collected from the field; Section 4 describes the digital image processing used to filter the original images in order to improve the performance of the system; Section 5 describes the relevance analysis of the variables collected from the processed images; Section 6 presents the computer-based experiments and results involving pattern recognition of the

prevalent disease in tomato plants, through the use of neural network techniques; Section 7 concludes the paper and presents future work.

2 Related work

Artificial Neural Networks (ANN) models have been successfully used in the prediction of problems in different domains such as dealing and forecasting economic issues [9,10], bio-processing and chemical engineering [11], and Agriculture [12,13].The semi-automatic detection of diseases and pests with the aid of algorithms supported by ANN has been addressed in several previous works. However, these studies used different species of plants other than tomato. Thus, as far as we are concerned, this is a shortcoming because tomato leaves have particularities that cannot be found at other vegetables.

Vieira [14] used a multi-layer perceptron neural network (MLP) to detect the lesions in tomato leaves. However, their success rate was only 77.8 % in the identification of lesions by late blight. For instance, only 9 samples of leaves with lesions of a total set of 65 samples of injured leaves of three different types of illness were used. Vieira´s digital images were manually pre-processed: the user defines the most relevant areas of the original image caused by disease and only those subsets of images are analyzed. Finally, the author does not clearly specify what disease was being evaluated on his experimentation. Movaghamejad and Nikzad [15] used MLP techniques to describe the drying behavior of tomato in different experimental conditions.

Sanyal and Patel [16] analyzed the texture and color of the rice leaves (*Oryza sp.*) to identify diseases as brown spot and blast diseases. The simulation performed by the authors was based on 400 samples, and the ranking was performed by an MLP neural network. Images of normal and infected leaves were used, and the authors reached a performance of 89.26 % of accuracy in the classification of individual pixels of the images, (considering the injured leaves or not). We stress that neither the leave as a whole nor the plant were evaluated. Such work is different of our approach because the authors only investigated individual pixels while we investigated the whole amount of pixels.

Camargo [17]defined a classifier based on support vector machine (SVM) to operate on color images of cotton (*Gossypium sp.*). He used only 127 images of the cultures. The best classification of his model was found with 45 different characteristics, reaching a performance of 93.1 %.

Our approach is quite different from Vieira´s [14] and other related works. We have used simple digital samples collected directly by mobile phone in the field. Besides, the images that we have collected *in situ* have many details of leaves, floor, fruits and, sometimes, parts of the sky and the environment itself. Such characteristics reinforce the quality of the filters and the pattern recognition system that we developed to detect the disease. In addition, our study used a larger number of samples and the hit rates achieved were greater than related work.

3 Material and methods

For the task of identification the leaf area of the affected tomatoes by the late blight, we have used a large set of real images of young and old plants. The images were taken directly from the experimental field using a mobile phone digital camera where the final resolution of the images has been reduced to only 96 dpi, for improving the speed of the recognition process.

The experimental setup, as it is composed by ordinary mobile phones and low end personal computers at the computer science lab. Besides, we did not used sophisticated machinery because farmers and workers may not be willing to afford expensive equipments or even cannot operate it properly. Finally, we used mobile phones because the represent a promise new opportunities for reaching farmers with agricultural information. The initial set of samples available contains 226 color digital images of leaves collected from 66 different genotypes of tomatoes, grown under the principles of organic agriculture. The tomatoes were cultivated in the experimental fields of the horticulture sector of the Horticultural Department of the IA/UFRRJ in an cropping area historically associated with the natural occurrence of late blight. The genotypes include a commercial cultivar susceptible to illness, two resistant to it, and also 63 cultivars still under review at the UFRRJ [8].

The original images of the tomatoes were collected from the experimental field, and their quality (image definition) were reduced, using the simple algorithms such as the one coded in *PlanarImage* class (from *javax.media.jai* package) to reduce about 70% in its size, in order to accelerate the performance of further processing. Next, the reduced colored images were converted to black and white images, also using the simple algorithms such as the one coded in the *BufferedImage* class (from *java.awt.image* package).

The prototype is a java application that was developed to be as simple as possible because the hardware and software platform for exchanging data in areas with limited Internet and electrical grid connectivity is hard to be achieved. This goal was a need to fulfill our target users, which were families of farmers that could not afford for sophisticated computer-based technology. In this way, the proposed system may achieve good results using ordinary hardware and free license softwares. We used the Eclipse IDE for Java and some free packages like Neuroph [18], Java Advanced Imaging (JAI), AWT Image, plus the MySQL.

4 Treatment of the images

This section describes the two methods used to treat the images, where each image was submitted to a red/green color filter, and to a grayscale filter. The red/green filter consists of a simple automated procedure previously developed by the research team [2], which considers the

principle that the color of a healthy tomato plant is bright Green. The yellow or brown tones may indicate the occurrence of an injury whereas all different tones are considered part of the background and must be discarded. After this first image filtering process, the result was a set of images having only pure red, pure green, or black pixels (Figure 1). The red pixels identify the injured areas, while the green and black pixels represent the healthy regions of leaves, and the background image, respectively.

The second digital image processing was used to convert the original colored images into grayscales images. This was made to identify the different hues that represent the injury and healthy areas of the leaves.

Figure 1. Examples of tomatoes leaf (images were taken at the experimental field) infected by *P. infestans*. The blue arrows indicate injured areas of the leaves. The images illustrate the situation before and after application of the Red/Green filter**.**

4.1 Red/Green filtering

The analysis of the tones of each pixel of the image was performed by classifying the pixels according to the values of their R, G and B components of the RGB color system. We established the several rules:

(i) A *pixel* is classified as injury and will be converted to pure red (255,0,0), in case it is yellowed or browned, which indicates the occurrence of the lesion. For the tone calculation, it was used the following rule:

$$\begin{aligned} &\forall R,G,B: \\ &(25 \le R \le 120) \wedge \\ &(60 \le G \le 160) \wedge \\ &(25 \le B \le 100) \wedge \\ &(G \ge R) \wedge (G \ge B) \\ &\wedge \big((R+B) * 0.7 \le G\big) \\ &\Rightarrow pixel = Red \end{aligned} \quad (1)$$

(ii) A *pixel* is classified as healthy and will be converted to pure green (0,255,0) in case it has a green tone, from any bright to dark green hue, according to the rule:

$$\begin{aligned} &\forall R,G,B: \\ &((135 \le R \le 180) \wedge (130 \le G \le 160) \\ &\qquad \wedge (40 \le B \le 60)) \vee \\ &((10 \le R \le 120) \wedge (25 \le G \le 110) \\ &\qquad \wedge (10 \le B \le 95)) \\ &\Rightarrow pixel = Green \end{aligned} \quad (2)$$

(iii) A pixel of any different tone will be considered as part of the background image and will be converted to black (0,0,0).

After the filtering process, each image had its pixels counted. This first counting process simple counts, separately, the amount of Red, Green and Black pixels. It was interesting to notice how the proposed method keeps its performance in discriminating brown spots, even when the focuses of an image was poor.

4.2 Grayscale converting

Similar to the process described in the previous section, the colored images were converted to grayscale images, using a standard method of Java language. After this monochromatic conversion, each image has had its pixels counted again. This second counting process focuses on counting how many pixels were in predetermined grayscales bands. These bands are determined as follows:

$$class(x) = \begin{cases} Black, & x \ge 200 \\ Dark\ Gray, & x \in [150,200) \\ Medium\ Gray, & x \in [100,150) \\ Bright\ Gray, & x \in [50,100) \\ White, & x < 50 \end{cases} \quad (3)$$

Each value, of each pixel *x* is analyzed and associated to a color spectrum and, for each new association, a specific counter is updated. In addition, it will also accounted the mean value of all gray tones, called *MeanGrayTone* variable, which includes all *DarkGray*, *MediumGray* and *BrightGray* pixels values, as follows:

$$\begin{aligned} &MeanGrayTone = \\ &Where: \\ &Where: \\ &DG = \{x | x \in Dark\ Gray\}, \\ &MG = \{x | x \in Medium\ Gray\}, \\ &BG = \{x | x \in Bright\ Gray\} \\ &pixelValuel(x) = nominal\ value\ of\ pixel\ x \\ &n = number\ of\ pixels\ classified\ as\ Gray \end{aligned} \quad (4)$$

5 Analysis of relevance

Further the different pixels counting over the Red/Green and Grayscale images, the mean values of the original images RGB components are also accounted. Those R, G and B means were calculated in a similar way as the *MeanGrayTone* (4); The *MeanRed* is obtained by calculating the arithmetic mean of all R (red) components, from the RGB representation, among all pixels that form the image. A similar procedure is used to calculate the *MediumGreen* and *MediumBlue* values. It was noted during the analysis that the order of magnitude of each variable is quite heterogeneous; thus all variables were normalized by applying a linear normalization in an interval between 0 and 1.

To improve the results of the first experiments, the dimension of the samples were reduced by a relevance analysis of available variables, leading to a subset that contains the most significant [19]. The first step was to separate the normalized set of samples in two different groups: one having those that shown visual evidences of problems and the other containing the ones that were considered healthy. The two groups will be identified as G_1 and G_0, respectively. Afterwards, we analyzed the statistical moments of 1st and 2nd order of each variable, in each of the groups G_1 and G_0, in order to check the discriminative capacity of them, by considering the spatial separation of sets G_1 and G_0 [20]. Thus, given a variable X, for this one to be considered a discriminative variable, it must follow the condition:

$$\begin{aligned} &\{\ X \in D\ \mid \\ &(\ \mu(X_1) > \mu(X_0)\ \ \wedge (\mu(X_0) + \delta^2(X_0) + tol\ <\ \mu(X_1)) \\ &\vee \\ &(\mu(X_0) > \mu(X_1)\ \ \wedge (\mu(X_1) + \delta^2(X_1) + tol\ <\ \mu(X_0))\ \} \end{aligned} \quad (5)$$

Where:
$X_1 = \{x | x \in X \cap G_1\}$
$X_0 = \{x | x \in X \cap G_0\}$

Thus, for a variable to be considered discriminative, it is expected that its distribution curves of G_0 and G_1 values are fitted in a way that the confusion area between the classes are as small as possible. In the equation (5), only the variables where the G_0 and G_1 mean values are separated by a distance bigger than one standard deviation plus a tolerance value, represented by the parameter *tol*, will be considered discriminative. In the automated evaluating process that we developed, only variables with curves separated by a measure beyond the minimum distance, controlled by *tol*, will be selected. As a consequence, *tol* depends on the general characteristics of the set, and its definition is an empirical process. Moreover, different sets could select different discriminative variables, depending on image quality (in terms of light, contrast, noise, etc.). After this initial analysis, we defined *tol* as 0.1, which conducted to the selection of eight variables, a 30% discard rate of the original variable set.

6 Pattern recognition experiments

In this section, semi-automatic detection of diseases of tomatoes was investigated experimentally. A comparative study was performed using MLP networks to estimate their abilities for detecting the late blight. Among all the networks configuration analyzed, the best-suited models were chosen based on the highest recognition score.

We tested 160 distinct neural network configurations, by varying the learning rate from 0.2 up to 0.8 (with a step of 0.2), the *momentum* from 0.1 up to 0.9 (with a step of 0.2), and the number of hidden neurons from 2 up to 16. Some features of the architecture of all tested networks were identical: 8 input neurons, one output neuron, sigmoid transfer function and a maximum of 50,000 *epochs* of training, for a global maximum error of 0.01. For each configuration, 50 tests were made, in a total of 8,000 batches of training and testing, so the calculating of average performance of each configuration has statistical validity. For all experiments, the training and testing sets were split in a ratio of 9:1.

From these initial batches, the average of performance was calculated to each architecture, and the architectures that showed the highest classification power were identified (Table 1). These results were obtained by providing to the networks, during the testing phase, only validation samples. that had not been presented to the network during the training phases.

Table 1: Best recognition performances among all different neural networks configurations tested, using a 0.5 class separation threshold, and tol=0.1.

Hidden Neurons	Recog. Score (Mean)	Best Recognition Score (Learning Rate / Momentum)
2	69.36	77.94 (0.4 / 0.7)
4	70.32	77.65 (0.2 / 0.5)
6	70.66	78.82 (0.2 / 0.7)
8	71.15	79.71 (0.2 / 0.1)
10	68.21	79.41 (0.2 / 0.9)
12	70.96	76.18 (0.6 / 0.3)
14	70.49	80.00 (0.2 / 0.7)
16	71.36	78.24 (0.2 / 0.1)

As the majority of the neural networks presented a higher performance with a learning rate of 0.2, this was the value used for the subsequent calculation of the ideal separation threshold. The calibration of the outputs threshold of the neural networks could affect the system performance as a whole. In the first tests, we used a standard value of 0.5 for the boundary between the outputs: for a given input sample, if the network output was lower than 0.5, the sample would be considered belonging to class G_0. For values above this threshold, the sample would be classified as belonging to class G_1.

Bayes' theorem can be applied to decision-making criteria for classifications made by neural network systems [21]. To calibrate the threshold used for each network configuration and to improve the hit rate in the classification, each network was presented to 50 testing subsets, each subset consisting of 20 samples, randomly chosen from the testing set. All answers, provided by each group of networks with the same configuration, were separated into two subsets: the first containing the answers for samples of affected plants (class S_1) and the second containing the answers for healthy samples (class S_0). Again, we calculated the mean and standard deviation for each outputs subset, and used both to determine the optimal threshold for classes' separation. The distribution analysis was done by taking the whole set of outputs from the 50 networks trained with 2 hidden neurons, learning rate equal to 0.2, and *momentum* equals to 0.7. Based on the distribution analysis, each threshold was defined as being a point inside the intersection area between the distribution curves, that would best select the sets. In this example, the optimal threshold value is 0.67, which increased the classifying performance from 73.55% to 82.24% of hits.

The same visual analysis of outputs distribution graphs was performed for the eight networks selected in the previous step. For each configuration, we defined an individual threshold value. Now using these new thresholds and also the same trained networks, we achieved better results. It can be seen that the threshold adjustment for already trained networks improved the average performance for all configurations. Finally, we selected the configuration with the best performance: the 8-14-1 network, with training rate equal to 0.2, *momentum* equal to 0.3, and threshold of 0.63. When we analyzed each network of all 50 with this configuration, we found networks that have achieved an accuracy rate of 94.12%.

A sensitivity analysis of the results obtained by each network configuration can bring more information about the performance as a whole. Table 2 shows the details of the classifications made, taking into account the percentage of correct answers for each network, in each category of sample, i.e.; the percentage of correct responses for the samples labeled 0 (no evidence of disease) or 1 (with evidence of late blight). The table also shows the percentage of false positive and false negative classifications. A false positive is a sample of healthy plant that was classified as having a disease, and a false negative is the opposite. The results on Table 2 were calculated upon the tests described above (each network was presented to 50 testing sets, where each set consisted of 20 randomly selected samples). Plus, we performed the analysis using the optimal classification thresholds for each configuration, as explained before

Table 2: Sensitivity Analysis of tested neural networks configurations.

Hidden Neurons	% Score Healthy Samples	% Score Injury Samples	% False Positives	% False Negatives	Available Healthy Samples	Available Injury Samples
2	75.70	84.43	24.30	15.57	251	749
4	83.25	84.23	16.75	15.77	238	762
6	83.98	87.56	16.02	12.44	271	729
8	81.97	85.74	18.03	14.26	274	726
10	85.09	88.10	14.91	11.90	268	732
12	85.78	86.53	14.22	13.47	240	760
14	84.55	89.30	15.45	10.70	274	726
16	85.28	88.67	14.72	11.33	231	769

7 Conclusion and future work

This work presented an ANN computational approach, based on MLP, aimed at detection of late blight, the most common tomato disease on Brazilian small properties and farms.

In this work, we reported a set of quantitative experiments and their initial results based on pattern recognition of late blight through the use of simple techniques to treat im real field mages and MLP neural networks. The Red/Green filter is a simple technique to process massive amount of digital images if compared to more sophisticated digital images algorithms, such as edge detection treatments or trimming the bottom, among others, but it has showed to be strong enough to overcome the focus, blur, lightning and definition limitations of the digital images.

We stress that, despite its simplicity, the propoesed approach may be suitable and may benefit small tomato workers. The ANN model is also able to be retrained, and the range of experimental conditions may be expanded by addition of new sets of experiments. Thus, it can be concluded that the application of ANN models and pattern recognition may be considered as an alternative for the detection of late blight in tomato crops.

8 Acknowledgement

We are grateful to FAPERJ (E-26/112.588/2012 and E-26/110.928/2013) and Brazilian Fund For Development of Education FNDE-MEC/SeSU by the Financial Support.

9 References

[1] IBGE – Instituto Brasileiro de Geografia e Estatística. Retrieved in October 18, 2013, from http://www.ibge.gov.br/home/estatistica/economia/pam/2010/PAM2010_Publicacao_completa.pdf, 2010. (in portuguese)

[2] Vianna, G. K., Cruz, S. M. S. "Análise Inteligente de Imagens Digitais no Monitoramento da Requeima em Tomateiros"; Anais do IX Congresso Brasileiro de Agroinformática. 2013 (in portuguese).

[3] Mizubuti, E. S. G., Maziero, J. M. N., Maffia, L. A., Haddad, F. , Lima, M. A. "CGTE Program: Simulation, Epidemiology and Management of Late Blight. Global Initiative on Late Blight Conference". Germany, 2002.

[4] Filgueira, F. A. R. "O novo manual de olericultura"; Editora da UFV, 3ª edição, 2008. (in portuguese)

[5] Nakano, O. "As pragas das hortaliças: seu controle e o selo verde"; Horticultura Brasileira, Vol. 17, n.1 (pp. 4-5), 1999. (in portuguese)

[6] Andrade, D. F. A. A. "Previsão e controle químico de pinta-preta (Alternaria solani) sob dois sistemas de condução do tomateiro (Lycopersicon esculentum Mill.)"; Master´s Thesis, Universidade Federal de Viçosa, 1997. (in portuguese)

[7] Vale F. X. R., Zambolim L., Paul P. A., Costa H. "Doenças causadas por fungos em tomate". In: ZAMBOLIM, L; VALE FXR; & COSTA, H (eds).

"Controle de doenças de plantas – Hortaliças", Vol. 2 (pp. 699-755), 2000. (in portuguese)

[8] Correa, F.M., Bueno Filho, J.S.S., Carmo, M.G.F. "Comparison of Three Diagrammatic Keys for the Quantification of Late Blight in Tomato Leaves". Plant Pathology, Vol. 58 (pp. 1128-1133), 2009.

[9] Terasvirta, T., D. Dijk. "Linear models, smooth transition autoregressions, and neural networks for forecasting macroeconomic time series: a re-examination." working paper. http://www.econ.puc-rio.br/pdf/td485.pdf. 2014.

[10] Gavidia, J. V, Gupta, V. K. "A comparison of neural networks and econometric discrete dependent variable models in prediction of occupational attainment." Journal of the Academy of Business and Economics, URL: http://www.findarticles.com/p/articles/mi_m0OGT/is_1_3/ai_n8690370. 2004.

[11] Baughman, D.R., Liu, Y.A. "Neural Networks in Bio-Processing and Chemical Engineering", Academic Press, New York, 1995.

[12] Tasdemir, K., Wirnhardt, C. "Automatic assessment of land parcel identification systems for agricultural management". IGARSS, Pg. 5697-5700, 2012.

[13] Erenturk, K., Erenturk, S. "Comparison of genetic algorithm and neural network approaches for drying process of carrot". J. Food Eng., 78, pp. 905–912, 2007.

[14] Vieira, F. S. et. al. "Utilização de Processamento Digital de Imagens e Redes Neurais Artificiais para Diagnosticar Doenças Fúngicas na Cultura do Tomate". Anais do XX Seminário de Computação, (p. 58-69), 2011. (in portuguese)

[15] Movagharnejad, K., Nikzad, M.. "Modeling of tomato drying using artificial neural network". Computers and Electronics in Agriculture. Volume 59, Issues 1–2, Pp 78–85, November 2007.

[16] Sanyal, P., Patel, S. C. "Pattern recognition method to detect two diseases in rice plants"; Imaging Science Journal, Vol. 56, n. 6 (pp. 319-325), 2008.

[17] Camargo, A., Smith, J.S. "Image Pattern Classification for the Identification of Disease Causing Agents in Plants"; Computers and Electronics in Agriculture, Vol. 66 (pp.121-125), 2009.

[18] Sevarac, Z. "Neuroph - Java neural network framework"; Retrieved in January, 2012 from: http://neuroph.sourceforge.net/. 2012.

[19] Vianna, G. K., Thomé, A. C. "Neuro-Fuzzy System for Diagnosis of Engines, Based on Oil Samples Analysis"; Annals of 4th World Multiconference On Systemics, Cybernetics And Informatics, (pp.290-295), 2000.

[20] Spiegel, M. R., et al. "Schaum's Outline of Theory and Problems of Probability and Statistics"; McGraw-Hill, 1977.

[21] Johnson, R., Wichern, D. "Applied Multivariate Statistical Analysis". Prentice-Hall International, Inc., 1992.

SESSION

PARTICLE SWARM OPTIMIZATION ALGORITHMS AND APPLICATIONS

Chair(s)

TBA

Teeth Selection Using Particle Swarm Optimization

A. N. Marana[1], E. B. Barboza[2], A. Z. Peixinho[2], and D. T. Oliveira[3]
[1]Department of Computing, Faculty of Sciences, UNESP - São Paulo State University, Bauru, São Paulo, Brazil
[2]UNESP Graduate Program on Computer Sciences, São Paulo, Brazil
[3]Bauru School of Dentistry, USP - University of São Paulo, Bauru, São Paulo, Brazil

Abstract— *Usually, the forensic human identification through dental biometrics is performed manually by human experts. An important alternative to facilitate and expedite this task is to use an ADIS (Automated Dental Identification System). In ADIS, the selection of most discrimintive teeth can improve substancialy the subject identification rates. In this paper we propose the use of BPSO (Binary Particle Swarm Optimization) in order to select optimum subsets of most discriminative teeth shape descriptors, and, consequently, to improve the rates of correct person identification. Results obtained in our experiments have confirmed our hypothesis that some teeth are more discriminative than others and should be priorized during the forensic human identification based on dental recognition. The use of BPSO have reduced the EER from 20% to 10%, and have reduced the rank-k from k=15 to k=4 in order to reach 100% of correct teeth retrieval.*

Keywords: Biometrics, Forensics, Dental Recognition, Feature Selection, Particle Swarm Optimization, Image-Foresting Transform.

1. Introduction

Biometrics is commonly used in Forensics for human identification. Many physical and behavioral features can be used for human identification, including face, fingerprints, voice, retina, iris, and vein patterns [1]. However, for deceased individuals, in many situations only dental biometrics is possible. In this paper we address the problem of chosing the best teeth for human forensic identification through dental recognition.

Usually, the process of human identification through its teeth features is carried out by forensic experts, who perform manual comparison between *antemortem* and *postmortem* records, requiring long period of time and making this practice susceptible to errors [2].

In a previous work [3], we have proposed an interactive and semiautomatic dental identification system (ADIS) for forensic human identification based on dental recognition. Our ADIS uses tooth shape dental descriptors, which are calculated by Beam Angle Statistics (BAS) or Shape Context (SC) methods [4], [5], on images segmented by using a Differential Image Foresting Transform (DIFT) segmentation algorithm [6].

In this work, we propose the use of Binary Particle Swarm Optimization (BPSO) technique [7] in order to select the most discriminative teeth among the individuals, and, thus, to improve the human identification rates in Forensics. To the best of our knowledge, no previous work have carried out automatic teeth selection by using BPSO for biometrics purposes in the same way that we have done in our work.

In order to asses our proposal, we have performed experiments on a database composed of 40 panoramic radiograph images (20 subjects, 2 images per subject). Experimental results showed that some teeth shapes are more discriminatives than others and should be the first choices for analysis, during the forensic human identification based on dental recognition. In our experiments, the use of the most discriminative teeth have reduced the equal error rate from 20.0% to 10.0% in the best case, and the 100% of correct retrieval in the Rank-k metric was reduced from $k = 15$ to $k = 4$.

2. Preprocessing

In order to carry out teeth shape analysis, it was necessary a preprocessing phase, including noise removal, brightness adjustments, region of interest selection, teeth edge detection, and, most important, teeth shape segmentation.

The teeth segmentation in the panoramic dental radiograph images was achieved by usin the DIFT-based algorithm proposed by [6].

Figure 1 shows the interface of our interactive ADIS, in which it is possible to observe the results obtained in the preprocessing phase. First the ROI is marked manually on the panoramic radiograph image (left image). Next, internal and external seeds are chosen by the user (center image). Finally, the tooth shape points are automaticaly found by the DIFT-based algorithm (top right image).

3. Shape Analysis

After the segmentation, it is necessary to extract shape descriptors from the teeth, in order do carry out the human identification. In our work, we have used two shape descriptors: Shape Context (SC), proposed by [5], and Beam Angle Statistics (BAS), proposed by [4].

In the Shape Context (SC) method, given a pixel p belonging to the shape, the polar-logarithmic distribution of

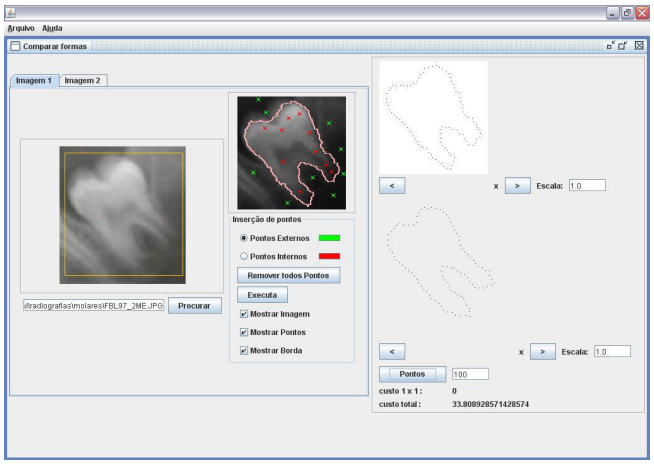


Fig. 1: Interface of our interactive ADIS (Automated Dental Identification System). It is possible to see the results of the preprocessing phase: the ROI in the left image; the tooth shape automatically found (points in central image), and the segmented tooth (points in the top right image).

the remaining shape points in relation to p is calculated, providing a global and discriminative characterization of the shape. Thus, the corresponding points of two similar shapes will have similar shape contexts. The similarity measurement between two objects is computed as being the minimum cost of alignment among the points of their shapes. The more different are the two shapes of the objects, greater will be their alignment cost.

In the Beam Angle Statistics (BAS) method, shape description is based on beams, which are the lines connecting a reference shape point p with all other shape points q. The characteristics of each shape point are extracted by using the angles formed between each pair of beams. From these angles, taken as random variables at each point on the shape, the moment theorem is applied to provide statistical information of the shape. In the functions that express the moments, the valleys and hills correspond, respectively, to concave and convex parts of the object shape. The moments describe the statistical behavior of the beam angles at the shape points and are stored in a feature vector. One of the advantages of BAS is that it is naturally invariant to rotation and scale transformations, since it is based on beam angles. The similarity measure between two shapes is determined by the best alignment cost between them, obtained in general from the features of the first two moments.

4. Distance Function

After extracting the feature vectors from the teeth shapes, it is necessary to compare them, in order to stablish a distance between a query readiography image (postmortem) and a database radiograph image (antemortem). Equation 1 shows the distance function D adopted in our work:

$$D(PM, AM) = \frac{1}{n}\sum_{i=1}^{n} dist(AM_i, PM_i) \qquad (1)$$

where PM and AM are the antemortem and postmortem radiographs, n is the maximum number of teeth being compared, and $dist(AM_i, PM_i)$ is the shape distance of the ith tooth of the antemortem and postmortem radiographs.

5. Particle Swarm Optimization

The concept of optimization based on computational simulation of social and cognitive behaviors, called PSO (Particle Swarm Optimization) was first introduced by Kennedy and Eberhart in 1995 [8].

PSO consists of a technique with low computational cost and information sharing innate to the social behavior of the composing individuals. These individuals, also called particles, flow through the multidimensional search space looking for feasible solutions for the problem. The position of each particle in this search space represents a possible solution whose feasibility is evaluated using an objective function. The particle movement is influenced by two factors: the best solution found by the particle so far ($pbest$), and the best solution found so far, considering all the particles ($gbest$). These factors, called cognitive and social components, respectively, influence in the movement of the particles through the search space creating attraction forces. As result, the particles interact with their neighbors and store in their memories the locations of the optimal solutions.

After every iteration, $pbest$ and $gbest$ are updated if a better solution is found by the particle or by the population, respectively. This process is repeated until the desired result is obtained or a certain number of iterations is reached or even if the solution possibility is discarded.

In 1997, Kennedy and Eberhart adapted the standard continuous PSO algorithm to binary spaces [7]. This variant of the continuous PSO was termed binary PSO (BPSO). In our work we have used BPSO, since our goal was to select or not (a binary decision) a specific tooth from the complete set of teeth, in order to obtain the best teeth subset for forensic human identification.

6. Experimental Results

6.1 Database

The database used in our experiments was composed of 40 panoramic dental radiograph images captured from 20 different subjects, who provided two radigraphs in two distinct sessions.

Figure 2) shows two panoramic dental radiograph images from the same subject of our database. In order to simulate a real scenario, he oldest radiograph was considered the *antemortem*, whilst the newest one was considered the *postmortem*.

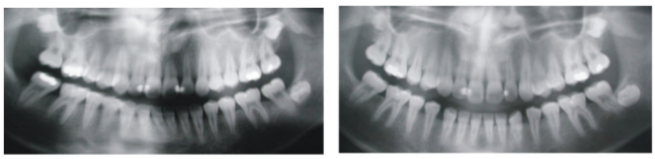


Fig. 2: Two panoramic dental radiograph images from the same subject of our database. The oldest image (left) was considered the *"antemortem"* and the newest image (right) was considered as the *"postmortem"* image.

6.2 Optimization Strategies

Two distinct strategies for selecting the most discriminative teeth were used in our work, varying the objective function during the optimization process, aiming:

- Equal Error Rate minimization;
- Rank-k minimization.

For both strategies we performed experiments composed by 20 tests, generating solutions that indicate the selected teeth in each run (a selected tooth was represented by the digit 1, and a discarded tooth by the digit 0).

In these tests, each *antemortem* radiograph was matched against all *postmortem* radiographs.

During the first 10 tests those solutions containing smaller numbers of teeth were priorized, so that only the most discriminative teeth were highlighted.

Over the next 10 tests, the same procedure was applied, however, those solutions containing higher numbers of teeth were priorized, so that the less discriminative teeth were highlighted.

The output results followed the teeth labeling notation adopted by FDI - World Dental Federation, also known as ISO-3950:2009 notation [9], as showed in Figure 3.

6.2.1 Equal Error Rate minimization

This strategy aimed to minimize the equal error rate (EER).

Without the best teeth selection, the following EER values were obtained:

- Shape Context (SC): 20.00%;
- Beam Angle Statistics (BAS): 15.52%.

The results obtained with the Shape Context descriptor, after the optimization, prioritizing solutions with minimum number of teeth and maximum number of teeth are showed in Tables 1 and 2, respectively.

Analyzing the teeth that were selected in all tests for Shape Context descriptor, using the approach that priorize a minimum number of teeth, we can highlight the importance of the following teeth: upper right central incisor (11), first upper left molar (26), lower right canine (43), and first lower right molar (46). See Figure 3, to locate these teeth in the mouth, according to the FDI notation.

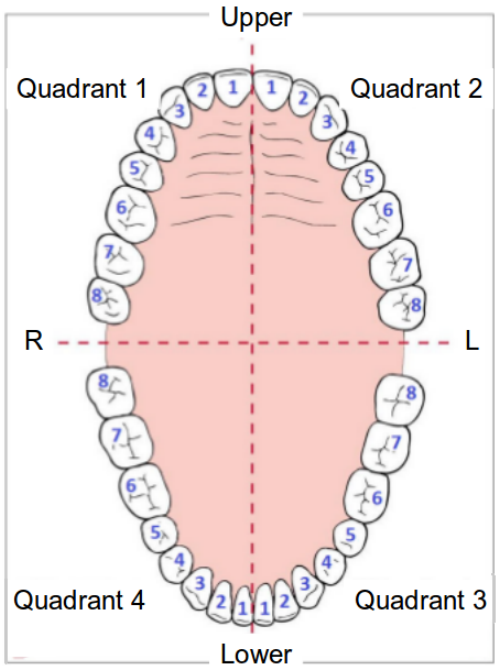


Fig. 3: Teeth FDI notation [9].

Table 1: Minimize EER - Shape Context. Solutions with minimum number of teeth.

	Shape Context (Less Teeth)				
	Teeth				
Quadrant	1	2	3	4	EER (%)
Number	12345678	12345678	12345678	12345678	
Test 1	10000000	00000100	00010000	01110110	10
Test 2	10000000	00000100	00010000	10110110	10
Test 3	10000000	00000100	00010000	01110110	10
Test 4	10000000	00000100	00110000	00110110	10
Test 5	10000000	00000100	00010000	10110110	10
Test 6	10000000	00000100	00100100	01100100	10
Test 7	10000000	00000100	00100100	01100100	10
Test 8	10000010	00000100	00010000	01110100	10
Test 9	10000000	00000100	10010000	00110110	10
Test 10	10000000	00000100	00010000	01110110	10

Analyzing the teeth discarded in all tests for Shape Context descriptor, using the approach that priorize a maximum number of teeth, we can highlight the following teeth that were less discriminative: first upper rigth premolar (14), first upper left premolar (24), second upper right premolar (15), second upper left premolar (25), second lower right premolar (45), first upper right molar (16), and the lower left central incisor (31). See Figure 3, to locate these teeth in the mouth, according to the FDI notation.

The results obtained with the BAS descriptor, after the optmization, prioritizing solutions with minimum number of teeth and maximum number of teeth are showed in Tables 3 and 4, respectively.

Analyzing the teeth that were selected in all tests for BAS descriptor, using the approach that priorize a minimum

Table 2: Minimize EER - Shape Context. Solutions with maximum number of teeth.

	Shape Context (More Teeth)				
	Teeth				
Quadrant	1	2	3	4	EER (%)
Number	12345678	12345678	12345678	12345678	
Test 1	11000011	11000101	00111111	11110111	10
Test 2	11000011	11000101	00111111	11110111	10
Test 3	11000011	10100101	00111111	11110111	10
Test 4	11000011	10100101	00111111	11110111	10
Test 5	11100000	11000111	01111111	01110111	10
Test 6	11000011	10100101	00111111	11110111	10
Test 7	11000011	10100101	00111111	11110111	10
Test 8	11100001	11000110	01111111	01110111	10
Test 9	11100001	11000110	01111111	01110111	10
Test 10	11100000	11000111	01111111	01110111	10

Table 3: Minimize EER - BAS. Solutions with minimum number of teeth.

	BAS (Less Teeth)				
	Teeth				
Quadrant	1	2	3	4	EER (%)
Number	12345678	12345678	12345678	12345678	
Test 1	10010001	00100000	01001010	01100100	10
Test 2	10010000	00100101	01001000	01100110	10
Test 3	00010000	00100000	00010001	00101110	10
Test 4	01001000	00100101	00011010	00100100	10
Test 5	00001010	00100101	01010010	00100100	10
Test 6	01010011	00100010	00001010	01100100	10
Test 7	10010001	00100000	01001010	01100100	10
Test 8	10010001	00100000	01001010	01100100	10
Test 9	10010001	00100000	01001010	01100100	10
Test 10	01001001	00100001	00011010	01100000	10

number of teeth, we can highlight the importance of the following teeth: upper left canine (23), and lower right canine (43). See Figure 3, to locate these teeth in the mouth, according to the FDI notation.

Table 4: Minimize EER - BAS. Solutions with maximum number of teeth.

	BAS (More Teeth)				
	Teeth				
Quadrant	1	2	3	4	EER (%)
Number	12345678	12345678	12345678	12345678	
Test 1	11001011	10100111	11011110	01110111	10
Test 2	01111000	01100000	11011011	01101111	10
Test 3	11011011	10100111	01011110	11110101	10
Test 4	11011011	10100111	01011110	11110101	10
Test 5	11001011	10100111	11011110	11110101	10
Test 6	11001011	10100111	11011110	01110111	10
Test 7	01111100	01100000	10011011	10101111	10
Test 8	11011001	11100101	01011010	11110101	10
Test 9	11011011	10100111	01011110	11110101	10
Test 10	11001011	10100111	11011110	01110111	10

Analyzing the teeth discarded in all tests for BAS descriptor, using the approach that priorizes a maximum number of teeth, we can highlight the teeth that were less discriminative for the descriptor: first upper right molar (16), first upper left premolar (24), second upper left premolar (25), lower left canine (33), third lower left molar (38), second lower right premolar (45). See Figure 3, to locate these teeth in the mouth, according to the FDI notation.

Observing the experimental results, we can conclude that the use of BPSO optimization strategies leaded to better dental recognition results, since the EER was reduced to 10% for both descriptors. Before the optmization, the Shape Context EER was 20%, and the Beam Angle Statistics EER was 15.52%.

6.2.2 Rank-k minimization

Rank-k identification rate means the number of times the identity is in the top k most likely candidates.

Without the selection of the most discriminative teeth, 100% of recovery was reached by both shape descriptors (SC and BAS) only for $k = 15$ ($Top15$). It means that only considering the 15 top matches it was possible, before the BPSO optimization, to ensure the retrieval of all subjects included in the group.

The results obtained with the Shape Context descriptor, after the optimization, priorizing solutions with a minimum number of teeth and a maximum number of teeth are showed in Tables 5 and 6, respectively.

Table 5: Minimize Rank - Shape Context. Solutions with minimum number of teeth.

	Shape Context (Less Teeth)				
	Teeth				
Quadrant	1	2	3	4	Rank
Number	12345678	12345678	12345678	12345678	
Test 1	00100000	11000100	00010000	00001100	Top5
Test 2	11000000	01000100	00110001	00011100	Top4
Test 3	11000000	00000101	10110001	00001100	Top4
Test 4	10100000	11000100	00100001	00001100	Top4
Test 5	10100000	01000100	00100011	00010000	Top4
Test 6	11000000	01000100	00110001	00011100	Top4
Test 7	01000000	01000100	00100011	00000100	Top4
Test 8	10100000	11000100	00100001	00001100	Top4
Test 9	11000000	01000100	00110001	00011100	Top4
Test 10	10100000	01000100	00101001	00011100	Top4

Analyzing the teeth that were selected in all tests for Shape Context descriptor, using the approach that priorizes a minimum number of teeth, we can highlight the following teeth that were more discriminative: upper right central incisor (11), upper left lateral incisor (22), first upper left molar (26), lower left canine (33), third lower left molar (38), and first lower right molar (46). See Figure 3, to locate this tooth in the mouth, according to the FDI notation.

Analyzing the teeth discarded in all tests for Shape Context descriptor, using the approach that priorizes a maximum number of teeth, we can highlight the following teeth that were less discriminative: first upper right molar (16), second upper right molar (17), second upper left molar (27), upper left central incisor (21), lower left central incisor (31), lower right central incisor (41), first lower left molar (36), lower right canine (43), and lower left lateral incisor (32). See Figure 3, to locate these teeth in the mouth, according to the FDI notation.

Table 6: Minimize Rank - Shape Context. Solutions with maximum number of teeth.

	Shape Context (More Teeth)				
	Teeth				Rank
Quadrant	1	2	3	4	
Number	12345678	12345678	12345678	12345678	
Test 1	10111000	01000101	00111001	00000111	Top4
Test 2	10111001	01011101	00111011	01011111	Top5
Test 3	01101000	01100101	00111000	01011111	Top4
Test 4	11101001	01100101	00110011	00001111	Top4
Test 5	01101000	01100101	00111000	01011111	Top4
Test 6	10111001	01011101	00111011	01011111	Top5
Test 7	10111001	01011101	00111011	01011111	Top5
Test 8	01101001	01100100	00111000	01011111	Top4
Test 9	01101000	01100101	00111000	01011111	Top4
Test 10	11101001	01100101	00110011	00001111	Top4

The results obtained with the BAS descriptor, after the optimization, priorizing solutions with a minimum number of teeth and a maximum number of teeth are showed in Tables 7 and 8, respectively.

Table 7: Minimize Rank - BAS. Solutions with minimum number of teeth.

	BAS (Less Teeth)				
	Teeth				Rank
Quadrant	1	2	3	4	
Number	12345678	12345678	12345678	12345678	
Test 1	01001000	00100000	00010000	01010100	Top4
Test 2	11001000	00100000	00001000	01000100	Top3
Test 3	10110010	00100000	00001000	01001100	Top4
Test 4	01001000	00000100	00011000	00001110	Top4
Test 5	01101000	00000100	00001000	00000110	Top4
Test 6	11001000	00100000	00001000	01000100	Top3
Test 7	01001000	00100000	00010000	00010100	Top4
Test 8	11001000	11000000	00011010	00000100	Top3
Test 9	11001000	11000000	00011010	00000100	Top3
Test 10	01101000	00000100	00001000	00000110	Top4

Analyzing the teeth that were selected in all tests for BAS descriptor, using the approach that priorizes a minimum number of teeth, we can highlight the importance of the following teeth: upper right lateral incisor (12), second upper right premolar (15), second lower left premolar (35), and first lower right molar (46). See Figure 3, to locate these teeth in the mouth, according to the FDI notation.

Analyzing the teeth discarded in all tests for BAS descriptor, using the approach that priorizes a maximum number of teeth, it is possible to highlight the following teeth that were less discriminative: first upper right molar (16), third upper right molar (18), first upper left premolar (24), second upper left premolar (25), second upper left molar (27), first lower left molar (36), and second lower left molar (37). See Figure 3, to locate these teeth in the mouth, according to the FDI notation.

The rank minimization strategy leaded to an important gain, ensuring 100% of accuracy rate considering only the $Top4$ most similar matches, while before the optimization the same accuracy was obtained with $Top15$ most similar matches.

Table 8: Minimize Rank - BAS. Solutions with maximum number of teeth.

	BAS (More Teeth)				
	Teeth				Rank
Quadrant	1	2	3	4	
Number	12345678	12345678	12345678	12345678	
Test 1	11111010	11100101	11010001	11011111	Top4
Test 2	11111010	11100100	11011001	11011111	Top4
Test 3	11111010	11100100	11011001	11011111	Top4
Test 4	11111010	11100100	01111001	01111111	Top4
Test 5	11111010	11100100	11011001	11011111	Top4
Test 6	11111010	11100101	11010001	11011111	Top4
Test 7	11111010	11100101	11010001	11011111	Top4
Test 8	11111010	11100100	11011001	11011111	Top4
Test 9	11111010	11100100	01111001	01111111	Top4
Test 10	11111010	11100100	01111001	01111111	Top4

6.3 Best teeth subsets for human authentication and recognition

In order to choose the best subset of teeth for forensic human identification we have used an approach based on the frequency of selection of each tooth.

For each tooth, it was counted the number of times it was selected in the n tests performed, generating a frequency vector $\vec{f}$.

Then, it was determined a threshold t so that the teeth with occurrence frequencies lower than t were discarded. The optimal threshold t was obtained by testing all possible values of t in the range $[0, n]$, where n represents the number of performed tests.

After performing this process for the results presented in Section 6.2, the best solutions for each teeth shape descriptor (SC and BAS) were obtained for the purposes of authentication (using the EER optimization), and recognition (using EER optimization). The best solution for each purpose and their performances (EER or Rank) are shown in Tables 9 and 10, respectively.

Table 9: Best solution for authentication purpose.

	Teeth				
Quadrant	1	2	3	4	EER (%)
Number	12345678	12345678	12345678	12345678	
SC	10000000	00000100	00110000	01110110	10
BAS	01001000	00100000	01011010	01100101	11.31

Table 10: Best solution for recognition purpose.

	Teeth				
Quadrant	1	2	3	4	Rank
Number	12345678	12345678	12345678	12345678	
SC	00101000	01000101	00111000	00001111	Top5
BAS	11111010	11100100	11011001	11011111	Top4

7. Conclusion and future work

In our previous work [3], we have proposed an interactive ADIS (Automatic Dental Recognition System), which obtained good results by using the DIFT-based algorithm for tooth segmentation and Shape Context or BAS shape description methods for semiautomatic dental recognition.

In this paper we have extended our previous work, by using the BPSO (Binary Particle Swarm Optimization) method in order to select the best teeth (most discriminative teeth) to be used for forensic human identification based on dental recognition using panoramic dental radiograph images.

Experimental results obtained in this work showed that the using of subsets of teeth present in the human body can increase significantly the identification performance. In our experiments, the Equal Error Rates (EER) was reduced from 20% to 10% for both shape description methods (Shape Context and Beam Angle Statistics), and in the Rank-k analysis, in order to obtain 100% of retrieval the rank k was reduced from $k = 15$ to $k = 4$.

Another result observed in our experiments is that the lower teeth (quadrants 3 and 4) seems to be more discriminative than the upper teeth (quandrants 1 and 2). This suggests that even for manual analysis the human experts shoud priorize the analysis of the lower teeth. This result can be due to the noise present in the upper area of the panoramic dental radiographs or due to the intrinsic characteristics of the teeth anatomy. Further investigations will be carried out on a large panoramic dental radiograph image database in order to confirm these findings.

References

[1] A. K. Jain, A. Ross, and K. Nandakumar, *Introduction to Biometrics*. Springer, 2011.

[2] H. Chen and A. Jain, "Tooth contour extraction for matching dental radiographs," in *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, vol. 3. IEEE, 2004, pp. 522–525.

[3] E. B. Barboza, A. N. Marana, and D. T. Oliveira, "Semiautomatic dental recognition using a graph-based segmentation algorithm and teeth shapes features," in *Biometrics (ICB), 2012 5th IAPR International Conference on*. IEEE, 2012, pp. 348–353.

[4] N. Arica and F. Yarman Vural, "Bas: a perceptual shape descriptor based on the beam angle statistics," *Pattern Recognition Letters*, vol. 24, no. 9-10, pp. 1627–1639, 2003.

[5] S. Belongie, J. Malik, and J. Puzicha, "Shape context: A new descriptor for shape matching and object recognition," in *Neural Information Processing Systems Conference (NIPS)*, December 2000, pp. 831–837.

[6] A. Falcão and F. Bergo, "Interactive volume segmentation with differential image foresting transforms," *Medical Imaging, IEEE Transactions on*, vol. 23, no. 9, pp. 1100–1108, 2004.

[7] J. Kennedy and R. Eberhart, "A discrete binary version of the particle swarm algorithm," in *Systems, Man, and Cybernetics, 1997. Computational Cybernetics and Simulation., 1997 IEEE International Conference on*, vol. 5. IEEE, 1997, pp. 4104–4108.

[8] ——, "Particle swarm optimization," in *Neural Networks, 1995. Proceedings., IEEE International Conference on*, vol. 4. IEEE, 1995, pp. 1942–1948.

[9] ISO, *ISO 3950:2009 Dentistry – Designation system for teeth and areas of the oral cavity*, 2009.

SPSO-TVAC Congestion Management in a Practical Electricity Market Based Generator Sensitivity

H. A. Shayanfar*
Department of Electrical Engineering
College of Technical and Engineering
South Tehran Branch
Islamic Azad University, Tehran, Iran

H. Shayeghi
Technical Engineering Department
University of Mohaghegh Ardabili
Ardabil, Iran

A. Ghasemi
Technical Engineering Department
University of Mohaghegh Ardabili
Ardabil, Iran

hashayanfar@gmail.com, hshayeghi@gmail.com, ghasemi.agm@gmail.com

Abstract- *Congestion management is one of the major tasks performed by system operators to ensure the operation of transmission system within operating limits. In the emerging electric power market, the congestion management becomes extremely important and it can impose a barrier to the electricity trading. This paper presents a new method of Speed Particle Swarm Optimization with Time-Varying Acceleration Coefficients (SPSO-TVAC)-based Congestion Management (CM) by optimal rescheduling of active powers of generators. In the proposed method, generators are selected based on their sensitivity to the congested line for efficient utilization. The task of optimally rescheduling the active powers of the participating generators to reduce congestion in the transmission line is attempted by SPSO-TVAC. The SPSO-TVAC algorithms are tested on the IEEE 30-bus, IEEE 118-bus and Practical Indian 75-bus systems, after which the results are compared with conventional PSO to determine the effectiveness of CM. Compared with PSO-TVAC, PSO-TVIW, FPSO and PSO, SPSO-TVAC can better perform the optimal rescheduling of generators to relieve congestion in the transmission line.*

Keywords: Electricity Market, SPSO-TVAC, Optimal Congestion Management, Efficiency and Security of Distribution Network Improvement, practical study.

List of symbols

g: Participating generator
N_g: Number of participating generators
IC_g: Incremental and decremented cost of generator g
ΔP_g Active power adjustment at bus g.
ΔP_g^{min}: Minimum adjustment limit of generator g.
ΔP_g^{max}: Maximum adjustment limit of generator g.
P_g: Active power output.
P_g^{min}: Minimum generation limit of generator g.
P_g^{min}: Maximum generation limit of generator g.
F_l^0: Power flow caused by all contracts requesting the transmission service.
F_l^{max}: Power flow limit of line l.
n_l: Number of transmission lines in the system.
ΔP_{ij}:Changed in active power flow on the line connected between buses i and j.
ΔPG_g: Changed in active power of generator g.
n: Number of all the buses in the system.
V_i, V_j: Voltage magnitude at buses i and j.
θ_i, θ_j: Phase angle at buses i and j.
G_{ij}: Conductance of the line connected between buses i and j.
B_{ij}: Susceptance of the line connected between buses i and j.
K: Current iteration number

I. INTRODUCTION

Transmission lines are often driven close to or even beyond their thermal limits in order to satisfy the increased electric power consumption and trades due to increase of the unplanned power exchanges. If the exchanges were not controlled, some lines located on particular paths may become overloaded, this phenomenon is called congestion. Political and environmental constraints make the construction of new transmission lines difficult and restrict the electrical utilities from better use of existing network. It is attractive for electrical utilities to have a way of permitting more efficient use of the transmission lines by controlling the power flows [1].

In irregular electrical environment, Independent System Operators (ISO) managed system line congestion. Temperature, voltage and stability are system limitations. Selective method for congestion management should be accepted from economical aspect and the necessary incentive to increase investment to provide network capacity for transmission network and generation division.

In [1] presented a number of CM approaches. In [2-4], a technique which is called congestion factor of distribution network is described. Ranking zone categorized by sensitivity index is divided to active and reactive power. This technique in computational aspect is complex. In [5], an optimal load flow with minimization congestion cost and service is presented. In [6], is described OPF for synchronization between the producing company and ISO using disconnected port. Ref. [7], used OPF can be injected power to adjust the system condition for instability state and over load heat. In [8],

* Corresponding Author. E-Mail Address: hashayanfar@yahoo.com (H. A. Shayanfar)

is described the concept of Relative Electrical Distance (RED) to alleviate overloaded lines by the timing of active power is presented. This method minimized system loss and improved voltage profile. However, in this paper has been not raised schedule cost has not been raised. In [9], a method of production scheduling optimization considering scheduling based on cost minimization objective PSO is presented.

Although, these methods seem to be good approaches for the solution of the CM optimization problem, however, when the system has a highly epistatic objective function (i.e. where parameters being optimized are highly correlated), and the number of parameters to be optimized is large, then they have degraded effectiveness to obtain the global optimum solution. In order to overcome these drawbacks, a new improved version of PSO (SPSO-TVAC) technique is proposed in this paper. The proposed algorithm is a typical swarm-based approach to optimization and has emerged as a useful tool for the engineering optimization. Unlike the other heuristic techniques, the PSO algorithm has a flexible and well-balanced mechanism to enhance the global and local exploration abilities.

II. PROBLEM EXPRESSION

The optimal congestion management minimizing re-dispatch cost can be expressed as [9].

$$Min \sum_{g=1}^{Ng} Ic_g\,(\Delta P_g).\Delta P_g \tag{1}$$

where, ICg and ΔPg are incremental, decremental cost, respectively and active power adjustment is of generator g. N_g the is number of participating generators.

Subject to:

Power balance constraint:

$$\sum_{g=1}^{Ng} \Delta P_g = 0 \tag{2}$$

Operating limit constraints:

$$\Delta P_g^{\min} \le \Delta P_g \le \Delta P_g^{\max}; g = 1,2,...,Ng \tag{3}$$

where, $\Delta P_g^{\min} = P_g^{\min} - P_g; \Delta P_g^{\min} = P_g^{\min} - P_g$.

Line flow constraints:

$$\sum_{g=1}^{Ng} (GS_g^{ij}.\Delta P_g) + F_l^0 \le F_l^{\max} \tag{4}$$

2.1. Selecting Re-dispatched generators

The Generator Sensitivity (GS) technique indicates the change of active power flow due to change in active power generation. The GS value of generator g on the line connected between buses i and j can be written as [9].

$$GS_g^{ij} = \frac{\Delta P_{ij}}{\Delta P_{Gg}} = \frac{\partial P_{ij}}{\partial \theta_i}.\frac{\partial \theta_i}{\partial P_{Gg}} + \frac{\partial P_{ji}}{\partial \theta_j}.\frac{\partial \theta_j}{\partial P_{Gg}} \tag{5}$$

The power flow equation on congested lines can be calculated by:

$$\frac{\partial P_{ij}}{\partial \theta_i} = -V_i V_j .G_{ij} .\sin(\theta_i - \theta_j) + V_i V_j .B_{ij} .\sin(\theta_j - \theta_i) \tag{6}$$

$$\frac{\partial P_{ji}}{\partial \theta_i} = +V_i V_j .G_{ij} .\sin(\theta_i - \theta_j) - V_i V_j .B_{ij} .\sin(\theta_j - \theta_i) = -\frac{\partial P_{ij}}{\partial \theta_i} \tag{7}$$

The relation between the change in active power at each bus and voltage phase angles can be written as:

$$[\Delta P]_{n\times 1} = [H]_{n\times n} \times [\Delta]_{n\times 1} \tag{8}$$

$$[H]_{n\times n} = \begin{bmatrix} \frac{\partial P_1}{\partial \theta_1} & \frac{\partial P_1}{\partial \theta_2} & \cdots & \frac{\partial P_1}{\partial \theta_n} \\ \frac{\partial P_2}{\partial \theta_1} & \frac{\partial P_2}{\partial \theta_1} & \cdots & \frac{\partial P_2}{\partial \theta_n} \\ : & : & & : \\ . & . & & . \\ \frac{\partial P_n}{\partial \theta_1} & \frac{\partial P_n}{\partial \theta_2} & \cdots & \frac{\partial P_n}{\partial \theta_n} \end{bmatrix}_{n\times n} \tag{9}$$

where, $[M] = [H]^{-1}$ therefore $[\Delta\theta]_{n\times 1} = [M]_{n\times n} \times [\Delta P]_{n\times 1}$. Since bus 1 is the reference bus, the first row and first column of [M] can be eliminated. Therefore, the modified [M] is written as:

$$[\Delta\theta]_{n\times 1} = \begin{bmatrix} 0 & 0 \\ 0 & [M-1] \end{bmatrix}_{n\times n} \times [\Delta P]_{n\times 1} \tag{10}$$

In (10), the modified [M] represents the values of $\frac{\partial \theta_i}{\partial P_{Gg}}$ and $\frac{\partial \theta_j}{\partial P_{Gg}}$ in (5) to calculate GS values. Large GS generators will be selected for redispatched since they are more influential on the congested line.

III. SPSO-TVAC

3.1. CPSO Technique

Classic PSO (CPSO) has been found to be robust in solving problems featuring nonlinearity and non-differentiability, multiple optima, and high dimensionality through adaptation, which is derived from the social-psychological theory. The features of the method are as follows [10]:

The method is developed from research on swarm such as fish schooling and bird flocking. It is based on a simple concept. Therefore, the computation time is short and requires few memories [11]. It was originally developed for nonlinear optimization problems with continuous variables. CPSO is basically developed through simulation of bird flocking in two-dimension space. The position of each agent is

represented by X_Y axis position and also the velocity is expressed by V_X (the velocity of X axis) and V_Y (the velocity of Y axis). Modification of the agent position is realized by the position and velocity information [12]. Bird flocking optimizes a certain objective function. Each agent knows its best value so far (pbest) and its XY position. This information is analogy of personal experiences of each agent. Moreover, each agent knows the best value so far in the group (gbest) among pbest. This information is analogy of knowledge of how the other agents around them have performed. Namely, each agent tries to modify its position using the following information [12]:

The current positions (x, y),

The current velocities (V_X, V_Y),

The distance between the current position and p_{best}

The distance between the current position and gbest

This modification can be represented by the concept of velocity and the place of that. Velocity of each agent can be modified by the following equation:

$$x_i(t+1) = x_i(t) + v_i(t+1) \tag{11}$$

$$V_i(t+1) = \omega v_i(t) + c_1 r_1(t)[pbest_i(t) - x_i(t)] + c_2 r_2(t)[leader_i(t) - x_i(t)] \tag{12}$$

Where,

x_i: position of agent i at iteration k

v_i: velocity of agent i at iteration k

w: inertia weighting

$c_{1,2}$: tilt coefficient

$r_{1,2}$: rand random number between 0 and 1

leader: archive of unconquerable particles

p_{besti}: pbest of agent i

g_{best}: gbest of the group

Convergence of the PSO strongly depended on w, c1 and c2. While c1,2 are between 1.5 till 2, however the best choice to these factors is 2.05. Also, $0 \le w < 1$ whereas this value is really important factor to the system convergence and this is better that this factor define dynamically. While it should be between 0.2 and 0.9 and it should decrease linear through evolution process of population. Being extra value of w at first, provides appropriate answers and small value of that help the algorithm to convergence at the end [12].

3.2. PSO with Time-Varying Inertia Weight (PSO-TVIW)

The PSO-TVIW method is capable of locating a good solution at a significantly faster rate, when compared with other evolutionary optimization methods; its ability to fine tune the optimum solution is comparatively weak, mainly due to the lack of diversity at the end of the search [13]. Also, in PSO, problem-based tuning of parameters is a key factor to find the optimum solution accurately and efficiently. The main concept of PSO-TVIW is similar to CPSO in which the Eqs. (11), (12) are used. However, for PSO-TVIW the velocity update equation is modified by the constriction factor C and the inertia weight w is linearly decreasing as iteration grows [14].

$$V_i(t+1) = C\{\omega v_i(t) + c_1 r_1(t)[pbest_i(t) - x_i(t)] + c_2 r_2(t)[leader_i(t) - x_i(t)]\} \tag{13}$$

3.3. PSO-TVAC

Consequently, PSO-TVAC is extended from the PSO-TVIW. All coefficients including inertia weight and acceleration coefficients are varied with iterations. The equation of PSO-TVAC for velocity updating can be expressed as [15]:

$$V_i(t+1) = C\{\omega v_i(t) + ((c_{1f} - c_{1i})\frac{k}{k_{\max}} + c_{1i}).r_1(t)[pbest_i(t) - x_i(t)] + ((c_{2f} - c_{2i})\frac{k}{k_{\max}} + c_{2i}).r_2(t)[leader_i(t) - x_i(t)]\} \tag{14}$$

3.4. Speed PSO-TVAC

Consequently, SPSO-TVAC is extended from the PSO-TVIW and PSO-TVAC. All coefficients including inertia weight, acceleration coefficients and speed for any step are varied with iterations. The equation of SPSO-TVAC for velocity updating can be expressed as:

$$\omega = (\omega_{\max} - \omega_{\min}).\frac{(k_{\max} - k)}{k_{\max}} + \omega_{\min} \tag{15}$$

$$C = \frac{2}{\left|2 - \varphi - \sqrt{\varphi^2 - 4\varphi}\right|}, where 4.1 \le \varphi \le 4.2$$

$$V_i(t+1) = C\{\omega v_i(t) + ((c_{1f} - c_{1i})\frac{k}{k_{\max}} + c_{1i}).r_1(t)[pbest_i(t) - x_i(t)] + ((c_{2f} - c_{2i})\frac{k}{k_{\max}} + c_{2i}).r_2(t)[leader_i(t) - x_i(t)]\} \tag{16}$$

and

c_{1i} =2.7; c_{1f} =0.3; c_{2i} =0.4; c_{2f} =2.6;

$c_1 = ((c_{1f} - c_{1i}) \times (k/k_{max})) + c_{1i}$;

$c_2 = ((c_{2f} - c_{2i}) \times (k/k_{max})) + c_{2i}$;

$v_{max}(j) = \left(\max\left(X(:, j)\right)\right) \times penalty\ factor$

```
if particle_velocity(i,j)==0
   if rand<0.5
      V(i,j)=rand × v_max(j)
   else
      V(i,j)=-rand × v_max(j)
   end
end
X(i,j) = X(i,j) +V(i,j)
```

(17)

Where, i and j is size of population and dimension of problem, respectively. Penalty factor for this study is 0.1. The procedure of SPSO-TVAC for CM problem is described in Fig. 1.

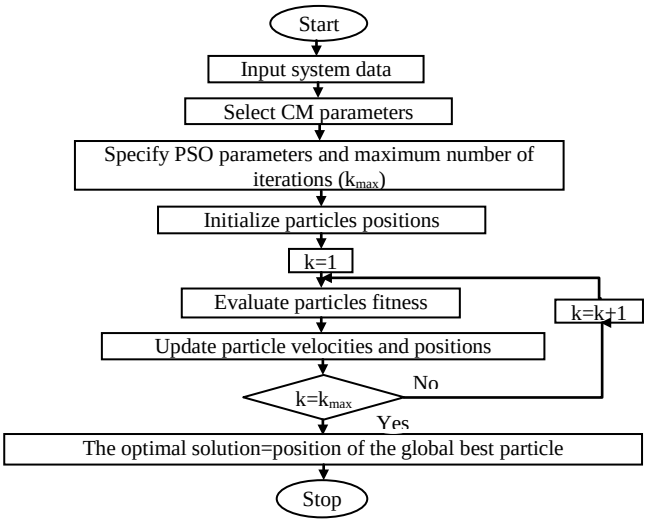


Figure 1. Flowchart of the SPSO-TVAC

IV. SIMULATION RESULTS

The different methods discussed earlier are applied to two cases to find out the minimum cost for any demand. One is IEEE 30-bus and other is 118-bus systems [16]. Results of proposed algorithm are compared with Particle Swarm Optimization (CPSO [17], PSO-TVAC [17] and PSO-TVIW [17]).

4.1. IEEE 30-bus system

In the first case study, the IEEE 30-bus system with six generators and forty one lines is used. The system configuration of the proposed case study is shown in Fig. 2 and the system data can be found in [16]. In this case, Bus 1 is considered as the reference bus or slack. A congested line between buses 1 and 2 exists as shown in Table 1. The maximum number of iterations and honey bee are set as 200 and 30, respectively.

TABLE 1. A CONGESTED LINE ON THE IEEE 30-BUS SYSTEM

Congested line	*Active power flow (MW)*	*Line limit (MVA)*	*Overload (MW)*
1 to 2	170	130	40

Table 2 shows the GS values of 6 generation units. Considering GS values, all generators are selected for re-dispatched.

TABLE 2. GENERATION SENSITIVITY OF 6 UNITS ON THE IEEE 30-BUS SYSTEM

Gen no.	*1*	*3*	*5*	*8*	*11*	*13*
GS 1_2	0	-0.8908	-0.8527	-0.7394	-0.7258	-0.6869

The GS values of all six generators in the IEEE 30-bus system are high therefore it is needed to use all generators for redispatch to relieve the congested line. To achieve this goal, selected group of generators having the largest GS values may be used to save the computational effort. In Fig. 3, the average active power adjustment and GS values of each generator have been shown. With 50 trial simulation, statistical results with SPSO-TVAC approaches are compared in Table 3. SPSO-TVAC provides the minimum redispatch cost solution of $ 209.075, whereas PSO-TVAC $ 237.9/h, CPSO and PSO-TVIW provide $ 240.3/h and $ 239.2/h, respectively. In addition, the solutions of SPSO-TVAC optimization have the lowest standard deviation 1.1; whereas this value in PSO-TVAC is 1.6, in CPSO and PSO-TVIW provide 48.2 and 3.8, respectively.

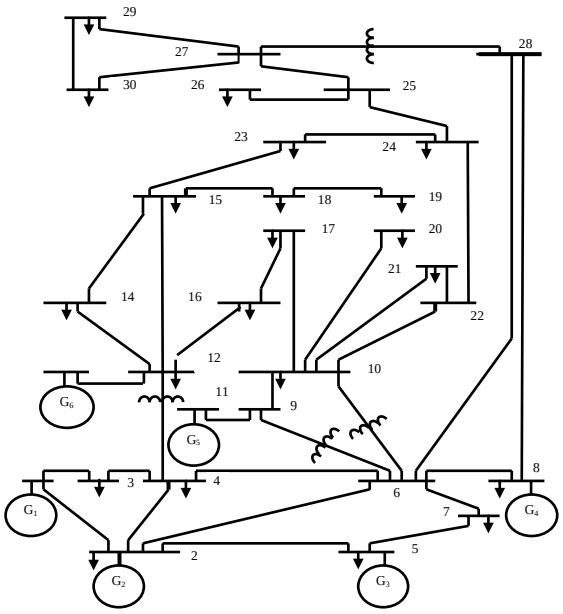


Fig. 2. The IEEE 30-bus system configuration.

4.2. IEEE 118-bus system

The system configuration of the IEEE 118-bus system with 54 generators and 186 lines [16] is used as second case study. Bus 1 is assigned as the reference bus. The congested line data is shown in Table 4.

The GS values are shown in the Fig. 4, the results of GS values for all generator buses are presented in Table 5. This results shows that the generator buses 85, 87, 89, 90, and 91 are among the largest magnitude of GS. This implies that these generators could significantly affect to the congested line. Thus, they are chosen as redispatched generators. Using the largest GS values, only 6 generators out of 54 are used for redispatching by proposed algorithm, requiring a much less computational effort.

With 50-trial simulation, the solutions from supposed algorithm and different PSO approaches are shown in Table 6. From the results, supposed algorithm provides the lowest redispatch cost of $ 821.1h, while PSO-TVAC is 829.5/h and CPSO and PSO-TVIW provide the minimum $ 875.0/h and $ 853.8/h, respectively. Mean and standard deviation values of supposed algorithm is 83.1, however PSO-TVAC is 94, CPSO and PSO-TVIW provide 196.4 and 165.8, respectively.

TABLE 4. A congested line on the IEEE 118-bus system.

Congested line	Active power flow (MW)	Line limit (MVA)	Overload (MW)
89 to 90	260	200	60

TABLE 3. COMPARISON OF SPSO-TVAC SOLUTIONS ON THE IEEE 30-BUS SYSTEM.

MW	ΔP_1	ΔP_2	ΔP_5	ΔP_8	ΔP_{11}	ΔP_{13}	*Total* ΔP	*Cost ($ /h)*
CPSO[17]								
Max	-66.1	28.9	23.3	18.1	6.2	3.7	146.3	403.1
Min	-47.9	18.6	16.5	11.3	2.8	0.1	97.2	240.3
Mean	-55.9	22.6	16.2	10.5	5.6	2.6	113.2	287.1
SD	8.3	7.6	3.5	3.3	3.2	3.3	15.9	48.2
PSO-TVIW[17]								
Max	-58.5	16.7	13.0	11.8	8.6	5.7	114.2	288.0
Min	-47.3	20.1	14.5	10.5	4.8	0.5	97.7	239.2
Mean	-50.1	18.9	13.2	9.2	5.9	4.1	101.4	253.1
SD	2.8	3.5	5.4	3.3	3.5	6.1	13.3	3.8
PSO-TVAC[17]								
Max	-51.1	22.0	14.7	8.8	6.2	1.0	103.8	254.9
Min	-47.3	25.1	16.0	7.6	0.6	0.0	96.7	237.9
Mean	-49.3	17.5	14.0	9.9	6.8	3.0	100.5	247.5
SD	0.8	2.1	2.1	2.2	2.3	2.4	4.6	1.6
SPSO-TVAC								
Max	-49.6	19.2	10.2	9.3	7.8	0.2	96.3000	230.8173
Min	-44.5	18.3	11.10	7.50	7.30	0.100	88.8000	209.0750
Mean	-40.2	19.2	12.30	8.20	8.10	0.20	88.2000	211.0210
SD	0.5	1.3	1.6	1.6	1.6	1.7	2.3	1.1

TABLE 5. GS VALUES OF 54 GENERATORS ON THE IEEE 118-BUS SYSTEM

Gen no.	GS (10^-3)	Gen no.	GS (10^-3)	Gen no.	GS (10^-3)
1	0	42	-0.0375	80	-0.9250
4	-0.0005	46	-0.0242	85	50.068
6	-0.0001	49	-0.0460	87	50.654
8	-0.0014	54	-0.0838	89	74.455
10	-0.0014	55	-0.0871	90	-701.15
12	0.0004	56	-0.0854	91	-427.90
15	0.0021	59	-0.1100	92	-28.411
18	0.0051	61	-0.1160	99	-9.391
19	0.0046	62	-0.1130	100	-12.915
24	0.1350	65	-0.1350	103	-12.737
25	0.0484	66	-0.0983	104	-12.854
26	0.0337	69	0.2120	105	-12.772
27	0.0451	70	0.3690	107	-12.202
31	0.0339	72	0.2326	110	-12.274
32	0.0477	73	0.3400	111	-12.07
34	-0.0323	74	0.5410	112	-11.747
36	-0.0329	76	0.8650	113	0.0110
40	-0.0343	77	0.0012	116	-0.1750

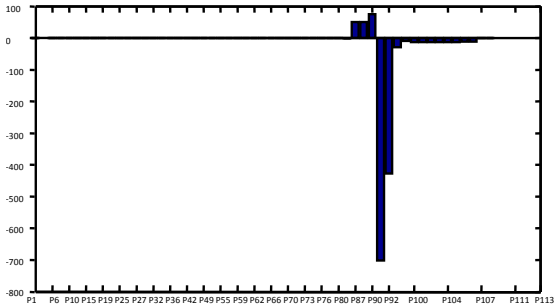


Figure 4. GS values of 54 units on the IEEE 118-bus system.

TABLE 6. COMPARISON OF SPSO-TVAC SOLUTIONS ON THE IEEE 118-BUS SYSTEM

MW	ΔP_1	ΔP_{85}	ΔP_{87}	ΔP_{89}	ΔP_{90}	ΔP_{91}	Total ΔP	Cost ($ /h)
CPSO[17]								
Max	-5.1	-6.4	-8.6	-122.9	117.8	18.9	279.8	1604.5
Min	-5.1	-27.3	-27.5	-28.9	68.1	25.9	182.7	875.0
Mean	-5.9	-15.3	-31.5	-62.0	85.1	26.8	226.6	1183.8
SD	4.4	8.4	11.4	17.5	23.2	14.6	30.5	196.4
PSO-TVIW[17]								
Max	-2.7	-13.8	-23.4	-97.7	121.4	10.4	269.4	1497.8
Min	-6.8	-18.2	-28.2	-33.1	78.3	8.9	173.5	853.8
Mean	-5.5	-12.1	-28.2	-59.8	76.4	29.8	211.7	1088.4
SD	4.3	6.7	10.7	16.9	21.1	13.5	26.3	165.8
PSO-TVAC[17]								
Max	-5.9	-6.2	-6.5	-96.2	80.1	30.5	225.5	1229.6
Min	-0.8	-12.1	-13.9	-52.3	81.6	3.3	163.8	829.5
Mean	-4.4	-10.3	-22.0	-58.5	69.4	24.7	189.3	970.7
SD	2.9	5.0	10.0	15.1	9.8	16.1	16.5	94.5
SPSO-TVAC								
Max	-48.1	-2.8	-38.65	-2.13	79.65	12.3	183.6	906.1
Min	-2.04	-14.1	-11.53	-52.0	75.52	5.42	160.9	821.1
Mean	-3.50	-15.3	-15.4	-50.1	79.60	5.876	169.9	861.40
SD	2.7	4.7	9.8	14.2	8.7	15.2	15.7	83.1

The relationship between GS values and power re-dispatch are shown in the Fig. 5. As the GS at bus 85, 87, and 89 are positive, the generation output at these buses are reduced. By contrast, the generators at bus 90 and 91 have negative GS values, thus the generation is increased. Moreover, the GS magnitude affects the amount of active power adjustment. The reference bus is used to maintain the power balance.

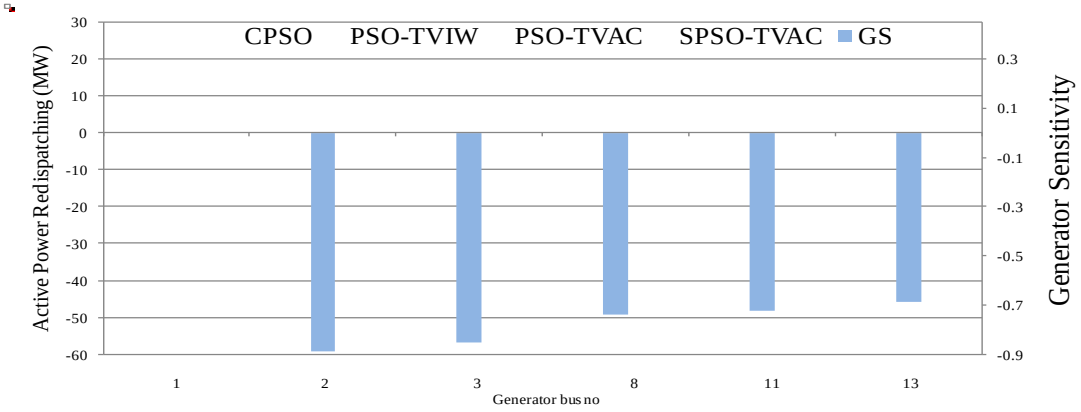


Figure 3. GS values and generation redispatch on the IEEE 30-bus system.

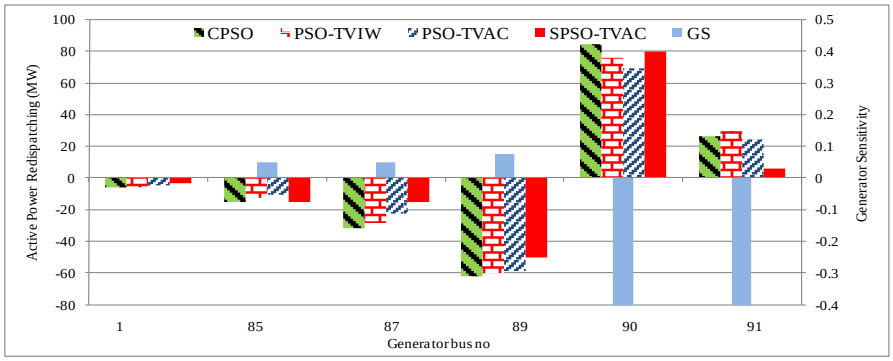


Figure 5. GS values and power redispatch on the IEEE 118-bus system.

4.3. Practical Indian 75-bus System

The Practical Indian 75-bus system consists of fifteen generator buses and 60 load buses. Bus 12 has been assigned as the Slack Bus. The single line diagram of the Practical Indian 75-Bus system is shown in Fig from ref [18]. The characteristics of the generating units and their constraints are available in [18,19]. In this test system, congestion occurred in Line 71 connecting Bus 26 and Bus 41. When the base case power flows in various branches were computed, Line 71 was already overloaded. The unconstrained scheduled power flow of 401.65 MW is recorded in Line 71, whose power flow limit is 400.00 MW. Hence, congestion has to be relieved by rescheduling active power generation of the participating generators. Accordingly, GSs are computed for the congested Line 71 using Equation (5). This is plotted in Fig. 6. Generators that are participating in CM must be selected depending on their sensitivities to the congested line. In this test system, only 10 generators have shown strong influence on the congested line; thus, these have been selected for congestion management. The evolutionary algorithms are employed to optimally reschedule the active power of the selected generators for relieving congestion in Line 71.

Table 7 gives the active power generation of the 10 participating generators before and after CM employing SPSO-TVAC, PSO, FPSO, and FDR-PSO.

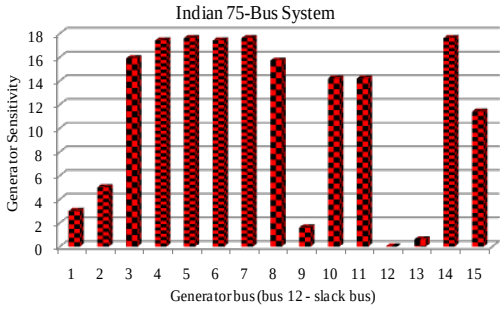


Figure 6. Generator Sensitivity Factors of Line 71 for the Practical Indian 75-Bus System.

The evolutionary algorithms have been implemented ten times on the Practical Indian 75-bus system in order to determine the robustness and effectiveness of the proposed methods. Table 8 shows the best, worst, and mean values after CM for optimal rescheduling of the active powers of the participating generators. It is observed from Table 8 that the SPSO-TVAC algorithm gives minimum cost for rescheduling the active power of participating generators in relieving congestion. It is further observed that the losses incurred for relieving congestion are also comparatively low in the case of SPSO-TVAC. However, the time taken for the best run among the 10 runs of evolutionary algorithms for congestion management is less in the case of FDR-PSO and FPSO [18].

TABLE 7. ACTIVE POWER GENERATION BEFORE AND AFTER CM FOR THE PRACTICAL INDIAN 75-BUS SYSTEM

Gen No.	Active Power Generation (pu) before Congestion Management	Active Power Generation (pu) after Congestion Management			
		PSO[18]	FPSO [18]	FDR-PSO [18]	SPSO-TVAC
3	1.9248	1.7731	1.8000	1.9310	1.9982
4	1.1653	0.9649	1.0000	0.9143	0.9829
5	1.7572	1.9576	1.8000	1.9776	1.8277
6	0.9680	1.0534	1.2000	1.0819	1.2122
7	0.7005	0.5693	0.6000	0.9220	0.6144
8	0.7469	0.9666	0.8000	0.8994	0.7977
10	1.0237	0.7376	0.8000	0.6925	0.8000
11	1.2258	1.2485	1.0900	1.0858	1.1203
14	1.3312	1.3950	1.5000	1.2989	1.6255
15	4.4229	4.5217	4.5400	4.4250	4.5302

TABLE 8. COMPARISON OF CM METHODS FOR THE PRACTICAL INDIAN 75-BUS SYSTEM

	PSO [18]	FPSO [18]	FDR-PSO [18]	SPSO-TVAC
Best(Rs/MWh)	5189.47	5075.44	5189.1	4987.98
Worst(Rs/MWh)	5243.81	5133.08	5213.77	5126.91
Mean(Rs/MWh)	5203.92	5098.34	5198.36	5076.41
Time(Seconds)	2.1207	2.4600	1.9573	2.0500
Losses(MW)	207.8246	205.1068	206.6673	204.1333
Slack Bus Power(MW)	1793.975	1788.776	1792.843	1783.415

V. CONCLUSIONS

In this paper we have used the method of re-dispatching generation otherwise referred to as counter trade as practiced in Indian. In this paper a new algorithm is used in optimization problem of congestion management in electricity market regarding to distribution network constraints to reduce cost and increase efficiency and security of power system distribution network. Purposed algorithm had an appropriate convergence rate compared with other techniques. SPSO-TVAC algorithm convergence rate is really less than in comparison other methods in solving complex mathematical problems. The numerical results demonstrate that the proposed method has better ability in finding optimal answers and possibility of particle placed in local zone.

REFERENCES

[1] Kumar. A. Srivastava, S.C., Singh S.N., "Congestion management in competitive power market: a bibliographical survey", Electric Power Systems Research, Vol. 76, pp. 153_164, 2005.

[2] Yu, C.N., M. Ilic, M., "Congestion clusters-based markets for transmission management", Proceeding of IEEE PES Winter Meeting, Vol. 2, pp. 821_832, 1999.

[3] Kumar, A., Srivastava, S.C., Singh, S.N., "A zonal congestion management approach using ac transmission congestion distribution factors", IEEE Transactions on Power Systems, Vol.72, pp. 85_93, 2004.

[4] Meena, T., Selvi, K., "Cluster based congestion management in deregulated electricity market using PSO", Proceeding of IEEE Indicon Conference, December Vol.11, No.13, pp. 627_630, 2005.

[5] Fu, J., Lamont, J.W., "A combined framework for service identification and congestion management", IEEE Transactions on Power Systems Vol.16, pp. 56_61, 2001.

[6] Yamina, H.Y., Shahidehpour, S.M., "Congestion management coordination in the deregulated power market", Electric Power Systems Research Vol. 65, pp. 119_127, 2003.

[7] Capitanescu, F., Cutsem, T.V., "A unified management of congestions due to voltage instability and thermal overloads", Electric Power Systems Research, Vol. 77, pp. 1274_1283, 2007.

[8] Yesuratnam, G., Thukaram, D., "Congestion management in open access based on relative electrical distances using voltage stability criteria", Electric Power Systems Research, Vol.77, pp. 1608_1618, 2007.

[9] Dutta, S., Singh, S.P., "Optimal rescheduling of generators for congestion management based on particle swarm optimization", IEEE Transactions on Power Systems, Vol. 23, pp. 1560_1569, 2008.

[10] H. Shayeghi, H. A. Shayanfar, A. Safari, R. Aghmasheh, "A robust PSSs design using PSO in a multi-machine environment", Energy Conversion and Management, Vol. 51, pp. 1656-1670, 2010.

[11] AT. Al-Awami, YL. Abdel-Magid, MA. Abido "A particle-swarm based approach of power system stabilizer enhancement with unified power flow controller", Elect Power Energy Syst. Vol. 29, pp. 251-250, 2007.

[12] A. M. El-Zonkoly, A. A. Khalil and N. M. Ahmied, "Optimal tuning of lead-lag and fuzzy logic power system stabilizers using particle swarm optimizatio"", Expert System with Applications, Vol. 10, pp. 1-10, 2008.

[13] Y. Shi, R.C. Eberhart, "Empirical study of particle swarm optimization", In Proceeding of the Congress on Evolutionary Computation, Vol. 3, pp. 1945_1950, 1999.

[14] A. Ratnaweera, S. K. Halgamuge, H. C. Watson, "Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients", IEEE Transactions on Evolutionary Computation, Vol. 8, No. 3, June 2004.

[15] A. Ratnaweera, S.K. Halgamuge, H.C. Watson, "Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients", IEEE Transactions on Evolutionary Computation. Vol. 8, pp. 240_255, 2004.

[16] Alsac, O., Stott, B. ", Optimal load flow with steady-state security", IEEE Transactions on Power Apparatus and Systems, Vol. 93, pp. 745-751, 1974.

[17] Panida, B., Chanwit, B., Weerakorn, O., "Optimal congestion management in an electricity market using particle swarm optimization with time-varying acceleration coefficients", Computers and Mathematics with Applications, Vol. 3, pp. 1-10, 2010.

[18] Ch Venkaiah and D M Vinod Kumar," Fuzzy PSO Congestion Management using Sensitivity-Based Optimal Active Power Rescheduling of Generators", Journal of Electrical Engineering & Technology Vol. 6, No. 1, pp. 32-41, 2011.

[19] I Jacob Raglend and Narayana Prasad Padhy, "Solutions to Practical Unit Commitment Problems with Operational Power Flow and Environmental Constraints", in Proceedings of IEEE, pp. 1-6, 2006.

Heidarali Shayanfar received the B.S. and M.S.E. degrees in Electrical Engineering in 1973 and 1979, respectively. He received his Ph. D. degree in Electrical Engineering from Michigan State University, U.S.A., in 1981. Currently, he is a Full Professor in Electrical Engineering Department of Iran University of Science and Technology, Tehran, Iran. His research interests are in the Application of Artificial Intelligence to Power System Control Design, Dynamic Load Modeling, Power System Observability Studies, Voltage Collapse, Congestion Management in a Restructured Power System, Reliability Improvement in Distribution Systems and Reactive Pricing in Deregulated Power Systems. He has published more than 490 technical papers in the International Journals and Conferences proceedings. He is a member of Iranian Association of Electrical and Electronic Engineers and IEEE.

Hossein Shayeghi received the B.S. and M.S.E. degrees in Electrical and Control Engineering in 1996 and 1998, respectively. He received his Ph.D. degree in Electrical Engineering from Iran University of Science and Technology, Tehran, Iran in 2006. Currently, he is an Associate Professor in Technical Engineering Department of University of Mohaghegh Ardabili, Ardabil, Iran. His research interests are in the application of robust control, artificial intelligence and heuristic optimization methods to power system control design, operation and planning and power system restructuring. He has authored and co-authored of 6 books in Electrical Engineering area all in Farsi, two book chapters in international publishers and more than 260 papers in international journals and conference proceedings. Also, he collaborates with several international journals as reviewer boards and works as editorial committee of eight international journals. He has served on several other committees and panels in governmental, industrial, and technical conferences. He was selected as distinguished researcher of the University of Mohaghegh Ardabili several times. In 2007, 2010, 2011 and 2013 he was also elected as distinguished researcher in engineering field in Ardabil province of Iran. Also, he is a member of Iranian Association of Electrical and Electronic Engineers (IAEEE) and IEEE.

Ali Ghasemi received the B.Sc. and M.Sc. (Honors with first class) degree in electrical engineering from Isfahan University of Technology (IUT), Isfahan, and University of Mohaghegh Ardabili (UMA), Ardabil, Iran, in 2009 and 2011, respectively. Currently he is pursuing the Ph.D. degree in the electrical engineering and computer science of UMA. His research interests are application of forecast methods, operation adaptive and robust control of power systems, Planning, Power System Restructuring and applications of heuristic techniques. He has authored of a book in Electrical Engineering area in Farsi, and more than 100 papers in reputable international journals and conference proceedings. Also, he collaborates as editorial committee and reviewer of 12 international journals.

He is a member of the Iranian Association of Electrical and Electronic Engineers (IAEEE). In 2012 and 2013, He received the award of the *4th Electric Power Generation Conference* and *UMA* for his M.Sc. thesis. He is the recipient honor M.Sc and Ph.D student Award of UMA, 2014. He received the 2013 best young researcher award of the *Young Researcher and Elite Club*.

Optimal Allocation and Sizing of Capacitors by BPSO Algorithm with Consideration of Annual Load Profile: A Real Case Study

H. Shayeghi
Department of Technical Engineering
University of Mohaghegh Ardabili Ardabi, Iran

H. A. Shayanfar*
College of Electrical Engineering
Center of Excellence for Power System Automation and Operation, Iran University of Science and Technology, Tehran, Iran

A. Ghasemi
Department of Technical Engineering
University of Mohaghegh Ardabili Ardabi, Iran

hshayeghi@gmail.com, hashayanfar@gmail.com, ghasemi.agm@gmail.com

Abstract: *In the present work, a swarm search optimization-based approach as known Binary Particle Swarm Optimization (BPSO) has been developed to find the optimal capacitor placement on the Meshkinshahr distribution network. Harmonic distortion of voltage sources in network caused a double harmonic current injection, increasing network losses and incidence of resonance phenomenon. Moreover, the optimal capacitor placement problem of radial distribution system for variable load level is considered as a non-linear optimization problem with a non-differentiable objective function due to the fact that the costs of the capacitor vary in a discrete manner and the system load also varies continuously throughout the day. Therefore, BPSO is proposed to minimize harmonic distortion effects in the distribution network based on optimal capacitor placement. To check the feasibility, the proposed framework is applied on real distribution network. Numerical experiments are included to demonstrate that the proposed method can obtain better quality solution than many existing techniques as view of minimizing the power loss and improving of voltage profile.*

Keywords: Optimal capacitor placement, Voltage source distortion harmonic, BPSO and Annual load profile.

I. INTRODUCTION

As the electrical loads are growing, the distribution system is developing to supply the customers with higher levels of demand. Growing the loads increases the lines current leading to the increase of loss. This also decreases the voltage level in the distribution network. Capacitors are used commonly to solve these difficulties. However, the investment cost is an issue which prevents their wide use. This highlights the importance of optimal allocation and sizing of the capacitors (OASC). Given the discrete nature of the allocation and sizing problem, local minima are the main difficulty in the OASC problem. There are several publications on solving the OASC problem. New strategies to reduce losses, improve voltage profile and liberate the feeder capacity can be used on radial distribution networks [1].

Whatever we wish to take up, the security of system should have more investment, which means higher costs but lower profits. Therefore, in practice there are two factors, namely the system security and economic issues to find an optimal operating point. In the study of capacitor placement to be investigated in such subject, the study different aspects of the problem will be discussed [2].

Nowadays, using artificial intelligence techniques and hybrid method has increased in solving optimal capacitor placement problem, such as it can be applied to Genetic Algorithms (GA) [3,4], Ant Colony Optimization (ACO) [5], Immune Algorithm (IA) [6], Simulating Annealing (SA) based on Fuzzy Logic (FL) [7] and intelligence algorithms noted in parallel capacitor placement and filter [8] problem. Based on our review of the scientific researches, the effect of harmonic currents and especially harmonic distortion of voltage resources on network are not considered. Also, none of the above research considered discrete nature of placement problems and solved this problem in continuous form. For this reason none of the methods presented guarantees the accuracy of the final response. In this paper, the optimal capacitor placement problem is solved with BPSO. The advantage of this method in comparing to the other methods can be expressed as:

First, the discrete manner of this method can be captured the discrete nature of the placement problem and second, the size and cost of capacitors based on the IEEE standard are determined. Guarantied method convergence

* Correspanding Author. E-Mail Address: hashayanfar@yahoo.com (H. A. Shayanfar)

and accuracy of the ultimate answer considered the effect of harmonic distortion voltage sources on network.

The rest layout of this paper is organized as follows: section 2 gives the objective function of capacitor placement problem with equality and inequality constraints. In this way, the effect of voltage source harmonic distortion on network is considered. Section 3, the BPSO is discussed. Section 4, the BPSO algorithm is used in the capacitor placement problem on the Meshkinshahr distribution network and the results are discussed. Finally, the simulation results are presented in section 5.

III. PROBLEM FORMULATION

In this section, the objective function of capacitor placement problem with the equality and inequality constraints are proposed as follows. In this formulation, the effect of voltage sources harmonic distortion on the network is considered.

2.1. Network Model at Harmonic Frequencies

For modelling of a distribution system in Fig.1, fundamental and harmonic frequencies based on [9], the formulation, notations and assumption steps are as follows:

1) Calculate the magnitudes and the phase angles of the Bus voltages at fundamental frequency by using back-forward power flow method and calculate the power losses at the power frequency Eq. (1).

$$P^1_{lo\,\mathbf{s}(i,i+1)} = R_{i,i+1}[|V^1_{i+1} - V^1_i \,||\, y^1_{i,i+1}\,|]^2 \tag{1}$$

2) Calculate the h^{th} harmonic frequency load admittances, shunt capacitor admittances and feeder admittances Eqs. (2-4) respectively.

$$y^h_{li} = \frac{P_{li}}{|V^l_i|^2} - j\frac{Q_{li}}{h\,|V^l_i|^2} \tag{2}$$

$$y^h_{ci} = h y^l_{ci} \tag{3}$$

$$y^h_{i,i+1} = \frac{1}{R_{i,i+1} + jhX_{i,i+1}} \tag{4}$$

where, P_{li}, Q_{li} denotes the load active and reactive powers at bus *i*. Similarly, $R_{i,i+1}, X_{i,i+1}$ shows the resistance and inductive reactance of feeder section between buses *i* and *i+1*.

3) Calculate the harmonic voltages, the rms voltage and the total harmonic distortion from Eqs. (5) to (7), respectively.

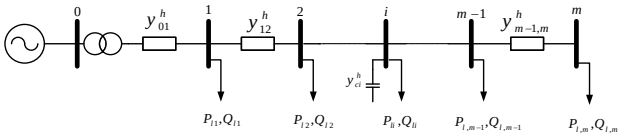


Figure 1. one-line diagram of radial distribution feeder

$$\begin{bmatrix} V^h_1 \\ V^h_2 \\ \cdot \\ \cdot \\ V^h_{m-1} \\ V^h_m \end{bmatrix} = \begin{bmatrix} Y^h_{11} & Y^h_{12} & 0 & & & 0 \\ Y^h_{21} & Y^h_{22} & \cdot & & & \\ 0 & \cdot & \cdot & & & \\ \cdot & & & \cdot & \cdot & 0 \\ \cdot & & & \cdot & Y^h_{m-1,m-1} & Y^h_{m-1,m} \\ 0 & & & 0 & Y^h_{m,m-1} & Y^h_{m,m} \end{bmatrix}^{-1} \begin{bmatrix} I^h_1 \\ I^h_1 \\ \cdot \\ \cdot \\ I^h_{m-1} \\ I^h_m \end{bmatrix} \tag{5}$$

$$I^l_i = \left[\frac{P_{hi} + jQ_{hi}}{V_{il}}\right]^* \tag{6}$$

$$I^h_i = C(h) I^l_i \tag{7}$$

where, *C(h)* is the the nonlinear portion of the load at bus *i*.

4) Calculate the rms voltage from Eq. (8) respectively.

$$|V_i| = \sqrt{\sum_{h\neq 1}^{H} |V_i^h|^2} \tag{8}$$

where, *H* is the upper limit of considered harmonic order.

2.2. Objective Function Proposed

For a real radial distribution system, the objective function used in optimal capacitor placement, including cost reduction of capacitor placement and minimizing cost of system losses with consideration of problem constraints such as maximum amount of voltage and Total Harmonic Distortion (THD) of network voltage. This objective function can be defined by:

$$F = Min\left[K_P P_{loss} + \sum_{j=1}^{nc} K^C_j Q^C_j\right] \tag{9}$$

Eq. (9) is composed of two parts. First part expressed cost of energy loss. Where, K_P is the cost of energy loss and P_{loss} can be defined as follows:

$$P_{loss} = \sum_{h=1}^{H}\left[\sum_{i=0}^{m-1} P^h_{loss(i,i+1)}\right] \tag{10}$$

Second part is the cost of capacitor placement including cost of buy and install capacitor and *nc* shows the number of used capacitor.

2.3. Problem Constraints

Capacitor placement problem with the equality and inequality constraints are expressed as follows

2.3.1 Load Flow Constraints

All equations of load flow at the main frequency of system Calculate from Eqn. (11)

$$\begin{aligned} P_i &= |V_i|^2 G_{ii} + \sum_{\substack{n=1 \\ n\neq i}}^{N} |V_i|\,|V_n|\,|Y_{in}|\cos(\theta_{in} + \delta_n - \delta_i) \\ Q_i &= -|V_i|^2 B_{ii} + \sum_{\substack{n=1 \\ n\neq i}}^{N} |V_i|\,|V_n|\,|Y_{in}|\sin(\theta_{in} + \delta_n - \delta_i) \end{aligned} \tag{11}$$

2.3.2 Rms Voltage at Each Bus Constraints

Total rms voltage (main and harmonic components) in node *j* at level *i* is given in Eq. (12):

$$|V_{i,j}| = \sqrt{|V_{i,j}^{1}|^2 + \sum_{h\neq 1}^{H} |V_{i,j}^{h}|^2} \quad (12)$$

One goal of capacitor placement in this paper voltage profile improvement, to achieve this goal should be bus voltage range placed at domain Eq. (13).

$$0.95\,pu \le |V_{i,j}| \le 1pu \quad (13)$$

2.3.3 Max THD Constraint

Total harmonic distortion of network voltage in node *j* at level *i* can be calculated by:

$$THD_{i,j}(\%) = \frac{100}{|V_{i,j}^{1}|}\sqrt{\sum_{h\neq 1}^{H} |V_{i,j}^{h}|^2} \quad (14)$$

$$THD_{i,j}(\%) \le THD_{\max} \quad (15)$$

IV. Binary Particle Swarm Optimization (BPSO)

The PSO is a population-based optimisation method first proposed by Kennedy and Eberhart. Some of the attractive features of the PSO include the ease of implementation and the fact that no gradient information is required. This algorithm is based on two essential basis: Collective intelligence (categories like birds, fish groups and group movements) and dynamic calculation.

PSO began his work with a random population of d dimensional space, Where i particles as shown be $X_i = (x_{i1}, x_{i2}, ..., x_{id})$. Each particle allocated coordinate in space, According to the fitness level may be a little bit far. Also, $P_i = (p_{i1}, p_{i2}, ..., p_{id})$ shows the amount of particle fitness, *pb* shows the best position of individual particles and represents the best particle of space around the best position will provide the lowest fitness entire community. In each step updated population particle velocity, where move from *pb* to *gb* based on:

$$v_{id}^{k+1} = \omega v_{id}^{k} + c_1 r_1 (pb_{id}^{k} - x_{id}^{k}) + c_2 r_2 (gb_{gd}^{k} - x_{id}^{k}) \quad (16)$$

In $v_i = (v_{i1}, v_{i2}, ..., v_{id})$, the subscript *i* is the particle velocity applying the random parameters represents the weight caused acceleration of particles can be moved *pb* toward *gb*; Thus, i^{th} particle position is updated by:

$$x_{id}^{k+1} = v_{id}^{k} + v_{id}^{k+1} \quad (17)$$

The c_1 and c_2 are fixed parameters to accelerate convergence of particles *pb* toward *gb*.(In this study $c_1 = c_2 = 2$ and $\omega = 0.8$). Also, r_1 (.), r_2 (.) are two random functions they are distributed with uniform probability in the interval [0 1]. The ω denotes the inertia weight parameter to adjust the velocity of the particles and also to make balanced local and global search; So for faster convergence of the algorithm this parameter must be carefully specified finally, if parameters set up was selected correctly then increase velocity of algorithm convergence. Fig 2 shows this process obviously. Therefore, Kennedy and Eberhart improved PSO algorithm for using in discrete problems. This algorithm named binary PSO (BPSO). New approach function according to Eq. (18) makes it possible to search in discrete space and final velocity of particles can be located in the interval [0 1]. Eventually ultimate particles location are updated by replacing, Eq. (19) instead of Eq. (17)

$$Sigmoid(v_{id}^{k}) = \frac{1}{1+e^{-v_{id}^{k}}} \quad (18)$$

$$x_{id}^{k} = \begin{cases} 1 \text{ if } r < Sigmoid(v_{id}^{k}) \\ 0 \quad \text{Otherwise} \end{cases} \quad (19)$$

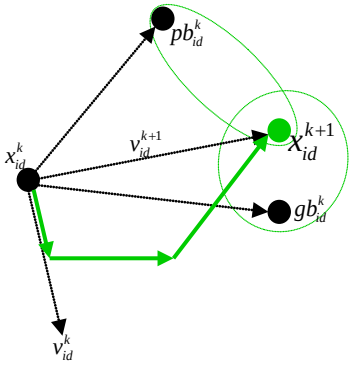


Figure 2: Velocity and location of particle updating process

V. Capacitor Placement Using BPSO

The system shown in Fig.1 is *m*-bus radial distribution system. Table 1 shows a sample of the yearly cost of fixed capacitor sizes. To select the capacitor size Q_j^c to be placed at bus *j*, a combination of capacitor sizes (R-size) is chosen from Table 1, as an example,

$$Q_j^c = (b_1 sz_1 + b_2 sz_2 + ... + ... + b_r sz_r + ... + b_R sz_R) \quad (20)$$

where, $j \in J$ is a selective bus to install capacitor, $b_r = \{0,1\}$, size of capacitor selected from Table 1 and $Q_j^c \le Q_{\max}^c$ and $Q_{\max}^c$ is the maximum size of capacitor takes place in the specified bus.

Fig.3 shows flowchart of problem solving using BPSO algorithms.

Table 1. Network capacitor placement cost in discrete form

	1	2	3	4	5	6
Capacitor size (kVAr)	150	300	450	600	750	900
Capacitor cost ($/kVAr)	0.5	0.35	0.253	0.22	0.276	0.183
	7	8	9	10	11	12
Capacitor size (kVAr)	1050	1200	1350	1500	1650	1800
Capacitor cost ($/kVAr)	0.228	0.17	0.207	0.201	0.193	0.187
	13	14	15	16	17	18
Capacitor size (kVAr)	1950	2100	2250	2400	2550	2700
Capacitor cost ($/kVAr)	0.211	0.176	0.197	0.17	0.189	0.187
	19	20	21	22	23	24
Capacitor size (kVAr)	2850	3000	3150	3300	3450	3600
Capacitor cost ($/kVAr)	0.183	0.18	0.195	0.174	0.188	0.17

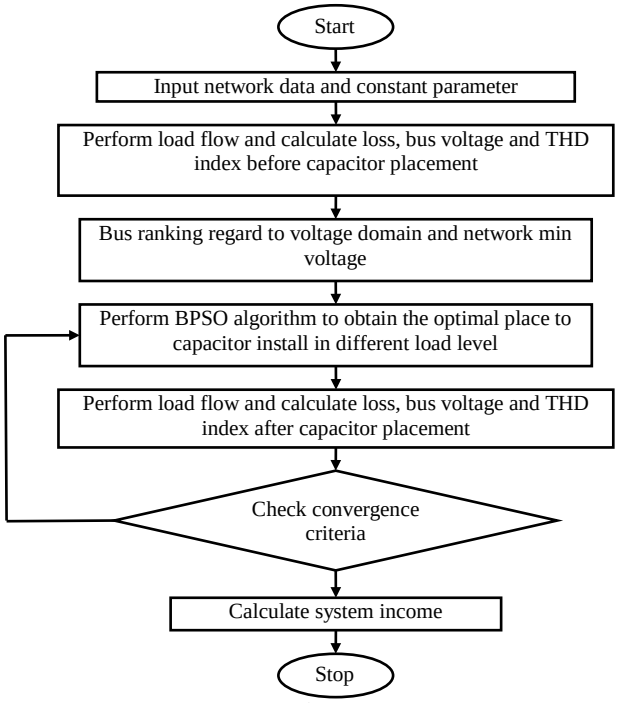


Figure 3. Solving problem flowchart using BPSO algorithm

VI. SIMULATION RESULTS

In order to show the performance of the proposed method, simulation is executed on the Meshkinshahr radial distribution network shown in Fig.4.

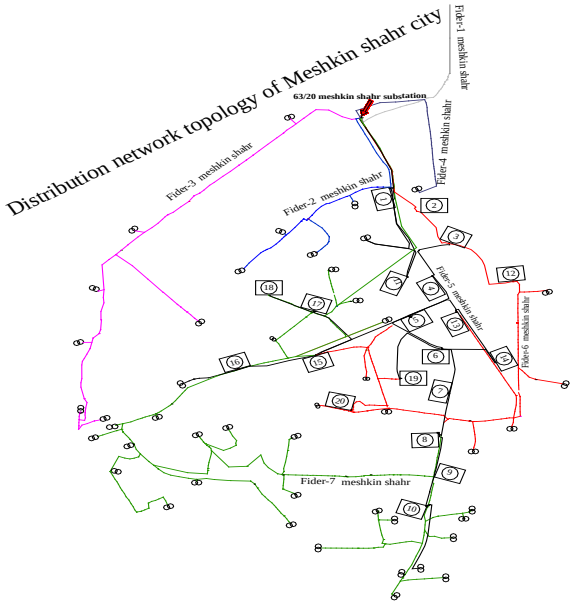


Figure 4. Meshkinshahr radial distribution network topology

Fig.5 shows network load in March 2010. If we focus on this figure it can be found that the 30th March is the most loads in month. Thus capacitor placement will be done on Meshkinshahr network in 30 March 2010.

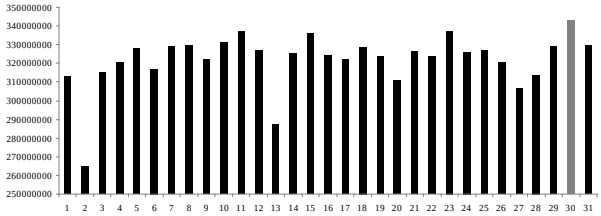


Figure 5. Network load in March 2010

Fig.6 shows network configuration in line diagram plan. According to Figs. 4 to 6 Meshkinshahr distribution network has 1 substation and 20 bus stations.

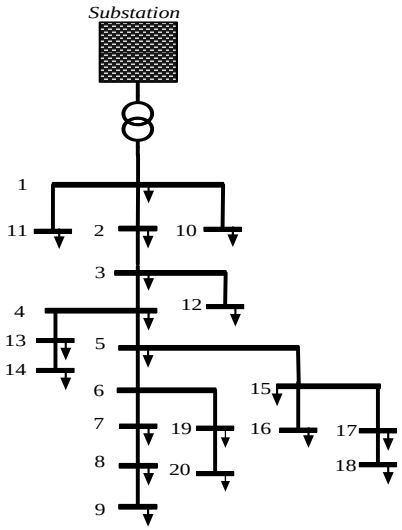


Figure 6. Line diagram of Meshkinshahr distribution network

Fig.7 shows network load in active and reactive power.

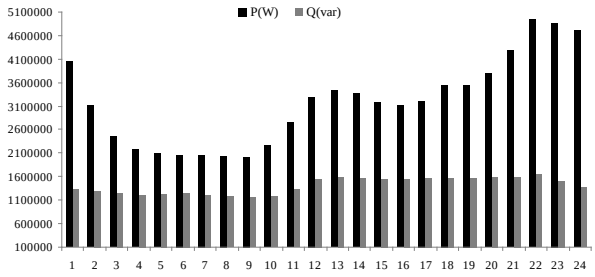


Figure 7. Network load in 30th March 2010

Figure 8 shows load profile of network in 30 March 2010. It is obvious that minimum load of the Meshkinshahr network at 9 hr (low), nominal load at 16 hr (nominal) and maximum load of network at 22 hr (peak). So in continue, we solved the optimal capacitor

placement on the Meshkinshahr network in three load level.

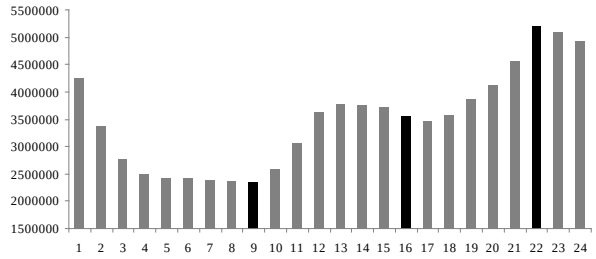


Figure 8. Load profile of network in 30 March 2010

Parameters used in the problem to minimizing network loss and calculating objective function based on reference [12] to be under possession. Fixed parameters of the objective function are:

$K_P=168$ \$/KW, K_j^C obtain from Table 1 and Q_j^C is calculated based on Eq. (20).

Problem constraints are:

$V_{min} = 0.95pu$, $V_{max} = 1.1pu$ and $THD_{max} = \%5$

Finally, calculate harmonic distortion of the voltage source from Eq. (21).

$$V_1 = \alpha V_0 \quad (21)$$

Size of α for different level of harmonic obtain from Table. 2.

Table 2. α size for different level of harmonic

Harmonic level	1	3	5
α size	1	0.04	0.03

5.1. Network Condition before Capacitor Placement

The simulation results are tabulated in Table. 3, it shows system offend from THD and voltage constrains in all of 3 load level. In next section, the power loss reduction, voltage profile improvement and network cost reduction by optimal capacitor placement on the network is given.

Table 3. Network characterize before capacitor placement

Load level / Characterize	Low	Nominal	Peak
Loss(KW)	2970	7351	19195
Loss cost($)	475200	1176160	3071200
$Voltage_{min}$(pu)	0.9593	0.9375	0.9022
THD_{max} Percent (%)	10.47	10.73	11.3

5.2. Capacitor Placement Using BPSO Algorithm

In this section, the optimal capacitor placement on the network based on load conditions at 30 March 2010 is performed. The numerical results are given in Table 4, it is clearly concluded the better performance of system after capacitor placement at last network loss measure and will be reduce to 1185 kW. It should be mentioned that capacitors installed on the network in all three modes in order to accelerate the convergence of algorithms are chosen based on the values in Table 1.

Table 4. Network characterize after capacitor placement

Load level / characterize	Low	Nominal	Peak
Loss(KW)	2853	7047	18431
Loss cost($)	456480	1127520	2948960
$Voltage_{min}$(pu)	0.9797	0.9688	0.9511
Capacitor Cost($)	2070	2277	2458
THD_{max} Percent (%)	4.36	4.9915	4.8842
Economic Providence($)	16650	46363	119782

5.3. Economic Providence Due to Network Loss Reduction

According to the results of Tables 3 and 4 can be seen that the total loss costs at all levels of network load is 4722560\$ this amount after capacitor placement reduce to 4% it is 4532960\$ including 33898\$ for the capacitor installation on network; system income due to loss reduction per year will be 182795\$.

Table 5 shows amount and location of installed capacitor used on the network to reduce losses and improve voltage profile.

Table.5. Amount and location of installed capacitor

Bus Number / Capacitor size (kVAr)		
Low	**Nominal**	**Peak**
13/1320	9/1138	7/1320
18/750	16/1138	18/1138
2070	**2276**	**2458**
Total Capacitor		

Figure 9 display effect of the BPSO algorithm on the convergence of defined objective function.

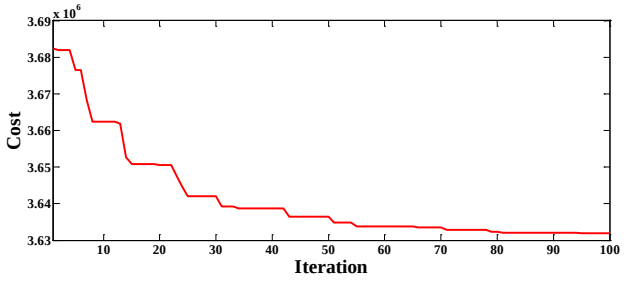


Figure 9. Convergence of defined objective function

Fig.10 shows network voltage profile after optimal capacitor placement using BPSO algorithm. It is clear that after applying the voltage constraint on network, all bus voltages placed above 0.95 pu.

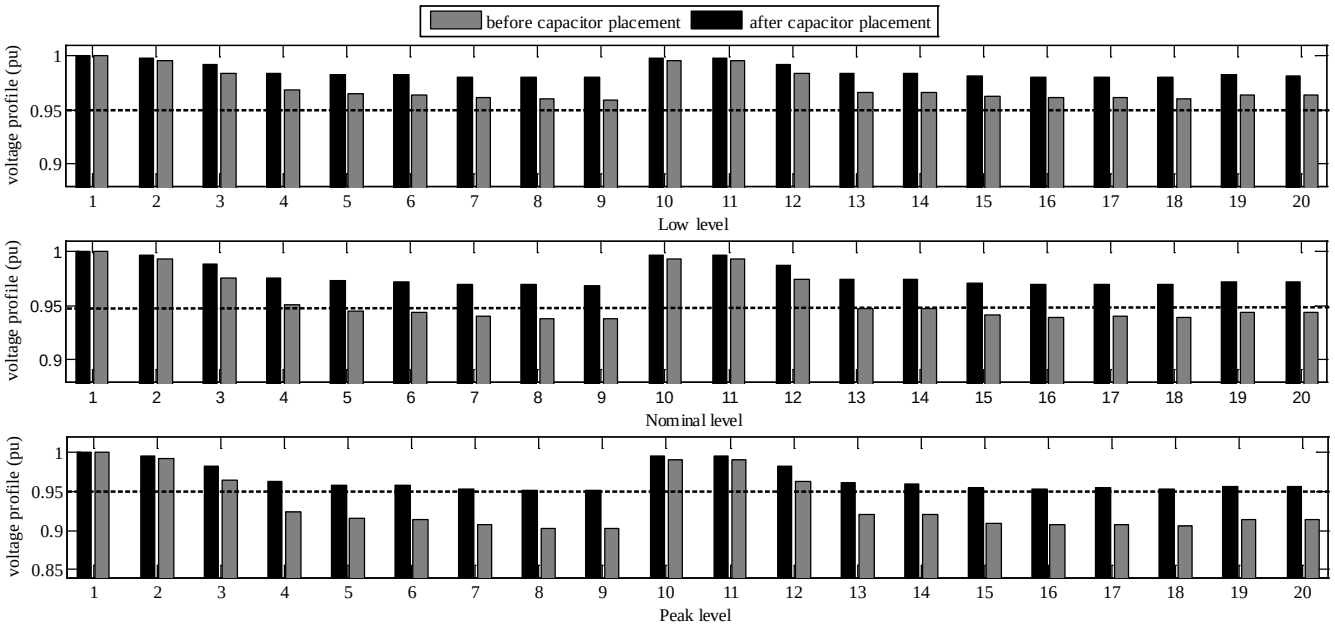


Figure 10. Network voltage profile after capacitor placement in 3 load level

VII. CONCLUSION

In this paper, a modified PSO based on binary theory namely BPSO is comprehensively enhanced and effectively applied for solving a optimal placement of capacitors problem to minimize power loss objective function while satisfying the operation constraints such as harmonic distortion of voltage sources on radial distribution network. In this problem our aim was to ultimate loss reduction and improvement of network voltage profile with consideration of increasing economic benefits resulting from loss reduction. To achieve this goal voltage domain, maximum voltage source harmonic distortion (max_{THD}) constraints and network annual load profile in determining the optimal capacitor size and location on the network was considered. THD index role is control, to reduce harmonic currents and to prevent the resonance phenomenon occurrence on the distribution networks. The proposed method is a powerful framework for solving the optimal capacitor placement problem because it benefits from the advantages of PSO and binary law. It has some notable advantages which have the following characteristics:

- By using binary theory to complete locally oriented search for the solutions resulted with original PSO.
- The BPSO develops the idea of binary between p_{best} and g_{best}. Thus, the global searching ability is redeemed about on average in the proposed method.

Consequently, the numerical result shows that the good performance of method accuracy at different level of network load (low load, nominal load and peak load) during a 30 March 2010.

REFERENCES

[1] H. N. Ng, M. M. A. Salama, and A.Y. Chikhani, "Classification of capacitor allocation techniques" IEEE Trans. Power Delivery,Vol.15, pp. 387-392, Jan.2000.

[2] M. F. AlHajri, M. R. AlRashidi, M. E. El-Hawary, "A novel discrete particle swarm optimization algorithm for optimal capacitor placement and sizing" in Proc. IEEE Conf. Vol. 5, pp. 1286-1289, apr 2007.

[3] M. Dlfanti, G. P. Granelli, P. Maranninio, "Optimal capacitor placement using deterministic and genetic algorithm" IEEE Trans. On Power Systems, PWRS-15; Vol. 15, No. 3, pp. 1041-1046, 2000.

[4] M. D. Reddy, V. C. Reddy, "Optimal capacitor placement using fuzzy and real coded genetic algorithm for maximum savings" Journal of Theoretical and Applied Information Technology, Vol. 4, No. 3, pp. 219-226, 2008.

[5] Ying Wang, Ruiye Liu, Shiwei Xia, Liying Liu, "Reactive power optimal planning of distribution network based on ASPO with harmonic distortion consideration" ICIEA 2009, pp. 3515-3521, 2009.

[6] T. Huang, Y. Hsiao, C, J. Jiang. "Optimal placement of capacitors in distribution systems using an immune multi-objective algorithm" International Journal of Electrical Power and Energy Systems, Vol. 30, No. 3, pp. 184–192, 2008.
[7] S. K. Hattacharya and S. K. Goswami, "A new fuzzy based solution of the capacitor placement problem in radial distribution system" Expert Systems with Applications, Vol. 36, No. 3, pp.4207-4212, 2009.
[8] C. T. Hsu, Y. H. Yan, C. S. Chen, and S. L. Her, "Optimal reactive power planning for distribution systems with nonlinear loads" in Proc. IEEE Region 10 Int. Conf. Computer, Communication, Control and Power Engineering, Vol. 5, pp. 330-33, 1993.
[9] C. S. Cheng and D. Shirmohammadi, "A three-phase power flow method for real-time distribution system analysis" IEEE Transactions on Power System, Vol. 10, No. 2, pp. 671– 679, May 1995.
[10] J. Kennedy and R. Eberhart, "Particle swarm optimization" in Proc. IEEE Int. Conf. Neural Networks, Vol. IV, Perth, Australia, 1995, pp.1942–1948.
[11] J. Kennedy and R. C. Eberhart, "A discrete binary version of the particle swarm algorithm" Proc. of the conference on Systems, Man, and Cybernetics SMC97, pp. 4104-4109, 1997.
[12] T. M. Khalil and H.K.M. Youssef, "A binary particle swarm optimization for optimal placement and sizing of capacitor banks in radial distribution feeders with distorted substation voltages" AIML 06 International Conference, 13 - 15, Sharm El Sheikh, Egypt, June 2006.

BIOGRAPHIES

Hossein Shayeghi received the B.S. and M.S.E. degrees in Electrical and Control Engineering in 1996 and 1998, respectively. He received his Ph.D. degree in Electrical Engineering from Iran University of Science and Technology, Tehran, Iran in 2006. Currently, he is an Associate Professor in Technical Engineering Department of University of Mohaghegh Ardabili, Ardabil, Iran. His research interests are in the application of robust control, artificial intelligence and heuristic optimization methods to power system control design, operation and planning and power system restructuring. He has authored and co-authored of 6 books in Electrical Engineering area all in Farsi, two book chapters in international publishers and more than 260 papers in international journals and conference proceedings. Also, he collaborates with several international journals as reviewer boards and works as editorial committee of eight international journals. He has served on several other committees and panels in governmental, industrial, and technical conferences. He was selected as distinguished researcher of the University of Mohaghegh Ardabili several times. In 2007, 2010, 2011 and 2013 he was also elected as distinguished researcher in engineering field in Ardabil province of Iran. Also, he is a member of Iranian Association of Electrical and Electronic Engineers (IAEEE) and IEEE.

Heidarali Shayanfar received the B.S. and M.S.E. degrees in Electrical Engineering in 1973 and 1979, respectively. He received his Ph. D. degree in Electrical Engineering from Michigan State University, U.S.A., in 1981. Currently, he is a Full Professor in Electrical Engineering Department of Iran University of Science and Technology, Tehran, Iran. His research interests are in the Application of Artificial Intelligence to Power System Control Design, Dynamic Load Modeling, Power System Observability Studies, Voltage Collapse, Congestion Management in a Restructured Power System, Reliability Improvement in Distribution Systems and Reactive Pricing in Deregulated Power Systems. He has published more than 490 technical papers in the International Journals and Conferences proceedings. He is a member of Iranian Association of Electrical and Electronic Engineers and IEEE.

Ali Ghasemi received the B.Sc. and M.Sc. (Honors with first class) degree in electrical engineering from Isfahan University of Technology (IUT), Isfahan, and University of Mohaghegh Ardabili (UMA), Ardabil, Iran, in 2009 and 2011, respectively. Currently he is pursuing the Ph.D. degree in the electrical engineering and computer science of UMA. His research interests are application of forecast methods, operation adaptive and robust control of power systems, Planning, Power System Restructuring and applications of heuristic techniques. He has authored of a book in Electrical Engineering area in Farsi, and more than 100 papers in reputable international journals and conference proceedings. Also, he collaborates as editorial committee and reviewer of 12 international journals.

He is a member of the Iranian Association of Electrical and Electronic Engineers (IAEEE). In 2012 and 2013, He received the award of the *4th Electric Power Generation Conference* and UMA for his M.Sc. thesis. He is the recipient honor M.Sc and Ph.D student Award of UMA, 2014. He received the 2013 best young researcher award of the *Young Researcher and Elite Club*.

SESSION

AGENT TECHNOLOGIES AND APPLICATIONS

Chair(s)

TBA

Manipulation of Nodes in a Connected Car Network

Henry Hexmoor[1], Guy Fraker[2]
[1]Computer Science Department
Southern Illinois University
Carbondale, IL 62901, USA
[2]AutonomousStuff.com

Abstract - *The future of electronic, connected, and networked automobiles is upon us. This new technology reshaping the way we drive and communicate on motorways. The technology cars maintain connections and communicate via Wi-Fi. The Connected cars initiative envisions cars that form a dynamic social network. Each car exchanges local information such as speed and travel direction with peers. Other than increased safety and traffic improvement, cars are anonymous in the network. In this network, it is possible to use neighborhood information to blend in and evade traffic from law enforcement authority. For example, if there are a hundred cars driving in unison, it is difficult to apprehend speeding vehicles. If the police vehicle is interested in a specific car, it needs to enter the network and positions itself behind the target car. This project explores techniques for infiltrating a connected car network to form ties with a specific node.*

Keywords: *connected cars, networks, law enforcement, multiagent simulation, netlogo*

1.0 Introduction

A connected cars network is a network of IP equipped automobiles. This enables a vehicle to share data with other devices both inside and outside the vehicle. Connected vehicles can vastly improve the transportation system in the areas of safety, mobility and environment by providing connectivity among the vehicles to enable crash prevention and infrastructure. Wireless devices will provide continual real-time connectivity to all system users. It also provides connectivity among the vehicles and the infrastructure to enable safety mobility and environmental benefits. Connected cars network relies on wireless ad-hoc networks and Wi-Fi direct technologies [2][3][4][6][7]. It is a decentralized type of wireless network where each node participates in routing by forwarding data for other nodes, and so the determination of which nodes forward data dynamically is made based on the network topology. Wi-Fi direct is independent of access point to communicate with a mobile agent [5]. It works by embedding a limited wireless access point into the devices [1]. Each vehicle is linked to a node in a wireless ad-hoc network. Whenever other nodes come in communication range of a node, it uses Wi-Fi direct technology to communicate with that node. It sends its vehicle ID number to the other node. It also receives the same information from other nodes. The goal of this project is to enable law a enforcement vehicle to scan for and position itself near vehicles that violate the local speed limit. The vehicle will then search the database for the ID number, which contains an entity is in the format of <id, location, speed, distance from a police vehicle, ID of the alleged suspect vehicle>. If the vehicle finds a speeding vehicle or intruder vehicle, it stores the offender vehicle ID and forwards this information to its neighbor node and to the neighbor police node. If a police node is not a neighbor of a node then it compares the shortest distance to police by comparing the hop-distance to police through other nodes. When a police node receives the tuple <ID, location, direction of a speeding/intruder node> through a network of nodes or a single node then it starts pursuing the intruder node. We used the concepts of Wireless ad-hoc network and Wi-Fi Direct technology. Ours is a decentralized Wireless network where each node participates in routing information by forwarding data to other nodes. Hence, the determination of which nodes will forward data uses the network structure. Wi-Fi direct does not depend on access points for communication with a mobile agent. It works by embedding a limited wireless access point into the devices. Each vehicle is linked to a node in a wireless ad-hoc network. Whenever other node comes in communication radius of a node, it uses Wi-Fi direct technology to communicate with that node. It sends its vehicle ID number to the other node. It also receives the same information from other node. The vehicle will then search the database for the ID number, which contains an entity is in the format of <car id, location, speed, its distance from a police vehicle and the ID of the alleged suspect vehicle>. If the vehicle finds a speeding vehicle or an intruder vehicle, it stores the offender vehicle ID and forwards this information to its neighbor node and the neighbor police node. If a police node is not a neighbor of a node then it compares the shortest distance to police by comparing the hop-distance to police through other nodes.

2.0 System Analysis

Every node broadcasts car id to its immediate neighbors, which are in its Wi-Fi range. A single node sends and receives a set of information from its neighbors. The information includes the ID number. Every agent has an ID number [1]. The number distinguishes it from other agent. Like a registration number of a vehicle, a node has a unique ID number. In Wi-Fi environment, this serves as a MAC address for a node in the network. Whenever a vehicle enters into the range of the network it automatically connects with the network as shown in Figure 1. In this design, the wireless network is always available.

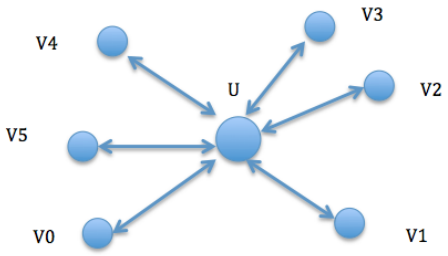


Fig 1. Illustrating how the connected cars system work by taking an example network

A mobile agent changes it place but ID number remains the same. Every single agent sends its ID number to its immediate neighbors and receives its neighbors ID number as well. If a node finds an intruder node or speeding node as its immediate neighbor then it stores the information of the intruder node as:
1. Speed: Current speed of the node.
2. Location: Current location of a node.
3. Target Node: If a node finds an intruder node or speeding node as its immediate neighbor then it stores the information of the intruder node as Target Node. And in later broadcast of the information along with other information it sends the Target Node information to its neighbors. If a neighbor node finds the information of the Target Node from its neighbor then it also updates its own information about Target Node with this and looks for the shortest distance to police to forward Target Node information.
4. Hop distance to police: In this proposed design, all nodes are mobile. Instead of maintaining static information about hop distance to police, the routing table always updates itself. Suppose a mobile node having a police as its immediate neighbor. Then its hop distance to police is 1. After a while the node changes its place where police is not 1. Other than itself an immediate neighbor of its own is connected with a police. Then the hop distance to police will be 2. So a mobile agent in this system always updates its hop distance to police.

2.1 Simulation Setup

- Every vehicle (i.e., agent) node in the network has its own database table in the MySQL database.
- Technically, a Netlogo agent cannot forward its information table to another agent [8].
- In a simulation of Netlogo [8], there are multiple agent nodes with multiple interactions between them which needs database update many times a second. So, connecting to MySQL is somehow technically convenient but database triggering, fetching, storing take extra overhead to maintain. As shown in the Figure 2, SQL database is connected to Netlogo by using extensions keyword [9].

NetLogo-sql (or NetLogo SQL Wrapper) is a JDBC based extension for the NetLogo modeling environment to access databases using SQL queries. It allows NetLogo modeling environment to use a relational database by providing better and easier use of interface for storing and retrieving data than currently provided by NetLogo, using lists and files during the execution of a model simulation and speed up the execution of models. The SQL Wrapper extension provides a way for NetLogo simulations to access existing data from, or update or inserts data into a relational database with an existing database schema. As shown in Figure 2, there are four components to enable this:

- NetLogo with its extension-mechanism;
- The SQL Wrapper implementing the extension-mechanism;
- A database access layer (i.e., JDBC and a suitable driver);
- A database system.

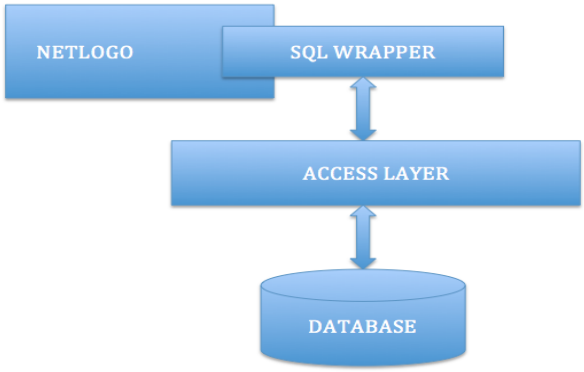


Fig 2. Sql wrapper acts as a bridge between NetLogo and the database

The NetLogo layer is concerned with accepting commands from NetLogo and returning data. Similarly, the database layer is concerned with communicating with the database. To separate these, a third layer is provided as a bridge between these two: The wrapper layer. This layer provides mapping between Netlogo data primitives and SQL primitives, and other mappings that are translations from Netlogo to SQL and vice versa. The context layer takes care of resource management and additional configuration. In this case, the connection pooling as implemented using an external software library called BoneCP. To allow the Netlogo user to use the connection pool, specific command syntax will added to enable and configure it when needed. All layers are depicted in Figure 3.

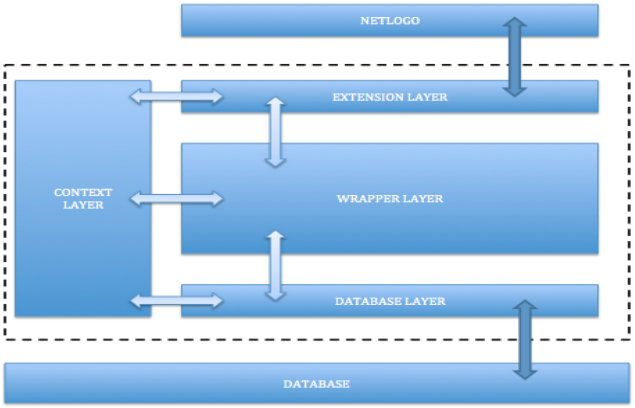


Fig 3. The internal architecture of sql wrapper has four layers

3.0 Implementation and Results

For validating the connected vehicles concept, we implemented a prototype using Netlogo 5.0.1 simulation rapid prototyping platform.

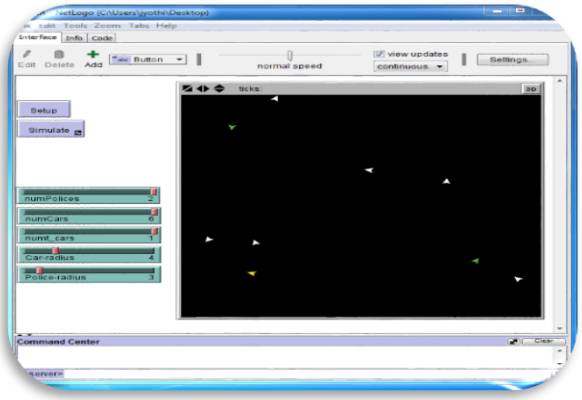


Fig 4. The interface of the present project

Netlogo library tools are included for the interface as shown in Figures 4, 5, and 6. In order to generate result for various input environment, a controller from number of simple vehicle, intruder vehicle and police vehicle are incorporated. A slider tool is provided to adjust the radius of the commutation range for police cars. We monitored the simulation output in multiple case studies. The range of adjustment tools includes the following:

1. numPolice vehicless: Number of police for the simulation environment
2. numCars: Number of cars for the simulation environment
3. numt_cars: Number of target in the simulation environment, and
4. Police-radius: Police communication range with its neighbors.

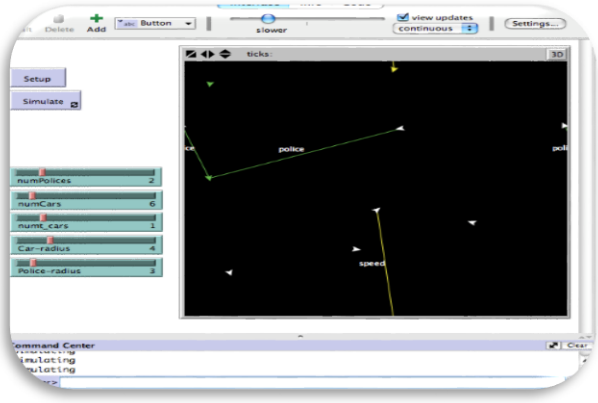


Fig 5. Agents connections in the data base and exchange data

In Figure 5 at a simulation moment, we observe wireless connection among nodes. Police, regular vehicles and intruder car are connected with one another and form a mobile network of information with neighbors. When an agent connecting with high speed agent then we name the link as 'speed'. When connecting with police then we name the link 'police'. A node connects with a police node and sends its information to the police node. Upon receipt, a police node searches for a target node information in the set. If a police node finds information about a target node, then it changes its direction toward the target Node's direction and location and starts to chase the target node. But in the simulation, target node's speed is much larger than the police's. When a police node approaches the target node within its sensing radius distance, then it connects with the target node with a link. It apprehends the target node agent.

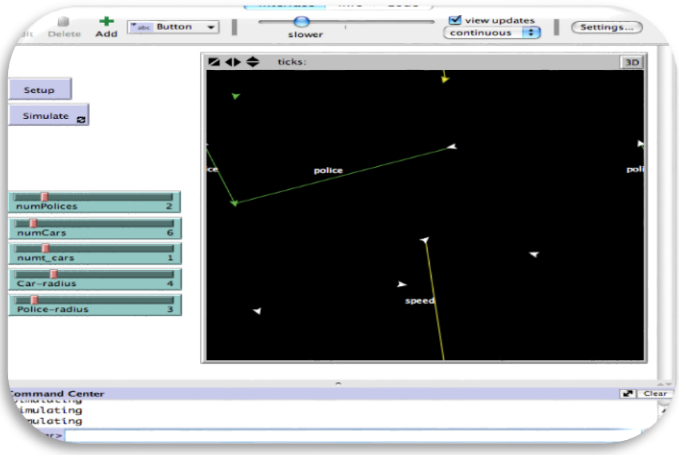


Fig 6. Agents connecting in the database and exchanging data

In Figure 6 at a simulation moment, we can see the wireless connection between agent nodes. Police, vehicles and intruder car are connecting with one another and forming a mobile network of information with its neighbor. When a white agent node connecting with high-speed yellow agent then naming the link as 'speed'. When connecting with green police then naming the link 'police'. A white node connects with a green police node and sends its information to the police node. On receiving, a police node looks searches for Target Node information in the set. If a police node finds information about a Target node, then it immediately changes its direction towards Target Node direction and location and starts chasing the Target Node. However, in the simulation Target Node speed is much higher than the police vehicle. When a green police node closes to the Target Node radius distance then it connects with Target Node with a Red link. The police vehicle apprehends the Target Node, yellow agent. The database is continually updated with the status of the vehicles. The target node is set to the highest speed car and the hop distance to the police is updated along with them. When the vehicle lost its connection with other vehicle, immediately the database is updated and the values of that particular vehicle are updated.

4.0 Conclusions

Connected car technology is a rather novel arena that enables exploration of possibilistes for innovation. These are useful for networked cars on the road and the space. Soon, there will be large networks of cars that can be seen moving as liquids yielding car lakes. As such, we need to use network properties to analyze dynamics of car networks. These can be useful for individual nodes as in measuring latency of effects of events reaching a node of interest. We have implemented a prototype and initiated navigation within it. This work creates a foundation for future studies. Managing networkwd cars as lakes to be monitored and directed is a promising direction for exploration.

Acknowledgement: Ms. Jyothi Vaddula implemented the netlogo model

References

[1] Woolridge, M. & Wooldridge, J. M., (2001) , Introduction to Multi Agent Systems, New York,NY, John Wiley & sons,
[2] Aruba networks (2011) , Next Generation Wireless Mesh Network
[3] Camps-Mur, D., Perez-Cost, X., & Sallent-Ribes, S., (2011), Designing Energy Efficient Access Points with Wi-Fi Direct
[4] Frodigh, M., Johansson, P., & Larsson, P., (2000), Wireless ad hoc networking -The art of networking without a network
[5] Aneiba, A., & Rees, S.J., Mobile Agents Technology And Mobility
[6] Kumar, P.R., Borkar, V., Gupta, P., Kawadia, V., Narayanaswamy, S., Rozovsky, R., Sreenivas, R. S., & Xie, L. (2002), Ad Hoc Wireless Networks :Analysis, Protocols, Architecture andTowards Convergence.
[7] Sanai, D., (2001), Detection of Promiscuous nodes using ARP packets.
[8] Tisue, S., & Wilensky, U., (2004), NetLogo: A Simple Environment for Modeling Complexity.
[9] Blom, J., Quakkelaar, R., & Rotteveel, M., NetLogo SQL Wrapper User manual

A Brownian Agent Approach for Modeling and Simulating the Population Dynamics of the Schistosomiasis

R. L. Cagnin, I. R. Guilherme, F. M. F. Gonçalves, J. F. P. Queiroz, and A. Baldassin
DEMAC/IGCE, UNESP – Univ Estadual Paulista, Rio Claro, São Paulo, Brazil

Abstract - *Multi-agent simulations have been successfully employed in the studies of population behavior. In this work we present a Brownian agent approach for simulating and modeling the Schistosomiasis infection. Early results suggest that the proposed model describes relevant aspects of the infection conditions.*

Keywords: Simulation; Schistosomiasis; Brownian Agents; Population Dynamics

1 Introduction

Multi-agent simulations (MABS) have been successfully adopted in the simulation of various problems [1] such as ecological systems modeling, population behavior for predicting crime, biological population dynamics, information spreading, formation of groups, among others [2]. Another important use of Multi-agent based simulations has been made in the context of infection and disease contention such as malaria [3].

In this kind of simulations, it is assumed that individual agents belonging to a major and complex system are endowed with a certain autonomy and intelligence. These agents interact for a purpose, a global behavior, or a specific goal. These interactions between different agents influence the dynamics of the entire system.

Among the various approaches for modeling MABS, a possible one is the Brownian agent. A Brownian agent is characterized by being able to generate a local field of information that can also influence the agent itself or other agents. The advantage of using Brownian agents is given in the context of problems that presents information spreading over time and space, for example, disease spreading [4-5].

In this work we use the Brownian agents to model and simulate the spread of Schistosomiasis disease and also to represent the dynamics of the population involved in the infection.

2 Schistosomiasis

The Schistosomiasis is a debilitating disease that can be lethal and it is caused by a worm type parasite of the genus Schistosoma [6]. The disease cycle has two main hosts: the *Biomphalaria* snail which acts as intermediary host, and men as a definitive host. Schistosomiasis occurs in regions where there are streams, ponds and low or nonexistent sanitation. In Brazil, it is still considered a serious public health problem because it affects millions of people each year causing a significant number of serious infection and even death [7].

The disease cycle begins when an infected man releases the worm eggs as excretes in a favorable water region for the parasite development. These eggs hatch in ciliated larvae called miracidium, which penetrate the epidermis and contaminate the *Biomphalaria* snail. After a period of maturation within the snail, these larvae suffer metamorphosis and are released through the skin of the snail, now as tailed larvae, called cercariae, the infective form of the disease. The cercariae seek the definitive host, men, and penetrate their skin. Now without the tail, they are loaded into the bloodstream and when they become adult parasites, they lodge in the veins of the intestine or bladder of the host, thereby causing Schistosomiasis. The adult worms reproduce and generate new eggs that are expelled by men, thus closing the cycle of the disease [6].

The contamination cycle depends on water resources, because the parasites (miracidium and cercariae) survive on this ambient for a short period of time, one or two days long, consuming their internal nutrients reservation. Also the snail usually lives in aquatic habitat.

The measures against the Schistosomiasis disease can be applied by combining two strategies: combating the *Biomphalaria* snail with improvements in sanitation, health care, and application of molluscicide; or applying educational sanitation campaigns for the population. These measures can be taken because the life cycle of the parasite depends on both interactions between larvae and hosts, and on the sanitation conditions for larval development [7].

3 Related work

Over the past years there has been an increasing trend in the application of mathematical and simulation models in the studies and control of infectious disease [8]. One of the advantages in using these models is the access of public health professionals and managers, allowing them to analyze and

apply prophylactic measures. Mathematical models have been created to understand the schistosome epidemiology and to the design of control programs.

In this sense, the literature provides a set of works that applies mathematical models for the Schistosomiasis contagion. Complex and detailed models can be used in simulation studies to assist the interpretation of data and to prevent the occurrence of epidemics [9].

Some models of the Schistosomiasis focus on the conditions of the environment and the prophylactic measures, trying to predict the potential efficacy of targeted chemotherapy or focal molluscicide application [10]. A mathematical model for Schistosomiasis transmission must integrate data from diverse sources to aid in designing control strategies, and evaluating surveillance programs [11]. The identified prophylactic measures against Schistosomiasis are: selective mass treatment, targeted mass treatment, mass treatment for animal reservoirs, and health education [12].

Detailed and complex models focus on realistic features including drug treatment for human hosts, infection age in snail hosts, density-dependent birth rate of snails, distribution of schistosomes within human hosts, and disease-induced mortality in both human and snail hosts [13], representing the biological consequences of applying some control strategies.

In this context, these mathematical models have been applied in stochastic simulations to investigate whether it would be possible to detect patterns predicted by theoretical models [14].

The population dynamics of the Schistosomiasis infection involves human and intermediate snail hosts as well as other possible species competing or serving as intermediary hosts. Some of these models consist in deterministic differential equations for the infected and susceptible subpopulations [15], or the possibility of migration between human habitats [16]. The deterministic system is generalized to a stochastic system of differential equations to account for the random behavior of variables which influences population dynamics [17].

All mathematical models presented in literature represents relevant features of the contagion process [18], like the population dynamics of each species and the environmental, social and economical conditions.

4 Brownian agents and the mathematical modeling

The model of this problem can be implemented by the Brownian agents approach. A Brownian agent is characterized by a set of state variables u_i^k where i = 1, ..., N refers to an individual agent i, and k indicates their different state variables [4]. In the agents modeling used in the simulation it is adopted two observable variables: space (Eq. 1) and velocity (Eq. 2), presented by all agents as it is assumed that every agent has mobility in space.

There are also two other internal variables: information and energy. The variable internal information (Eq. 3) allows characterizing the nature of the vector agents (men and snails) on the problem of the contagion; it characterizes the state of the agent as infected or healthy. The internal energy of the agent (Eq. 4), allows defining how active an agent will be at a given stage of the simulation. In the case of the proposed problem, this is equivalent to the metabolic reserve of the different parasite population individuals. As pointed out, parasites (Miracidium and Cercariae) survive temporarily consuming their nutrient reserves [19].

$$u_i^1 = \mathbb{r}_i, \quad (1) \qquad u_i^2 = d\frac{\mathbb{P}_i}{dt} = \mathbb{v}_i, \quad (2)$$

$$u_i^3 = \theta_i, \quad (3) \qquad u_i^4 = e_i. \quad (4)$$

The behavior of each agent is modeled by a deterministic and a stochastic component. The deterministic behavior is based on factors that influence the environment where they are inserted. In the case of biological populations, this influence is modeled as constraints associated with the water resources (lakes, rivers and ponds), whereas in the case of human populations, refers to the biological needs of the local population that lead them to an environment of possible contagion. The stochastic component is specified to represent free search along the geographical environment, i.e., each agent may randomly move around a mean position, simulating a wandering in search of resources, such as food. Thus, the dynamics of the variables may be influenced by stochastic and deterministic factors (Eq. 5).

The model of the environment corresponds to a geographical environment, where water resources, essentials for the dynamics of contagion, are modeled as deterministic attractive potentials which affect the three agent populations of miracidium, cercariae and snails. Thus, the water resource can be modeled from a mathematical function that describes an area of influence for the agents mentioned. For characterizing a lake or a pond along the environment, these potentials are represented by circular or elliptical geometry.

According to the discussed models of agent and environment, it is possible to determine the variation of the speed of an individual agent (Eq. 5) and also its energy consumption (Eq. 6).

$$d\frac{u_i^k}{dt} = f_i^k + F_i^{stoch}, \quad (5)$$

$$d\frac{u_i^4}{dt} = -m_i(t), \quad (6)$$

where f_i^k, represents the deterministic influences (water resources and information fields, described in the following) over the agent, F_i^{Stoch}, is the stochastic influences and m_i the parasite metabolic rate.

The energy dynamics will be applied only to the parasite population (miracidium and cercariae) because it is assumed that the lifetime of the parasite agent population is much

smaller than the lifetime of the vector population (men and snails). Thus, the number of agents of vector type remains constant during each simulation.

The internal information of the agent is presented only in snails and human agents, and varies according to the conditions for contagion. This variable contributes for building local areas of influence in certain regions of the environment, called information field.

The information field models the interaction between different populations. It is constructed from the individual contribution of each agent according to the value of its current internal information. Thus, the greater the number of agents in a given region of space, the greater is the effect of this field over the other parasite agents that constitute the system. This represents a higher chance for contamination.

Only uncontaminated vectors are assumed to become contaminated and produce regions that have higher chances of contagion for parasite populations. For example, a vector agent already contaminated does not influence the information field accessible to the parasite agents. To simulate the need for searching a host, the parasites agents feel the influence of the field, characterized as a deterministic force that attracts them to the vector agents. Moreover, vectors are generators of the information field contributing to the value of its state at a given point in space.

5 Computational model

The simulator development was performed using the oriented-object paradigm. As support technologies, the simulator uses Java based frameworks and Web technologies. Also, it is intended to create a framework as result of this work.

As the initial step, a conceptual model of the agents and the environment was developed and also their relationships. The whole computational model was performed by the MVC design pattern, allowing independence between the interfaces and the business model implemented. It means that it is possible to implement the model presented in this work in any other system architecture.

The simulator modeling process must take into account the elements which compound the problem of the contagion, thus, more attention should be taken in a consistent manner to represent the processes and conditions under which individuals react or modify their behavior during the simulation. For this purpose, the characteristics that distinguish individuals as well as their behavior define the attributes and methods of classes that represent them.

For the specific problem, the classes of the system correspond to the agents of multi-agent simulation, their environment and their interaction. A description of the classes is given in the next sections.

5.1 Agents modeling

In the following it is presented a description of the classes that represent the agents of the Multi-agent based simulator (Figure 1):

- Class BrownianAgent: This class represents Brownian agents, which have as attributes the set of state variables (internal energy and information) and observable variables (position and velocity), necessary for modeling this problem.

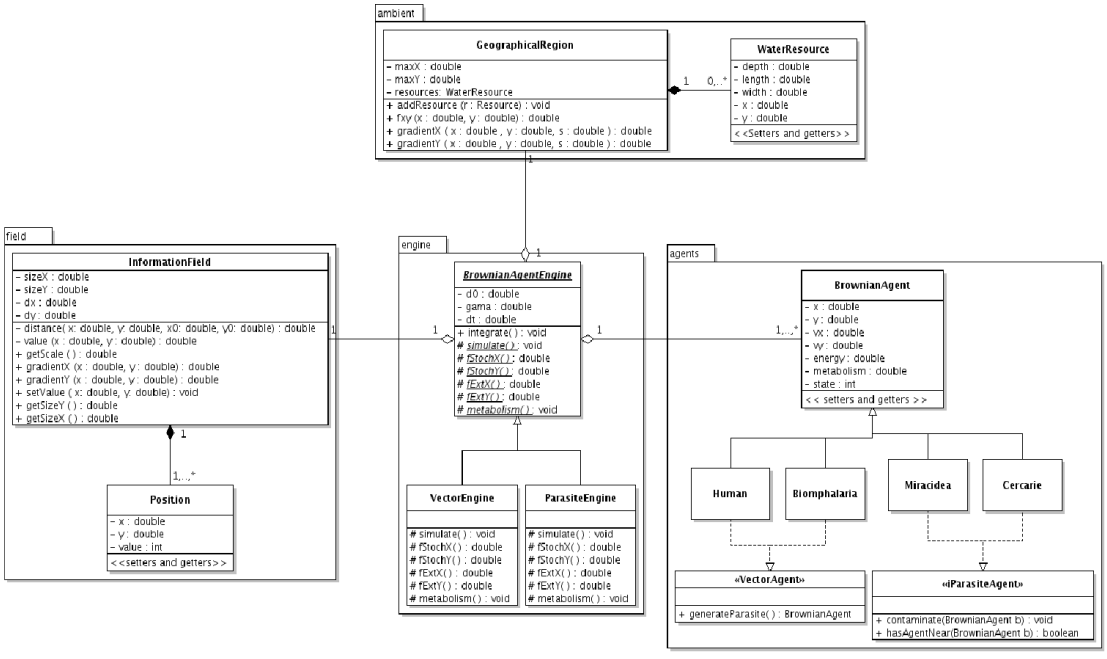


Fig. 1.Class diagram presenting the classes of the model used in the simulator.

- Interface VectorAgent: Presents an abstract method that constructs new Cercariae or Miracidium agents. This method is executed if the Human or *Biomphalaria* agent is infected. It is a method implemented by the classes *Human* and *Biomphalaria* by polymorphism. In particular, according to the life cycle of the *Shistossoma mansoni*, human agents must generate Miracidium agents, in the presence of water resource, and *Biomphalaria* agents must generate Cercariae agents.
- Interface ParasiteAgent: Presents two polymorphic methods to be implemented by classes that represent the parasites (Cercariae and Miracidium). The "hasAgentNear" method is the function that computes in a given simulation step if there is a Vector agent in vicinity. As a polymorphic method, it presents different actions for the different classes. In particular, it is necessary that Miracidium agents only sense the *Biomphalaria* agents and Cercariae agents, only sense the Human agents. The other method called "contaminate" causes a change in the internal information (attribute "state") of the Vector agent previously selected by the "hasAgentNear" method, thus contaminating the vector agent.
- Class Human and Class Biomphalaria: These classes represent the modeling of each vector agent necessary in the life cycle of the *Schistosoma*.
- Class Miracidium and Class Cercariae: Represent a population of parasite agents (Miracidium and Cercariae).

5.2 Environment modeling

The environment modeling is carried out by the classes "GeographicalRegion" and "WaterResource". Computer modeling of the environment requires that the species of *Biomphalaria* snails, Cercariae and Miracidium parasites must be found in water resources. Therefore the geographical environment presents a set of water resources.

The environment is modeled as a class called "GeographicalRegion". This class consists of a composition with another class called "WaterResource" which represents the water resources, such as ponds, lakes, standing water, etc. [6-8]. By simplifying assumptions, it is assumed that these resources have circular or elliptical geometry, thus its attributes "length" and "width" represent the semi-axes of an elliptical shape.

In the model, *Biomphalaria*, miracidium and cercariae agents feel the influence of water resource as an attractive deterministic potential, which traps them into the region of the modeled water resource.

The fact described before justifies the method called "fxy" presented in the "GeographicalRegion" class which calculates a component of the deterministic force presented in equation 5 (see section III). This method computes the value of the attractive force felt by the agents, resulting from the sum of all water resources of the geographical area, at a given point with coordinates x, y. The agents which are located near such resource and feel its influence becomes trapped along the elliptical or circular region. It represents the needs of water resources presented by the *Biomphalaria*, cercariae e miracidium population.

In a future model it will also be considered the existence of other resources present in the environment, as well as the motivation for human populations, such as housing, stores, hospitals, schools, etc.

5.3 Simulator engine

The simulator engine is composed of some classes which implement the Brownian agents approach and handle the events during the simulated scenarios. The events handled by the engine include computation of all the agents dynamics (Eq. 5 and 6), creation and removal of agents, changes in agents states, creation of the desired environment and creation of the initial population.

The "Simulation" class is the interface between the controller classes and the problem model discussed above (Figure 2). This class presents all the configuration values such as demographic data (number of individuals of all population), the geographical region data and timing sets. Also, it is composed of the "BrownianAgentEngine" class which describes all the computational model of the Schistosomiasis contagion problem.

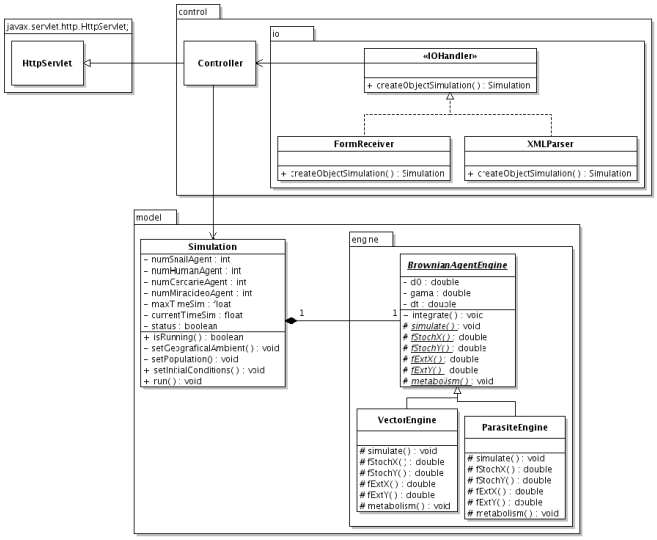


Fig. 2.Class diagram representing the classes that handles the input parameters and the classes responsible for the model simulation.

The "BrownianAgentEngine" class has the task of simulating the dynamics of the populations and handling the events that occur during a scenario simulation. It is an abstract class which presents a set of abstract methods implemented by the classes "VectorEngine" and "ParasiteEngine".

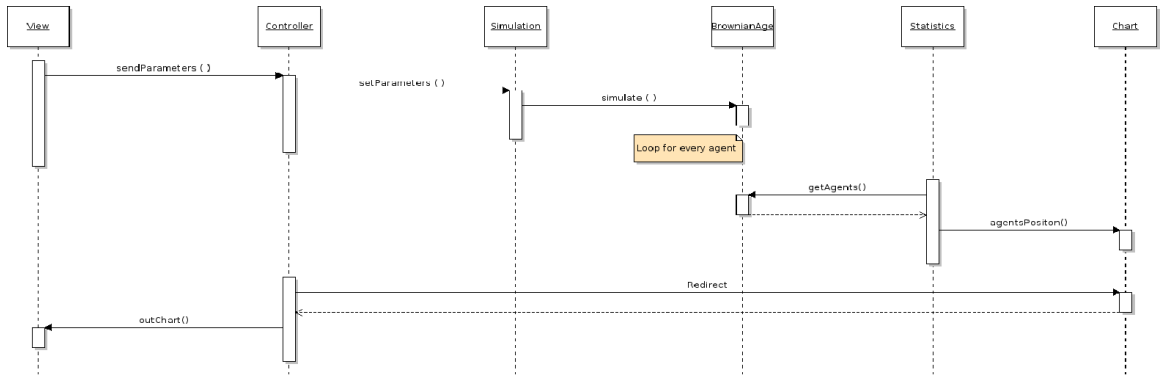


Fig. 3.Sequence Diagram representing the simulation process and all the involved classes.

The "VectorEngine" and "ParasiteEngine" class implements all the abstract methods of the "BrownianAgentEngine" class. As pointed out earlier, the dynamics of each population may be different according to the constraints implied for each population. For example, parasite agents have their energy consumed, simulating the behavior of the real ones. Each of these classes will handle a specified population with their constraints and specific features.

The method "fExtY" and "fExtX" presented in the "BrownianAgentEngine" class computes the sum of all components of the external features which influence the agent, (Eq. 5). When implemented in the "ParasiteEngine" it has also a component related to the water resource and a component related to the information field generated by the Vector agents.

Another method which has great importance in this problem is called "integrate", which integrates the equation 5 of the model based on the Brownian agent approach. The execution of this method changes the state of the attributes "x" and "y" of all the agents during the simulation, representing its motion along the geographical space.

5.4 Population interaction and the information field

The interaction between the agents is essential for the contagion modeling. As discussed earlier, the contagion occurs when different species interact and some specific conditions are satisfied.

In this model it is used the information field as a mechanism for interaction between different population. Recall that the information field is a dynamical field created by the vector agents with individual contributions, what means that the larger the number of vector agents in a given space coordinate the greater will be the chances for vectors to become infected.

This mechanism is created by the following feature: a class named "InformationField" holds the coordinates of all the uncontaminated vectors. A parasite agent in a given coordinate "x" and "y" will sum the contribution of the gradient of the created field along the simulation. This contribution is given by the methods "gradientX" and "gradientY".

As an uncontaminated vector agent changes its position coordinate, the value of the field also changes and so the contribution of the gradients calculated by the methods "gradientX" and "gradientY". Also, if an agent becomes contaminated its contribution to field is zero.

This class takes into account which of the vector agent and parasite agent are involved because the real problem involves the interaction of specific agents in different population. So there must be only one field for the "Human" and "Cercariae" agents and another different for the "Biomphalaria" and "Miracidium" agents.

The mechanism described above is responsible for the disease spreading along the geographical space.

6 The simulation process

This section describes the simulation process from the input parameters to the reports generated by the simulator. The sequence diagram in Figure 3 illustrates the process.

The user enters the input data from an XML file to the simulator. These data are sent to a Java class that adjusts the input parameters of the simulation. In general, the parameters are set by the user and grouped into three categories: geographical region, demographics and timing data.

Demographic data is based on population density of the region considered taking into account the population size of humans and snails of the genus *Biomphalaria* inhabiting the region as well as its distribution in geographic space, thus, define a population density space around some points representing, for example, the housing occupation area.

The other group of input parameters is based on the dimensions of the geographical region and the scale used during the simulation. Also, the user sets the parameters associated with water resources, like the size of the resource, the depth and its coordinate location.

Finally, the last set of parameters is associated with the simulation time, for example, duration of the simulation. The

configuration performed by the user, and the initialization of the simulation are given by the sequence diagram shown in the Figure 3.

During the simulation a list of all the agents is kept in memory and must be covered, so all agents who are active in the simulation are modified by the classes "VectorAgentEngine" or "ParasiteAgentEngine".

1. Update the information fields: the list of agents is iterated over for each vector agent (Human or snails) and the associated field is updated, i.e., the value associated with the new positions of the vector agents is changed according to the events that occurred;
2. Proliferation and contamination: contamination occurs as follows: if the agent is the type of Miracidium or Cercariae, and there is a snail or human respectively, in a given radius, these agents are contaminated. A contaminated agent remains in this state, since such agent is not cured. The proliferation occurs as follows: every agent of human type, which is contaminated, generates a new Miracidium agent, which is added to the end of the agents list. The same occurs with the snails that are infected, however the new agent created is a Cercariae agent.
3. Update the positions of agents: position, velocity and energy of all agents in the system are updated. If the energy of a given agent is depleted, its agent is removed from the simulation.

The simulation consists in the implementation of phases 1, 2 and 3, in that order, for a predetermined number of simulation steps.

Initially, it is assumed the existence of at least one human infected by the parasite *Schistosoma* agent that releases its feces or urinates near a water resource. From a computational point of view, when the position of a human agent is domain of the function that describes a water feature, new agents of the class “Parasite” are created.

The Parasite agents alter their dynamics influenced by the information field generated by the population of Vectors agents. This occurs because the information field generates an attractive force that directs the parasites to the vector agents.

If a Parasite agent reaches a Vector agent, the contagion will occur and will trigger the creation of new Parasite agents and will change the attribute "state" of the Vector agent. Vectors agents who are infected will not contribute to the information field. This ensures that Parasite agents will always aim to search for different healthy Vector individuals.

After running the simulation, the data from each step of the simulation are sent to the framework JFreeChart for subsequent data rendering.

6.1 Results

A series of simulations were performed aiming the investigation of the population dynamics of the Schistosomiasis contagion. To do this, we analyze the influence of the locations, water resources and population humans and snails. As we have already discussed, the cycle of transmission of the disease is complex and involves the interaction of different populations. Thus, an initial set of simulations were executed.

All simulations were performed in a maximum of 500 steps. The speed of spread of infection was calculated at the end of each simulation. For each scenario tested, 30 simulations were performed. The simulations are initiated with the diverse populations of agents arranged along the geographic space. The human agents will be arranged randomly, simulating a disordered occupation usually common in regions with low sanitation. The other populations that depend on water resources are allocated along the dimensions of these resources.

The factors mentioned before (size and distribution of the water resources and population density) directly influence the spread of information over time and space. In Figure 4, it is shown that the size of vector populations influences the propagation speed of the contagion. The distribution and size of water resources, influences directly the snails distribution. This means that a greater number of humans living around these resources become susceptible to the infection.

A high occupancy of humans allows rapid snails infection (Figure 5). The same behavior is observed with the number of snails. In both cases it appears that the significant increase of the population of individuals of parasites increases the possibility of infection from the individual vectors. These factors also increase the spread of the disease over the geographical space. This happens because the infected vectors create new parasites, closing the cycle of disease. This behavior was expected according to the characteristics of the disease cycle as previously discussed.

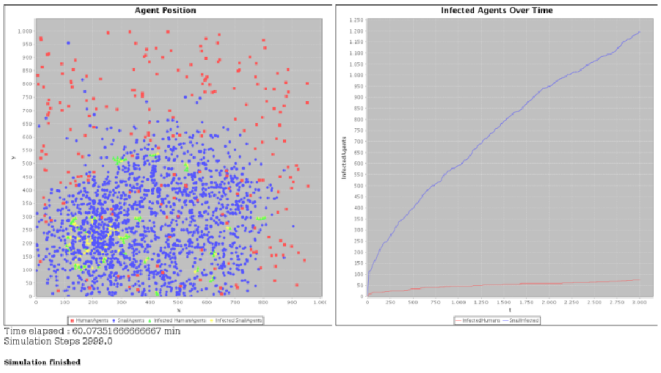


Fig. 4. The simulator interface presenting the calculated data. In red, it is shown the healthy human population, in green contaminated human population, in blue healthy snail population, and in yellow the contaminated snail population.

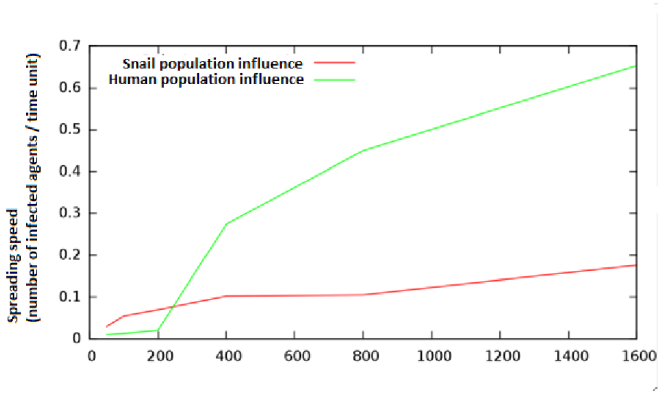


Fig. 5. Influence of the human and snail populations on the contagion spread rate.

7 Conclusions

A multi-agent approach based on the Brownian agent model has been used for modeling and simulating the population dynamics in the Schistosomiasis contagion. In general, the results obtained from the simulation showed that the location is a determining factor for infection, and this factor is influenced by the population density of vectors found along the geographic space. The simulator has undergone a series of tests checking various parameters, and in general, it represented relevant aspects to demonstrate the dynamics of populations in the Schistosomiasis contagion.

Model validation with real data will be performed in future work. Due to the complexity of real-world scenarios, the simulation will require the connection with models of real spatial data, and due to the high number of agents a version of the simulator that uses parallel processing will also be needed.

8 References

[1] F. Bousquet and C. Le Page, Multi-agent simulations and ecosystem management: a review. Ecol. Model . 176. 2004. pp. 313–332.

[2] V. Furtado, Modelagem e Simulação Multiagente da Criminalidade, Comp.Scien. 0002. Dec. 7, 2008.

[3] C. Linard, N. Poncon, D. Fontenille and E. F. Lambin. A multi-agent simulation to assess the risk of malaria re-emergence in southern France, Ecological Modelling 220 (2009) pp. 160–174.

[4] F. Schweitzer. Brownian Agents and Active Particles: Collective Dynamics in the Natural and Social Sciences, Ed. Springer-Verlag, 2003 Berlin.

[5] H. Hu, P. Gong, B. Xu. Spatially explicit agent-based modelling for schistosomiasis transmission: Human–environment interaction simulation and control strategy assessment. Epidemics 2 (2010) 49–65.

[6] V. Schall, C. L. Massara, M. J. Enk, H. S. Barros. Os Caminhos da Esquistossomose. Parte I Dentro do nosso corpo. Centro de Pesquisas René Rachou/Fiocruz. 2007. (Série Esquistossomose, 8).

[7] J. Shin, Y. Park. Brownian agent-based technology forecasting. Technological Forecasting & Social Change 76 (2009) 1078–1091.

[8] C. Minggang, U. F. Zheng (1999) Schistosomiasis control in China (INTRO).

[9] S. Ouaro, A. Traoré (2013) Deterministic and Stochastic Schistosomiasis Models with General Incidence

[10] M. E. J. Woolhouse (1996) Mathematical models of transmission dynamics and control of schistosomiasis

[11] Spear RC, Liang S (2011) Mathematical modeling as an aid in understanding transmission and control of Schistosoma japonicum in hilly and mountainous regions of China.

[12] J. Xiang, H. Chen, H. Ishikawa (2012) A mathematical model for the transmission of Schistosoma japonicum in consideration of seasonal water level fluctuations of Poyang Lake in Jiangxi, China.

[13] Z. Feng, C.-Che Li, F. A. Milner (2002) Schistosomiasis models with density dependence and age of infection in snail dynamics.

[14] M. S. Chan, F. Mutapi, M. E. J. Woolhouse and V. S. Isham (1999) Stochastic simulation and the detection of immunity to schistosome infections.

[15] E.J. Allen, H.D. Victory, Jr. (2003) Modelling and simulation of a schistosomiasis infection with biological control.

[16] Z. FENG, C. - C. LI, F. A. MILNER (2005) Schistosomiasis M o d e l s with Two Migrating H u m a n Groups

[17] G. M. Williamsa, A. C. Sleigha, Y. Lia, Z. Fengb, G. M. Davisd, H. Chene, A. G.P. Rossa, R. Bergquistf, D. P. McManusa (2002) Mathematical modelling of schistosomiasis japonica: comparison of control strategies in the People's Republic of China

[18] D. Gurarie, X. Wang, A. L. Bustinduy, and C. H. King (2011) Modeling the Effect of Chronic Schistosomiasis on Childhood Development and the Potential for Catch-Up Growth with Different Drug Treatment Strategies Promoted for Control of Endemic Schistosomiasis

[19] Ruppert, E., Fox, R. and Barnes, R. (2005) "Zoologia dos Invertebrados", Editora Roca, 7ª edição.

A Pong Game Agent Using the Neuronal Network Model

Robert Hercus and Hong-Shim Kong
Neuramatix Sdn. Bhd., No. 27-9 Level 9, Signature Office Bandar Mid-Valley,
59200 Kuala Lumpur, Malaysia

Abstract - *This paper presents an alternative approach to intelligent agent development in video games through the use of the Neuronal Network Model (NNM). The NNM is an integrated network of multiple networks of neurons capable of learning from and adapting to its environment. The pong game is simulated to evaluate the feasibility of implementing the NNM as an agent controller. The goal of the controller is to place the agent which is the paddle, in the correct position to return the pong ball. The learning process of the model is based solely on its failures in predicting the correct motor actions to return the pong ball. Experimental results show that the NNM is capable of acting as an intelligent agent controller.*

Keywords: agent, control, game, neuronal network, pong

1 Introduction

Today, there is a huge market for the video games industry. In the United States alone, it was reported that consumers spent US$20.77 billion in 2012 [1] on video games, hardware and gaming accessories. Of these sales, 40 percent of them were purchases of digital content, which includes games, add-on content, mobile apps, subscriptions and social networking games. Video games are going through massive enhancements and changes driven by technology advancements. At present, games can be played on a variety of media such as dedicated consoles, personal computers, handheld devices and smartphones. At the same time, the demand for realistic gaming experiences has also increased. Players today expect games to be intelligent as well as engaging. It is only through such experience that a player's interest in the games can be continuously piqued and sustained.

In the pioneering stages of the video games industry, the agent or opponent in a game is often a preset computer program with finite strategies. Once these strategies have been discovered, the players can respond to them. As such, games begin to lose their appeal. Furthermore, if the difficulty in games is increased, the number and complexity of strategies may increase as well. Encoding strategies into these games can thus prove to be a daunting task.

On the other hand, when a game is played against another person, the games are more interactive and entertaining. Therefore, efforts have been made to develop believable, intelligent games. The advent of artificial intelligence (AI) technology provides solutions to this challenge. AI methods such as neural networks, Bayesian models and behaviour trees have been used in many ways to enhance a player's satisfaction in terms of gaming experience and entertainment value.

One such usage is the simulation of human-like intelligence and behaviour in an agent or non-player character (NPC). An NPC is an entity or character that is not controlled by a player in a game, such as the hostile entities in Far Cry 3 [2]. Far Cry 3 is a first-person shooter (FPS) game, wherein a player needs to combat numerous mercenaries. In this game, AI is programmed into the enemies so that they can react in many situations adaptively. For example, if the enemy is injured, he can shout for help or release emotional distress. Another example is the Drivatar opponent drivers in Forza Motorsport [3]. Drivatar employs neural networks to control drivers. Each driver is assigned a different neural network. As such, each driver has different skill levels and driving tendencies, such as how they approach a particular corner or how aggressively they try to overtake.

Matchmaking in online gaming is another example of the usage of AI in games. In Ghost Recon Online [4], an FPS game, a neural network has been used to select players from a pool of players with matching skills in a multiplayer online match. TrueSkill [5], which is based on Bayesian networks, has been developed by Microsoft for the Xbox Live online gaming for the same purpose as well. The matching is done by considering each player's profile, which includes information about the player's behaviour and personal preference. The information on players is derived from historical data, taking into account both previous match results and player attributes collected by tracking the players' behaviour over time. Neural networks make it easy to include additional parameters, and allow for continuous updating of the model in real time as new data is collected. A good matchmaking process can increase the chance of players having fun together, thus improving player retention.

AI can also be used to plan the path of agents in a game environment, such as in real time strategy (RTS) games [6], [7]. RTS games usually require positioning and movement of units and structures to secure game areas. When deciding the movement of units, the complexity of the terrain and obstacles may be taken into account by AI to decide future courses of action.

In recent years, game-AI contests such as Unreal Tournament [8] and StarCraft [9], have been held to apply artificial intelligence algorithms on commercial gaming platforms. Such contests offer researchers and developers a platform to evaluate their AI algorithms in robust game environments. For example, in the AIIDE (AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment) StarCraft AI tournament, AI approaches such as finite state machines (FSM), decision trees and probabilistic inferences have been employed for micromanagement and/ or macromanagement of agents in the game [10], [11].

Numerous researches on the development and application of artificial intelligence for games can be found in many publications. For example, Kotrajaras and Kumnoonsate presented a tool that uses genetic algorithms and steepest ascent hill-climbing to learn and adjust map properties [12]. In [13], Recio *et al.* have developed a generic framework based on swarm intelligence, specifically ant colony optimisation for controlling agents in a dynamic environment. These show that research on artificial intelligence in games is gaining momentum. Moreover, game environments provide a range of excellent platforms for fundamental AI research.

In this paper, the Neuronal Network Model (NNM) is introduced as an intelligent agent's controller in games. The NNM is a data structure consisting of interconnected nodes in a multi-level network. A group of networks in the model can be interconnected to perform a task. In this work, the feasibility of the NNM is demonstrated by its application in a simulated pong game. The objective of the NNM in the game is to predict the trajectory of the pong ball, and to actuate the paddle to the predicted final position that will result in the ball being returned successfully. The NNM is implemented in the NeuraBase toolbox, which can be downloaded at [14].

The remainder of the paper is organised as follows: Section 2 describes the fundamental design of NNM, followed by the description of the pong game and its NNM controller designs in Section 3. In Section 4, the learning algorithm of the model is presented. Section 5 analyses and discusses the experimental results. Finally, Section 6 presents conclusions and future work to be done.

2 Neuronal network model

The NNM is a multi-node, multi-level, and multi-network neuronal network which is based on a concept of how the human brain may work. The basic node in the network is termed an elementary neuron, which represents an event. A couple of elementary neurons can be associated temporally or spatially to form a sequence of events, such as shown in Fig. 1. A detailed description of the NNM is available in [15].

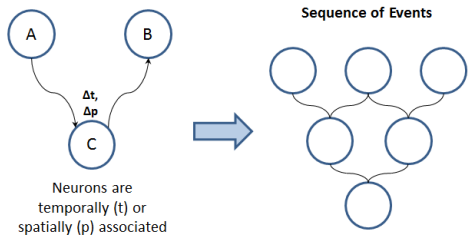


Figure 1. The NNM fundamental design

Generally, neurons can be classified into three categories: sensory neurons, motor neurons and controller neurons. Sensory neurons represent the data acquired by sensors, whereas motor neurons represent the motor actions that are performed in response to the sensory stimuli. Controller neurons associate a sensory neuron sequence to a motor neuron action.

A group of sensory or motor neurons can be associated in a multi-level network, and form a network of sensory or motor event sequences respectively. These networks represent sequences of events. In other words, a history of events is stored in the NNM. As for the controller neurons, they are grouped in a network known as an interneuronal network. The relationship between all the networks is illustrated in Fig. 2, which shows the general architecture of NNM. This general architecture has been applied in the balancing of an inverted pendulum [16], navigational control of an unmanned aerial vehicle [17] and control of an antagonistic muscle actuated manipulator [18].

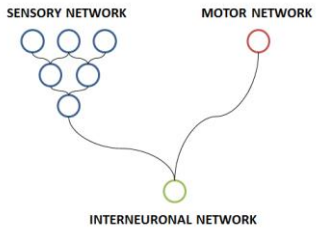


Figure 2. The NNM architecture

A detailed description of how the general NNM architecture applied in the pong game is presented in Section 3.

3 Pong game and NNM controller model

This section describes the design of the pong game, and more essentially the design of the NNM controller.

3.1 Pong game design

A pong game environment is simulated with the 2D robotics simulation platform, Stage [19], [20], as shown in Fig. 3. The blue rectangular object is the simulated paddle, and the red spherical object is the pong ball. The green area is the playable area, which is the table for the game. The paddle

is only allowed to move horizontally, while the ball is allowed to move anywhere within the table area, in a linear fashion.

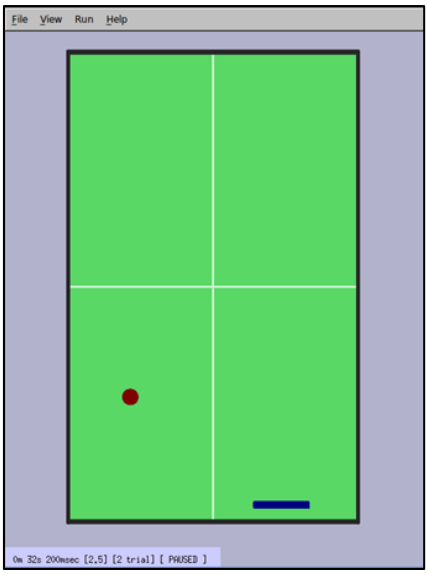


Figure 3. Simulated pong game in Stage

In this application, the blue paddle is controlled by the NNM. Its opponent is a preset computer program, which returns the ball in random directions from the other end of the table. In this simulation, the ball can start from any of 15 given positions at the top of the table. The ball can move at any angle, between 21 to 159 degrees. The NNM will learn to position the paddle at the correct position at the bottom of the table in order to return the ball successfully. The paddle can be placed at 14 possible horizontal positions. The discrepancy between the ranges of the paddle (14 discrete values) and pong (15 discrete values) positions is due to the size and shape of the objects. These attributes determine the ranges of the objects. The ranges should confine the movement of the objects within the table area.

The deflection of the ball at the sides of the table is simulated as a perfect-reflection. However, the deflection of the ball at the paddle and at the top of the table need not be of perfect-reflection, but can be anywhere between 21 to 159 degrees.

3.2 Pong game NNM architecture

The structure of the NNM in this pong game is shown in Fig. 4. It is a 4-network-NNM architecture (denoted by 4 different colours), which is based on the general 3-network-NNM architecture in Fig. 2. The sensory events of the pong game are the ball positions on the pong table, in terms of chessboard-based coordinate system (*x* and *y* axis, as column and row respectively). These events are captured in a network known as a sensory network. On the other hand, the positions of the NNM controlled paddle are the motor events in the model. These paddle positions are stored in another network, known as motor network. A sensory event and a motor event are associated through a controller event in another network defined as an interneuronal network.

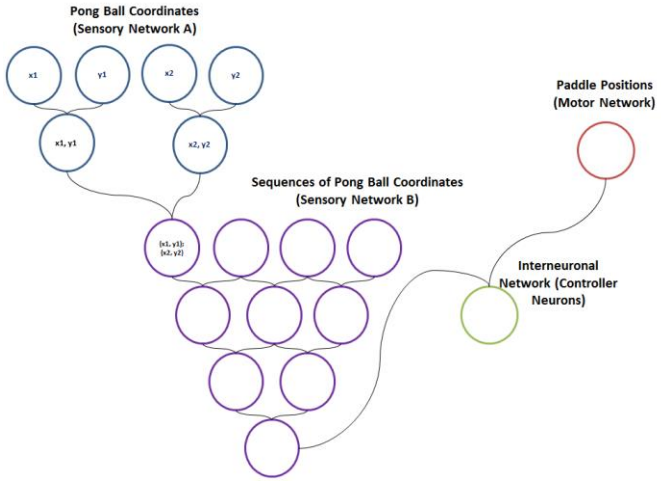


Figure 4. The NNM architecture of pong controller

Fig. 5 shows the coordinate system of the pong table. The range of *x* axis is of lower case alphabets: *{a, b, c, …, n, o, p}*, while the *y* axis is of upper case alphabets: *{A, B, C, …, W, X, Y}*. So, the positions of the ball and paddle in the figure are *(b, B)* and *(h, Y)* respectively.

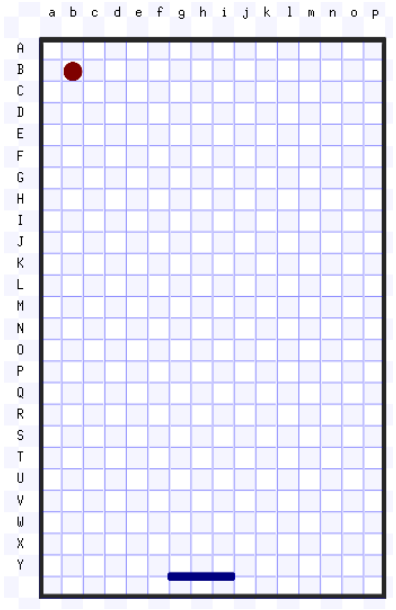


Figure 5. Grid map of simulated pong game

3.2.1 Sensory network

The coordinates of the pong ball are sampled at 5 pre-defined positions on the *y* axis. The five *y* positions are *{D, F, I, N, V}*, wherein each position comes after another position in a sequence heading towards the NNM-controlled paddle's

end. When the ball has reached the defined *y* positions, the coordinate is taken and stored in the NNM.

The first two coordinates acquired by the sensor provide the information on the initial trajectory of the ball. The next three coordinates refine the trajectory of the initial information, as the ball trajectory will be changed if it hits the sides of the table.

The sensory events are defined by two stacked sensory networks, such as shown in Fig. 4. One of the sensory networks, Sensory Network A builds the coordinates from a set of pre-defined elementary sensory neurons. The elementary sensory neurons contain the basic representations to define the coordinates. Based on the possible squares on the *x* and *y* axes, there are only 21 elementary sensory neurons (16 *x* squares + 5 *y* squares).

Sensory Network A captures the coordinates at interval *y* axis squares that the ball has "glided over" when the game is running. Therefore, there are only 2 levels in this network. The first level contains the elementary neurons, and the second level stores the coordinates. This network will serve as the basis to build the sequences of ball positions in the other sensory network, Sensory Network B.

Sensory Network B has only 4 levels of neurons. This is because there are only 5 coordinates from the start of the ball at the opponent's end to the NNM's end. Two coordinates are required to form a sequence in the first level neuron of this network. Hence, only 4 levels of neurons exist.

3.2.2 Motor network

In the motor network, there are 14 basic motor events: *{b, c, d, …, m, n, o}*. Hence, there are only 14 elementary motor neurons. The motor event, which is also known as motor action, is the position of the paddle. The position corresponds to the *x* axis squares where the paddle can be placed, while being fixed at the bottom of $y = Y$ (*y* axis square). The range is set between *b* and *o* because the length of the paddle is of 3 units, and the need of keeping the paddle within the table.

The motor actions are stored in a single level in the network. Each motor action is associated with at least a controller event.

3.2.3 Interneuronal network

The relationship between a motor action and a sensory event is defined by a controller event in the controller interneuronal network, through a controller neuron. A sensory event can have multiple controller events, wherein each controller event is associated with a different possible motor action.

The strength of the relationship between a sensory event sequence and a motor action is weighted in terms of being the correct action. By using the frequency attribute to define this, correct motor actions can be predicted for a sensory event.

4 Learning algorithm

The overall learning model of the NNM controller is depicted in Fig. 6. The learning starts when the pong ball has been hit by the opponent, and on its way towards the NNM's end of the table. The coordinate of the ball is acquired sequentially at a series of *y* axis squares.

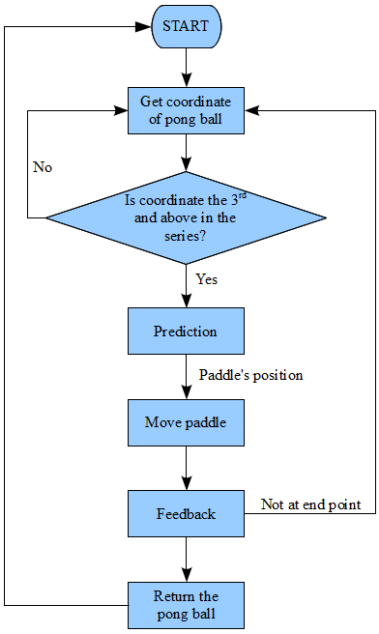


Figure 6. The overall learning process flow

4.1 Prediction

The prediction starts when the third coordinate in the *y* axis series is acquired. The prediction does not begin from the first two coordinates because at least two coordinates are required to gauge the direction of the pong ball. Therefore, it is reasonable to begin the prediction at the third coordinate. Furthermore, at the third coordinate, the ball is still within the opponent's area (about a quarter way of the pong table).

The NNM controller will attempt to return the pong ball based on the sequence of sensory events received. A matching sensory event is searched within its network. If a match is not found, it means that the event is a new sensory event. The new event will be encoded into the NeuraBase sensory networks; and a random motor action to position the paddle is executed.

On the other hand, if a match is found, a number of potential motor actions will be given through the related controller neurons. The controller neuron has to meet a

certain prediction threshold before the corresponding motor action is considered for execution. In this model, the prediction threshold, which is the frequency of the controller neuron, has to be equal or greater than five *(≥ 5)*.

The controller neurons which have met the threshold will be processed with the weighted average method to predict the most sensible motor action. If none of the controller neurons related to the sensory event has met the threshold condition, a random motor action will be executed.

4.2 Feedback

Once the recommended motor action is given, the paddle is moved accordingly to position. When the paddle is in position, feedback is provided to the controller for learning. Fig. 7 shows the feedback learning process of the model. If the pong ball has not reached the paddle *y* axis position, which is also the end point, learning is not required on the feedback since the pong ball is still moving forward.

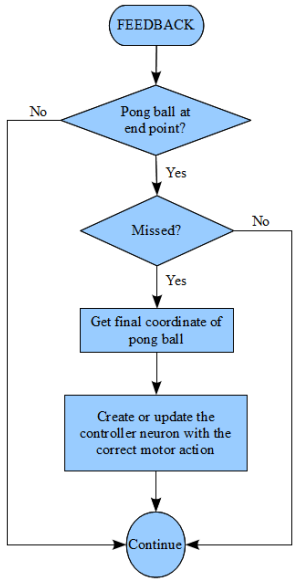


Figure 7. The feedback process flow

When the pong ball has reached the end point, the feedback is analysed. In this model, the NNM is learning from its failures to predict correctly. If the ball is returned successfully, nothing is updated in the NeuraBase and the game continues. However, if the paddle misses the ball, the final position of the ball is taken into consideration for learning. The final position is treated as the expected position of the paddle if given the same sensory sequence in future, subject to meeting threshold requirements.

NeuraBase registers this knowledge by creating or updating a controller neuron. If the controller neuron has already been registered, the corresponding controller neuron is updated instead. With this, knowledge is reinforced. A reward is given to the associated controller neuron through the increment of the frequency attribute. In this case, the reward is 5.

When learning is based on failed predictions, the previous failures in prediction are corrected by creating or updating the correct controller neuron. In this case, the frequency of the erroneous controller neurons is not penalised. Instead, the correct controller neuron's frequency is increased.

As such, the problem of overfitting is minimised or avoided altogether. Overfitting may occur in the pong game when the high number of successful returns may be misconstrued as good prediction, even though it is actually not.

5 Experimental results

The pong game was started with only the elementary sensory and motor neurons defined in NeuraBase, and run for 10000 trials. Each trial begins with the ball at a random position from the opponent's end, with a random angle as the trajectory. The trial ends when the NNM-controlled paddle misses the pong ball. Fig. 8 shows the number of times the ball was returned by the NeuraBase-controlled paddle in each trial.

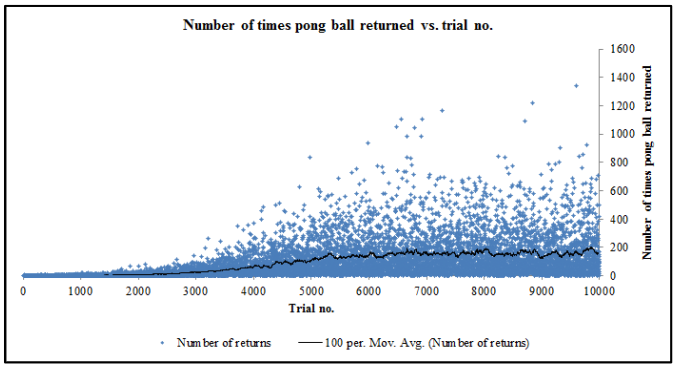


Figure 8. Frequency of pong ball returns per trial

A 100-trial moving average trend line of the frequency of ball returns per trial is shown in Fig. 9. As observed in the figure, the learning process is on an uptrend, and begins to show saturation at around 6000 trials, between 125 and 200 returns.

Fig. 10 shows the total number of neurons used in NeuraBase after 10000 trials. Altogether, 30341 neurons were created in the NeuraBase. This requires approximately 1.3 Mbytes of physical memory as each neuron uses 40 bytes of memory. Out of these neurons, 16401 neurons are built in the extended sensory network, 14 neurons in the motor network and 13926 neurons in the controller interneuronal network.

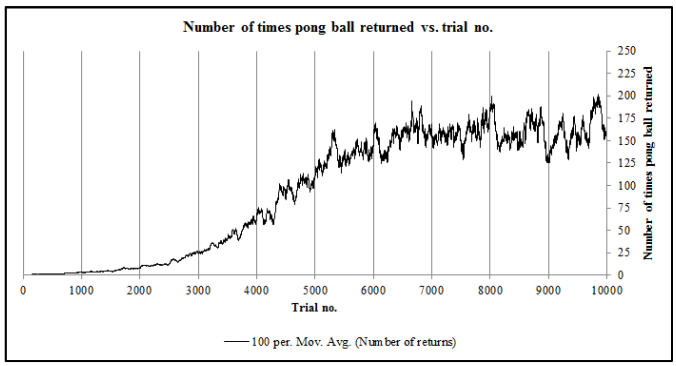


Figure 9. Moving average of frequency of pong ball returns per trial

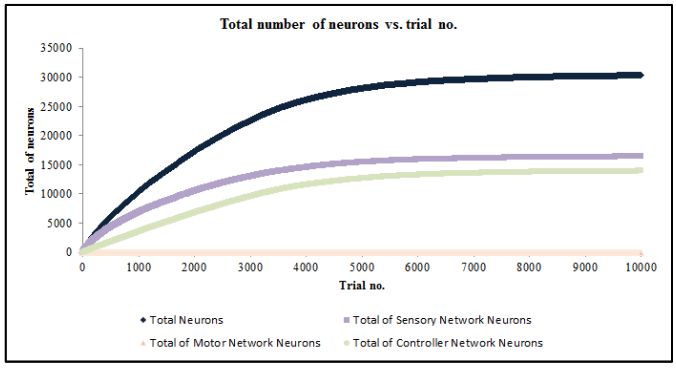


Figure 10. Neuron statistics in NeuraBase

Initially, the number of neurons in the sensory and controller networks is growing in a steep slope because of the high occurrence of new sensory events or experience. This growth eventually approaches saturation as the number of new sensory events decreases. The number of controller network neurons is lower than sensory network neurons all the time; as the links between sensory network neurons and motor network neurons are only created by learning the final pong ball position when it is missed. Therefore, only the correct motor actions are associated with sensory neurons. However, each sensory neuron can possibly have a few correct motor actions due to the variation of pong ball trajectories in a square.

Another experiment was the study of the failure of prediction from NeuraBase. In the failure study, a hundred trials were run with the NeuraBase that has been trained earlier with 10000 trials. As each trial ends with a miss, hence, there are 100 failures. Suppose that an instance is a series of predictions from NeuraBase to prevent an opponent's attempt to score. Overall, 15584 such instances of prediction were performed. So, the failures occurred at 0.6417% (99.3583% successes) of the time. Out of the 100 trials, 45 trials failed because the predictions were still random, indicating that there is still room for NeuraBase to learn. Therefore, only 0.3529% of instances are actually failures from NeuraBase's predictions.

One reason found for the failures in NeuraBase's predictions is the approximation of the ball position to a square on the pong table. The ball can be in any position in a square. As such, a square may consist of a number of different paths, even if the trajectory angles may be the same. This may result in a miss because the predicted motor action may only tolerate a certain variance in the path. Even if another motor action is considered, it may fail again as the variance may have just shifted.

Table I shows how the path can be different for the same sensory event, *(b, V)* square. In the *x* axis, any value between *-6* and *-7* is considered as *b*. As for the *y* axis, any value between *-8* and *-9* is mapped to *V*. As depicted in the table, there are multiple ball coordinates corresponding to the same square. The predicted motor action, which is *d*, is positive for most coordinates, except for an anomaly in the first row (negative feedback).

TABLE I
PATH OF PONG BALL AND SENSORY DATA REPRESENTATION

x Axis	Sensory X	y Axis	Sensory Y	Trajectory Angle	Feedback
-6.09	b	-8.15	V	-45	NEG-
-6.62	b	-8.31	V	-129	POS+
-6.66	b	-8.31	V	-129	POS+
-6.64	b	-8.29	V	-129	POS+
-6.22	b	-8.18	V	-45	POS+
-6.16	b	-8.21	V	-45	POS+
-6.18	b	-8.23	V	-45	POS+

In addition, failure may also be caused by the fact that the paddle is sometimes slow in moving towards the predicted position. The pong ball may have a shorter distance towards the end of the table; and the paddle may coincidently be further away from the point that it is supposed to be. So, the ball might reach the end before the paddle is in its final position. However, the chances of this happening are small, and are based on the Stage simulation model specified for velocity control of the paddle.

The causes of failures described above may be solved by having more sensory information, for example by having extra angles allowed to be travelled by the ball. Or, an extra coordinate in addition to the five already in the series. Speed may be also factored into the NeuraBase design so that the paddle can move with variable speeds.

6 Conclusions

The research described in this paper has demonstrated that the Neuronal Network Model performs well as a controller in a game environment. The high percentage of successful predictions is a testimony to this. The

experimental results also show that the total amount of required memory is small.

As future work, more parameters such as velocity and acceleration can be incorporated into the controller of the pong ball. Besides that, a 3-dimensional model of the pong game could be developed to examine how the model performs in a more complex environment. Two independent NeuraBases could be created for a two-player game as well.

References

[1] Entertainment Software Association. (2013). Essential facts about the computer and video game industry [Online]. Available: http://www.theesa.com/facts/pdfs/esa_ef_2013.pdf

[2] Wikipedia. (2014). Far Cry 3 [Online]. Available: http://en.wikipedia.org/wiki/Far_Cry_3

[3] Microsoft Research. (2014). Drivatar in Forza Motorsport [Online]. Available: http://research.microsoft.com/en-us/projects/drivatar/forza.aspx

[4] Olivier Delalleau, Emile Contal, Eric Thibodeau-Laufer, Raul Chandias Ferrari, Yoshua Bengio, and Frank Zhang. "Beyond skill rating: advanced matchmaking in Ghost Recon Online" ; IEEE Trans. Computational Intelligence and AI in Games, vol. 4, no. 3, pp.167-177, Sept 2012.

[5] Micrsoft Research. (2014). TrueSkill Ranking System [Online]. Available: http://research.microsoft.com/en-us/projects/trueskill

[6] Renato Luiz de Freitas Cunha and Luiz Chaimowicz. "An artificial intelligence system to help the player of real-time strategy games," in 2010 Brazilian Symp. Games and Digital Entertainment (SBGAMES), pp. 71-81.

[7] Sindre Berg Stene. "Artificial intelligence techniques in real-time strategy games – architecture and combat bahavior," M. S. thesis, Dept. Comput. and Inform. Sci., Norwegian Univ. of Sci. and Technology, Norway, 2006.

[8] Wikipedia. (2014). Unreal Tournament [Online]. Available: http://en.wikipedia.org/wiki/Unreal_Tournament

[9] 2014 StarCraft AI Competition [Online]. Available: http://webdocs.cs.ualberta.ca/~cdavid/starcraftaicomp/

[10] Santiago Ontañón, Gabriel Synnaeve, Alberto Uriarte, Florian Richoux, David Churchill, and Mike Preuss. "A survey of real-time strategy game AI research and competition in StarCraft" ; IEEE Trans. Computational Intelligence and AI in Games, vol. 5, no. 4, pp. 293-311, Dec 2013.

[11] David Churchill. (2013). 2013 AIIDE StarCraft AI Competition Report [Online]. Available: http://webdocs.cs.ualberta.ca/~cdavid/starcraftaicomp/report2013.shtml

[12] Vishnu Kotrajaras and Tanawat Kumnoonsate. "Fine-tuning parameters for emergent environments in games using artificial intelligence" ; Int. J. of Comput. Games Technology, Jan 2009.

[13] Gustavo Recio, Emilio Martin, César Estébanez, and Yago Sáez. "AntBot: Ant colonies for video games" ; IEEE Trans. Computational Intelligence and AI in Games, vol. 4, no. 4, pp. 295-308, Dec 2012.

[14] NeuraBase Generic Toolbox [Online]. Available: http://www.neuramatix.com

[15] Robert George Hercus. "Neural networks with learning and expression capability". U.S. Patent 7412426 B2, 2008.

[16] Robert Hercus, Kit-Yee Wong, and Kim-Fong Ho. "Balancing of a simulated inverted pendulum using the NeuraBase network model" ; Lecture Notes in Comput. Sci., vol. 8131, pp. 527-536, 2013.

[17] Robert Hercus, Hong-Shim Kong, and Kim-Fong Ho. "Control of an unmanned aerial vehicle using a neuronal network," in 2013 IEEE Symp. Computational Intelligence, Cognitive Algorithms, Mind, and Brain (CCMB), Singapore, 2013, pp. 73-79.

[18] Robert Hercus, Kit-Yee Wong, and Kim-Fong Ho. "Control of a muscle actuated manipulator using the NeuraBase Network Model" ; J.Automation and Control Eng, vol. 2, no. 3, pp. 302-309, 2014.

[19] The Player Project [Online]. Available: http://playerstage.sourceforge.net/

[20] Brian Gerkey, Richard T. Vaughan, and Andrew Howard. "The Player/Stage Project: Tools for Multi-Robot and Distributed Sensor Systems", in Proc. 11th Int. Conf. Advanced Robotics, Coimbra, Portugal, June 2003, pp. 317-323.

Towards a Multi-Agent Architecture for Process Supervision and Control System

J. F. P. Queiroz[1], **I. R. Guilherme**[1], **and R. L. Cagnin**[1]
[1]DEMAC/IGCE, UNESP – Univ Estadual Paulista, Rio Claro, São Paulo, Brazil

Abstract - *Technological advances have leveraged the use of devices in industrial environments, which have contributed to the increase in their complexity. In this sense, there is a need for new technologies and approaches to address involved complexity and develop computational systems that provide tools to integrate heterogeneous components, manage and analyze the huge amount of data produced, in order to automate the activities of process supervision and control. In this context, the approach of multi-agent systems, service-oriented architecture and semantic technologies offer promising and novel ways to design and develop complex, distributed, flexible and scalable systems. This work presents a multi-agent architecture based on these technologies aiming to provide a reference model for the development of process supervision and control system. The proposed architecture was specified based on a literature survey and brings together the main requirements and features related to the use of these technologies for process supervision and monitoring.*

Keywords: Multi-agent System; Semantic Web; Service-Oriented Architecture; Supervision and Control System

1 Introduction

The technological advances in software and hardware have caused a cost decrease and consequently an increase on the industrial automation and information systems. With the increased use of controllers and measurement devices, the amount of generated data on the industrial environment has also suffered this effect. Many of the produced data are simply stored and underused or wasted, mainly because it is necessary a great effort, time demand and specialized knowledge for their integration and analysis. Most often, they represent an important information resource that can be used to improve the processes management and control.

In this context, new technological approaches based on computer systems need to be studied and proposed for processes supervision and control, not only for retrieving and analyzing the large volume of data, but also to automate other tasks related to the process monitoring and control support, assisting engineers and operators in decision making.

Such software systems need to deal with the complexity inherent to industrial environments, characterized primarily by the distribution and amount of heterogeneous components, as well as the large volume of produced data. This problem arises from the different communication technologies and interfaces used by these components. Therefore, the integration and management of processes and components are essential and a major requirement for supervisory and control systems. Other characteristics such as flexibility and scalability are also important, because these environments often require changes by updating or adding new resources such as devices and software modules [1].

The design and development of systems with these features need to be based on robust architectures and at the same time flexible enough to support system and environment upgrades and changes. Also they should provide tools for automating some user tasks and support them in decision making, improving their supervision and control capabilities.

In this sense, the multi-agent systems (MAS) approach provides more appropriate abstractions to deal with complex and distributed problems [2]. It aims to simplify the understanding, design, development and management of distributed, open, dynamic and heterogeneous computer systems [3]. Also in this context, web technologies such as service-oriented architectures (SOA) have been adopted to support the interoperability among heterogeneous components and systems. Ontologies have also been used to model the application knowledge and to provide better data integration and processing.

The goal of this work is to describe an architecture model that integrates the MAS approach, SOA and ontologies in order to be used as basis for the design and development of more open, flexible and scalable software solutions for the process supervision and control in industrial environments. In this sense, we present a multi-agent architecture defined and specified based on a survey of a set of related works that have presented approaches in this domain. Therefore, the proposed architecture aims to bring together the main requirements and features of supervisory and control systems. It focuses mainly in the process supervision and monitoring aspects related to data acquisition, integration and processing, in order to automate tasks and support users with relevant information in abnormalities prediction, detection, diagnostics and recovery.

2 Related work

Several approaches reported in the literature apply the agent-based technology in different industrial domains covering manufacturing [4] and continuous process industry [6]. The MAS approaches have been used not only in academic research, but also in industrial projects [4-6], to solve problems ranging from the integration of heterogeneous and distributed data and components [7-8], to the automation of monitoring and control of processes and environments, in order to identify and recover from the occurrence of abnormalities [9-11].

As a complementary solution to agent-based technology, some approaches have also proposed the use of SOA [12-13], as well as semantic web technologies such as ontologies [7, 9, and 14]. Service-based architectures can further improve the system flexibility and interoperability, while ontologies allow semantic annotation of data in order to make it machine readable and interpretable also contributing to automate data searching, accessing, integration and processing.

Based on a literature survey a set of works which uses the MAS, SOA and semantic web technologies to develop supervisory and process control systems was analyzed. In these approaches were identified and analyzed the main requirements and features, which have been used as basis for the definition and specification of the agents and the infrastructure concerning the proposed MAS architecture. The following are discussed the main features identified.

2.1 Components and data integration

The integration of components (devices and legacy systems) and data is one of the primary requirements in complex environments, characterized by several distributed heterogeneous components. On such environments, the majority of components have their own interfaces which are generally incompatible with each other and work with different communication protocols and standards, contributing to the wide variety of data models and formats.

The agent-based approaches have been used to integrate and manage the different components and the produced data in order to improve and automate the execution of process and analysis tasks. This integration can be done at low-levels, among components, and at high-levels between components and high-levels systems. In high-levels ontologies have also been employed to further improve the integration and processing of data by machines.

It has been identified two main solutions which lead to the components integration: (1) agents implementing the required communication interfaces to access their resources and services, (2) adoption of service-oriented architectures, such as web services, where components are wrapped and provide their resources through standardized interfaces that can be easily accessed by agents and other systems. In both cases the data will be provided by standardized interfaces and can be semantically annotated, allowing agents to find and consume these data dynamically and automatically, thus minimizing the human intervention.

2.2 Monitoring and data analysis

The monitoring and data analysis is a feature related to the capability to detect, find, diagnose and predict the occurrence of abnormalities, by continuous monitoring and analysis of process measured variables. The monitoring process can be embedded into devices, if they support resource constraints, or by high-level monitoring systems. In the surveyed approaches were identified a set of agents employed to perform monitoring tasks. They can also perform the pre-processing and data analysis.

2.3 Planning and control

This feature is related to the management of the processes, comprising the tasks to determine and execute control plans, assessment and simulation of control actions, avoid and recover from abnormalities, and adapt to environmental changes. In the surveyed approaches these activities are performed by a set of agents who have knowledge about the processes and often capabilities for cooperation and negotiation with other agents.

2.4 User interface

This feature comprises the supervisory and control system GUIs, responsible for present to the users the functionalities and the resources of the application. These interfaces can be customized and developed with different technologies. Through the GUIs users can interact with the application in order to monitor and control all the related processes. Many approaches use a personal agent for managing these GUIs, which makes the interface between the users and the other agents of the MAS, retrieving and integrating the information in a customized manner.

2.5 Agents interaction

Another aspect indentified is the interaction between agents that is related to the way they communicate with each other to negotiate and exchange resources and services. In this context, the use of an agent communication language and common protocols are responsible for ensuring the agents inter-operation. The agent communication language defines the message structure while a shared vocabulary describes the message content. The vocabulary is usually defined by ontologies, used to represent the application domain concepts in such a way the message content can be understood by agents.

2.6 Multi-agents system organization

The system organization is related to the MAS architecture, i.e. the way the agents are distributed, the interaction among them and also between external components. In the surveyed approaches have been identified two types of MAS organization.

In the first approach, the agents do not have an explicit organization, instead they have specific roles and they are related according to their capabilities. In order to agents find and interact with each other, a widely used resource consists in a broker agent that provides a way for agents to register and search other agents which offers required resources and services. Thus, in these approaches all the application agents just need to know this agent.

In the other approach, the agents are organized hierarchically by layers. Each layer determines the roles played by the agents and hierarchical relationship among them. In this organization, the agents of the lower layer always offer services to the agents of the upper layers. In the supervision and control scenario, the agents in lower layers are responsible for data acquisition, in the intermediate layer they perform the data analysis and processing, and in the upper layer they present the results to the users.

3 Multi-agent architecture

Please The MAS architecture (Figure 1) comprises a set of agents organized in seven different roles that define their behaviors and capabilities for the management and automation of the processes supervision and control activities. Depending on the need and complexity of the application there can be multiple agents performing the same role, where each one is responsible for specific activities related to its role. These roles and their behaviors are presented in detail in Section 4.

The Figure 1 shows how the multi-agent architecture can be applied in a distributed industrial environment in order to realize the supervision and control of the processes. In Figure 1, lower layers comprise mainly existing physical devices, such as the process instrumentation (sensors, actuators and controllers), data acquisition or providers systems, control systems, and other systems used to retrieve data and control the process. All these components are distributed along industrial environments and can be accessible through standardized interfaces, such those defined by web services (e.g. following industrial standards as OPC-UA and DPWS [15]), or other specific proprietary interfaces.

Above the physical layer are the supervisory and control systems, deployed in local operational units and often connected to physical devices by a local network. In the architecture these systems comprise a set of agents. The agents can be physically distributed along the industrial environment, deployed on local or remote operational units. The Resource and Controller agents can also be deployed on monitoring and control devices if they support the resource constraints. In the same way, the User Interface agents can be deployed on user mobile devices in a manner they can connect and access the MAS of different operational units at any time through a local or even remote network connection.

Each local operational unit can have an application database (DB) to store the process historical data, retrieved from measurement devices, and information about supervision and control activities. The central remote operation unit can has a central database that keeps and integrates data from local units.

In the same way, each local unit can keeps a knowledge base (KB) responsible for store all the application knowledge, comprising the information about the processes such as devices, operational parameters, abnormalities that may occur, and also information about the agents and their capabilities. All this knowledge is defined by experts in the application domain and it is used by the agents to perform their tasks.

Ontologies can be used to support the semantic representation and sharing of the application knowledge. In this sense, domain ontologies should be specified to define a common vocabulary that describes the domain concepts such as devices, data, abnormalities, agents, and their respective properties. Application ontologies also should be defined to describe the processes, monitoring and control plans, supervised abnormalities and diagnostics. There are already several ontologies developed for different domains that can be used and adapted for this purpose.

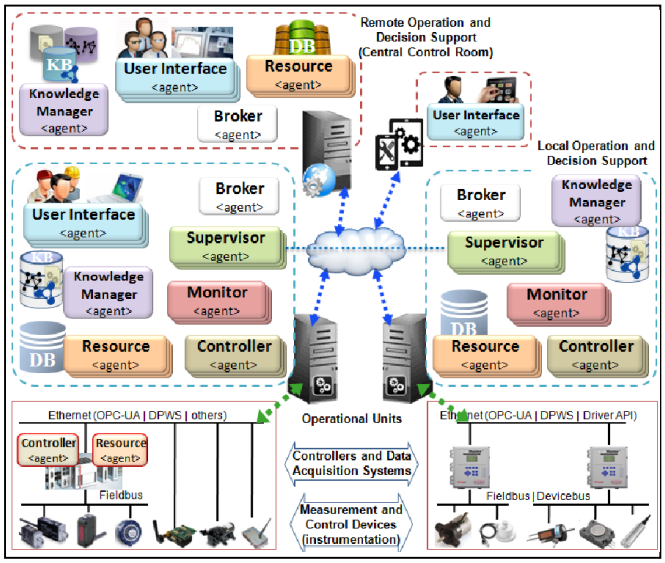


Fig. 1. Multi-agent architecture applied in a distributed application.

4 Agents specification

In the proposed multi-agent architecture, agents can perform seven roles, defined based on surveyed approaches. The roles determine the behaviors and capabilities of the agents (Figure 2). The same role can be performed by one or more agents. The following are described each defined role.

4.1 Resource

The main function that characterizes this role is the capability to manage the application data resources (historical and real-time measured process data), and make them available to the other agents of the MAS. Thus, agents that perform this role are responsible for retrieve, integrate and publish the data, hiding the complexity of the communication with the various components to the rest of the MAS. In addition, they must know the application metadata model, and use it to semantically annotate the data through domain ontologies.

In an application there can be several agents performing this role, each one responsible for provide access to a particular data resource or a set of specific data resources.

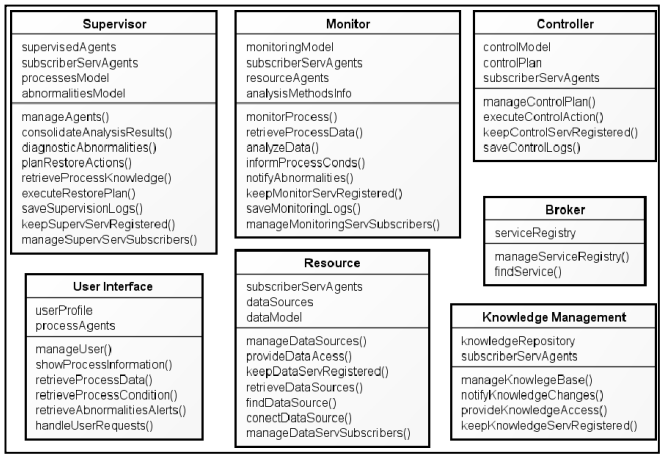


Fig. 2. Main capabilities defined for each role. It represents the knowledge and main tasks performed by agents.

4.2 Monitor

The main function that characterizes this role is the capability to continually analyze the process data, in order to determine the process or devices states and conditions, identifying or predicting the occurrence of abnormalities. Agents that perform this role must also generate alarms and notifying other agents about the detection of unwanted conditions that can bring the process to a critical situation which need to be treated.

The monitoring activity consists in the analysis of process variables, measured by sensors and retrieved by Resource agents. The method of analysis ranges from simple check of the variation of the measured values to the use of more complex methods, such as machine learning to find patterns in the behavior of these variables. These methods comprise algorithms which may be implemented by the agent itself through the use of third party code libraries. For data analysis can also be necessary techniques for pre-processing, feature extraction, prediction or noisy data avoidance.

In an application there can be several agents performing this role, each one responsible for monitoring one or a set of different process operations or devices in order to identify or predict their conditions.

4.3 Controller

The main function that characterizes this role is the capability to perform control actions on the process to modify its operational conditions. These agents can control devices directly through their access interfaces or through control systems. For this purpose, like Resource agents they should implement the communications interfaces with such systems and devices. In an application there can be several agents performing this role.

Agents performing this role can operate as device controllers, maintaining parameter values inside the set points, or just performing changes in the settings of the control parameters or even executing specified control actions. These actions can be organized into control plans, where each plan is defined for a given situation, in such a way that these agents are able to follow and execute the plans when needed or even through user requests. The control can be executed automatically when there aren't security constraints, otherwise control actions will be executed only under user request or agreement.

4.4 Supervisor

The main function that characterizes this role is the capability to coordinate Resource, Monitor and Controller agents in carrying out the tasks of supervision and control of processes. The agent that performs this role should be able to carry the necessary tasks to initialize and maintain the agents working properly, even when faced with changes in the environment.

The capabilities defined by this role also comprises integrating partial information about the abnormalities identified by Monitor agents, which monitoring different operations or devices in the process, using different analysis methods for monitoring the same or related operations. Under these conditions the Supervisor agent is responsible for consolidating the information from these agents and diagnosing the abnormalities. The diagnostics should be reported to the User Interface agents, and must provide decision support information such as the probable causes, consequences and corrective measures. Based on abnormalities diagnostics Supervisor agent should have

capabilities to determine the recovery plans and counter-measures. The recovery plans can be executed through requests to the Controller agents.

Depending on the complexity of the application, a single agent with this role is sufficient to manage the entire process. In other cases there can be several Supervisor agents responsible for managing a set of related processes or different operational units. In this case, Supervisor agents should have capabilities to negotiate between them in order to determine the local control actions or plans to be executed. This negotiation is necessary because in many cases the supervised processes or operational unit are related to each other in such way that a local action could affect other related or subsequent process. They also could interact between them to diagnose abnormalities related to other processes or when local knowledge is not enough.

Before execute any change in the process, this agent can also have the capabilities to simulate and assess the consequences and impacts of these actions. Thus, it can assist engineers in decision making.

4.5 User Interface

The main function that characterizes this role is the capability to retrieve, integrate and present the processes information to the users, concerning their interests. Agents that perform this role operate as an interface between the users and the MAS, interacting with other agents to answer the user requests and providing users with information about supervised processes.

For each new user in the application an agent must be created to manage its respective profile and preferences, related to the supervision and control process activities, presenting information and notifying the users according to their interests, acting in a personalized way and reducing the information overload.

4.6 Knowledge Manager

The main function that characterizes this role is the capability to manage the knowledge used in the application. Agents performing this role are responsible for managing the application knowledge bases, providing access to other agents. For this purpose, these agents must implement the inference mechanisms necessary to manipulate this knowledge. This agent can also monitor changes on the stored knowledge, for example, updates or inclusion of information about monitoring settings or new abnormalities. The changes should be reported to the Supervisor agents to perform the necessary actions to adjust the application settings to the new conditions.

Depending on the application size and distribution there can be various agents performing this role, each one responsible for managing the local knowledge.

4.7 Broker

The main function that characterizes this role is the capability to manage a registry of the services performed by agents in the MAS, in such a way that services can be found dynamically by other agents. So, the Broker agent must provide mechanisms for agents publish their services and search for other agents which provide required services. It means that all application agents must also have capabilities to register and search for services in the registry of Broker agents. The Broker agent must allows the agents of the MAS to find and consume, at runtime, a service provided by another agent without have to be directly bounded to this one, improving the system openness and flexibility. The services search and discovery can be improved through the use of ontologies to semantically describe them.

Depending on the size of the application and the distribution of the agents in different platforms, there can be more than one Broker agent. The Broker agent can also have an active behavior, reporting other agents about updates in the registered services and also about new agents and services in the MAS. This behavior contributes to the adaptation of agents to cope with changes that may occur in the MAS.

5 MAS organization

The MAS social organization determines the relationships between agents that perform the roles specified in the previous section. The agents interact to accomplish their tasks through messages for exchanging information and resources.

5.1 Agents interaction

The relationships among the agents are shown in Figure 3. In this figure the Broker agent does not appear related to any other agents for clarity, although it has relationships with all other agents of the MAS. All the application agents can register and search services through a Broker agent. Therefore, the messages exchanged between the Broker and other agents may be a request to register or search a service, or a response informing the list of agents that provide the sought service. In case of Broker agent has an active behavior, other agents can also subscribe for it to be informed about new agents that provide services and resources of interest.

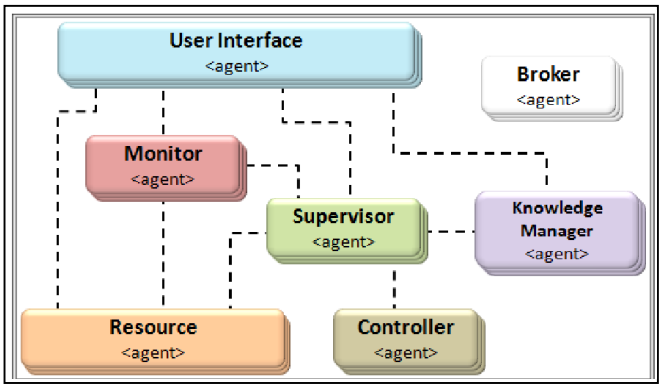


Fig. 3. Relationships between roles.

The User Interface agents interact with all other agents except with Controller. The interaction with the Resource, Monitor and Supervisor agents are basically to retrieve information about the supervised processes concerning user interests such as monitoring data, diagnosis of abnormalities and the decision support information that can help users to manage process, recover or avoid abnormalities. The interaction with the Knowledge Manager is to query or insert the information and knowledge of the processes, abnormalities and agents.

The Supervisor agent interacts with all other agents. The interaction with the Resource agents is to store the records of diagnostics, with the Monitor agents to request abnormalities monitoring tasks and be notified when abnormal situations are identified. The interaction with the Controller comprises requests to perform control actions. The interaction with the Knowledge Manager is to query the knowledge base to retrieve necessary information for the supervision and control of the processes, and also be notified when changes occur in application knowledge. The Supervisor agent can also interact with other Supervisors to negotiate control plans or diagnose abnormalities.

The Monitor interacts with the Resource agents to retrieve the data used in the monitoring tasks and also store the records generated during monitoring activities in the historical application database. The other relationships among the agents have already been presented above.

5.2 MAS functional and implementation aspects

In the industrial environments there are different interconnected units responsible for performing a set of operation. Based on this, the proposed MAS can be implemented for each local and remote operational unit. This organization ensures a local autonomy and independence between units for the process supervision and control. It also allows easily integration with new units, components and agents, without have to change the entire application infrastructure. The MAS must be developed and deployed on agent platforms, compliant with wide accepted standards.

In this MAS organization Resource and Controller agents most often performs only reactive tasks that do not require processing performance, making these agents suitable to be embedded into low-level devices. However, the Supervisor, Monitor and Knowledge Manager agents have more proactive and cognitive behaviors requiring more powerful platforms to be deployed. The User Interface agents can be deployed on user mobile devices and through them users can connect to any operational unit to manage the processes, concerning the required privileges. The GUIs can be implemented by these agents or by a separated application that interfaces with them.

In an application, for each MAS deployed on different operational units, there are one Broker agent, in order to allow the other agents to find each other dynamically. The Broker agents should know each other and share their service records or must have mechanisms to search agents registered on other Broker registry. These mechanisms are important because through them the Supervisor and User Interface agents can find and interact with agents from different MAS.

Likewise, for each MAS there are a Knowledge Manager agent to maintain a knowledge base, storing only the local knowledge, required for local process management, or all the application knowledge. Both strategies contribute to the autonomy of each MAS. Thus even during a communication loss among operational units, each MAS can continue working with their local resources. The Knowledge Manager agent of the central control room should synchronize all the application knowledge, in order to provide a global scenario and assist experts to manage the entire industrial environment. The knowledge should be defined and structured by domain experts in the development phases.

New agents can be developed and deployed in the application. When it happens, the new agent need to register their services through a Broker agent, from this moment the agent will be automatically integrated in the MAS, since other agents can find and interact with it. Also considering service-based technologies, such as OPC-UA and DPWS [15], a specific Supervisor agent can be in charge of search for new devices providing services in the network. Thus, when new devices are discovered, this agent can get more information about it with Knowledge Manager agent and initialize a new related agent or request for existing ones to wrap this device resources. For devices or other component that provide both data and control interfaces, only one agent can be deployed to manage it, so this agent will implement both Resource and Controller role capabilities.

6 Conclusions

In this paper was presented a multi-agent architecture that provides a reference model for the development of applications to process supervision and control. The architecture brings together the main requirements and features for process supervision and monitoring. They were identified in a literature survey and used as basis for the specification of the agent roles and MAS organization. Additionally, the architecture is specified in general terms and can be used to develop solutions for different domains.

In the multi-agent architecture the integration of agent-based, SOA and Semantic Web technologies allows the design of a more appropriate infrastructure to deal with the complexity of industrial environments, contributing to the development of flexible, open, dynamic and scalable systems. In this sense, the modularity and autonomy presented by the agents allow better management, control and integration of the various components. Standardized interfaces and mechanisms for publishing and discovering services offered by SOA, allow interoperability, reusability and dynamic integration of heterogeneous components. The representation of the application knowledge through the semantic Web technologies ensures the autonomy of agents to perform their tasks as well as to discover, integrate, and process the data in a seamless and automatic manner.

For future work we intend to develop applications based on proposed architecture, in order to validate and show how it can be developed and deployed. Further, improve the architecture specification and develop a framework to support the design of applications for process supervision and control.

7 References

[1] J. A. Hossack, J. Menal, S. D. J. McArthur e J. R. McDonald, "A multiagent architecture for protection engineering diagnostic assistance". Power Systems, IEEE Transactions on power systems, vol. 18, n. 2, pp. 639–647, 2003.

[2] M. Wooldridge, An Introduction to MultiAgent Systems, John Wiley & Sons, 2002.

[3] N. R. Jennings e M. Wooldridge, "Applications of Agent Technology". In Agent Technology - Foundations, Applications, and Markets, Springer-Verlag, pp. 3–27, 1998.

[4] P. Leitão, "Agent-based distributed manufacturing control: A state-of-the-art survey". Engineering Applications of Artificial Intelligence, vol. 22, n. 7, pp. 979–991, 2009.

[5] M. Pechouček e V. Marík, "Industrial deployment of multi-agent technologies: review and selected case studies". Autonomous Agents and Multi-Agent Systems, vol. 17, n. 3, pp. 397–431, 2008.

[6] M. Metzger e G. Polaków, "A Survey on Applications of Agent Technology in Industrial Process Control". IEEE Transactions on Industrial Informatics, vol. 7, n. 4, pp. 570–581, Nov. 2011.

[7] M. Purvis, S. Cranefield e M. Nowostawski, "A distributed architecture for environmental information systems". In Environmental Software Systems-Environmental Information and Decision Support, 2000.

[8] M. A. M. Capretz e M. C. Hryb, "Software integration using a dynamic wrapper agent". In WSEAS Conferences, 2005.

[9] L. Bunch, M. Breedy, J. M. Bradshaw, M. Carvalho e N. Suri, "KARMEN - Multi-agent monitoring and Notification for Complex Processes". In LNAI n. 3593, 2005.

[10] M. Cerrada, J. Cardillo, J. Aguilar e R. Faneite, "Agents-based design for fault management systems in industrial processes". Computers in Industry, vol. 58, n. 4, pp. 313–328, 2007.

[11] Y. S. Ng e R. Srinivasan, "Multi-agent based collaborative fault detection and identification in chemical processes". Engineering Applications of Artificial Intelligence, vol. 23, n. 6, pp. 934–949, 2010.

[12] J. M. Mendes, F. Restivo, P. Leitão e A. W. Colombo, "Injecting Service-Orientation into Multi-Agent Systems in Industrial Automation". Artifical Intelligence and Soft Computing, pp. 313–320, 2010.

[13] H. Zhou, J. Cao, C. Guo e J. Qin, "The architecture of intelligent distribution network based on MAS-SOA". Power and Energy Society General Meeting, 2010 IEEE, p. 1–6, 2010.

[14] S. Natarajan, K. Ghosh, R. Srinivasan, "An ontology for distributed process supervision of large-scale chemical plants". Computers & Chemical Engineering, v. 46, p. 124–140, Nov. 2012.

[15] G. Candido, F. Jammes, J. Oliveira, A. W. Colombo, "SOA at device level in the industrial domain: Assessment of OPC UA and DPWS specifications". In 8th IEEE International Conference on Industrial Informatics (INDIN 2010), pp. 598–603, jul. 2010.

SESSION

FUZZY LOGIC AND SYSTEMS AND APPLICATIONS

Chair(s)

TBA

Processing of ICARTT Data Files Using Fuzzy Matching and Parser Combinators

Matthew T. Rutherford[1], **Nathan D. Typanski**[2], **Dali Wang**[3], **Gao Chen**[4]
[1,2,3] Department of Physics, Computer Science & Engineering,
Christopher Newport University, Newport News, VA, United States
[4] Science Directorate, NASA Langley, Hampton, VA, United States

Abstract— *In this paper, the task of parsing and matching inconsistent, poorly formed text data through the use of parser combinators and fuzzy matching is discussed. An object-oriented implementation of the parser combinator technique is used to allow for a relatively simple interface for adapting base parsers. For matching tasks, a fuzzy matching algorithm using Levenshtein distance calculations is implemented to match string pairs, which are otherwise difficult to match due to the aforementioned irregularities and errors in one or both pair members. Used in concert, the two techniques allow parsing and matching operations to be performed which had previously only been done manually.*

Keywords: fuzzy matching, Levenshtein distance, parser combinator

1. Introduction

Enabling a computer to recognize and parse irregular and poorly formatted data can be handled in multiple ways. Many traditional parsing approaches include an initial lexical analysis phase, in which a given character sequence is tokenized, followed by a parse phase [1], [2]. These techniques suffice for compilers and other common parsing applications, but when faced with inputs in which unique edge cases are common, the grammar and parse rules must be adapted often in order to properly handle this kind of input. We found the lexxer-parser combination approach sluggish and cumbersome when novel errors in the input were arising frequently throughout the parsing process.

Parser combinators offered a viable solution, allowing the consolidation of multiple simple parsers to form a series of conglomerate parsers, which are then able to collectively address irregular, complex input. For our purposes, the combinator approach proved faster compared to frequently writing and rewriting lexxers and parsers when our program encountered input that would not parse properly. We could write a parser quickly, test it on the new input, and rewrite it as needed.

In order to produce the intended output, our program needed to reference an external knowledge base. However, this knowledge base was not guaranteed to correspond perfectly with the input data. Due to this varying degree of correspondence, fuzzy matching was used in multiple stages of this process to determine which portions of the knowledge base apply. The program had to ensure that no overlapping matches occurred: it needed a way to perform imperfect string matching such that the best one-to-one mapping with the knowledge base could be found.

1.1 The Extended ICARTT Data Format Standard

The International Consortium for Atmospheric on Transport and Transformation (ICARTT), a scientific airborne research group, created the Extended ICARTT Data Format Standard [3] in 2004 to provide a well-defined, text-based file format for atmospheric data to help ensure consistency across the several hundred files produced per data collection mission. ICARTT files are broken into two main sections: a data section and a corresponding metadata section. The metadata section is where all contextual information corresponding to the data section is given. This portion of the file is made up of several subsections. The data section is where the actual data recorded by the various aircraft-mounted instruments as well the corresponding time data are given. Raw data are given as either comma delimited or space delimited columns of decimal values. Each data column corresponds to a variable given in the above metadata section and can be several hundred to several thousand lines long, depending on the length of the data collection period for a given file.

1.2 Purpose of Research

The decentralized structure of the ICARTT data collection campaigns resulted in errors and inconsistencies—some quite subtle—among data files. Thousands of raw data files have been produced, which need to be edited and reformatted in order to meet the Extended ICARTT File Format Standard before being added to the central Toolset for Airborne Data (TAD) archive.

In the past, reformatting of ICARTT files had been done manually, file by file. The purpose of the program described here is to automate this process as much as possible in order to quickly reformat large sets of raw ICARTT files with little-to-no manual effort or editing. To this end, the aforementioned parser combinator and fuzzy matching algorithms were implemented.

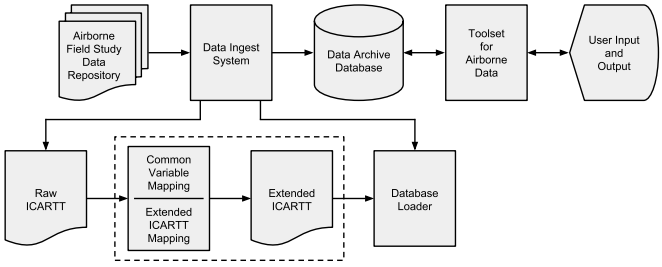


Fig. 1: Outline showing the process of converting a distributed set of raw ICARTT data files to the extended format standard. The role of our program is enclosed within the dotted box.

2. Related Literature

2.1 A Flexible Parser with Backtracking

In his 1963 paper [4], E.T. Irons describes his concept of an adaptive parsing schema, which could be used for pattern recognition as well as computer code correction and optimization. He points out weaknesses of the parsers in ALGOL and FORTRAN compilers at that time. Weak error handling being his main concern, he cites the frequent inability of compilers to correct and move past errors, specifically flawed object strings in the code. The schema he proposes would be capable of working through errors by using a more dynamic parsing structure.

He presents an approach which uses a slightly simplified equivalent of Backus-Naur Form (BNF) to describe the strings handled by the parser. Based on this pre-defined BNF-like grammar, the parser would work through and correct strings. When the parser encounters a string object where multiple parse sequences are applicable, it would attempt all possible sequences until one worked, and then move on to the next string. If, however, none of the attempted sequences work, the parser would backtrack to a previous point in the parsing sequence. This process of attempting different parse paths repeats until the parser works its way through the input.

Despite the age of Irons' paper, it is quite relevant to our chosen approach. He discusses an early conception of a recursive descent parser. Our program's parsing schema derives much of its functionality from recursive descent parsers, using varying combinations of parsers to work through the input.

2.2 USGS Metadata Pre-Parser

Peter N. Schweitzer of the United States Geological Survey addresses a somewhat similar parsing problem to our own with his Chew and Spit (cns) program [5]. The cns module precedes the primary parser module, mp [5], in cases where the input, a metadata file, is initially too poorly formatted to be parsed by mp.

When working through a metadata file, cns forms a parse tree, which it populates with parsed metadata elements based on user-defined element aliases as well as its own inference. These aliases are held in the alias file, which cns uses as a simple expert system. This alias file performs a somewhat similar role to that of the Extended Variable Map (see Section 3.2) in our program's parse and match system.

3. Description of Methodology

Our chosen approach had two distinct phases: 1. parsing and correcting raw file metadata contents based on the Extended ICARTT Data Format Standard, and 2. using fuzzy matching to properly identify and define variables from raw data files based on a corresponding variable map. We built the program in Python 3, which has multiple robust artificial intelligence libraries [6], [7] and parsing libraries [8], [9] available.

3.1 Using Parser Combinators

Due to the often irregular and poorly structured data found in many raw ICARTT files, a consistent, static grammar was impossible to define from the specification alone. Writing a parser capable of understanding the majority of these files required fast, iterative development. The maintenance costs of a separate lexical phase or library of dense regular expressions were unacceptable.

Ideally, we wanted a library that offered one of the benefits of monadic parsers [10], [11]: sufficient parser combinators that eliminate the need for a separate lexical phase. Pyparsing [8] is one such library. It approximates the compositional nature of monadic parsers in an object-oriented context. For example, `many1` from [11] corresponds with `OneOrMore()` in Pyparsing, `sepby1` with `delimitedList()`, and so forth for a number of other useful combinators.

The Pyparsing ParseResults object [8] offers a versatile data structure for the output of the parsing process, as it can be treated as a either a list or a dictionary containing parsed strings. These features gave our program the necessary level of flexibility to handle poorly formatted, inconsistent input from the raw ICARTT files. After a file has been parsed, it is rebuilt according to the specifications in the Extended ICARTT File Format Standard. For some parts of the file, this process is straightforward; though in other places, like the variable mapping described in Section 3.2, complicated inference techniques and the ability to reference a knowledge base is necessary.

3.2 Variable Mapping with Fuzzy Matching

The metadata subsection containing variables adheres to different specifications for content and formatting than the other metadata subsections. ICARTT variables proved to be particularly difficult to parse, as they were prone to subtle and irregular errors, and required our program to reference an external knowledge base for verification of content. As such,

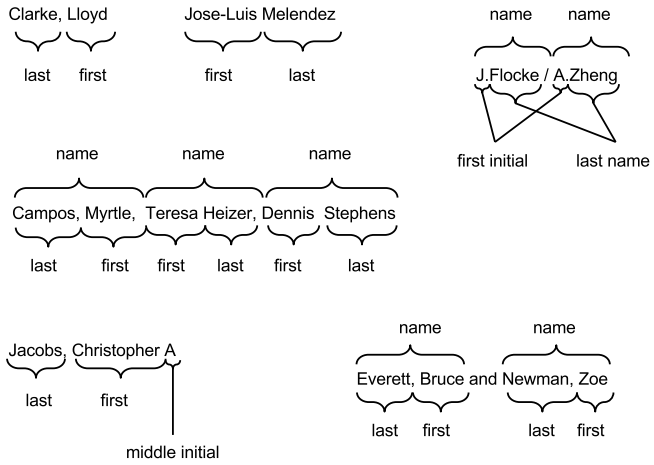


Fig. 2: Common input sequences for the "PI Names" field. These inputs can be automatically parsed and converted into a standardized format.

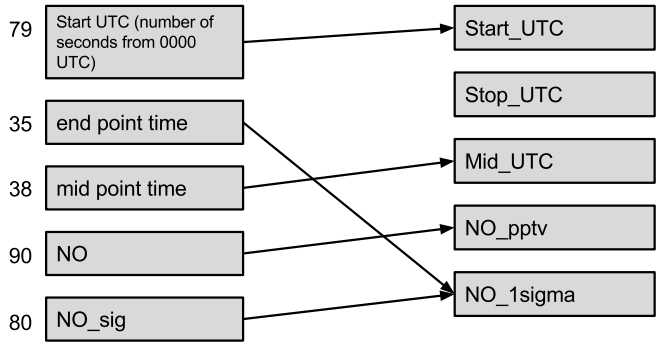


Fig. 3: Column header/variable info pairing, first pass. Confidence is shown to the left of variable info. Note contention for NO_1sigma.

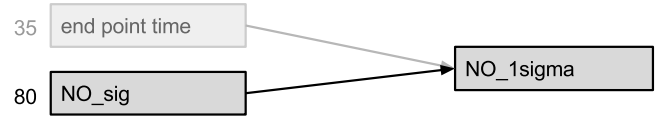


Fig. 4: NO_sig steals ownership of NO_1sigma because it has a greater confidence rating.

the implementation of a more intelligent, adaptable algorithm to properly handle the variable subsection was required.

ICARTT files contain two distinct variable types: time variables and dependent variables. Ideally, both have certain discernible traits denoting their type, but this is often not the case with raw files. In order to separate time variables from dependent variables, we use a configuration file against which fuzzy matching is performed to categorize the variable.

A corresponding Extended Variable Map was created as the aforementioned knowledge base for each of the ICARTT mission file sets. These maps contained all the information a human would normally need to correct the dependent variables in a file. All time variables are considered "independent variables" for these purposes and thus not included in the maps. Each row in an Extended Variable Map corresponds one-to-one to a variable in an ICARTT file, but both the raw files and the Variable Maps were created manually, resulting in irregularities stemming from human error. For example: the extended map and variable information lines often contained different names for the same variable, but the extended map would only list one of the two, or might not perfectly match either—perhaps using underscores where there could be hyphens, or zeros ("0") where there should be the letter "O". Thus, the information lines had to be matched to column headers before the extended map could be applied.

3.2.1 Fuzzy Match Pairing Algorithm

The Fuzzywuzzy Python library [6] gave our program the necessary matching capabilities to make it far more sensitive to subtle and odd edge cases resulting from human error. The library centers its operations around a fairly simple implementation of Levenshtein distance [12] to find the closest matches between pairs of strings.

The following algorithm is used to find the optimal one-to-one fuzzy matching between two lists ("left" and "right") of strings:

1) If the right list is empty, pair each element in the left list with the null element.
2) For each element in the left list, find its best confidence match in the right list.
3) If any two or more elements in the left list matched to the same element in the right list, then choose the one with the greater confidence.
4) If an element in the right list only has one match in the left list, then pair those two elements.
5) If there are remaining elements that are unmatched in the left list, then recursively repeat this procedure on those lists.

Figures 3–5 illustrate the matching algorithm.

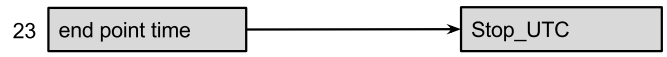


Fig. 5: Final step. The fuzzy match algorithm gets called again this time. Even though the confidence is low, this is the only match left—and it happens to be the right one. This is the recursive step. In this case the recursive step just instantly hits a base case. In more complex matches it might recurse a few times.

3.3 Results

Out of the 723 ICARTT files to be processed from the INTEX-A mission, our program was able to successfully parse and automate most of the formatting cleanup of 706 of them. Success in variable mapping varied depending on the quality of the variable names chosen. In cases where the name listed as the raw file's variable name, the variable info name, and its corresponding header name were all somewhat similar, the fuzzy matcher had little problem matching all of these together. In some cases, adding information to our knowledge base, e.g., a new line in the extended map or a "time variable name" listing in the configuration files, could be enough for the rest of the file cleanup to be fully automated.

These results are extremely promising, and a huge advantage over the previous manual cleanup of data files. Automated matching of variables, thanks to the fuzzy matching algorithms and a sufficiently advanced parser, meant even complex pairings of variables and column headers like "Carbon dioxide mixing ratio (ppmv)" and "CO2(ppmv)" could be performed automatically. Where this was not the case, and some manual editing was needed to guide the code toward the correct output, the speed benefits were still substantial compared to doing the same task by hand.

Overall, the program serves as an example of how artificial intelligence techniques and somewhat modern parsing techniques can automate a real, practical process previously performed only by manual human work.

4. Discussion and Future Work

Our program has successfully processed nearly 1,500 raw ICARTT data files with relatively little manual editing needed. Moving forward, we plan to continue enhancing the parsing and matching functionality for progressively increased automation and reliability in order to further minimize the need manual editing of raw files. Along with the continued ICARTT file conversion effort, we plan on exploring automation techniques for producing the knowledge base used here as well.

5. Acknowledgements

The authors would like to thank Lindsay Parker, Amanda Benson, and Aubrey Beach at NASA Langley Research Center, as well as Elliot Rieflin, Antonio Siochi, Clare Maliniak, and Anton Riedl at Christopher Newport University for their contributions, feedback, and support throughout the project.

References

[1] M. E. Lesk and E. Schmidt, *Lex: A Lexical Analyzer Generator*, 1975.

[2] S. C. Johnson, *Yacc: Yet Another Compiler Compiler*. Bell Laboratories Murray Hill, NJ, 1975, vol. 32.

[3] A. Aknan, G. Chen, J. Crawford, and E. Williams, *ICARTT File Format Standards V1.1*, 2013.

[4] E. T. Irons, "An error-correcting parse algorithm," *Communications of the ACM*, vol. 6, no. 11, pp. 669–673, 1963.

[5] P. E. Schweitzer, *A Pre-parser for Formal Metadata*, 2012. [Online]. Available: `http://geology.usgs.gov/tools/metadata/tools/doc/cns.html`.

[6] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, *et al.*, "Scikit-learn: machine learning in python," *The Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[7] A. Cohen, *Fuzzy String Matching in Python*, 2014. [Online]. Available: `https://github.com/seatgeek/fuzzywuzzy/`.

[8] P. McGuire, *Getting Started with Pyparsing*. O'Reilly Media, Inc., 2007.

[9] A. Vlasovskikh, *Funcparserlib: Recursive Descent Parsing Library for Python Based on Functional Combinators*, 2013. [Online]. Available: `http://code.google.com/p/funcparserlib/`.

[10] G. Hutton and E. Meijer, "Functional pearl: monadic parsing in haskel," *Journal of Functional Programming*, vol. 8, no. 4, pp. 437–444, 1998.

[11] D. Leijen and E. Meijer, "Parsec: direct style monadic parser combinators for the real world," Technical Report UU-CS-2001-27, Department of Computer Science, Universiteit Utrecht, Tech. Rep., 2001.

[12] V. I. Levenshtein, "Binary codes capable of correcting deletions, insertions and reversals," in *Soviet Physics Doklady*, vol. 10, 1966, p. 707.

Optimal Damping of Oscillation in Multi-machine Power System by Renewable Energy Effects

H. A. Shayanfar *
Center of Excellence for Power System Automation and Operation, College of Elect. Eng., Iran University of Science and Technology, Tehran, Iran

O. Abedinia
Electrical Engineering Department, Semnan University, Semnan, Iran

N. Amjady
Electrical Engineering Department, Semnan University, Semnan, Iran

hashayanfar@yahoo.com, oveis.abedinia@gmail.com, n_amjady@yahoo.com

Abstract— *In this paper a robust Power System Stabilizer (PSS) is proposed based on inverse additive perturbation in a power system with wind farms. For this purpose a new algorithm of Harmony Search (HS) with Fuzzy Mechanism (FM) is proposed to achieve robustness of proposed controller strategy which is formulated based on an enhancement of system robust stability margin. To demonstrate the capability of proposed technique, ten machine 39 buses power system of New England is presented as a case study. The achieved results demonstrate the superiority of proposed technique in comparison of conventional PSS.*

Keywords*:* HS, Fuzzy controller, PSS, Multi-machine.

NUMENCLUTURE

δ	Rotor angle
ω	Rotor speed
P_m	Mechanical input power
P_e	Electrical output power
E'_a	Internal voltage behind x'_d
E_{fd}	Equivalent excitation voltage
T_e	Electic torque
T'_{do}	Time constant of excitation circuit
K_A	Regulator gain
T_A	Regulator time constant
v_{ref}	Reference voltage
v	Terminal voltage

I. Introduction

In power system, the Low Frequency Oscillations (LFOs) are related to the small signal stability are detrimental to the goals of maximum power transfer and power system security. Once the adjustment of using damper windings on the generator rotors and turbines to control these oscillations was found to be satisfactory, the stability problem was thereby disregarded for some time [1]. But, as power systems began to be operated closer to their consistency limits, the weakness of a synchronizing torque among the generators was distinguished as a major cause of system instability [2]. Automatic Voltage Regulators (AVRs) helped to improve the steady-state stability of the power systems, but transient stability started a concern for the power system operators. The addition of a supplementary controller into the control loop, such as the introduction of Power System Stabilizers (PSSs) to the AVRs on the generators, supplies the means to reduce the inhibiting effects of low frequency oscillations [3].

In the recent years, renewable electrical energy such as wind power generations, have achieved a significant level of penetration in the power systems due to infinite availability and low impact to environment. However, wind power generation is staggering in nature. Matching the supply and the demand is often a problem. The power output fluctuations from wind power generations cause a problem of low frequency oscillation, deteriorate the system stability and make the power system operation more difficult. The power frequency and the tie-line power deviations persist for a long term. In this status, the governor system may no longer be capable to absorb the frequency fluctuations due to its slow response [4].

Actually, several plants prefer to employ conventional lead-lag structure PSSs, due to the ease of online tuning and reliability [5]. However, the revenue of these controllers doesn't have good behavior in different load conditions. So, different intelligent algorithms have been introduced to optimal tuning of the PSSs parameters such as Ant Colony (AC) [6], Genetic Algorithm (GA) [7], Particle Swarm Optimization [8], Artificial Bee Colony [9] and etc.

Accordingly, this paper proposes a new algorithm of Harmony Search (HS) to solve the mentioned problem.

* Correspanding Author. E-Mail Address: hashayanfar@yahoo.com (H. A. Shayanfar)

HSA is a recently developed powerful evolutionary algorithm, inspired by the improvisation process of musicians, for solving single or multi-objective optimization problems. In the proposed technique, each musician plays a note for finding a best harmony all together [10-11].

The remaining parts of the paper are organized as follows. The second section formulates the power system modeling. Section three proposes the controller structure and the HAS is described in next section based on its application for the solution of the low oscillation problem with wind. At the end the achieved numerical results from New England power system are presented. Section six concludes the paper.

II. Problem Statement

In this paper a multi-machine power system is considered as a test case where the third order model is presented in [12]. Actually, the proposed power system consists of ten generators and the electrical and mechanical part of i^{th} generator is modeled as follow:

$$\dot{\delta}_i(t) = w_i(t) - w_0 \tag{1}$$

$$\dot{w}_i(t) = \frac{w_0}{M_i}(P_{mi} - P_{ei}(t)) - \frac{D_i}{M_i}(w_i(t) - w_0) \tag{2}$$

$$\dot{E}_{qi}(t) = \frac{1}{T'_{doi}}(E_{fi}(t) - E_{qi}(t)) \tag{3}$$

$$E_{qi}(t) = E'_{qi}(t) + (x_{di} - x'_{di}).I_{di}(t) \tag{4}$$

$$V_{ti}(t) = \frac{1}{x_{dsi}}[x_{si}^2 E_{qi}^2 + V_{si}^2 x_{qi}^2 + 2x_{si}x_{di}x_{dsi}P_{ei}\cot\delta_i]^{1/2} \tag{5}$$

$$P_{ei}(t) = \sum_{j=1}^{n} E'_{qi}(t)E'_{qj}(t)(B_{ij}\sin(\delta_{ij}(t)) + G_{ij}\cos(\delta_{ij}(t))) \tag{6}$$

$$Q_{ei}(t) = \sum_{j=1}^{n} E'_{qi}(t)E'_{qj}(t)(G_{ij}\sin(\delta_{ij}(t)) + B_{ij}\cos(\delta_{ij}(t))) \tag{7}$$

$$I_{di}(t) = \sum_{j=1}^{n} E'_{qj}(t)(G_{ij}\sin(\delta_{ij}(t)) - B_{ij}\cos(\delta_{ij}(t))) = \frac{Q_{ei}(t)}{E'_{qi}(t)} \tag{8}$$

$$I_{qi}(t) = \sum_{j=1}^{n} E'_{qj}(t)(B_{ij}\sin(\delta_{ij}(t)) - G_{ij}\cos(\delta_{ij}(t))) = \frac{P_{ei}(t)}{E'_{qi}(t)} \tag{9}$$

$$E_{qi}(t) = x_{adi}I_{fi}(t) \tag{10}$$

Also, it can be presented after mathematical transformers where, ΔP_e, Δw and ΔV_t are quantities;

$$\begin{bmatrix} \Delta\dot{P}_{ei} \\ \Delta\dot{w}_i \\ \Delta\dot{V}_{ti} \end{bmatrix} = \begin{bmatrix} \frac{S_{Ei} - S_{Vi}}{T'_{di}S_{Vi}} & S'_{Ei} & -\frac{R_{Vi}S_{Ei}}{T'_{di}S_{Vi}} \\ -\frac{w_{0i}}{H_i} & -\frac{D_i}{H_i} & 0 \\ \frac{S_{Ei} - S_{Vi}}{T'_{di}R_{Vi}S_{Vi}} & \frac{S'_{Ei} - S_{Vi}}{R_{Vi}} & -\frac{S_{Ei}}{T'_{di}S_{Vi}} \end{bmatrix} \begin{bmatrix} \Delta P_{ei} \\ \Delta w_i \\ \Delta V_{ti} \end{bmatrix} + \begin{bmatrix} \frac{R'_{Ei}}{T_{d0i}} \\ 0 \\ \frac{R'_{Ei}}{T_{d0i}R_{Vi}} \end{bmatrix} \Delta E_{fdi}$$

Where,

ΔP_{ei} is the state deviation in generator electromagnetic power for the i^{th} subsystem

$\Delta\omega_i$ is the state deviation in rotor angular velocity for the ith subsystem,

ΔV_{ti} is the state deviation in the terminal voltage of the generator for the ith subsystem

$$S_{Ei} = \frac{E_{qi}U_{si}}{X_{d\Sigma i}}\cos\delta_i + U_{si}^2\frac{X_{d\Sigma i} - X_{q\Sigma i}}{X_{d\Sigma i}X_{q\Sigma i}}\cos 2\delta_i$$

$$S'_{Ei} = \frac{E'_{qi}U_{si}}{X'_{d\Sigma i}}\cos\delta_i + U_{si}^2\frac{X'_{d\Sigma i} - X_{q\Sigma i}}{X'_{d\Sigma i}X_{q\Sigma i}}\cos 2\delta_i \tag{11}$$

$$R_{Ei} = \frac{U_{si}}{X_{d\Sigma i}}\sin\delta_i$$

$$R'_{Ei} = \frac{U_{si}}{X'_{d\Sigma i}}\sin\delta_i \tag{12}$$

$$S_{Vi} = S_{Ei} - R_{Vi}\frac{\partial U_{ti}}{\partial\delta_i}, R_{Vi} = S_{Ei}/\frac{\partial V_{ti}}{\partial E_{qi}}$$

A. Mechanical Oscillation Frequency

The linearized rotor motion equation for synchronous generator can be described as:

$$T_J\frac{d\Delta w}{dt} = \Delta T_m - \Delta T_e - \Delta T_D \tag{13}$$

Where,

ΔT_m is the mechanical input torque

ΔT_e is the electromagnetic torque and $\Delta T_e = K_1\Delta\delta + K_2\Delta\delta$,

By neglecting the $K_2\Delta E'_q$ the formulation can be described as; $\Delta T_e = K_1\Delta\delta + \Delta T_D$

D is the natural damping constant

Accordingly the above equation after Laplace transformer and $\Delta w = s\Delta\delta/w_0$ can be described as;

$$T_J\frac{s^2\Delta\delta}{w_0} = -K_1\Delta\delta - D\frac{s\Delta\delta}{w_0} \tag{14}$$

Which can be described as;

$$T_J s^2 + Ds + w_0K_1 = 0 \tag{15}$$

or

$$s^2 + 2\zeta_n w_n s + w_n^2 = 0 \tag{16}$$

Accordingly, we can achieve the following equation from above equations;

$$\zeta_n = D/2\sqrt{w_0T_JK_1}$$
$$w_n = \sqrt{w_0K_1/T_J} \tag{17}$$

Where,

ξ_n is the damping factor

w_n is the undamped mechanical oscillation frequency

Usually the value of D is small, so the system damping is low. In order to enhance the system damping for suppressing the low frequency oscillation, the positive damping term is added by PSS.

B. Power System Model

A 10-machine 39-bus power system with wind farms is considered as a test case for this paper which is presented in Fig. 1. To assess the effectiveness and robustness of the proposed method over a wide range of loading conditions, different operation conditions are considered. Details of the system data and operating condition are given in Ref. [13]. The proposed power system is divided to four area with connected wind farm as shown in Fig. 1.

C. Wind power model

It is clear that the output power of wind generator depends on wind velocity. The wind speed model chosen in this study consists of four-component model [14], that is defined as;

$$V_W = V_{WB} + V_{WG} + V_{WR} + V_{WN} \quad (18)$$

Where,

$WB\ V$ = Base wind velocity

$WG\ V$ = Gust wind component

$WR\ V$ = Ramp wind component

$WN\ V$ = Noise wind component

The base wind velocity component is represented in literature as: $V_{WB} = K_B$

Where, K_B is a constant and assumed to presenting the wind power. The gust wind velocity can be defined as;

$$V_{WG} = \begin{cases} 0 & t < T_{1G} \\ V_{\cos} & T_{1G} < t < T_{1G} + T_G \\ 0 & t > T_{1G} + T_G \end{cases} \quad (19)$$

Where,

$$V_{\cos} = (MAXG/2)\{1 - \cos 2\pi[(t/T_G) - (T_{1G}/T_G)]\} \quad (20)$$

MAXG is the gust peak

T_G is the gust period

T_{1G} is the gust starting time

(1-cosine) gust is an essential component of wind velocity for dynamic studies

The ramp wind velocity component is described as:

$$V_{WG} = \begin{cases} 0 & t < T_{1R} \\ V_{ramp} & T_{1R} < t < T_{2R} \\ 0 & t > T_{2R} \end{cases} \quad (21)$$

Where,

$$V_{ramp} = MAXR[1 - (t - T_{2G})/(T_{1R} - T_{2R})] \quad (22)$$

MAXR is the ramp peak

T_{1R} is the ramp start time

T_{2R} is the ramp maximum time

This component may be used to approximate a step change with $T_{2R} > T_{1R}$. Also, the random noise component can be described as:

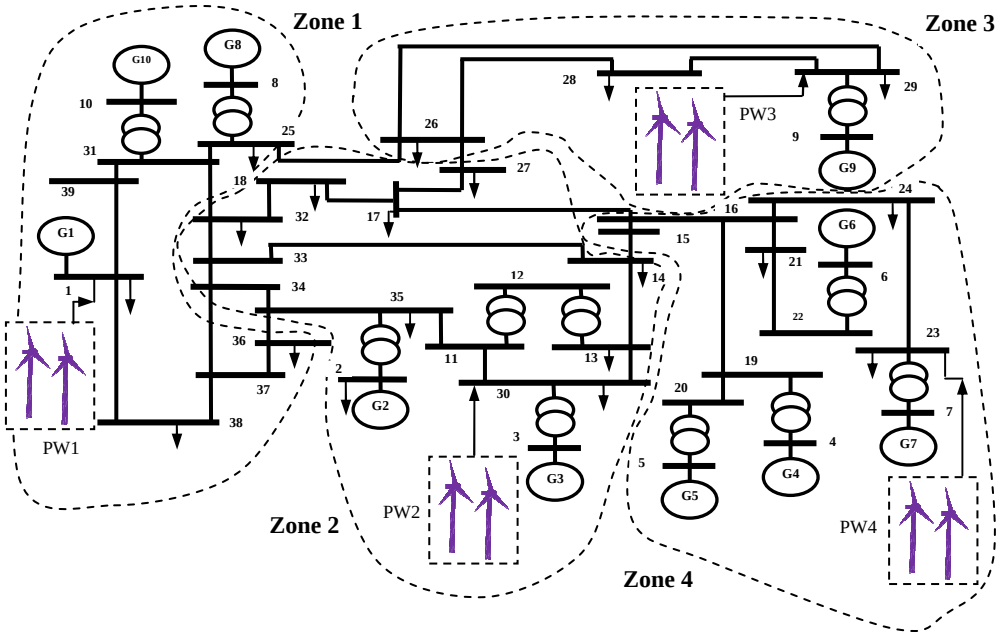


Figure 1. Structure of power system with wind farms

$$V_{WN} = 2\sum_{i=1}^{N}[S_V(\omega_i)\Delta\omega]^{1/2}\cos(\omega_i t + \phi_i)t < 0 \quad (23)$$

Where,

$$\omega_i = (i - 1/2)\Delta\omega \quad (24)$$

ϕ_i is a random variable with uniform probability density on the interval 0 to 2π and the spectral density function can be defined as:

$$S_V(\omega_i) = \frac{2K_N F^2[\omega_i]}{\pi^2[1+(F_{\omega_i}/\mu\pi)^2]^{4/3}} \quad (25)$$

Where, *KN* is the surface drag coefficient which is considered 0.004 and F is turbulence scale which is considered 2000, and μ is the mean speed of wind at reference height. Various study have shown that values of N=50, and $\Delta\omega$ = 0.5-2.0 rad/s provide results of excellent accuracy.

D. Characteristic of wind generator output power

The output power of studied wind generator is expressed by a nonlinear function of the power coefficient *Cp* as function of blade pitch angle, β, and tip speed ratio, γ. The tip speed ratio is presented as follow:

$$\lambda = \frac{R_{blade}\omega_{Blade}}{V_W} \quad (26)$$

The power coefficient can be presented as:

$$C_P = (0.44 - 0.0167\beta)\sin\left[\frac{\pi(\lambda-3)}{15-0.3\beta}\right] - 0.0184(\lambda-3)\beta \quad (27)$$

At the end, the output of mechanical power for wind generator is:

$$P_W = \frac{1}{2}\rho A_r C_P V_W^3 \quad (28)$$

Where, ρ (=1.25 kg/m^3) is the air density and A_r (=1735 m^2) is the swept area of blade.

III. Fuzzy Logic Controller

In classical control, the value of control is determined in relation to a number of data inputs using a set of equations to express the entire control process. Expressing human experience in the form of a mathematical formula is a very difficult task, if not an impossible one. Accordingly, Fuzzy Logic Controller (FLC) provides a simple tool to interpret this experience into reality [15]. FLCs are rule-based controllers with resembles structure with a knowledge based controller except that the FLC utilizes the principles of the fuzzy set theory in its data representation and its logic. To reduce fuzzy system effort cost, this paper proposed an intelligent technique for optimal tuning of fuzzy PID controller's rules.

IV. Multi-Objective Harmony Search Algorithm

A. Review of Harmony Search Algorithm

The brief procedure steps of harmony search for solving optimization problems are described in five steps as:

This procedure can be described as Fig. 2.

Step 1: Identify objective function and Equality & Inequality constraints by using:

$$\begin{aligned} &Minmize: \{f(x), x \in X\} \\ &s.t \\ &g(x) \geq 0 \\ &h(x) = 0 \end{aligned} \quad (29)$$

Where, f (x) is the objective function. X_i is the feasible set. x_i is the random choosing parameter. G(x) is the inequality constraint. h (x) is the equality constraint [10].

Step 2: Initialize Harmony Memory (HM) [11].

$$HM = \begin{bmatrix} x_1^1 & x_2^1 & \dots & x_{N-1}^1 & x_N^1 \\ x_1^1 & x_2^2 & \dots & x_{N-1}^2 & x_N^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_1^{HMS-1} & x_2^{HMS-1} & \dots & x_{N-1}^{HMS-1} & x_N^{HMS-1} \\ x_1^{HMS} & x_2^{HMS} & \dots & x_{N-1}^{HMS} & x_N^{HMS} \end{bmatrix} \frac{\Delta y}{\Delta x} \quad (30)$$

Step 3: Harmony memory initialization

The New Harmony Improvisation is applied in this step and consists of two stages of HMCR and PAR in literature as;

Step 3.1:.1 Harmony Consider Rated (HMCR)

$$x_i^{'} \leftarrow \begin{Bmatrix} x_i^{'} \in \{x_i^1, x_i^2, \dots, x_i^{HMS}\}(HMCR) \\ x_i^{'} \in X_i(1-HMCR) \end{Bmatrix} \quad (31)$$

Where $x_i^{'}$ is new value of x_i and *HMCR* is probability of choosing $x_i^{'}$ which PR means the probability function.

Step 3.2: pitch adjust rate (PAR)

$$x_i^{'} \leftarrow \begin{Bmatrix} Yes, \Pr(PAR) \\ No, \Pr(1-PAR) \end{Bmatrix} \quad (32)$$

Where, PAR is probability to shift x_i

$$x_i^{'} \leftarrow x_i^{'} \pm rand() \times bw \quad (33)$$

Where b*w* is range of x_i, rand is random number during 0-1.

Step 4: Update HM and check the stopping criterion Find value of f ($x_i^{'}$) from substitute $x_i^{'}$ [16].

Step 5: To check the stopping criterion, set the NI (Number of iteration) before begins to run the simulation; HS can stop calculation instantaneously when NI is reached.

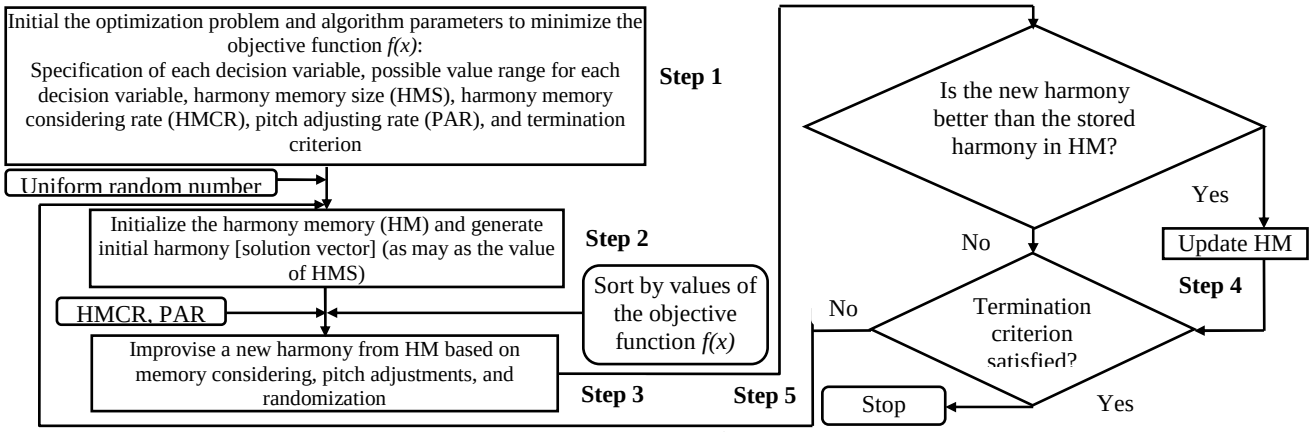


Figure 2. Flowchart of HSA

V. Simulation Results

In this paper, the HSA-FPSS is proposed, which combines the advantage of the HS and fuzzy control techniques to achieve good robust performance. It should be mentioned that obtaining the optimal decision-making logic for the proposed HS fuzzy control strategy is very important to achieve the desired level of robust performance [9], because it is a computationally expensive combinatorial optimization problem. Usually, the rule-base sets are determined by experience and control knowledge of a human expert. However, experts may not always be available and even when available it is a trial-and-error process that takes much time and cost [9]. The results of the fuzzy rule-base sets are listed in Tables 1-3. In the proposed rule-base optimization problem, the membership function sets for the K_{Pi}, K_{Ii} and K_{di} are defined as triangular partitions with five segments from 0 to 1 [17]. In this study the controllers are connected to $G_2 - G_{10}$ in the test system. Evaluation of the Integral of the Time multiplied Absolute value of the Error (ITAE) and Figure of Demerit (FD) based on the system performance characteristics are defined as [18]:

$$FD = (500 \times OS)^2 + (8000 \times US)^2 + 0.01 \times T_s^2$$

$$ITAE = 100 \times \int_0^{t_{sim}} t(|\Delta\omega|).dt \qquad (34)$$

Where, Overshoot (OS), Undershoot (US) and settling time of the rotor angle deviation of the machine is considered for evaluation of the FD. "Fig. 3", shows the plot of obtained fitness function value.

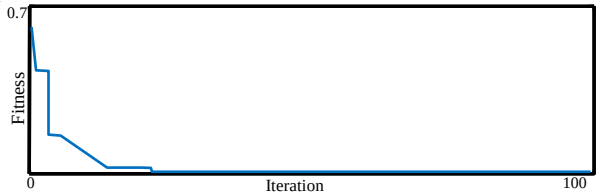


Figure 3. Variations of the fitness functions

TABLE I. OPTIMAL RULE-BASE FOR K_{Ii}

	NB	NS	PS	PB
NB	NM	PB	ZO	PB
NS	PB	NM	ZO	ZO
Z	PM	ZO	PM	ZO
PS	NS	PM	NS	NB
PB	PM	NB	NS	NS

TABLE II. OPTIMAL RULE-BASE FOR K_{Di}

	NB	NS	PS	PB
NB	NS	NS	PB	NM
NS	ZO	PM	ZO	NB
Z	NB	NS	ZO	ZO
PS	NS	NS	NB	NB
PB	NM	NB	PM	NS

TABLE III. OPTIMAL RULE-BASE FOR K_{Pi}

	NB	NS	PS	PB
NB	NS	NS	PB	NM
NS	ZO	PM	ZO	NB
Z	NB	NS	ZO	ZO
PS	NS	NS	NB	NB
PB	NM	NB	PM	NS

The wind velocity in four area is presented in Fig. 4. Also, wind power generations are presented in Fig. 5, and Fig. 6-8 shows tie-line power deviation in nominal load condition.

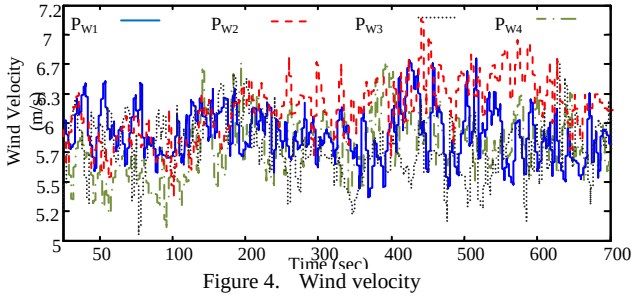


Figure 4. Wind velocity

- **Scenario**

For this section, it is very important that, the performance of the proposed controller is tested under transient conditions by applying a 6-cycle three-phase fault or increasing the mechanical torque. So, in this scenario a 6-cycle three-phase fault is applied in line 26-29 and bus 29. The responses of generators 1, 3, 7 and 9 are presented in Fig. 9. Also the numerical results of *FD* and *ITAE* are presented in Table 4.

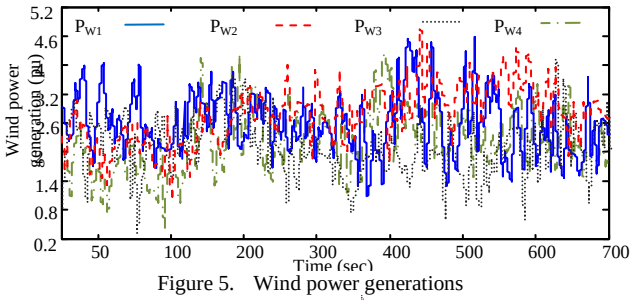


Figure 5. Wind power generations

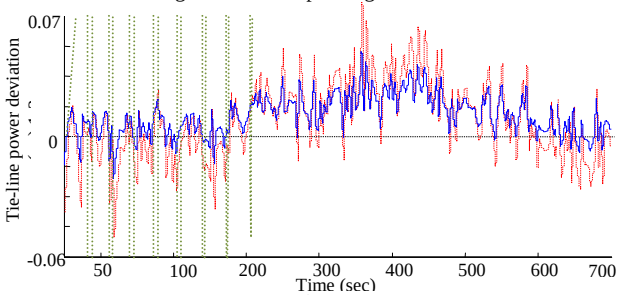


Figure 6. System response in first scenario under nominal load condition (Solid: Proposed, Dashed: CPSS, Doted: No-PSS)

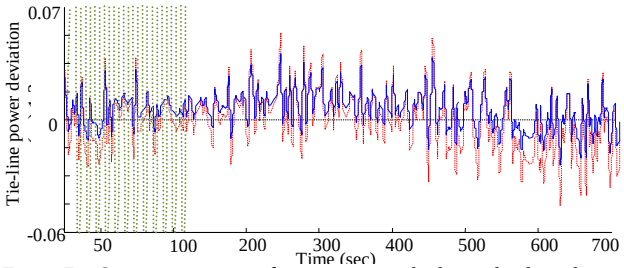


Figure 7. System response in first scenario under heavy load condition (Solid: Proposed, Dashed: CPSS, Doted: No-PSS)

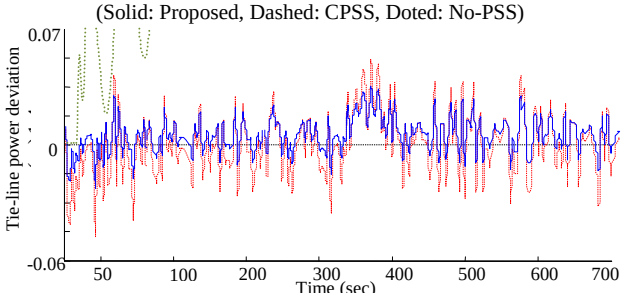


Figure 8. System response under light load condition (Solid: Proposed, Dashed: CPSS, Doted: No-PSS)

Table. 4. Calculate of FD and ITAE for different load changes.

Change load	HSA		CPSS	
	ITAE	FD	ITAE	FD
25%	2.8293	1.6251	9.656	8.455
20%	1.1093	1.2543	8. 856	8.490
15%	1.1110	1.0137	8.890	8.466
10%	1.0123	1.0225	8.265	8.349
5%	1.0313	1.0129	8.853	8.763
Nominal	1.0113	1.2421	8.876	8.223
-5%	1.2413	1.2663	8.243	8.324
-10%	1.1542	1.2514	8.866	8.365
-15%	1.1652	1.2665	8.234	8.387
-20%	1.1334	1.1322	8.734	8.376
-25%	2.1542	1.1652	9.029	8.376

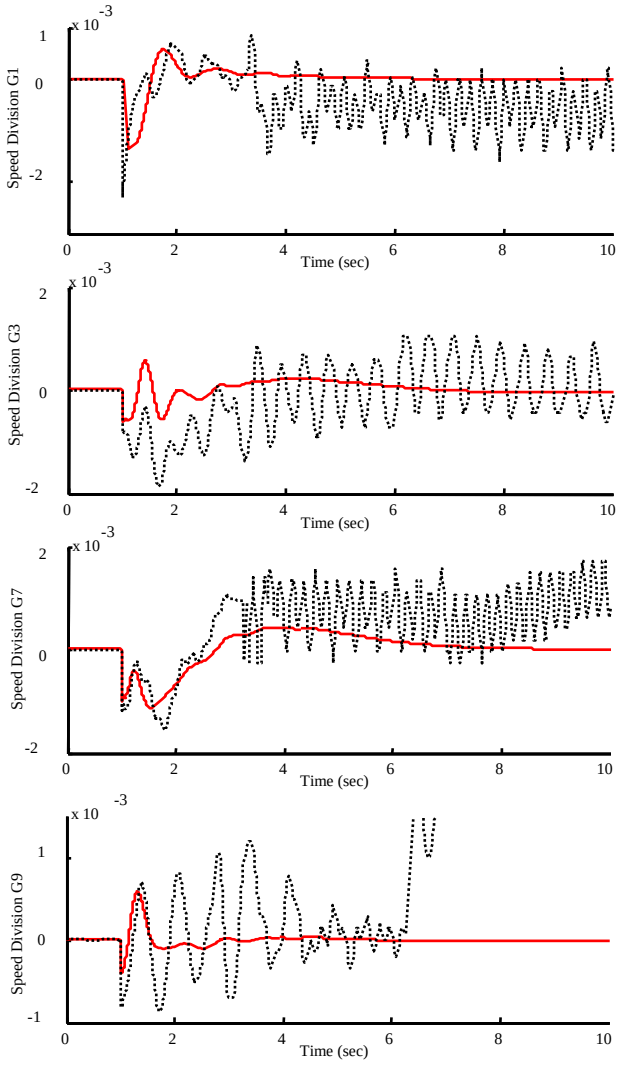


Figure 9. Generators response under nominal load condition (Solid: Proposed, Dashed: CPSS)

VI. Conclusion

In this paper, a design scheme of robust PSS ten-machine New England power system using by considering inverse additive perturbation in a power system with wind farms is presented through hybrid technique of fuzzy controller and HSA. By proposed HSA, the rule based of fuzzy controller is optimized to damp power system oscillation where. The proposed technique is tested in various load condition for the solution of the low frequency oscillation problem in power system. The proposed test case is compared with CPSS through different load conditions and numerical results of ITAE and FD which demonstrate the validity of mentioned strategy.

REFERENCES

[1] G. Shahgholian, A. Rajabi, B. Karimi, "Analysis and Design of PSS for Multi-Machine Power System Based on Sliding Mode

Control Theory", International Review of Electrical Engineering (I.R.E.E.), vol 5, Issue. 5, pp: 2241-2250, 2010.

[2] K. R. Padiyar, "Power System Dynamics-Stability and Control", BS Publications, Hyderabad, India, 2002.

[3] P. Kundur, Power system stability and control, New York: McGraw- Hill; 1994.

[4] E. Mahmoodi, M.M. Farsangi, "Design of stabilizing signals using model predictive control," International Journal on "Technical and Physical Problems of Engineering" (IJTPE), Issue2, Vol. 2, No. 1, pp: 1-4, Mar2010.

[5] M. J. Basler, R. C. Schaefer, "Understanding Power System Stability", IEEE Trans. Industry Applications, 2008, Vol. 44, pp. 463-474, 2008.

[6] M. Caner, N. Umurkan, S. Tokat, S. Vakkas Ustun, "Determination of Optimal Hierarchical Fuzzy Controller Parameters According to Loading Condition with ANN ", Expert Systems with Applications, Vol. 34, pp. 2650-2655, 2008.

[7] L. J. Cai, I. Erlich, "Simultaneous coordinated tuning of PSS and FACTS damping controllers in LARGE power system". IEEE Transactions on Power Systems, Vol. 20(1), pp. 294–300. 2005.

[8] Cai L.,Erlich I., Simultaneous coordinated tuning of PSS and FACTS damping controllers in large power systems. IEEE Trans. Power Syst., 20 (2005) No. 1, 294-300.

[9] Abido M., A novel approach to conventional power system stabilizer design using tabu search. Int. J. Elect. Power Energy Syst., 21 (1999) No. 6, 443-454.

[10] Z.W. Geem, J.H. Kim, and G.V. Loganathan, "A new heuristic optimization algorithm: harmony search", Simulation, vol. 76, no. 2, pp. 60–68, 2001.

[11] K.S. Lee and Z.W. Geem, "A new meta-heuristic algorithm for continuous engineering optimization: harmony search theory and practice", Computer Methods in Applied Mechanics and Engineering, vol. 194, pp. 3902–3933, 2004.

[12] Abido M. Thyristor Controlled Phase Shifter Based Stabilizer Design using Simulated Annealing Algorithm. in International Conference on Electric Power Engineering. 1999.

[13] Y. L. Abdel-Magid, M. A. Abido, S. AI-Baiyat, A. H. Mantawy, "Simultaneous Stabilization of Multi-machine Power Systems Via Genetic Algorithms", IEEE Trans. Power Syst., 1999, Vol. 4, No. 4, pp. 1428 – 1439, 1999.

[14] Dong-Jiang & Li Wang, Small-signal stability analysis of an autonomous hybrid renewable energy power generation/energy storage system part I: time-domain simulations, IEEE Transactions on Energy Conversion, Vol. 23, No.1, pp. 311-320, 2008.

[15] I. Boldea and S. A. Nasar, Electric Machine Dynamics. Macmillan Publishing Company, 1986.

[16] R. Parncutt, "Harmony: A Psychoacoustical Approach", Springer Verlag, 1989.

[17] Ghasemi A., Shayanfar H. A., Mohammad S. N., Abedinia O.," Optimal Placement and Tuning of Robust Multimachine PSS via HBMO ", in Proceedings of the International Conference on Artificial Intelligence, Las Vegas, U.S.A., 2011:201-8.

[18] O. Abedinia, Mohammad. S. Naderi, A. Ghasemi, Robust LFC in Deregulated Environment: Fuzzy PID using IHBMO, Proceeding of the IEEE International Power & Energy Society Power Systems Conference and Exposition, Italy, Rome (EEEIC), pp: 74-77, 2011.

BIOGRAPHIES

Heidarali Shayanfar received the B.S. and M.S.E. degrees in Electrical Engineering in 1973 and 1979, respectively. He received his Ph. D. degree in Electrical Engineering from Michigan State University, U.S.A., in 1981. Currently, he is a Full Professor in Electrical Engineering Department of Iran University of Science and Technology, Tehran, Iran. His research interests are in the Application of Artificial Intelligence to Power System Control Design, Dynamic Load Modeling, Power System Observability Studies, Voltage Collapse, Congestion Management in a Restructured Power System, Reliability Improvement in Distribution Systems and Reactive Pricing in Deregulated Power Systems. He has published more than 490 technical papers in the International Journals and Conferences proceedings. He is a member of Iranian Association of Electrical and Electronic Engineers and IEEE.

Oveis Abedinia received the B.S. and M.Sc. degrees in Electrical Engineering in 2005 and 2009, respectively. Currently, he is a Ph. D. student in Electrical Eng. Department, Semnan University, Semnan, Iran. His areas of interest in research are Application of Artificial Intelligence to Power System and Control Design, Load and Price Forecasting, Restructuring in Power Systems, Heuristic Optimization Methods. He has two industrial patents, authored of one book in Engineering area in Farsi and more than 70 papers in international journals and conference proceedings. Also, he is a member of Iranian Association of Electrical and Electronic Engineers (IAEEE) and IEEE.

Nima Amjady (SM'10) was born in Tehran, Iran, on February 24, 1971. He received the B.Sc., M.Sc., and Ph.D. degrees in electrical engineering from Sharif University of Technology, Tehran, Iran, in 1992, 1994, and 1997, respectively. At present, he is a Professor with the Electrical Engineering Department, Semnan University, Semnan, Iran. He is also a Consultant with the National Dispatching Department of Iran. His research interests include security assessment of power systems, reliability of power networks, load and price forecasting, and artificial intelligence and its applications to the problems of power systems.

FDM based MOP to Attune IPFC POD Controller and Dual-Input PSS in order to Improve Power System Dynamic Performance

H. Shayeghi

Technical Engineering Department
University of Mohaghegh Ardabili
Ardabil, Iran

Y. Hashemi

Technical Engineering Department
University of Mohaghegh Ardabili
Ardabil, Iran

H. A. Shayanfar*

College of Electrical Engineering
Center of Excellence for Power System Automation and Operation, Iran University of Science and Technology, Tehran, Iran

hshayeghi@gmail.com, yashar_hshm@yahoo.com, hashayanfar@gmail.com

Abstract – *With the presence of Interline Power Flow Controller (IPFC), three definite classes of the Power System Stabilizer (PSS) have been evaluated as: a) Conventional PSS (CPSS); b) dual-input PSS; and c) accelerating power PSS model (PSS2B). The design procedure of coordination is expressed as a Multi-Objective Problem (MOP) and the coordination is focused on concurrent selection of the controllers parameters utilizing Nonlinear Time-Varying Evolution (NTVE) based Multi-Objective PSO (MOPSO) algorithm. A Fuzzy Decision-Making (FDM) approach is proposed for finding best compromise solution from the set of Pareto-solutions obtained through MOPSO-NTVE. Finally, in a system having the IPFC, comparative analysis of the results obtained from application of the dual-input PSS, PSS2B and CPSS is presented. The simulation results show that the dual-input PSS & IPFC and PSS2B & IPFC coordination provide a better performance than the conventional single-input PSS & IPFC coordination.*

Keywords: Simultaneous coordinated design, IPFC, Power Oscillation Damping (POD), PSS.

I. INTRODUCTION

Among generator exciter control procedures for stability enhancement, PSS based active power is used as the input signal in numerous cases. These procedures damp out the local power oscillations between generators. With regard to minor information contained in active power signal, damping of low frequency power system oscillations is non-significant by this procedure. To solve this disadvantage of conventional PSS, rotor speed deviation is used as the input signal [1, 2]. These two signals are gathered after passing through compensators, and the resultant signal is applied to PSS that vastly can improve stability. Accelerating power PSS model [3] is the another model of the dual-input stabilizer which uses combinations of power and speed signals to derive the stabilizing signal. In this type of PSS, there is no need for a torsional filter in the main stabilizing path. This eliminates the exciter mode stability problem, thereby permitting a higher stabilizer gain that results in better damping of system oscillations.

Fig. 1 shows the fundamental formation of an IPFC. With this formation two lines can be controlled concurrently to optimize the network employment. Supplementary damping controller (SDC) design for an IPFC [4] can help damping of the low frequency oscillations in the power system while supply the primary purpose of the device, i.e., power flow control or voltage control. Control strategies for enhancing stability such as SDC based PI controller or nonlinear controllers as for example recurrent neural network controllers have been proposed in [5, 6]

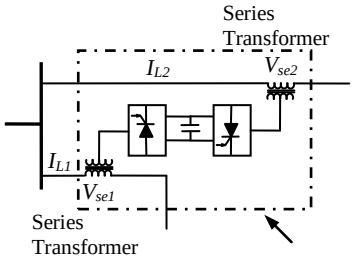


Fig. 1: Fundamental formation of IPFC

* Corresponding Author. E-Mail Address: hashayanfar@yahoo.com (H. A. Shayanfar)

However, uncoordinated local control of FACTS devices and PSSs may cause unwanted interactions that further results in the system destabilization. To improve overall system performance, many researches have been made on the coordination among conventional PSS and FACTS POD controller [7-11]. Some of these methods are based on the complex non-linear simulation [7, 8], and the others are linear approaches [9-11].

A comparative analysis between the results from application of the accelerating power PSS model, dual-input PSS, and CPSS in coordination with the IPFC as a FACTS device is presented in this paper. The subject of robust output feedback controller coordinated design is systematized as an multi-objective optimization problem and the coordination procedure is focused on concurrent searching of parameters of controllers using MOPSO-NTVE. A decision-making procedure based on FDM method is adopted to rank the Pareto-optimal solutions from best to worst and to determine the best solution in a deterministic environment with a single decision maker. First, the system eigenvalues without controllers, and then with proposed controllers is investigated. It is quite evident from the results that the system stability is greatly enhanced with the concurrent coordinated design of IPFC & dual-input PSS, and IPFC & PSS2B, as the damping ratio of the electromechanical mode eigenvalue has been greatly improved. In this study, the nonlinear time-domain simulation is carried out to validate the effectiveness of the proposed controllers. The controllers are simulated and tested under different operating conditions.

II. Model of Power System with IPFC

In order to consider the effects of IPFC in damping of the low frequency oscillation (LFO), the dynamic model of the IPFC (as shown in Fig. 2) is employed, while the resistance and transient of the transformers of the IPFC can be disregarded. By combining and linearizing equations the state variable equations of the power system equipped with the IPFC can be represented as [5]:

$$\dot{x} = Ax + Bu \tag{1}$$
$$x = \left[\Delta\delta \quad \Delta\omega \quad \Delta E'_q \quad \Delta E_{fd} \quad \Delta V_{dc}\right]^T$$
$$u = \left[\Delta u_{pss} \quad \Delta m_1 \quad \Delta\delta_1 \quad \Delta m_2 \quad \Delta\delta_2\right]^T$$

Where, $\Delta m_1, \Delta m_2, \Delta\delta_1$ and $\Delta\delta_2$ are the deviation of input control signals of the IPFC as explained previously.

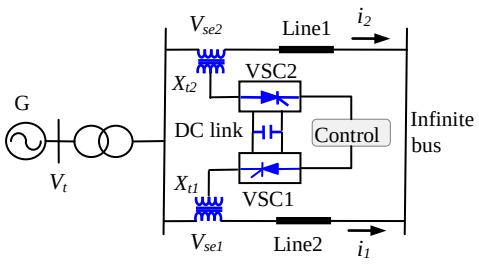


Fig. 2: A single machine connected to infinite bus with IPFC

III. IPFC Damping Controller

The four control parameters of the IPFC (m_1, m_2, δ_1, and δ_2) can be modulated (U_{ref} in Fig. 3) in order to produce the damping torque that the election of the best input of IPFC is based on Singular value decomposition (SVD) analysis. The construction of IPFC based damping controller is similar to the PSS controllers as shown in Fig. 3.

IV. Dual-Input Power System Stabilizer

The $p+\omega$ input PSS is shown in Fig. 4. p and ω are generator local signals which are selected as the PSS inputs. If p input of PSS and ω input of PSS are optimized independently and combined to make as $p+\omega$ input of PSS, an unexpected unstable oscillation mode may occur. In this paper, the parameters of the $p+\omega$ input of PSS with parameters of the IPFC controller are optimized all together.

The other model of the dual-input stabilizer described in IEEE Std 421.5 [12] is PSS2B (as shown in Fig. 5), which uses combinations of power and speed or frequency to derive the stabilizing signal. In this type of PSS in order to extract a signal proportional to rotor speed deviation, the following equation is used:

$$\Delta\omega_{eq} = \frac{1}{M}\int\left(\Delta P_m - \Delta P_e\right)dt \tag{2}$$

The objective is to derive the equivalent speed signal $\Delta\omega_{eq}$ so that it does not contain torsional modes. Torsional modes are weakened in the integral of ΔP_e signal. The problem is to measure the integral of ΔP_m free of torsional modes. In many applications, the ΔP_m component is neglected. This is satisfactory, except when the mechanical power changes. Under such conditions, a spurious stabilizer output is produced if ΔP_e alone is used as the stabilizing signal. The integral of mechanical power is related to shaft speed and electrical power as follows:

$$\int \Delta P_m dt = M\Delta\omega + \int \Delta P_e dt \quad (3)$$

According to (3), the integral of mechanical power change can be acquired by adding signals proportional to shaft-speed change and integral of electrical power change. This signal will contain torsional oscillations unless a filter is used. In Fig. 5 the ramp-tracking filter is a low-pass filter as torsional filter. Also, this filter minimizes the PSS output deviation that occurs when the mechanical power is changing rapidly.

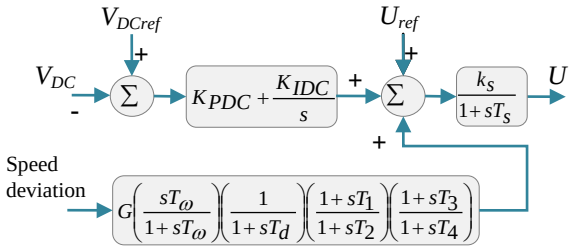


Fig. 3: Structure of IPFC based damping controller

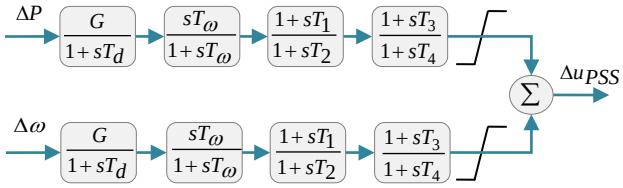


Fig. 4. Dual-input PSS

V. Function Optimization

To obtain the best values of controllers gains and time constants, the objective function and system constraints are formulated as follows:

$$Min\ F(G_m, T_n) = Min\ \{F_1, F_2\} \quad (4)$$

s.t. $G_{m\min} \le G_m \le G_{m\max}$

$T_{n\min} \le T_n \le T_{n\max}$

Where,

$$F_1 = \sum_{j=1}^{N} \sum_{\xi_i \le \xi_0} (\xi_0 - \xi_{i,j})^2 \text{ and } F_2 = \sum_{j=1}^{N} \sum_{\sigma_i \ge \sigma_0} (\sigma_0 - \sigma_{i,j})^2 \quad (5)$$

Where, $\sigma_{i,j}$, $\xi_{i,j}$ are the real part and damping ratio of the ith eigenvalue in the jth operating point. σ_0, ξ_0 are the desired minimum real part and damping ratio to be achieved. G_m, T_n are the optimization parameters and $f(G_m, T_n)$ is the objective function, where m and n are the total number of gains and time constants respectively. In this article, σ_0, ξ_0 are selected as -2 and 1 respectively. N is the total number of operating points that survey is carried out.

VI. Multi-objective Optimization Algorithm

The general multi-objective minimization problem, without constraints, can be stated as (18) [13]:

$$Minimize\ f(x) = (f_1(x), f_2(x), \cdots, f_m(x)) \quad (6)$$

subject to $x \in \Omega$, where: $x \in \Omega$ is a feasible solution vector, Ω is the feasible region of the problem, m is the number of objectives and $f_i(x)$ is the ith objective function of the problem. With assuming $f(x) = (f_1(x), f_2(x), \cdots, f_m(x))$ and $f(y) = (f_1(y), f_2(y), \cdots, f_m(y))$, $f(x)$ dominates $f(y)$, denoted by $f(x) \prec f(y)$, if and only if (minimization):

$$\forall i \in \{1,2,\ldots,m\}: f_i(x) \le f_i(y), \text{ and}$$
$$\exists i \in \{1,2,\ldots,m\}: f_i(x) < f_i(y) \quad (7)$$

$f(x)$ is non-dominated if there is no $f(y)$ that dominates $f(x)$. Also, if there is no solution y that dominates x, x is called Pareto Optimal and $f(x)$ is a non-dominated objective vector. The set of all Pareto Optimal solutions is called Pareto Optimal Set, denoted by P^*, and the set of all non-dominated objective vector is called Pareto Front, denoted by PF^*.

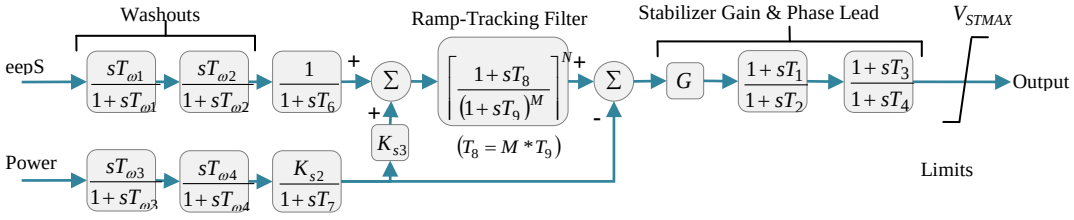


Fig 5: Accelerating power PSS model (PSS2B)

MOPSO-NTVE concept: Each element can be updated as follows:

$$\vec{v}(t+1)=C(\varphi)\left\{\varpi(iter)\vec{v}(t)+c_1(iter)\phi_1\left(\vec{p}_{best}(t)-\vec{x}(t)\right)+c_1(iter)\phi_1\left(\vec{R}_h(t)-\vec{x}(t)\right)\right\} \quad (8)$$

The variables ϕ_1 and ϕ_2, in (8), are the coefficients that determine the influence of the particle best position, randomly obtained in each iteration.

The inertia weight is given as described in (9). The cognitive parameter c_1 starts with a high value c_{1max} and nonlinearity decreases to c_{1min}. Also, the social parameter c_2 starts with a low value c_{2min} and nonlinearity increase to c_{2max} using the following equations [14]:

$$\omega(iter)=\omega_{min}+\left(\frac{iter_{max}-iter}{iter_{max}}\right)^{\alpha}(\omega_{max}-\omega_{min}) \quad (9)$$

$$c_1(iter)=c_{1min}+\left(\frac{iter_{max}-iter}{iter_{max}}\right)^{\beta}(c_{1max}-c_{1min}) \quad (10)$$

$$c_2(iter)=c_{2max}+\left(\frac{iter_{max}-iter}{iter_{max}}\right)^{\gamma}(c_{2min}-c_{2max}) \quad (11)$$

$$C(\varphi)=\frac{2}{\left|2-\varphi-\sqrt{\varphi^2-4\varphi}\right|} \text{ where, } 4.1\le\varphi\le 4.2 \quad (12)$$

where, $iter_{max}$ is the maximal number of iterations and $iter$ is the current number of iterations. And α, β and γ are constant coefficients.

VII. Decision Making Tool

Once the Pareto optimal set obtained, it is practical to choose one solution from all solutions that satisfy different goals to some extends. Due to the imprecise nature of the decision makers (DM) judgment, it is natural to consider that the DM may have fuzzy or imprecise nature goals of each target function. Hence, the membership functions are introduced to represents the goals of each objective function. Each membership function is defined by the experiences and intuitive knowledge of the decision makers [15] and [16]. A simple linear membership function was considered for each of the objective functions. The membership function is defined as follows:

$$\chi_i=\begin{cases}1, & OF_i\le OF_i^{min}\\ \dfrac{OF_i^{max}-OF_i}{OF_i^{max}-OF_i^{min}} & OF_i^{min}<OF_i<OF_i^{max}\\ 0 & OF_i\ge OF_i^{max}\end{cases} \quad (13)$$

For minimized objective functions and

$$\chi_i=\begin{cases}0, & OF_i\le OF_i^{min}\\ \dfrac{OF_i-OF_i^{max}}{OF_i^{max}-OF_i^{min}} & OF_i^{min}<OF_i<OF_i^{max}\\ 1 & OF_i\ge OF_i^{max}\end{cases} \quad (14)$$

for maximized objective functions. Where OF_i^{min} and OF_i^{max} are the minimum and the maximum value of ith objective function among all non-dominated solutions, respectively. Fig. 6 illustrates a typical shape of the membership function. For each non-dominated solution k, the normalized membership function χ^k is calculated as:

$$\chi^k=\frac{\sum_{i=1}^{N_{ob}}\chi_i^k}{\sum_{k=1}^{M}\sum_{i=1}^{N_{ob}}\chi_i^k} \quad (15)$$

where M is the number of non-dominated solutions and N_{ob} is the number of objective functions. The function χ^k can be considered as a membership function of non-dominated solutions in a fuzzy set, where the solution having the maximum membership in the fuzzy set is considered as the best compromise solution.

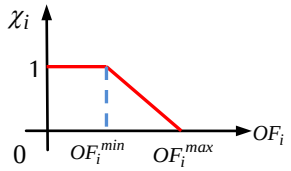


Fig. 6: Linear membership function.

VIII. Simulation Results

By using the linearized power system model and MOPSO-NTVE algorithm, interactions among IPFC damping controller and PSS controller are improved and the controllers parameters are optimized simultaneously to achieve a global optimal damping behaviour. Decision making procedure based on FDM theory is conducted to find the best compromise solution from the Pareto optimal archive generated using MOPSO-NTVE. As shown in Table 1, three operating points such as, nominal, light, and heavy is considered to study the controllers performance.

TABLE I: LOADING CONDITIONS

Loading	P_e (pu)	Q_e (pu)
Nominal	0.8	0.167
Heavy	1.2	0.4
Light	0.2	0.01

In order to illustrate the performance of the proposed controllers, simulation studies are carried out using MATLAB/Simulink and is verified by applying a three-phase fault of 100 ms duration at the infinite bus in the test system at t =1 s. To evaluate the performance of the proposed

controllers, the response of the coordinated tuning of the IPFC & dual-input PSS damping controllers and IPFC & PSS2B damping controllers, are compared with the response of the coordinated tuning of the IPFC & CPSS damping controllers. Fig. 7 shows the speed deviation for three loading conditions. It is clear from these figures that, the simultaneous design of the IPFC & dual-input PSS and IPFC & PSS2B damping controllers significantly improve the stability performance of the test power system, and low frequency oscillations are well damped out and the IPFC & PSS2B controller has the best performance. Integration time absolute error (ITAE) and shape of demerit (SD) are systematized that are indicated in the field of controllers evaluation [17, 18]. As shown in Fig. 8 the values of these system performance characteristics using the PSS2B & IPFC and dual-input PSS & IPFC are much smaller compared to CPSS & IPFC. This demonstrates that the overshoot, undershoot, settling time and speed deviations of the machine are greatly reduced. The obtained results confirms that proposed controllers can provide effective damping of oscillations for various operating conditions.

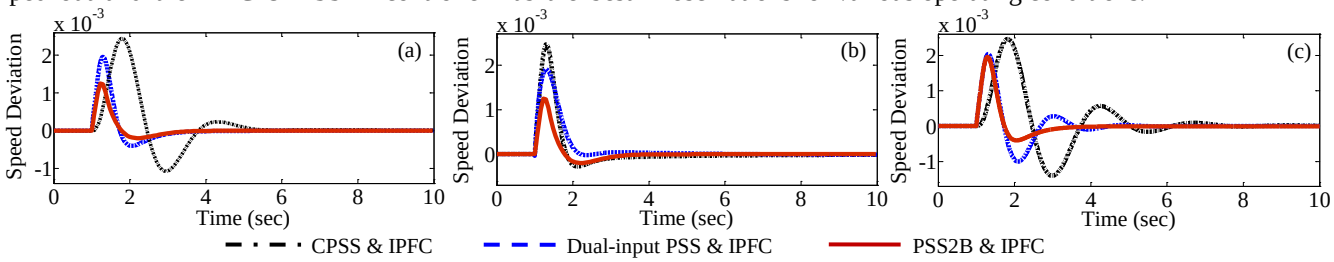


Fig. 7: Dynamic responses for speed deviation at (a) Nominal, (b) Light, (c) Heavy loading condition

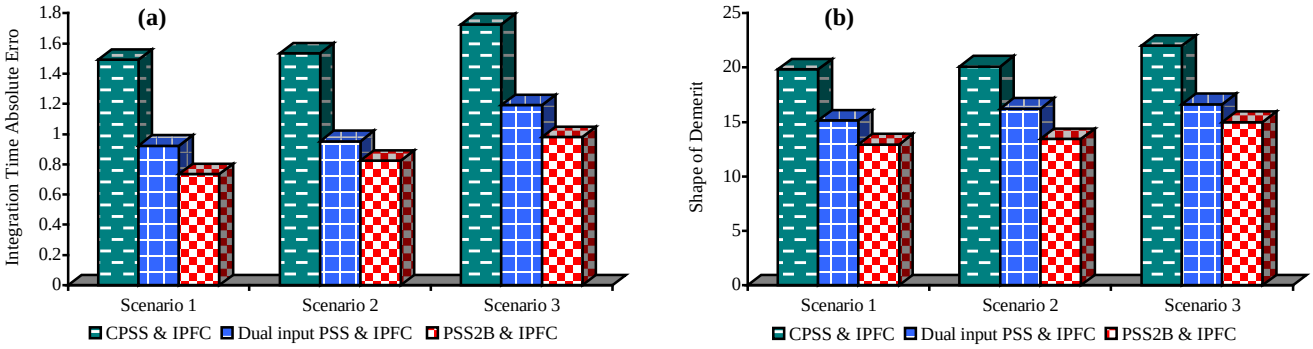


Fig. 8: Values of performance indices: (a) ITAE (b) SD.

IX. Conclusion

A comparative study of single-input PSS, dual-input PSS and accelerating power PSS model coordination with IPFC has been carried out to appreciate the effectiveness of dual-input PSS, and accelerating power PSS model in contrast to CPSS. For the proposed controllers design problem, an multi-objective function to minimize the power system oscillation has been used. Then, MOPSO-NTVE algorithm has been employed to optimally and coordinately tune the controllers parameters. From a decision maker perspective, FDM method is applied to determine solutions with respect to all relevant attributes from the set of Pareto-solutions obtained using MOPSO-NTVE. Simulation results have been presented for various loading conditions and disturbances to show the effectiveness of the proposed coordinated design. The proposed controllers are robust to fault in different operating conditions and generate appropriate stabilizing output control signals to improve stability. The simulation results show that the dual-input PSS & IPFC and PSS2B & IPFC coordination can more effectively damp the system oscillations under different operation conditions than the conventional single-input PSS & IPFC coordination. Also, the PSS2B & IPFC coordination has the best performance.

Appendix

Machine, Transformer, Transmission Line and Exciter:
H =6.5; T'_{do} :8.0 sec ; $D=0$; $x'_d=0.3$; $x_q=0.6$; $x_d=1.8$; $freq$ = 60; V_t = 1.05; V_b = 1; k_A=20; T_A=0.05; $x_t=0.01$; $x_{L1}=0.05$; $x_{L2}=0.05$;
MPSS: T_w = 10; $T_{i\text{-}min}$ = 0.05; $T_{i\text{-}max}$ = 1.5; i = 1, 2, 3, 4; G_{min} = -100; G_{max} = 100; T_d = 0.01;
PSS2B: T_{w1} = 10; T_{w2} = 10; T_{w3} = 10; T_{w4} = 0; T_6 = 0; T_7 = 10; K_{s3} = 1; $K_{s2}=T_7/H$; M = 5; N = 1;
IPFC: $x_{t1}=0.015$; $x_{t2}=0.015$; K_s = 1; T_s = 0.05; $K_{PDC}=-11$; $K_{IDC}=0$; C_{DC} = 3; V_{dc} = 2; $m_{1\text{-}min}$ = 0; $m_{1\text{-}max}$ = 2; $m_{2\text{-}min}$ = 0; $m_{2\text{-}max}$ = 2; T_w = 10; T_d = 0.01; $T_{i\text{-}min}$ = 0.05; $T_{i\text{-}max}$ = 1.5; G_{min} = -100; G_{max} = 100; i = 1, 2, 3, 4;

REFERENCES

[1] Y. Kitauchi, H. Taniguchi, T. Shirasaki, Y. Ichikawa, M. Amano, and M. Banjo, "Experimental verification of multi-input PSS with reactive power input for damping low frequency power swing," *Energy Conversion, IEEE Transactions on*, vol. 14, pp. 1124-1130, 1999.
[2] I. Kamwa, R. Grondin, and G. Trudel, "IEEE PSS2B versus PSS4B: the limits of performance of modern power system stabilizers," *Power Systems, IEEE Transactions on*, vol. 20, pp. 903-915, 2005.
[3] J. V. Burke, D. Henrion, A. S. Lewis, and M. L. Overton, "Stabilization via nonsmooth, nonconvex optimization," *Automatic Control, IEEE Transactions on*, vol. 51, pp. 1760-1769, 2006.
[4] A. Kazemi and E. Karimi, "The Effect of Interline Power Flow Controller (IPFC) on Damping Interarea Oscillations in the Interconnected Power Systems," in *Universities Power Engineering Conference, 2006. UPEC'06. Proceedings of the 41st International*, 2006, pp. 769-773.
[5] A. Kazemi and E. Karimi, "The Effect of Interline Power Flow Controller (IPFC) on Damping Interarea Oscillations in the Interconnected Power Systems," in *Universities Power Engineering Conference (UPEC).Proceedings of the 41st International* 2006, pp. 769-773.
[6] M. Banaei and A. Kami, "Interline power flow controller (IPFC) based damping recurrent neural network controllers for enhancing stability," *Energy Conversion and Management*, vol. 52, pp. 2629-2636, 2011.
[7] L. Cai and I. Erlich, "Fuzzy coordination of FACTS controllers for damping power system oscillations," 2002, pp. 251-256.
[8] X. Lei, E. N. Lerch, and D. Povh, "Optimization and coordination of damping controls for improving system dynamic performance," *Power Systems, IEEE Transactions on*, vol. 16, pp. 473-480, 2001.
[9] L. J. Cai and I. Erlich, "Simultaneous coordinated tuning of PSS and FACTS damping controllers in large power systems," *Power Systems, IEEE Transactions on*, vol. 20, pp. 294-300, 2005.
[10]P. Pourbeik and M. J. Gibbard, "Simultaneous coordination of power system stabilizers and FACTS device stabilizers in a multimachine power system for enhancing dynamic performance," *Power Systems, IEEE Transactions on*, vol. 13, pp. 473-479, 1998.
[11]G. Shahgholian and A. Movahedi, "Coordinated Control of TCSC and SVC for System Stability Enhancement Using ANFIS Method," *International Review on Modelling and Simulations (IREMOS)*, vol. 4, pp. 2367-2375, October 2011.
[12]"IEEE Recommended Practice for Excitation System Models for Power System Stability Studies (IEEE Std 421.5-2005)," ed. IEEE Power Engineering Society, 2006.
[13]C. A. C. Coello, G. B. Lamont, and D. A. Van Veldhuizen, *Evolutionary algorithms for solving multi-objective problems* vol. 5: Springer, 2007.
[14]C. N. Ko, Y. P. Chang, and C. J. Wu, "A PSO method with nonlinear time-varying evolution for optimal design of harmonic filters," *IEEE Transactions on Power Systems*, vol. 24, pp. 437-444, 2009.
[15]A. Lashkar Ara, A. Kazemi, and S. Nabavi Niaki, "Multiobjective optimal location of FACTS shunt-series controllers for power system operation planning," *IEEE Transactions on Power Delivery*, vol. 27, pp. 481-490, 2012.
[16]S. Christa and P. Venkatesh, "Multi-objective optimization problem for the thyristor controlled series compensators placement with multiple decision-making approaches," *European Transactions on Electrical Power*, vol. 23, pp. 249–269, 2011.
[17]H. Shayeghi, A. Safari, and H. Shayanfar, "PSS and TCSC damping controller coordinated design using PSO in multi-machine power system," *Energy Conversion and Management*, vol. 51, pp. 2930-2937, 2010.
[18]H. Shayeghi, H. Shayanfar, S. Jalilzadeh, and A. Safari, "Multi-machine power system stabilizers design using chaotic optimization algorithm," *Energy Conversion and Management*, vol. 51, pp. 1572-1580, 2010.

Biographies

Hossein Shayeghi received the B.S. and M.S.E. degrees in Electrical and Control Engineering in 1996 and 1998, respectively. He received his Ph.D. degree in Electrical Engineering from Iran University of Science and Technology, Tehran, Iran in 2006. Currently, he is an Associate Professor in Technical Engineering Department of University of Mohaghegh Ardabili, Ardabil, Iran. His research interests are in the application of robust control, artificial intelligence and heuristic optimization methods to power system control design, operation and planning and power system restructuring. He has authored and co-authored of five books in Electrical Engineering area all in Farsi, one book and two book chapters in international publishers and more than 260 papers in international journals and conference proceedings. Also, he collaborates with several international journals as reviewer boards and works as editorial committee of eight international journals. He has served on several other committees and panels in governmental, industrial, and technical conferences. He was selected as distinguished researcher of the University of Mohaghegh Ardabili several times. In 2007, 2010, 2011 and 2013 he was also elected as distinguished researcher in engineering field in Ardabil province of Iran. Also, he is a member of Iranian Association of Electrical and Electronic Engineers (IAEEE) and IEEE.

Yashar Hashemi received the B.Sc. and M.S.E. degrees in Electrical Engineering in 2009 and 2011 respectively. Currently, he is Ph.D. student in Electrical Engineering Department, University of Mohaghegh Ardabili,

Ardabil, Iran. His research interests include Power Systems Analysis, Wide Area Measurement and Control, Planning and Control of Renewable Energies, Dynamic Stability, Operation and Planning, Power System Restructuring and FACTS Devices Applications in Power System.

Heidarali Shayanfar received the B.S. and M.S.E. degrees in Electrical Engineering in 1973 and 1979, respectively. He received his Ph. D. degree in Electrical Engineering from Michigan State University, U.S.A., in 1981. Currently, he is a Full Professor in Electrical Engineering Department of Iran University of Science and Technology, Tehran, Iran. His research interests are in the Application of Artificial Intelligence to Power System Control Design, Dynamic Load Modeling, Power System Observability Studies, Voltage Collapse, Congestion Management in a Restructured Power System, Reliability Improvement in Distribution Systems and Reactive Pricing in Deregulated Power Systems. He has published more than 490 technical papers in the International Journals and Conferences proceedings. He is a member of Iranian Association of Electrical and Electronic Engineers and IEEE.

SESSION

NEURAL NETWORKS AND APPLICATIONS + DATA MINING + MACHINE LEARNING

Chair(s)

TBA

Application of Multilayer Perceptrons for Response Modeling

A. Nachev and M. Hogan
Business Information Systems, J.E. Cairnes School of Business & Economics, NUI, Galway, Ireland

Abstract - *This study explores the predictive abilities of multilayer perceptrons used for response modeling in direct marketing campaigns. We extend previous studies discussing how neural network design affects the model performance and also propose a simplified architecture, which outperforms the ones used before. We explore the variance in the neural network behaviour due to the randomness factor and validate the figures of merit by a rigorous testing procedure not applied in the previous studies. The model performance is estimated and analyzed using accuracy, ROC, AUC, lift, and precision-recall. We also compare multilayer perceptrons with logistic regression, naive Bayes, linear discriminant analysis, and quadratic discriminant analysis.*

Keywords: direct marketing, data mining, neural networks, multilayer perceptron.

1 Introduction

Marketing considers two main approaches to communicate with potential buyers: mass marketing, which targets the audience by broadcasting channels, and direct marketing, which uses one-to-one communication with potential buyers in the form of mailing, phone calls, delivery of promotional materials, etc. Nowadays, with increased competition, particularly in the financial and bank sector, direct marketing is a highly appreciated approach due to its efficiency, less waste of resources and cost-effectiveness. A well-executed direct marketing campaign can provide a positive return on investment by motivating customers to respond to the promoted services or goods. One of the preparatory stages ensuring a high return on investment consists of carefully processing the data available in order to generate an appropriate selection of potential buyers.

Formally, the direct marketing task is to use customers' historical purchase data in order to identify those who are most likely to respond positively to a new product/service.

In the terminology of data mining, this is a classification task: having historical data, which consists of attributes and class labels, a well trained model can classify, i.e. split a set of potential customers into two classes: the ones to be contacted, because of their high probability of response; and those not to be contacted, because they are unlikely to make a purchase. The term 'high' is subject of control – that is the cutoff point used to map probabilities into yes/no labels.

Many data mining and machine learning techniques have been involved to build decision support models capable of predicting the likelihood if a customer will respond to the offering or not. These models can perform well or not that well depending on many factors, an important of which is how training of the model has been planned and executed.

Recently, neural networks have been studied in [5, 6, 8, 14] and regarded as an efficient modelling technique. Decision trees have been explored in [5, 6, 8, 13]. Support vector machines are also well performing models discussed in [8, 11, 14]. Many other modelling techniques and approaches, both statistical and machine learning, have been studied and used in the domain.

In this paper, we focus to the neural network models, discussing the factors, which affect their performance and capabilities to predict. We extend the methodology used in [5, 6, 8] addressing issues, such as optimisation of the model hyper-parameters, data validation techniques and metrics used to estimate models behavior, and also propose an architecture, which outperforms the ones used before.

The remainder of the paper is organized as follows: section 2 provides an overview of the neural networks, in particular multilayer perceptrons; section 3 discusses the dataset used in the study, its features, and the preprocessing steps needed to prepare the data for experiments; section 4 presents and discuses the experimental results; and section 5 gives the conclusions.

2 Multilayer Perceptron

A variety of neural network models are used by practitioners and researchers for clustering and classification, ranging from very general architectures applicable to most of the learning problems, to highly specialized networks that address specific problems. Each model has a specific topology that determines the layout of the neurons (nodes) and a specific algorithm to train the network or to recall stored information. Among the models, the most common is the multilayer perceptron (MLP), which has a feed-forward topology. Typically, an MLP consists of a set of input nodes that constitute the input layer, an output layer, and one or more layers sandwiched between them, called hidden layers (Figure 1). Nodes between subsequent layers are fully connected by weighted connections so that each signal travelling along a link is multiplied by its weight w_{ij}. Hidden and output nodes receive an extra bias signal with value 1 and weight θ. The input layer, being the first

layer, has size (number of nodes), which corresponds to the size of the input samples. Each hidden and output node computes its activation level by:

$$s_i = \sum_j w_{ij} x_j + \theta \quad (1)$$

and then transforms it to output by an activation function. The MLPs we use in this study works with the logistic activation function:

$$f_i(s_i) = \frac{1}{1 + e^{-\beta s_i}} \quad (2)$$

where β is slope parameter. The overall model is given in the form:

$$y_i = f_i(w_{i,\theta} + \sum_{j=I+1}^{I+H} f_j(\sum_{n=1}^{I} x_n w_{m,n} + w_{m,\theta}) w_{i,n}) \quad (3)$$

where y_i is the output of the network for node i, $w_{i,j}$ is the weight of the connection from node j to i and f_j is the activation function for node j. For a binary classification, there is one output neuron with logistic activation function.

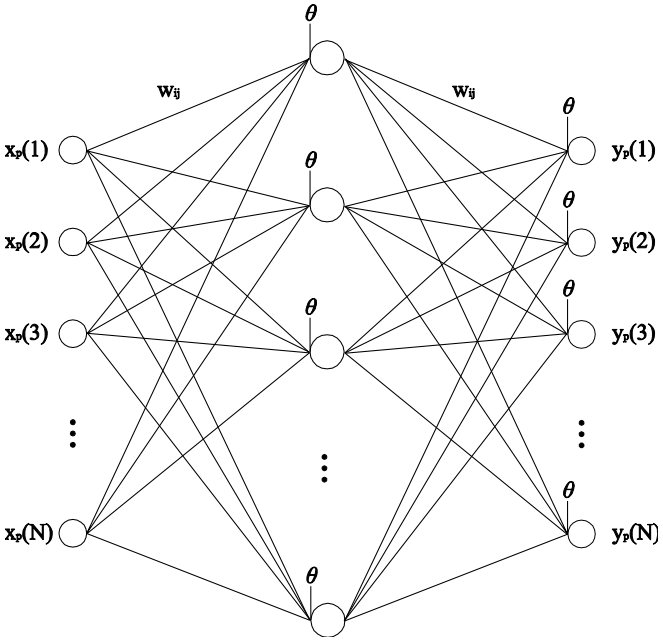


Figure 1. Architecture of an MLP neural network with one hidden layer.

The purpose of the hidden layer(s) is to extend the neural network abilities to learn. Several hidden layers can be placed between the input and output layers, although for nearly all problems, one hidden layer is sufficient. In that case, the hidden layer simply converts inputs into a nonlinear combination and passes the transformed inputs to the output layer. The most common interpretation of the hidden layer is as a feature extraction mechanism. That is, the hidden layer converts the original inputs in the problem into some higher-level combinations of such inputs. Using two or more hidden layers rarely improves the model, and it may introduce a greater risk of bad learning (local minima problem).

The number of input and output nodes is defined by the problem. The number of hidden nodes to use is far from clear. There is no theory yet to tell how many nodes are needed to solve a problem. Often, that number is chosen empirically by experiments. If an inadequate number of nodes are used, the network will be unable to model complex data, and the resulting fit will be poor. If too many nodes are used, the training time may become excessively long, and, worse, the network may overfit the data. When overfitting occurs, the network fits the training data extremely well, but it generalizes poorly to new, unseen data.

The training algorithm we use for the MLP is the Broyden–Fletcher–Goldfarb–Shanno (BFGS) algorithm [2, 9]. The BFGS method approximates the Newton's method, a class of hill-climbing optimization techniques. The algorithm stops when the error slope approaches zero or after a maximum of epochs.

For neural network, we adopt the popular multilayer perceptron, as coded in the R nnet package.

3 Dataset and Preprocessing

The direct marketing dataset used in this study was provided by Moro et al. [8], also available in [1]. It consists of 45,211 samples, each having 17 attributes, one of which is the class label. The attributes are both categorical and numeric and can be grouped as:

- demographical (*age*, *education*, *job*, *marital status*);
- bank information (*balance*, *prior defaults*, *loans*);
- direct marketing campaign information (*contact type*, *duration*, *days* since last contact, *outcome* of the prior campaign for that client, etc.)

A summary the attributes and their description is presented in Table 1.

The dataset is unbalanced, because the successful samples corresponding to the class 'yes' are 5,289, which is 11.7% of all samples. There are no missing values. Further details about data collection, understanding, and initial preprocessing steps can be found in [8].

The majority of out experiments used the dataset for training neural networks. They, however, process numeric data only in a fairly limited range, usually [0,1]. This presents a problem, as the dataset we use contains both numeric values out of the usual range and non-numeric. The data transformations needed sort that out are part of the data preparation stage of the CRISP-DM project model [3]. We did two transformations: mapping non-numeric data into binary dummies and normalization/scaling into the [0,1] interval.

Non-numeric categorical variables cannot be used as they are. They must be decomposed into a series of dummy binary variables. For example, a single variable, such as *education* having possible values of "unknown", "primary",

"secondary", and "tertiary" would be decomposed into four separate variables: *unknown* - 0/1; *primary* - 0/1; *secondary* - 0/1; and *tertiary* - 0/1. This is a full set of dummy variables, which number corresponds to the number of possible values. Note, however, that in this example only three of the dummy variables need - if the values of three are known, the fourth is also known. For example, given that these four values are the only possible ones, we can know that if the education is neither unknown, primary, nor secondary, it must be tertiary. Thus we map a categorical variable into dummies, which are one less than the number of possible values. Using reduced number of dummies we converted the original dataset variables into 42 numeric variables altogether, which is 6 less than the 48 variables used in [5, 6]. There are two benefits of that: first, the neural network architecture becomes simpler and faster; secondly, in some modeling algorithms, such as multiple linear regression or logistic regression, the full set of dummy variables will cause the algorithm to fail due to the redundancy. The model building utility we used converts categorical variables to binary dummies without redundancy.

Table 1: Dataset attribute names, types, descriptions and values.

#	***Name*** **(type)**	**Description: values**
1	*age* (numeric)	
2	*job* (categorical)	type of job: "blue-collar", "admin.", "student", "unknown", "unemployed", "services", "management", "retired", "housemaid", "entrepreneur", "self-employed", "technician",
3	*marital* (categorical)	marital status: "married", "divorced", "single"
4	*education* (categorical)	"unknown", "secondary", "primary", "tertiary"
5	*default* (binary)	has credit in default? "yes", "no"
6	*balance* (numeric)	average yearly balance, in euros
7	*housing* (binary)	has housing loan? "yes", "no"
8	*loan* (binary)	has personal loan? "yes", "no"
9	*contact* (categorical)	contact communication type: "unknown", "telephone", "cellular"
10	*day* (numeric)	last contact day of the month
11	*month* (categorical)	last contact month of year: "jan", "feb", "mar", ..., "dec"
12	*duration* (numeric)	last contact duration, in seconds
13	*campaign* (numeric)	number of contacts performed during this campaign and for this client
14	*pdays* (numeric)	number of days that passed by after the client was last contacted from a previous campaign
15	*previous* (numeric)	number of contacts performed before this campaign and for this client
16	*poutcome* (categorical)	outcome of the previous marketing campaign: "unknown", "other", "failure", "success"
17	*y* (binary)	output variable (desired target): "yes", "no"

The second data transformation we did is related to normalization/scaling. This procedure attempts to give all data attributes equal weight, regardless of the different nature of data and/or different measurement units, e.g. *day* (1-31) vs. *duration* in seconds (0-4918). If the data are left as they are, the training process gets influenced and biased by some 'dominating' variables with large values. In order to address this, we did normalization (z-scoring) according to the formula:

$$x_i^{new} = \frac{x_i - \mu}{\sigma}, \quad (4)$$

where μ is the mean and σ is the standard deviation of the variable in question. After the transformation, each variable has zero mean and unit standard deviation.

In order to scale down the values mapping them into [0,1], we used the linear transformation

$$x_i^{new} = \frac{x_i - a}{b - a}, \quad (5)$$

where [a,b] is the range of values for that variable.

The two transformations were applied to each of the variables independently and separately.

4 Experiments and Discussion

In order to build models for direct marketing application and compare their characteristics with those discussed in other studies [8, 5, 6], we used the same dataset as before and did experiments consistently. We, however, extended the methodology addressing the following issues:

- *Optimization of neural network architecture.* Given, the direct marketing task is a classic binary classification problem, which cannot be described as consisting of two or more clearly separable feature extraction stages, we could expect that the two-hidden layers architecture proposed in [5, 6] could be simplified to one hidden layer. We also reduce the input layer size by 6 nodes using alternative mapping of categorical variables into binaries, as discussed above. Simpler architectures are always preferable as they can be built and trained easily and run faster. One-hidden layer MLPs have proven both theoretically and empirically to be universal approximators for the majority of classification tasks.
- *Validation and testing.* With reference to the data preparation and evaluation stages of CRISP-DM, we extend the testing methodology by partitioning the data into training, validation and test sets. Using validation and test sets in a double-testing procedure helps to avoid overestimation of the model performance. If validation set is used only, it is involved in the model building process for benchmarking during the hyper-parameter optimization. Thus, the validation data become compromised for the purposes of model evaluation, because they have been 'seen' before and the model hyper-parameters have adapted to the validation data. Using validation data for testing usually leads to overestimation of the model performance. Further to [8, 5, 6], which use one-test

procedure with validation set (despite called test in [8]), we adopt double-testing.

- *Randomness and 'lucky' set composition*. With reference to the CRISP-DM data preparation stage, sampling procedure plays important role in model building. The most common partitioning procedure for training, validation, and test sets uses random sampling. Although, this is a fair way to select a sample giving each member of the original dataset equal chance of being selected, this is not an exact replica, in miniature, of the original dataset, and, as a consequence, does not preserve exactly the statistical parameters of the original dataset. In cases where data saturation is not sufficient for a good training, some 'lucky' draws train the model much better than others. Thus, the model instances show variance in behavior and characteristics influenced by the randomness. In order to address this issue and further to [8, 5, 6], we used a methodology, which combines cross-validation (CV), multiple runs over random selection of the folds and initial weights, and multiple runs over random selection of partitions.

All experiments were conducted using R environment [4, 10, 12].

In machine learning applications, classification performance is often measured using accuracy as the figure of merit. For a given operating point of a classifier, the accuracy is the total number of correctly classified instances divided by the total number of all available instances.

Accuracy, however, varies dramatically depending on class prevalence. It can be a misleading estimator in cases where the most important class is typically underrepresented, such as the class of 'yes' of those who respond positively to the direct marketing. For these applications, sensitivity and specificity can be more relevant performance estimators.

In order to address those accuracy deficiencies, we did Receiver Operating Characteristics (ROC) analysis [7].

In a ROC curve, the true positive rate (TPR), a.k.a. sensitivity, is plotted as a function of the false positive rate (FPR), a.k.a. 1-specificity, for different cut-off points. Each point on the ROC plot represents a sensitivity/specificity pair corresponding to a particular decision threshold. A test with perfect discrimination between the two classes has a ROC plot that passes through the upper left corner (100% sensitivity, 100% specificity). Therefore the closer the ROC plot is to the upper left corner, the higher the overall accuracy of the test. The area under the ROC curve (AUC) is a common measure for the evaluation of discriminative power. AUC represents classifier performance over all possible threshold values, i.e. it is threshold independent.

For the sake of consistency with the previous studies, we used 98 % of the dataset for training and validation, split randomly in ratio 2/3: 1/3. The rest of 2% were retained for test. Search of optimal NN architecture was made exploring models with one hidden layer of size from 0 to 13. Experiments showed that sizes above 13 are far from optimal and not considered. In order to validate the results and reduce the effect of lucky set composition, each architecture was tested 300 times: internally, the fit algorithm runs 10 times with different random selection of training and validation sets and initial weights. For each of those set compositions, the 3-fold cross-validation creates 3 model instances and average results. We iterated all that procedure 10 times per architecture recording and averaging accuracy and AUC. Table 2 and Figure 2 show the results.

Table 2 Validated performance metrics of neural networks with H hidden nodes and architecture 42-H-1, where accuracy (ACC) and area under ROC curve (AUC) values are average of 300 model instances of that architecture. ACC_{max} and AUC_{max} are maximal values obtained.

H	*ACC*	ACC_{max}	*AUC*	AUC_{max}
0	89.514	92.040	0.895	0.937
1	89.011	92.150	0.898	0.910
2	89.849	93.143	0.903	0.924
3	**90.489**	**93.143**	0.906	**0.939**
4	90.289	91.107	0.908	0.913
5	90.250	90.606	0.910	0.921
6	90.090	90.701	0.912	0.919
7	90.285	90.606	0.913	0.919
8	90.025	90.700	**0.915**	0.923
9	90.049	90.505	0.915	0.922
10	90.050	90.505	0.915	0.920
11	90.091	90.800	0.913	0.918
12	89.528	90.300	0.913	0.923
13	90.120	90.403	0.914	0.918

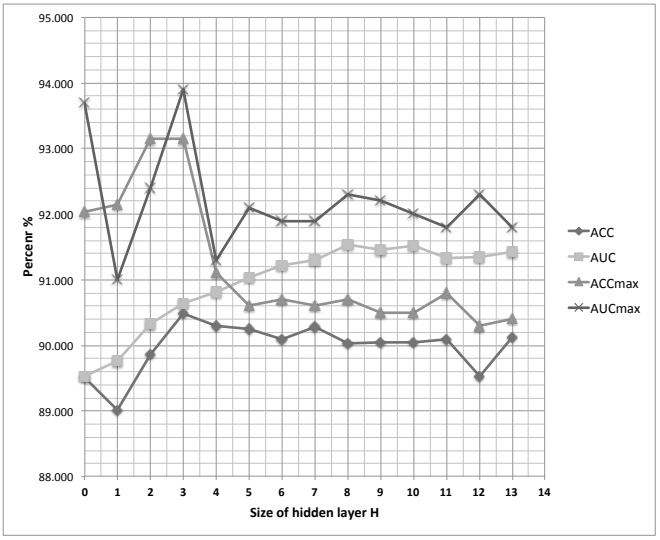


Figure 2. Average and max accuracy and AUC of 14 neural net architectures 42-H-1, H=0..13.

Models with 42-3-1 architecture show best average accuracy of 90.489%. They outperform the 48-20-15-1 architecture from [5]. There were also certain model

instances, which have higher accuracy (ACC_{max} in the Table 2). Models with 42-8-1 architecture have best average AUC of 0.915. Certain model instances achieved AUC=0.939.

The variance and instability of results can be explained by insufficient saturation of data for training. The model can't be trained well to discriminate between classes, particularly to recognize the under-presented 'yes' class. Nevertheless, experiments show that given the data saturation, a 42-3-1 neural net can be trained to reach accuracy 93.143%, which significantly outperforms the 48-20-15-1 one proposed in [5] and [6].

Figure 3 shows ten colour curves, each of which is a plot of a 42-8-1 model trained and validated by the 98% dataset. The colors represent different cutoff points with color bar shown on the right side of the box. The black curve is average of the 10 curves. The variance of TPR is depicted by the standard deviation bars.

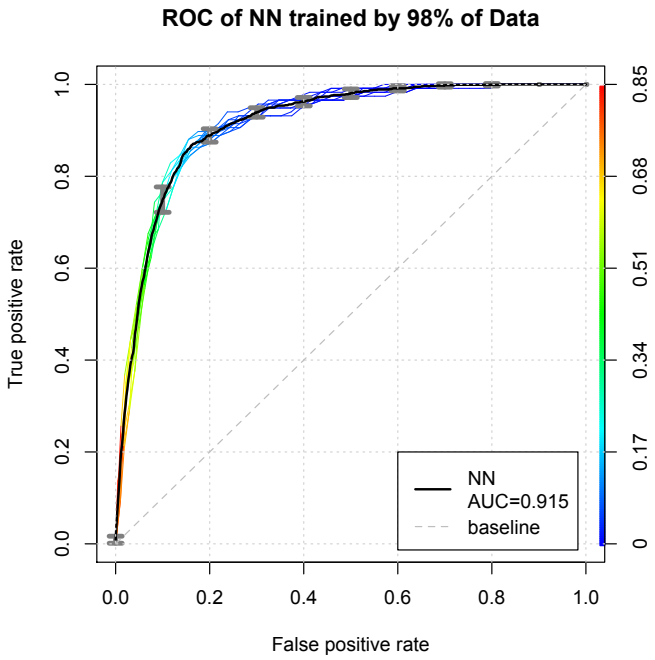


Figure 3. ROC curves of 10 neural network models with 42-8-1 architecture. Colors show values of the cutoff points applied. Black line represents average values of the 10 models. Standard deviation bars measure variance.

Lift is probably the most commonly used metric to measure performance of targeting models in marketing applications. A model is doing well if the response within the target is much better than average for the population as a whole. In a cumulative lift chart (gains chart), the y-axis shows the percentage of true positive responses (tpr). Formally,

$$tpr = sensitivity = \frac{TP}{TP + FN}. \quad (6)$$

The x-axis shows the rate of positive predictions (rpp). This is percentage of customers contacted by direct mailing, which is a fraction of total customers.

$$rpp = \frac{TP + FP}{TP + FP + TN + FN} \quad (7)$$

Figure 4 shows the cumulative lift charts of 42-8-1 neural networks. Ten colored curves and standard deviation bars illustrate variance caused by the randomness in partitioning. The black line is average of the ten.

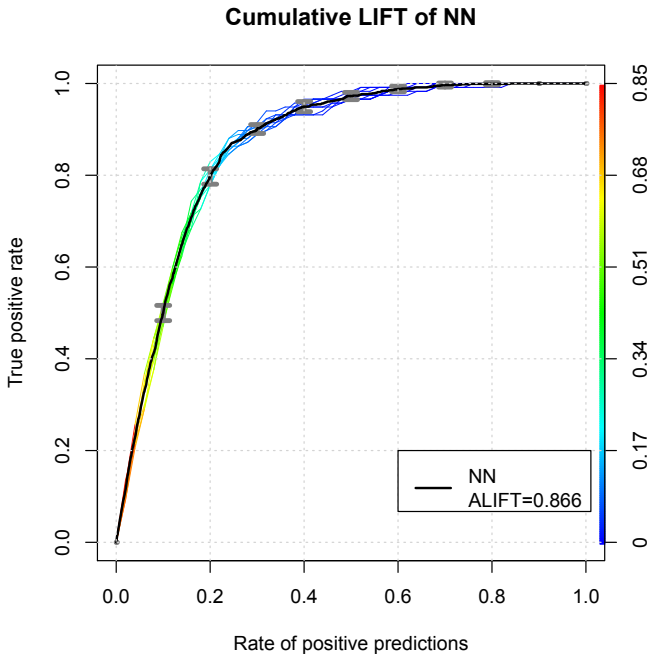


Figure 4. Cumulative LIFT curve of 10 neural network models with 42-8-1 architecture. Colors show values of the cutoff points applied. Black line represents average values of the 10 models. Standard deviation bars measure variance.

A good way to characterize performance of a classifier is to look at how precision and recall change as threshold changes. This can be visualized by precision-recall curve (Figure 5). The better the classifier, the closer its curve is to the top-right corner of the graph. Formally,

$$precision = \frac{TP}{TP + FP} \quad (8)$$

$$recall = \frac{TP}{TP + FN}. \quad (9)$$

In terms of the direct marketing task, precision is the percent of correctly identified 'yes' customers (who purchase the product) among all reported as 'yes'; recall is the percent of correctly identified 'yes' customers among those who are

'yes' in the test set. Recall and precision are inversely related: as recall increases, precision decreases and visa versa. Figure 5 shows 10 colored curves, each representing a randomly built NN model. The black line averages the 10 models. Standard deviation bar visualize variance of the models precision as result of randomness. Colors of the curves and the color bar show the threshold values relevant to the cutoff points.

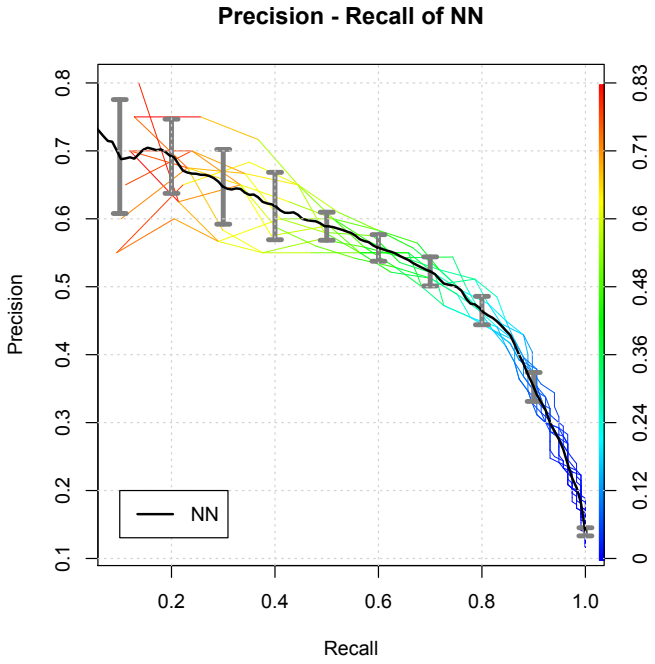


Figure 5. Precision - Recall curve of 10 neural network models with 42-8-1 architecture. Colors show values of the cutoff points applied. Black line represents average values of the 10 models. Variance is measured by standard deviation bars.

In order to compare neural networks (NN) as a modeling technique with other techniques, we built models based on logistic regression (LR), naive Bayes (NB), linear discriminant analysis (LDA), and quadratic discriminant analysis (QDA).

Figure 6 shows ROC curves of the models in one plot. Generally, if two ROC curves do not intersect then one model dominates over the other. When ROC curves cross each other, one model is better for some threshold values, and is worse for others. In that situation the AUC measure can lead to biased results and we are not able to select the best model. Common practice is to compare crossing ROC curves by restricting the performance evaluation to proper subregions of scores. Choice of the method and threshold value is based on the task goals. The figure shows that the curves of NN and LR intersect one another, but in most of the regions NN outperform LR being closer to the top-left corner. This is particularly visible in the most north-west regions, there maximal accuracy is achieved. NN entirely dominate over LDA, NB, and QDA, which performance can be ranked in that order.

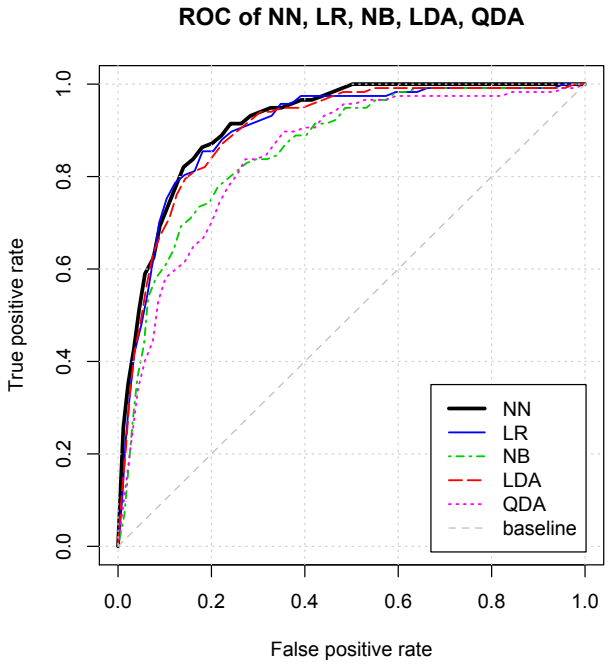


Figure 6. ROC curve of five models: Neural Network, Logistic Regression, Naive Bayes, Linear Discriminant Analysis, and Quadratic Discriminant Analysis. Each model runs with its optimal hyper-parameter values and size of the training dataset. Results are validated by 3-fold cross-validation.

5 Conclusions

Predictive models in direct marketing seek to identify customers/individuals most likely to respond to promotional solicitations or other intervention programs.

This paper presents a case study of data mining modeling techniques for direct marketing based on neural networks. We extend the studies of [5, 6, 8], addressing the following issues:

We explore how neural network design affect the model performance in order to find the optimal size of the hidden layer. We propose a simpler architecture with one hidden layer and three nodes. It achieves best accuracy and outperforms the 48-20-15-1 architecture proposed by [5]. When we measure the threshold independent figure of merit AUC, best performer is the 42-8-1 architecture.

We also did ROC analysis comparing neural nets, logistic regression, naive Bayes, linear and quadratic discriminant analysis techniques and found that neural nets prevail over logistic regression in most of the cutoff point intervals. Neural nets also entirely dominate over the rest of the techniques.

Another contribution of this study is validation of experimental results by a methodology that extends those of [5, 6, 8]. We used double-testing with both validation and

test sets and took into account performance variance and the effect of lucky set composition caused by randomness, not reported previously. Each model architecture was tested 300 times involving 3-fold cross-validation, random partitioning and iterations. We also did ROC, cumulative lift, and precision-recall analysis.

6 References

[1] Bache, K. & Lichman, M. (2013). UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science.

[2] Broyden, C., The convergence of a class of double rank minimization algorithms: 2. The new algorithm, J. Inst. Math. Appl., 6: 222–231, 1970.

[3] Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C. and Wirth, R. CRISP-DM 1.0 - Step-by-step data mining guide, CRISP-DM Consortium, 2000

[4] Cortez, P. "Data Mining with Neural Networks and Support Vector Machines using the R/rminer Tool". In Proceedings of the 10th Industrial Conference on Data Mining (Berlin, Germany, Jul.). Springer, LNAI 6171, 572–583, 2010.

[5] Elsalamony, H., Elsayad, A., Bank Direct Marketing Based on Neural Network, International Journal of Engineering and Advanced Technology, 2(6):392-400, 2013.

[6] Elsalamony, H. Bank Direct Marketing Analysis of Data Mining Techniques., International Journal of Computer Applications, 85 (7):12-22, 2014.

[7] Fawcett, T., An introduction to ROC analysis, Pattern Recognition Letters 27, No.8, 861–874, 2005

[8] Moro, S., Laureano, R., Cortez, P., Using Data Mining for Bank Direct Marketing: An Application of the CRISP-DM Methodology. In P. Novais et al. (Eds.), Proceedings of the European Simulation and Modelling Conference - ESM'2011, pp. 117-121, Guimarães, Portugal, October, 2011.

[9] Fletcher, R., A new approach to variable metric algorithms, Computer J., 13: 317–322, 1970.

[10] R Development Core Team. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria, http://www.R-project.org, 2009.

[11] Shin, H. J. and Cho, S., Response modeling with support vector machines, Expert Systems with Applications, 30(4): 746-760, 2006.

[12] Sing, T., Sander, O., Beerenwinkel, N., Lengauer, T., ROCR: visualizing classifier performance in R., Bioinformatics 21(20):3940-3941, 2005.

[13] Sing'oei, L., Wang, J., Data Mining Framework for Direct Marketing: A Case Study of Bank Marketing, International Journal of Computer Science Issues (IJCSI), 10(2):198-203, 2013.

[14] Yu, E. and Cho, S., Constructing response model using ensemble based on feature subset selection, Expert Systems with Applications, 30(2): 352-360, 2006.

A Comparison of Machine Learning Techniques for the Generation of River and Stream Water Quality Estimates

Alexander Maestre, Ph.D. P.E.[1] and Eman El-Sheikh, Ph.D.[2]
[1]Civil, Construction and Environmental Eng., The University of Alabama, Tuscaloosa, Alabama, USA
[2]Department of Computer Science, University of West Florida, Pensacola, Florida, USA

Abstract – Machine learning techniques can be used to classify water quality stations with similar concentration and discharge trends. Boosted Regression Trees (BRT) and Conditional Inference Trees (CIT) were considered as alternatives to conventional methods used by the United States Geological Survey to estimate daily concentrations of water constituents in rivers and streams based on continuous daily discharge data and discrete water quality samples collected at the same or nearby locations. The Weighted Regressions in Time, Discharge and Season (WRTDS) method is based on parametric survival regressions that generate unbiased estimates of the prediction errors. However, WRTDS needs a large number of samples collected during at least two decades. Alternatively, BRT and CIT can be used for water station classification by clustering data from nearby stations with similar concentration and discharge characteristics. This paper describes a machine learning tool that compares BRT, CIT, WRTDS, and clustering analysis for estimation of daily concentrations.

Keywords: Machine Learning, Clustering, Boosted Regression Trees, Conditional Inference Trees, Water Quality Modeling, Weighted Regressions in Time, Discharge and Season

I. INTRODUCTION

Several models are used by the United States Geological Survey (USGS) for predicting daily concentration of water constituents in rivers and streams using physical and temporal explanatory variables. A common method is Weighted Regressions in Time, Discharge, and Season (WRTDS) [1]. WRTDS provides a method for conducting regressions with censored information (non-detects) using parametric survival regressions. In addition, it uses a jack-knife cross validation approach that evaluates the importance of each survival regression by selecting subsets of the complete dataset. The cross validation approach is also used to identify trends of the constituent concentration in time. However, WRTDS needs a large number of samples (n > 200) collected at the specific station with daily water discharge records collected for at least 20 years without major gaps.

WRTDS has been created by a series of routines written in R, a free package for statistical computing and graphics [2]. The statistical method estimates the concentration using two libraries: dataRetrieval and EGRET. The dataRetrieval library [3] automatically downloads existing records of water discharge and water constituent concentrations from a dedicated server. Approximately 14,500 parameters are available for download using the dataRetrieval tool. The list of parameters available in the server includes nutrients, pesticides, organics, and physical properties among others. The second library, EGRET [4], was created to explore and generate graphics associated with river concentration trends. EGRET conducts the parametric survival regressions and estimates daily concentrations in those periods when samples were not collected.

WRTDS has become popular recently because it uses locally weighted regressions to estimate daily concentrations. It has been tested in major watersheds of the U.S. including the Mississippi and Chesapeake basins [1][5-11]. There are approximately 26,000 USGS stations installed throughout the U.S. A large percentage of these stations have long historical records of daily water discharge but only a few have more than the required 200 water quality samples. Fortunately, other agencies (including state and local environmental agencies) have been collecting

additional water quality samples for several decades. Improvements on these water quality stations include the installation of real time stations. Currently there are approximately 1,700 stations in the U.S. that collect water discharge with a frequency of 15 minutes or less. The information collected by these stations can be downloaded automatically via Internet.

Boosted Regression Trees (BRT) is a non-parametric technique that can successfully determine the influence of predictors in the response when the interaction occurs in a complex and non-linear way [12]. BRT has been used for a wide variety of applications, such as to investigate high variance traffic crash data in Taiwan [13], predict fishing effort distributions [14], and identify processes that drive the richness, composition, and occurrence of plants species in northwest Finland [15]. Conditional Inference Trees (CIT) estimate a regression relationship for continuous, censored, ordered, nominal and multivariate response variables by binary recursive partitioning in a conditional inference framework [16-18].

Machine learning methods such as BRT and CIT can be used as an alternative to WRTDS to identify nutrient concentration trends based on daily discharge data. Both BRT and CIT can provide information that improves the estimation of nitrate + nitrite-N concentrations in stations that have few samples. The approach involves generating a training data set with samples collected in stations with similar concentration and discharge characteristics. The selection of training set stations is obtained from an arbitrary set classified by a clustering algorithm. Once the subset of similar stations is identified, the tree model is created using the stations located within a cluster. To evaluate the estimates, lack of fit of predicted and observed concentrations are compared for WRTDS, BRT and CIT.

In this research, we explore BRT and CIT methods to improve the concentration estimates in stations with less than 200 samples. The training data set includes water quality records of samples collected by the USGS in major rivers and streams in Alabama and Florida. The dataset only includes stations that have at least 25 samples. The goal is to identify if there are any trends in nitrite + nitrate-N concentrations at different clusters based on discharge, nitrate + nitrate-N concentration, and drainage area watershed size. As the model analyzes new stations, a routine or program could identify patterns, similarities, and differences with previous runs and decide which combination of stations produces the best estimates.

II. Weighted Regressions in Time, Discharge, and Season (WRTDS) Method

Weighted Regressions in Time, Discharge, and Season (WRTDS) was developed by the United States Geological Survey (USGS) for analyzing long-term water quality data sets. WRTDS allows parameter adjustment of the mathematical model with changes that occur over time, and downloading data and metadata automatically from the National Water Information System (NWIS). In addition, WRTDS includes multiple routines that allow users to preprocess the original data sets and identify the presence of outliers and influential observations that may cause bias in the estimated concentrations.

Equation (1) shows the mathematical equation that serves as the foundation of the WRTDS method:

$$\ln(c) = \beta_0 + \beta_1 t + \beta_2 \ln(Q) + \beta_3 \sin(2\pi t) + \beta_4 \cos(2\pi t) + \varepsilon \quad (1)$$

where c is the concentration, the β terms are the unknown regression coefficients, Q is the discharge, t is the time, and ε are the independent random errors. For simplicity, each term of Equation 1 will be described with the same terminology used by the WRTDS method as follows:

$$\ln(c) = \beta_0 + \beta_1 \text{DecYear} + \beta_2 \text{LogQ} + \beta_3 \text{SinDY} + \beta_4 \text{CosDY} + \varepsilon \quad (2)$$

In WRTDS, each observed concentration is recalculated using a jack-knife cross validation procedure in which a subset is extracted based on windows that involve ranges in time, discharge, and season. This parametric survival regression enables WRTDS to accept the presence of censored information. Due to the generation of subsets, the number of samples and period of data collected must be sufficiently large to identify trends. Stations with few collected samples cause the method to calculate poor fitted coefficients.

III. Boosted Regression Trees (BRT)

Classification trees are an alternative to regression models to predict the concentration using the same terms included in Equation (1). They provide several advantages: (1) trees are very flexible and can accept broad types of responses including categorical, numerical, and survival data; (2) trees are invariant to monotonic transformations of the independent variables; (3) trees are easy to construct; and (4) trees are easy to interpret [19]. However, trees create poor predictors and may be difficult to interpret, especially as their size increases [20].

Trees with numeric responses are called regression trees, whereas trees with categorical responses are called classification trees. One advantage of classification trees is that they can be represented in a figure with branches and leaves representing the different homogeneous groups.

The tree is constructed by repeatedly breaking the data into exclusive subsets of homogeneous data to the extent possible. The splitting process continues until an overlarge tree is created, and then the tree is pruned to the desired size. In order to select the size of the tree that accurately predicts the prediction error, the method uses a procedure called cross validation. During cross validation, a portion of the observations is deleted and recalculated using the remaining observations. The recalculated values are compared with the original observations to calculate the prediction error.

Boosting appeared as a method to improve the poor prediction capabilities of classification trees [20-21]. Boosting is based on the idea that it is easier to find and average many weak classifiers than trying to find a single highly accurate prediction rule. The advantage of this method is that it is sequential. At each step the model is fitted iteratively to the training data by the current sequence of trees, and these classifications are used as weights to the next step. Incorrect classifications will have higher weights in the next step than cases that were hard to classify, increasing their chance to be correctly classified.

IV. Conditional Inference Trees (CIT)

Conditional Inference Trees (CIT) estimate a regression relationship for continuous, censored, ordered, nominal and multivariate response variables by binary recursive partitioning in a conditional inference framework [16]. The R package ctree method [17] was used for this research.

The algorithm starts by testing the global null hypothesis of independence between the input variables and response, and stops if this hypothesis cannot be rejected; otherwise, it selects the input variable with strongest association to the response. This association is measured by a p-value corresponding to a test for the partial null hypothesis of a single input variable and the response. Second, the algorithm implements a binary split in the selected input variable. It recursively repeat these steps. The implementation utilizes a unified framework for conditional inference, or permutation tests, developed by Strasser and Weber [18]. The approach ensures that the right sized tree is grown and no pruning or cross-validation is needed.

V. Clustering Analysis

Clustering analysis was conducted on the BRT and CIT method results. The initial step during generation of the BRT and CIT models is the selection of a training set for the model. Nitrate + nitrite-N concentration in rivers and streams varied greatly due to land use practices, location, and fluctuations in discharge [22-23]. The concentration of nitrate + nitrite-N at the test station can be estimated by selecting nearby stations with similar discharge and concentration distribution. As a result, a large database was created with nitrate + nitrite-N concentrations and discharge values for multiple stations located in Alabama and Florida. Stations with concentration distribution similar to the distribution observed at the test site were selected using a clustering method. In addition, the size of the drainage area (as an additional factor) was included in the generation of the clusters. The variation of flow and concentration in large watersheds was expected to be less evident than in small watersheds where discharge patterns are in general flashy and short in duration.

The R package mclust was chosen to select the nitrate + nitrite-N concentration and discharge values from those stations similar to the test station [24]. The mclust package implements a Gaussian hierarchical clustering algorithm and the expectation–maximization (EM) algorithm for a

parameterized mixture of models with the possible addition of a Poisson noise term [25]. One advantage of mclust is that it automatically selects among 10 different combinations of the covariance matrix parameterizations to identify clusters with the best Bayesian Information Criterion (BIC).

VI. Project Description

A Python program was combined with an R script to select information from desired stations and evaluate if there was an improvement in the estimation of nitrate + nitrite-N concentration using the BRT and CIT models. The program and script perform four steps during the process: (1) generation of a master dataset; (2) identification of stations with similar characteristics; (3) generation of BRT and CIT models; and (4) comparison among WRTDS, BRT and CIT models for stations in Florida and Alabama.

A. *Generation of Master Dataset*

Table I describes the data for stations used for generation of the models. The samples included in this table were collected during the period January 1, 2004 – December 31, 2009.

For all analyses conducted for this research, the stations included in Table I corresponded to the training dataset except for station 02446500 (Sipsey River near Elrod), which was the only station included in the test dataset.

B. *Identification of Stations with Similar Characteristics*

In general, the distribution of water discharge follows either power law or lognormal distribution [26]. Stations with similar median logarithm of discharge and median logarithm of nitrate + nitrite-N concentration could originate from areas of similar land use, catchment area, or times of concentration. Clustering analysis was conducted on stations that shared similar median and standard deviation of the natural logarithm of the discharge and nitrate + nitrite-N concentration.

It was hypothesized that, as the number of stations in the cluster increased, the results of BRT and CIT would improve by increasing the number of observations in the training set. The statistical program R was selected to calculate the median and standard deviation of the natural logarithm of the nitrate + nitrite-N concentration and discharge of all the stations included in the analysis.

TABLE I. Data for Selected Stations

Basin	USGS Station Number	Nitrate + Nitrite Concentration (mg-N / L)		Logarithm of Discharge (m^3/s)		Drainage Area (square miles)	Cluster
		$\bar{x}$	σ	$\bar{x}$	σ		
FLORIDA	02274505	0.02	0.13	-3.94	1.85	6	2
	02274005	0.18	0.22	-2.92	0.91	9	2
	02275197	0.23	0.33	-1.84	2.25	9	2
	02306774	0.07	0.03	-3.30	2.50	18	2
	02303350	0.08	0.10	-2.13	2.25	23	2
	02273630	0.01	0.07	-2.25	3.23	23	2
	02300700	0.12	0.10	-0.79	1.48	29	2
	02307359	0.03	0.02	-2.68	2.69	30	2
	02274490	0.07	0.07	-1.87	1.50	31	2
	02300100	0.10	0.22	-1.75	1.78	31	2
	02272650	0.03	0.21	-1.77	2.03	38	2
	02270000	0.21	0.13	-1.27	1.06	39	2
	02274010	0.41	0.48	-1.26	1.46	45	2
	02268390	0.07	0.19	-0.05	0.73	53	2
	02272676	0.02	0.06	-1.00	2.47	58	2
	02255600	0.02	0.06	-1.65	2.50	60	2
	02299950	0.07	0.09	-0.57	1.17	65	2
	02301990	0.97	0.36	-0.73	1.15	82	4
	02301300	0.23	0.27	0.50	1.50	107	4
	02271500	0.44	0.47	-0.27	1.45	109	4
	02302500	0.24	0.16	-0.54	1.83	110	4
	02269520	0.30	0.17	-0.37	1.00	120	4
	02295420	0.40	0.25	0.38	1.74	121	4
	02301000	0.76	0.68	0.82	1.10	135	4
	02310280	0.03	0.03	-3.04	2.26	150	4
	02303800	0.03	0.02	-0.82	2.23	160	4
	02310300	0.04	0.05	-3.12	2.75	182	4
	02303000	1.40	0.44	1.01	0.90	220	1
	02298123	0.09	0.11	0.58	1.71	233	1
	02310947	0.03	0.08	0.17	2.75	280	1
	02256500	0.04	0.05	0.23	2.24	311	1
	02303330	0.62	0.48	0.96	1.02	375	3
	02270500	0.12	0.19	1.69	1.17	379	3
ALABAMA	0242354750	1.49	1.46	-0.59	0.91	26	2
	0357479650	1.23	0.41	-1.44	1.26	33	2
	03575100	1.61	0.24	1.51	0.96	375	3
	02446500[a]	0.11	0.07	2.26	1.23	528	2

[a.] USGS Station 02446500 Sipsey River near Elrod, not included in the training dataset

The assumption behind clustering stations of similar characteristics was that all the stations in the clusters would be affected by the same phenomena that were regional or national in scope. For example, it was hypothesized that, if a specific year was wet, all the stations included in the cluster recorded large discharge values that year. These two conditions could impact the coefficients related to time and discharge in Equation (1). On the other hand, it was also considered that clustering stations located in regions with different climate patterns (i.e., northern versus southern U.S.) may affect the seasonal terms of the equation. For this reason, it was also considered preferable to select stations located within the same region.

C. Generation of the BRT Model

In the previous step the function mclust identified four clusters. In this step, mclust identified which cluster was associated with the station located in the Sipsey River (in this case, Cluster 1). The stations within the same cluster of the Sipsey River were selected for the generation of the Boosted Regression Tree. The BRT model was created using the library gbm for the General Boosted Model [27].

The R function gbm.step was used to generate the General Boosted Model. This function determines the optimal tree size using the k-fold cross validation procedure [28]. The default option in gbm.step uses 10 folds and a bag fraction of 0.5, which indicates that 50 percent of the observations of the observed variables are selected to construct the model. As indicated previously, since the distribution of nitrate + nitrite-N concentration and discharge followed a lognormal distribution, it was assumed that the logarithm of these parameters should follow a normal distribution. The model requires the selection of a method to calculate the loss function. Because both discharge and nitrate + nitrite-N concentration are continuous variables, it was decided to use the Gaussian option to focus on minimizing the square error between the observed and predicted values.

The last two parameters in the function gbm.step are the tree complexity and learning rate. The learning rate refers to how quickly the estimated value is calculated based on the previous estimated value plus a portion of the value obtained by the fitted regression model. The tree complexity refers to the depth of the tree (also known as the interaction depth), which is a function of the number of terminal nodes in the tree. It has been recommended for the learning rate to be as small as possible and obtain the optimum number of iterations by cross validation [27]. It is important in BRT models to avoid a large number of iterations because that can cause overfitting [29]. Overfitting occurs when the model starts depicting the random error instead of the relation between the predictors and response.

Preliminary analyses were conducted using sites located in Alabama, varying the tree complexity between 2 and 20 and the learning rate between 0.0001 and 0.05. The results indicated that, as the tree complexity increases, the number of trees decreases. The same pattern was observed between the learning rate and the number of trees. The lowest cross validation correlation standard error was observed when the tree complexity was 5 and the learning rate was 0.001.

D. Comparison between WRTDS, BRT and CIT Models

In WRTDS the estimates are based on the observations from the same station. On the other hand, BRT estimates are based on observations from other stations. The goal is to observe which method generates better estimates of nitrate + nitrite-N concentration for each of the observed concentrations. A perfect fit creates a straight line between the observed and predicted values. The sum of square errors (SSE) was selected as a measure of fitness between the WRTDS and BRT models. The model with the lowest SSE would produce the best estimates.

VII. Machine Learning Application

A machine learning application was developed in Python. The WRTDS, BRT, and CIT models, as well as the clustering analysis and comparison among models were completed using the statistical program R. The graphical interface tool was developed using the Tkinter/ttk package that provides dynamic interaction between the program and routines executed by R. The application performs two main tasks: (1) processes information about the stations and parameters included in the models; and (2) executes an R script that creates and compares the WRTDS, BRT and CIT models. The user entered the information for each station by

completing the fields on the main screen, as shown in Figure 1. Among the parameters needed by the model are the station number, parameter to be analyzed, discharge information, and period of analysis. The tool allows the user to either automatically download the information from the USGS website or access it from a text file that follows the format required by WRTDS.

The "Add Station" button adds the information to a text file that will be read by the R script. The user adds as many stations as needed to run the model. The "Start" button initiates the R script program. In the background, R reads the information from the text file created by the interface tool and creates a data frame with all the records obtained from the selected stations. During this process, the tool automatically generates three WRTDS figures for each station: concentration versus time, discharge versus time, and a multi-plot data overview, as shown in Figure 2. All the figures generated by the script are saved as images and pdf files in a separate folder.

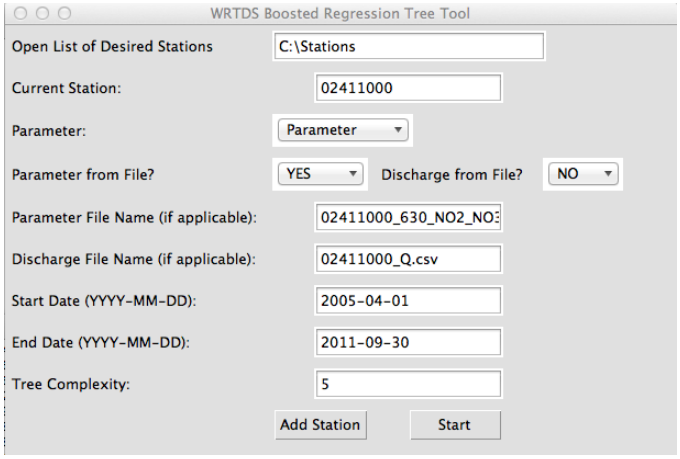


Fig. 1. Interface of the machine learning tool for comparison of WRTDS, BRT and CIT methods to estimate nitrate + nitrite-N concentration.

The script also generates three text files that could be used for further analyses: (1) a summary table with all observations from all stations; (2) a table that includes the drainage area size, median, standard deviation, and first and third quartiles of concentration and discharge for each station; and (3) a table that indicates the cluster assigned to each station during the cluster analysis.

VIII. Results

In this article, we present the results for the stations located in Alabama and Florida described in Table I. Figure 2 shows an example of one of the multi-plot data overview figures generated by WRTDS (station 02275197, Mosquito Creek near Okeechobee, FL). A multi-plot data overview was created for each station included in Table I. The multi-plot data overview allows the identification of gaps, outliers, as well as influential points in the datasets. It also provides a general idea of the number of samples collected by month.

The figure has four panels. In the upper left panel is a scatter plot of concentration versus discharge. This plot shows extreme events and potential correlations between discharge and concentration. Notice that both axes are in log scale matching the terms included in Equation (1).

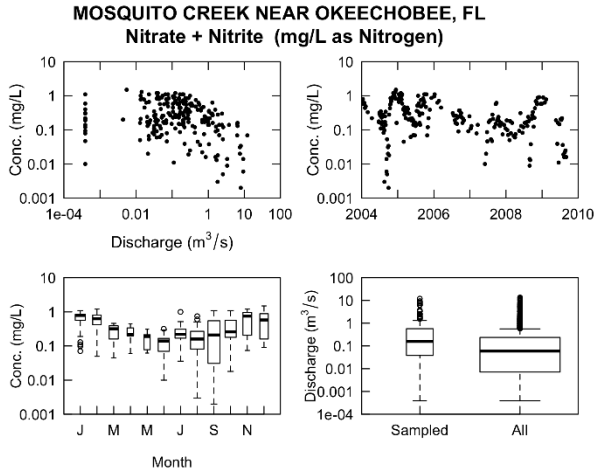


Fig. 2. Example multi-plot data overview figures generated by WRTDS.

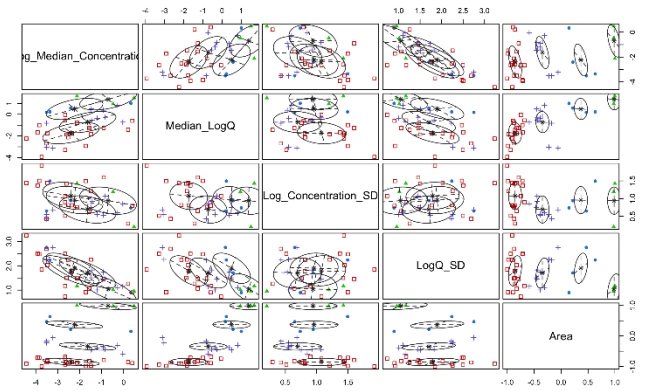


Fig. 3. Clustering example using the library mclust for USGS stations based on median and standard deviation of nitrate + nitrite-N concentration, logarithm of discharge, and drainage area size. The four clusters are: Cluster 1 (blue circle), Cluster 2 (red square), Cluster 3 (green triangle), and Cluster 4 (purple plus symbol).

These results indicate an inverse correlation between discharge and nitrate + nitrite-N concentration in this station. No significant gaps or censored observations are in the dataset and a uniform number of samples have been collected throughout the year. In the case of Mosquito Creek, for any given discharge, there is a range of nitrate + nitrite-N concentration that varies in almost two orders of magnitude.

The upper right graph shows the seasonal variation on concentration in time. This pattern was observed in many stations of the training dataset and is described by the SinDY and CosDY terms in Equation (2).

The lower left corner of Figure 2 shows the distribution of nitrate + nitrite-N concentrations by month. The box plots suggest that samples were collected throughout the year with a relative similar frequency for all the months of the year. The lack of samples at specific months could have a negative impact on seasonal components of the model.

Finally, the lower right corner shows two box plots that compare the discharge distribution for the period of analysis versus when samples were collected. In this case, the discharge values for samples collected do not seem to be different than the overall discharge distribution.

Figure 3 and Table I show the results of the clustering analysis based on the size of the drainage area, concentration/discharge median and standard deviation of all the stations included in the analysis. The analysis indicated that the best cluster model was the ellipsoidal, equal volume, shape and orientation (EEE). In this case, the model was generated with four components.

Figure 3 shows a positive correlation between the nitrite + nitrate-N concentration median and the size of the drainage area. The four clusters are clearly identified in the upper right panel. In addition, there is a positive correlation in the four clusters for drainage area and the median of the logarithm of discharge (LogQ) as expected. It seems that smaller watersheds tend to generate low nitrite + nitrate-N concentrations than larger watersheds. For example, in the panel that compares the median of the logarithm of the nitrite + nitrate-N concentration with the median of the logarithm of the discharge, the center of the ellipsoid for the cluster of small watersheds (Cluster 2, c = 0.01 mg/L) is about 50 times smaller than the center of the ellipsoid for large watersheds (Cluster 3, c = 0.49 mg/L).

Additional correlations were also observed in this figure. As the watershed size and concentration increase, there is a reduction in the standard deviation of the discharge. This inverse correlation is shown in the third scatter plot in the top row of Figure 3. The mean standard deviation of the stations located in Cluster 3 (approximately 2.8 m^3/s) was almost half of the mean standard deviation of Cluster 2 (approximately 6 m^3/s).

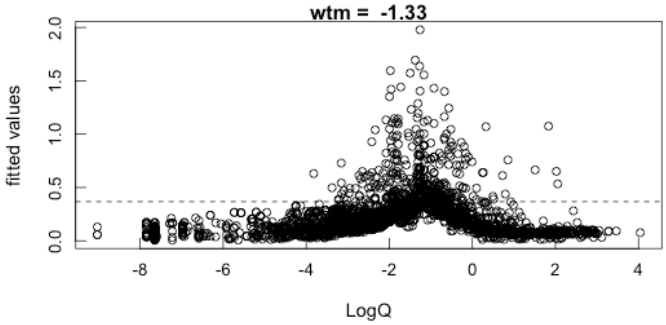


Fig. 4. Fitted values from the BRT model for Cluster 2 (low nitrate + nitrite-N concentration and watersheds with small drainage area). Discharge values in cubic meters per second (m^3/s).

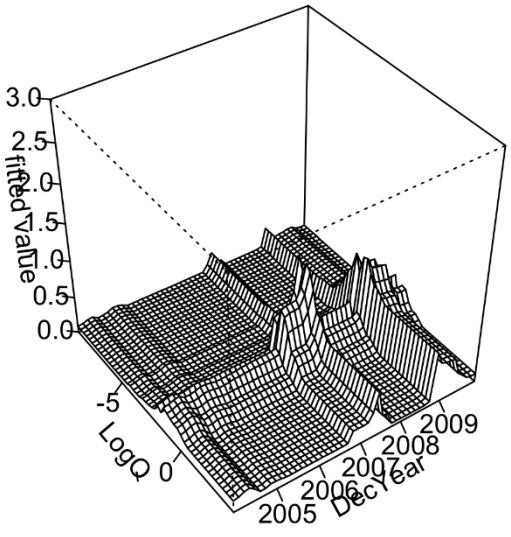


Fig. 5. Estimates of nitrate + nitrite-N concentration in Cluster 2 from the Boosted Regression Tree model at different conditions of discharge.

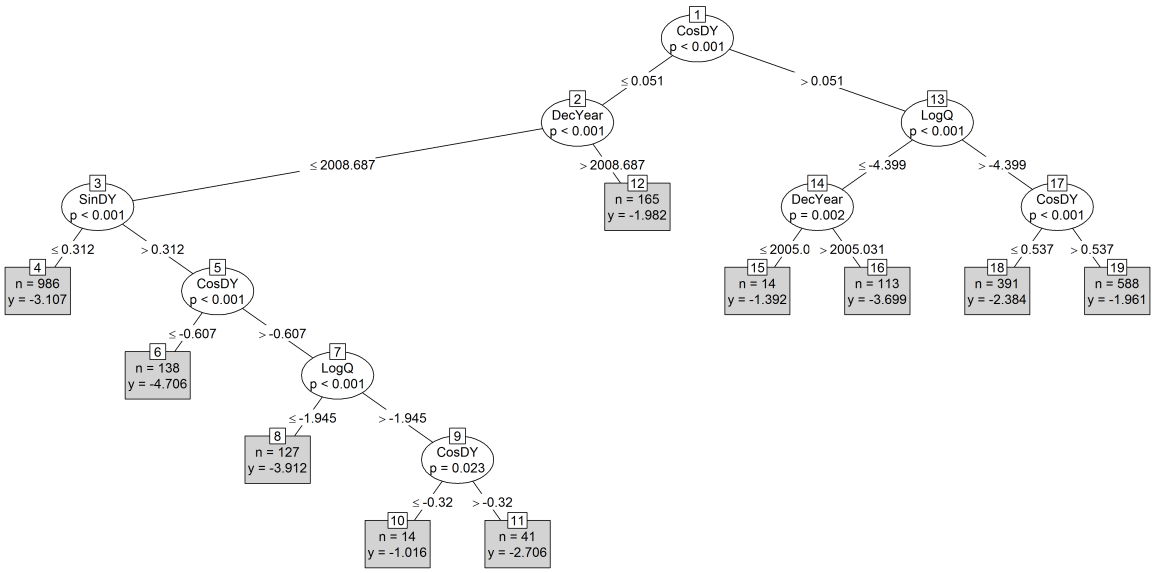


Fig. 6. Results of the Conditional Inference Tree for the estimation of nitrate + nitrite-N concentration using the stations located in Cluster 2.

Another observation is that, independently of the cluster, there is not a clear difference in the standard deviation of the logarithm of concentration (approximately 2.7 mg/L) for any change in area, discharge, or concentration except for Cluster 4, which was 25% smaller than the other clusters (approximately 2.0 mg/L).

Figure 4 shows the fitted values using BRT models for Cluster 2. It appears that nitrate + nitrite-N concentrations remain relatively constant except for discharge values near 0.25 m^3/s. This supports the hypothesis that, in small watersheds, nitrite + nitrate-N concentrations increase as runoff contributions from different watershed sources are transported during storm events as well as resuspension. Once a peak is reached, additional discharge contributions create a "dilution effect" reducing the concentration. This was observed in 19 stations located in Florida and Alabama, and might not be representative of other locations.

Figure 5 shows the Figure 4 results considering temporal changes. Between 2004 – 2009, there was always an increase in nitrite + nitrate-N concentrations for discharge values close to 0.25 m^3/s but there were two significant peaks in 2007 and 2009. This plot shows that temporal changes could be significant to changes in concentration.

Figure 6 shows the results of the Conditional Inference Tree for nitrate + nitrite-N concentration for the stations located in Cluster 2. The model generated 10 potential concentration estimates for all the combinations on time, discharge and season. The results indicate that the seasonal parameter CosDY was the most important criterion for classification, followed by the temporal variable and the logarithm of the discharge.

Interestingly, the break in CosDY term is highly associated with the logarithm of discharge values, and was for values smaller and greater than 0.051. This corresponds to the end of March and the beginning of October, which are the spring and summer seasons with high temperatures and elevated agricultural activity. In addition, an analysis of the observed discharges in the Cluster 2 stations indicated that low flows were observed in this period of time. On the other hand, the period between October and March was in general low in temperature and elevated water discharges.

High nitrate + nitrite-N concentration values were observed if the sample was collected after mid-2008 or approximately during the period November to March (CosDY > 0.537).

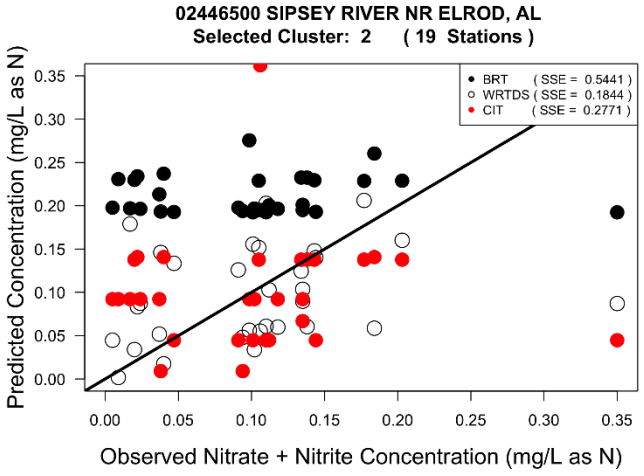


Fig. 7. Observed versus predicted nitrate + nitrite-N concentration values for the Sipsey River station at Elrod, AL. Predicted concentrations were obtained using WRTDS, CIT and BRT models.

Finally, Figure 7 shows the observed versus predicted values at the Sipsey River station near Elrod, Alabama using the CIT, BRT and WRTDS methods. It is expected that good estimates should all fall along a straight line with slope 1 that crosses the origin. During the clustering process, the Elrod station was assigned to Cluster 2. A total of 19 stations of the training dataset have similar discharge and nitrate + nitrite-N concentration characteristics to the Elrod station.

Figure 7 shows the concentration estimates using the WRTDS model in white circles, the BRT estimates in black, and the CIT estimates in red. Note that the estimates from the BRT and CIT are levels. For example in the case of the CIT estimates, the predicted values will correspond to one of the 10 grey boxes shown in Figure 6. The Sum of Square Error (SSE) for the three models indicated that the WRTDS model was better than CIT and BRT.

In general, the estimates of the BRT values are higher than the observed concentrations. On the other hand, CIT estimates have a better distribution than the BRT estimates, but the SSE was lower than the results obtained using the WRTDS method.

IX. Discussion

The use of machine learning techniques has great potential for improving the estimation of water constituents in rivers and streams using clustered data from multiple stations. WRTDS is a powerful tool because it uses a windowing procedure to include the non-stationarity component in the model. BRT, CIT, and other methods like Random Forest can easily identify changes in concentration with time. Unfortunately, large number of samples is needed to identify all potential combinations of factors that allow the use of time and discharge in the estimation of concentration.

The results described in this paper are initial steps toward the generation of complex and sophisticated tools that could assist in the identification of patterns and trends between discharge and concentration. WRTDS uses parametric survival regressions in the estimation of water constituents. The estimation of concentration can be greatly improved by combining the WRTDS method with clustering and machine learning methods including Boosted Regression Trees and Conditional Inference Trees.

This research work also explored the use of survival random forest methods currently available in the Comprehensive R Archive Network (CRAN). Unfortunately, the methods found were valid for right-censored data, whereas water quality involves left-censored data. Additional efforts are needed to explore the development of survival machine learning techniques applicable to left-censored data.

X. Conclusions

The use of the contributing drainage area size in the clustering procedure appears to be an important factor in the identification of clusters for water quality stations in rivers and streams. Indeed, there is an expected positive correlation between drainage size and magnitude of the discharge values. The advantage of using drainage area during clustering is that there is less overlap among clusters compared with clusters driven by discharge values.

Water quality stations in rivers and streams are usually located before or after where two or more major tributaries/rivers meet. The results reported in this paper suggest that there is a large number of water quality stations located in small watersheds. As rivers and major streams merge, the size of the watershed area increases, drastically driving the models to the identification of clusters with stations located in large watersheds. It is expected that, as the drainage area size of the stations located in the test data set increases, it is less likely to have variety in the stations selected during the clustering process.

Machine learning techniques that utilize clustering, classification, and survival information have great potential in the analysis of water quality data. Fortunately, there is great interest in developing techniques for such applications. The potential combination of trees, clustering, and windowing techniques, such as the one used in WRTDS, creates multiple opportunities for developing tools that improve early detection of water quality changes in our rivers and streams. Early detection of significant changes in the quality of the rivers and streams is critical for many sensitive water systems including drinking water sources and protected ecological areas.

Future work in this area includes the design of machine learning tools that determine and utilize the best methods to aggregate water quality data from multiple stations and reduce the overall bias of generated surface water quality concentration estimates.

REFERENCES

[1] R.M. Hirsch, D.L. Moyer, and S.A. Archfield, "Weighted regressions on time, discharge, and season (WRTDS), with an application to Chesapeake Bay River inputs," JAWRA Journal of the American Water Resources Association, vol 46, pp. 857–880, October 2010. doi: 10.1111/j.1752-1688.2010.00482.x

[2] R Core Team (2013). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL http://www.R-project.org/

[3] R.M. Hirsch and L. De Cicco, dataRetrieval: Retrieval functions for USGS data. 2012. R package version 1.2.1.

[4] R.M. Hirsch and L. De Cicco, Exploration and Graphics for River Trends (EGRET): An R-package for the analysis of Long term Changes in Water Quality and Streamflow, Including the Water Qauality Method Weighted Regressions on Time, Discharge, and Season (WRTDS) Draft Manual. 2013. R package version 1.2.3.

[5] R.M. Hirsch, 2012. Flux of nitrogen, phosphorus, and suspended sediment from the Susquehanna River Basin to the Chesapeake Bay during Tropical Storm Lee, September 2011, as an indicator of the effects of reservoir sedimentation on water quality: U.S. Geological Survey Scientific Investigations Report 2012–5185, 17 p. http://pubs.usgs.gov/sir/2012/5185/

[6] Hirsch, R.M., D.L. Moyer, and S. Phillips, 2013. Determining Nutrient and Sediment Loads and Trends in the Chesapeake Bay Watershed by Using an Enhanced Statistical Technique. U.S. Geological Survey Science Summary (January 2013).

[7] Sprague, L.A., R.M. Hirsch, and B.R. Aulenbach, 2011. Nitrate in the Mississippi River and Its Tributaries, 1980 to 2008: Are We Making Progress? Environmental Science & Technology 45(17):7209-7216. DOI:10.1021/es201221s

[8] Maestre, A. Williamson, D. Ward, A. 2012. "Nutrient Fluxes in Rivers of the Mobile – Alabama River System using WRTDS". 2012 Alabama Water Resources Conference. Orange Beach, Alabama.

[9] Moyer, D.L., Hirsch, R.M., and Hyer, K.E., 2012, Comparison of two regression-based approaches for determining nutrient and sediment fluxes and trends in the Chesapeake Bay watershed: U.S. Geological Survey Scientific Investigations Report 2012-5244, 118 p. http://pubs.usgs.gov/sir/2012/5244

[10] Markus, M., Demissie, M., Short, M., Verma, S., and Cooke, R. 2013. A Sensitivity Analysis of Annual Nitrate Loads and the Corresponding Trends in the Lower Illinois River. Journal of Hydrologic Engineering. doi:10.1061/(ASCE)HE.1943-5584.0000831

[11] Murphy, J.C., R.M. Hirsch, and L.A. Sprague, 2013. Nitrate in the Mississippi River and its tributaries, 1980 - 2010: An update. U.S. Geological Survey Scientific Investigations Report 2013–5169, 31p. http://pubs.usgs.gov/sir/2013/5169

[12] Lewin, W.-C., Mehner, T., Ritterbusch, D., and Bramick, U. (2014). The Influence of Anthropogenic Shoreline Changes on the Littoral Abundance of Fish Speces in German Lowland Lakes Varying in Depth as Determined by Boosted Regression Tress. Hydrobiologia, 724(1), 2014: 293-306. doi: 10.1007/s10750-013-1746-8

[13] Chung, Y. S. (2013). Factor complexity of crash occurrence: An empirical demonstration using boosted regression trees. Accident Analysis & Prevention, 61, 107-118. doi: 10.1016/j.aap.2012.08.015

[14] Soykan, C. U., Eguchi, T., Kohin, S., & Dewar, H. (2014). Prediction of fishing effort distributions using boosted regression trees. Ecological Applications, 24(1), 71-83. doi: 10.1890/12-0826.1

[15] le Roux, P. C., Luoto, M. (2014), Earth surface processes drive the richness, composition and occurrence of plant species in an arctic–alpine environment. Journal of Vegetation Science, 25: 45–54. doi: 10.1111/jvs.12059

[16] Strobl, C., Malley, J., and Tutz, G. (2009). An Introduction to Recursive Partitioning: Rationale, Application, and Characteristics of Classification and Regression Trees, Bagging, and Random forests. Psychological Methods, 14 (4), 323–348.

[17] Hothorn, T., Hornik, K., Strobl, C., and Zeileis, A. (2014). Package 'party': A Laboratory for Recursive Partytioning, CRAN Repository, http://cran.r-project.org/web/packages/party/party.pdf.

[18] Strasser, H. and Weber, C. (1999). On the asymptotic theory of permutation statistics. Mathematical Methods of Statistics, 8, 220–250.

[19] De Ath, G. and Fabricius, K. (2000). Classification and Regression Trees: A Powerful yet Simple Technique for Ecological Data Analysis. Ecology, 81 (11), 2000: 3178–3192. doi: 10.1890/0012-9658(2000)081[3178:CARTAP]2.0.CO;2

[20] De Ath, G. (2007). Boosted Trees for Ecological Modeling and Prediction. Ecology, 88 (1), 2007: 243–251. doi: /10.1890/0012-9658(2007)88[243:BTFEMA]2.0.CO;2

[21] Elith, J., Leathwick, J.R., Hastie, T. (2008). A Working Guide to Boosted Regression Trees. Journal of Animal Ecology, 77 (4), 2008: 802-813. doi: 10.1111/j.1365-2656.2008.01390.x

[22] Williams, G.P. (1989). Sediment Concentration versus Water Discharge During Single Hydrologic Events in Rivers. Journal of Hydrology, v. 111, p. 89 – 106

[23] Johnes, P.J. (1996). Evaluation and Management of the Impact of Land Use on the Nitrogen and Phosphorous Load Delivered to Surface Waters: the Export Coefficient Modelling Approach. Journal of Hydrology, 183: 323 – 349

[24] Fraley, C., Raftery, A., Murphy, T., Scrucca, L. (2012). MCLUST Version 4 for R: Normal Mixture Modeling for Model-Based Clustering, Classification, and Density Estimation. Technical Report No. 597. Department of Statistics, University of Washington. Seattle, USA.

[25] Fraley, C. and Raftery, A. (1999). MCLUST: Software for Model-Based Cluster Analysis. Journal of Classification, 16: 297-306

[26] Bowers, M. The Distributions of Seasonal Rivers Flows: Lognormal or Power-Law? Master of Science Thesis Purdue University, West Lafayette, Indiana. May 2012

[27] Ridgeway G. with contributions from others (2013). gbm: Generalized Boosted Regression Models. R package version 2.1. http://CRAN.R-project.org/package=gbm

[28] Hastie, T., Tibshirani, R., Friendman, J. (2009). The Elements of Statistical Learning. Data Mining, Inference, and Prediction. Springer Science+Business Media. New York, NY.

[29] Jun, S. (2013). Boosted Regression Trees and Random Forests. Statistical Consulting Report for Michael Melnychuck. University of British Columbia.

Towards Optimizing the Selection of Neurons in Affordable Neural Networks

J. T. Ausonio[1], R. J. Mitchell[1], and W. Holderbaum[1]
[1]School of Systems Engineering, University of Reading, Reading, United Kingdom

Abstract— *This paper considers variations of a neuron pool selection method known as Affordable Neural Network (AfNN). A saliency measure, based on the second derivative of the objective function is proposed to assess the ability of a trained AfNN to provide neuronal redundancy. The discrepancies between the various affordability variants are explained by correlating unique sub group selections with relevant saliency variations. Overall this study shows that the method in which neurons are selected from a pool is more relevant to how salient individual neurons are, than how often a particular neuron is used during training. The findings herein are relevant to not only providing an analogy to brain function but, also, in optimizing the way a neural network using the affordability method is trained.*

Keywords: ANN, neuron selection, optimization, saliency, redundancy.

1. Introduction

The brain's ability to reorganize and reinforce its functions is paramount to its ability to produce learned responses. As the biological brain loses neuron cells and the relationships they have made with surrounding cells (through either damage or decay) the inherent knowledge is ultimately lost. Yet, from a system perspective a brain can still produce acceptable output in spite of these changes within its foundation. Studies into brain neurogenesis, syntapogenesis, and sprouting show that new cells are being made and fresh connections established between sensory neurons, motor neurons, and neurons within the hippocampus. This neuroplasticity is the foundation for producing learned responses in the presence of a highly dynamic and structurally redundant network [1] [2] [3] [4].

The highly volatile neural network exhibited by the human brain presents a stark contrast to that of the structurally and functionally static designs of Artificial Neural Network (ANN) applications used in maths and engineering. It is worth considering whether mimicking this structural redundancy in ANN applications can provide any benefit. Structural redundancy within this study is defined as the duplication of highly salient neurons which, in the event of neuron loss, preserves network performance.

Various works by Uwate and Nishio [5][6][7] detail numerous methods and validations for the implementation of, what they have called, an Affordable Neural Network. The inspiration behind the Affordable Neural Network (AfNN) is based on providing an analogy to this neuron redundancy by detailing a mechanism which can manage a surplus of hidden layer neurons during training by. Affordability, in this sense, refers to the limited number of neurons which can be utilized at any given time. Using a Backpropagation (BP)-based Multilayer Perceptron (MLP) with more than the optimal number of neurons in a hidden layer, Uwate and Nishio were able to show that convergence was not only possible but, in some cases, improved when compared with a classic, structurally static, approach. The efficiency and benefits of this method were highly dependent upon the method by which an optimal number of neurons are selected from the hidden layer but detailed the use of chaotic oscillation as the best way of achieving this affordability selection [5].

However, the studies carried out by Uwate and Nishio did not provide sufficient analysis explaining why their preferred method works. Nor do they provide evidence that they have achieved the most optimal solution to affordable neuron selection.

Building upon these earlier studies, Uwate and Nishio make claims regarding the AfNNs ability to sustain damage (i.e. loss of structure) due to the duplication of weight values within the input-to-hidden and hidden-to-output connections [7]. This misconception regarding weight magnitude equaling saliency is the subject of LeCun's paper on "Optimal Brain Damage" [8]. In the paper by LeCun et. al. a method for measuring the actual saliency of a neuron's connections using the second derivative of the objective function is presented. A modification of this measurement is presented below as the means for not only measuring structurally redundancy within an AfNN but also to compare variations of this design.

This study revisits the findings presented by Uwate and Nishio and provides a more in-depth analysis towards, and quantifiable, non control-relative, measurements of, optimizing affordable neuron selection. This measurement is an extension on the saliency calculation presented by LeCun et. al.

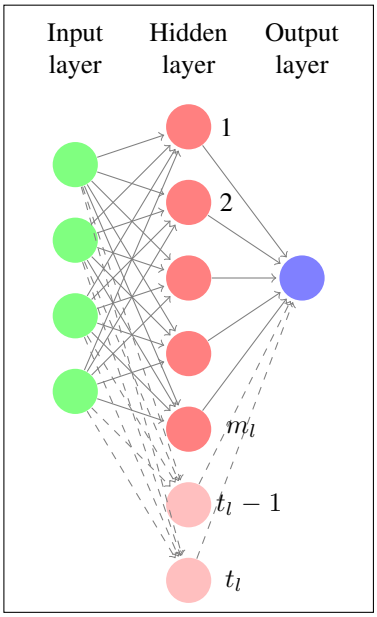
2. Affordable Neuron Selection

This study (re)defines the AfNN as a feed-forward MLP having extra neurons in a single hidden layer, thereby making the total number of neurons in the hidden layer

more than that which is considered optimal for a given data set. These neurons are provided at network design but the ratio of total neurons to selectable neurons is dependent upon the training data and is determined through trial and error. During BP training a subset of the total neurons are selected for participation whilst the rest remain non-contributory. This non-participation is preserved when computing errors, updating weight values and computing neuron saliency. The number of neurons selected is also a parameter to the system defined during network design and is equal to the optimal number of neurons in the hidden layer for a classic MLP approach.

Figure 1 provides a notional construction of an AfNN. In this diagram the hidden layer is designated l, where m_l is the number of optimal neurons in hidden layer l (i.e. the number of neurons chosen each time data is presented to the network), t_l represents the total number of affordable neurons in layer l where $t_l = m_l r_l$, and r_l is the ratio of t_l to m_l in layer l. As mentioned earlier, the optimal value of r_l is currently found through discovery but is always greater than 1 (i.e. $m \leq t$). If $m = t$ then the network is considered to be the classic MLP variant.

Fig. 1: Neural network with affordable neurons, modified from [5].



All neurons share the same local induction and activation functions and utilize a back-propagation learning method. Considering a training set N, where the training vector $n \in N$, the induced local field for neuron j within layer l, is defined as $v_j(n)$ and its corresponding activation is denoted φ_j. Using this activation, the output of neuron j (denoted $y_j(n)$) is as shown in equation 1.

$$y_j(n) = \varphi(v_j(n))\psi_j(n) \tag{1}$$

where $\psi_j(n)$ represents the affordability scalar, or participation designation, for neuron j at presentation of training vector n.

Similarly, when performing weight updates during the feedback process, the change in weight value between neuron i within layer $l-1$ and neuron j within layer l (denoted $w_{ij}(n)$) is defined as follows

$$\Delta w_{ij}(n) = \eta \delta_j(n) y_j(n) \tag{2}$$

where η is the learning rate and δ represents the local gradient for neuron j.

It should be noted that the inclusion of the affordability scalar, ψ_j, into the weight update is implied through the reuse of $y_j(n)$. Therefore, equation 1 is the primary focus for implementing the AfNN algorithm and will be shown with variance. Specifically, each of the affordability methods compared in this study will alter the calculation of ψ_j.

The following sections define the various neuron affordability methods to be studied. In particular, this study will revisit the random and chaotic selection methods mentioned by Uwate and Nishio in [5] but also provide a second chaotic map, and a new cyclic selection rule, for comparison. Following that, the learning results of each variant will be compared using a measurement of neuron saliency to highlight the relative benefits of each method against the goals set forth earlier.

2.1 Random Affordability

Following on from work by Uwate and Nishio, the comparison against a randomly selected affordability model is meant to provide comparison to, and credence towards, the use of a chaotic selector. We define $\psi_j^R(n)$ as the random affordability variant of $\psi_j(n)$ as follows

$$\psi_j^R(n) = \begin{cases} 1 \text{ if } r_j(n) \geq u_l(n) \\ 0 \text{ otherwise} \end{cases} \tag{3}$$

where $r_j(n)$ represents a random value associated with neuron j during the generalization of input n, $u_l(n)$ represents the m^{th} largest random value in layer l, and m denotes the optimal number of neurons in layer l as defined in section 2 .

The random number generator used for this study is provided by Boost libraries [9]. The seeding of the generator is done at system startup and this one seed is shared amongst all neurons.

2.2 Chaotic Affordability

The chaotic MLP implementation specified in [5] was described as using a skew tent map associated with each hidden layer neuron where all are given "different initial values" and are updated at every learning. This study utilizes more descriptive methods for initializing each chaotic value

and incrementing of the chaotic values. We define $\psi_j^T(n)$, the tent map affordability variant of $\psi_j(n)$, as follows

$$\psi_j^T(n) = \begin{cases} 1 \text{ if } \tau_j(n) \geq u_l(n) \\ 0 \text{ otherwise} \end{cases} \quad (4)$$

where $\tau_j(n)$ is the chaotic value associated with neuron j during generalization of training vector n, defined below. $u_l(n)$ is carried forward from the previous section.

$$\tau_j(n) = \begin{cases} \frac{2\tau_j(n^{-1})+1-\alpha}{1+\alpha}, & \text{if } -1 \leq \tau_j(n^{-1}) \leq \alpha \\ \frac{-2\tau_j(n^{-1})+1+\alpha}{1-\alpha}, & \text{if } \alpha < \tau_j(n^{-1}) \leq 1 \end{cases} \quad (5)$$

$$\tau_j(n^0) = rand(-1...1) \text{ where } \tau_j(n^0) \neq \alpha \quad (6)$$

where $\tau_j(n^{-1})$ is the chaotic value for neuron j during the previous training vector, denoted n^{-1}. The first chaotic value, $\tau_j(n^0)$, is set randomly. Finally, α is the tent map input parameter designating skewness. Equation 5 and the setting of α to 0.05 within this study are duplicated from [5].

A logistic map is also presented in place of the tent map in order to provide even more data for comparison. For the logistic tent map we define $\psi_j^L(n)$ as follows

$$\psi_j^L(n) = \begin{cases} 1 \text{ if } \iota_j(n) \geq u_l(n) \\ 0 \text{ otherwise} \end{cases} \quad (7)$$

$$\iota_j(n) = (\iota_j(n^{-1})\beta)(1 - \iota_j(n^{-1})) \quad (8)$$

$$\iota_j(n^0) = rand(0...1) \quad (9)$$

where β represents the input parameter to the logistic map. This value is set to 3.8275 but any value or β after the onset of chaos (between 3.56995 and 4.0) will suffice.

2.3 Cyclic Affordability

Through investigations this study has produced a simplified affordability method which has the benefit of being deterministic but also very predictable. Particularly, it will be shown that providing a consistent selection of groups of neurons will improve upon the reinforcement of a neurons saliency within the network. We define $\psi_j^C(n)$ as follows

$$\psi_j^C(n) = \begin{cases} 1 \text{ if } \gamma_j(n) \leq m \\ 0 \text{ otherwise} \end{cases} \quad (10)$$

$$\gamma_j(n) = \begin{cases} \gamma_j(n^{-1}) + 1, \text{ if } \gamma_j(n^{-1}) < t_l \\ 0, \text{ otherwise} \end{cases} \quad (11)$$

$$\gamma_j(n^0) = j \quad (12)$$

we use j as the initial value for γ in order to provide unique values to each neuron within the range 0 to $t-1$, inclusively.

3. Comparing Affordable Selection Methods

In order for an objective evaluation to occur a quantifiable measurement must be made against all variants of the AfNN design. For the purposes of this study, this measurement is in the form of neuron saliency using the second derivative of the objective function. After each AfNN variant has been trained each neuron will be measured against how salient it is towards the networks total error. The form of this saliency measurement is based on Optimal Brain Damage (OBD) by LeCun et. al. [8]. Whilst LeCun considered the use of entropy in order to purposefully remove the least salient items from a neural network the same calculation can be used in determining the most salient units.

However, a problem arises when considering the volatile nature of the network architecture. At any given time, a particular neuron may not be contributing towards total error and, therefore, will have no saliency against the subsequent output. To overcome this, the calculation of h_k below is modified to include the participation scalar ψ. This means that the saliency measure is only calculated for neuron j for the trials in which it contributed towards the total error, making the calculation of saliency relevant to that neuron without compromising the method in which h_k is generated within the AfNN method. The examples used in this research have only one hidden layer with one output neuron, resulting in simpler calculation for saliency. The following equations are used to calculate the saliency, s_j, of the hidden neurons, as derived from [8]

$$s_j = \frac{h_k w_{jk}^2}{2} \quad (13)$$

$$h_k = \sum_{n \in N} W(n) \quad (14)$$

$$W(n) = \frac{\partial^2 E}{\partial v_k^2} \varphi(v_k(n)) \psi_j(n) \quad (15)$$

$$\frac{\partial^2 E}{\partial v_k^2} = 2\varphi'(v_k(n))^2 - 2[d(n) - \varphi(v_k(n))]\varphi''(v_k(n)) \quad (16)$$

where w_{jk} denotes the weight connection neuron j in the hidden layer with the (only) output neuron k and $d(n)$ represents the desired output of the network for trail n within set N.

In addition to this, it is important to keep track of the saliency against unique neuron groupings. The following was derived for this study to calculate the sum of h_k relative to unique neuron groupings:

$$U(n) = \sum_j 2^j \psi_j(n) \quad (17)$$

$$U_p \in U(t_l) \quad (18)$$

$$U(t_l) = \{1, ..., 2^{t_l+1} - 1\} \tag{19}$$

$$O_p(n, U_p) = \begin{cases} \frac{\partial^2 E}{\partial v_k^2} \varphi(v_k(n)), \text{where } U(n) = U_p, \\ 0 \text{ otherwise} \end{cases} \tag{20}$$

$$h_p = \sum_{n \in N} O_p(n, U_p) \tag{21}$$

Where $U(t_l)$ represents the set of all unique hidden layer neuron group identifiers, $U(n)$ is the hidden layer neuron group identifier at presentation of vector n, and U_p is element p within set $U(t_l)$. O_p, therefore, represents the saliency of grouping p at presentation of vector n and h_p represents the summed saliency against unique group p for all training vectors in N.

The intention of tracking the saliency of each neuron, s_j is to confirm that a structural redundancy is being produced using the AfNN method whereby s_j distributions are being replicated within the hidden layer pool. The purpose of tracking each unique groupings saliency, h_p, is to show that group selections are more directly correlated to neuron saliency than individual selection. Together these two saliency measurements, s_j and h_k, are used to show that AfNN methods which more consistently select unique groups of neurons will exhibit more meaningful structural redundancy of highly salient neurons.

4. Simulated Results

Two data sets are used in testing the aforementioned affordability methods. The first is a reproduction of the x^2 dataset used by Uwate and Nishio [5]. The other is the well known Iris data set (IDS) [10]. The goal is to train five network variants (the five affordability methods mentioned earlier) with the two training sets for ten network configurations total. Table 1 shows, per network configuration and data set permutation, statistics on the average error, calculated over all trials in one epoch against a trained network. This data shows that, with regard to error rates, each AfNN variant produced different levels of accuracy on average. Further, whilst only somewhat portrayed in this table through the variance of total error, the learning rates of the affordability methods, particularly those of the random and chaotic types, were highly erratic throughout training.

Table 1: Average error per epoch after training - results per network configuration.

Data Set	Afford. Method	Avg. Err	Min Err	Max Err
IDS	Classic	2.96%	2.32%	3.44%
IDS	Random	7.65%	5.94%	9.85%
IDS	Chaos (log)	7.13%	6.31%	8.67%
IDS	Chaos (tent)	7.09%	6.08%	9.49%
IDS	Cyclic	3.48%	2.48%	4.57%
x^2	Classic	3.0%	2.96%	3.05%
x^2	Random	8.58%	7.12%	9.48%
x^2	Chaos (log)	8.4%	6.43%	10.43%
x^2	Chaos (tent)	8.12%	6.97%	10.09%
x^2	Cyclic	6.37%	2.94%	8.47%

After a network is trained the saliency, s_j is calculated using equation 13 for each neuron in the hidden layer for each network variant. This is duplicated across twenty sessions and a histogram of the results are presented in figures 2 and 3. The number of times a particular neuron is utilized within the network, c_j, based upon its relevant value for ψ is also retained. From this, the correlation $\rho(s_j, c_j)$, is calculated between the number of times a neuron was utilized, c_j, and that neurons s_j value.

Similarly, the value h_p is calculated for each unique grouping of hidden layers neurons, p, utilized during testing. A correlation, $\rho(h_p, c_p)$, is also calculate between the number of times a unique group is utilized, c_p, and its associated value h_p. The histogram of the "group" correlation coefficients are presented alongside the "individual" correlation values, per network configuration and data set in figures 4 through 11. They are organized by data set and affordability method in order to compare, relative to one affordability method and one data set, the importance on individual neuron reinforcement versus group reinforcement and selection.

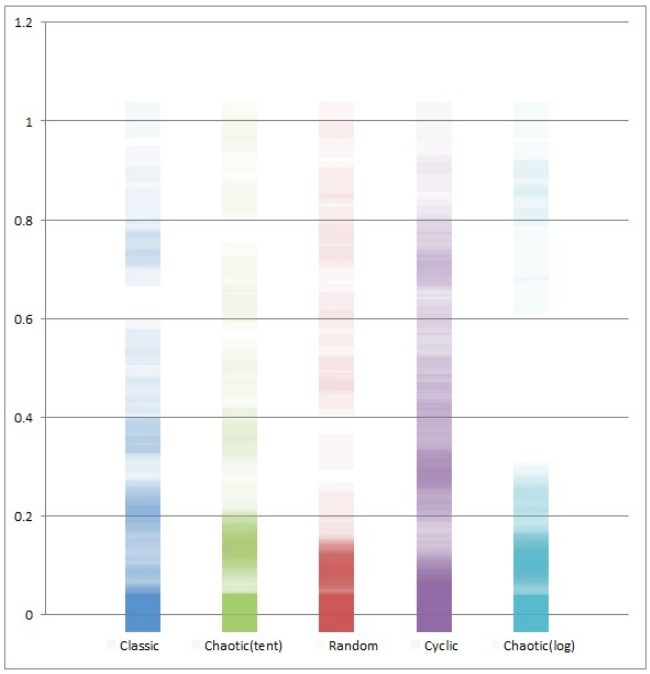


Fig. 2: Distribution of s_j for each affordability method, normalized between zero and one, against the Iris Data Set

In comparing both datasets within figures 2 and 3, the classic, or "full-affordability" model is used as a control in that its saliency measurements provide the rubric against which the random, chaotic, and cyclic affordability methods are measured.

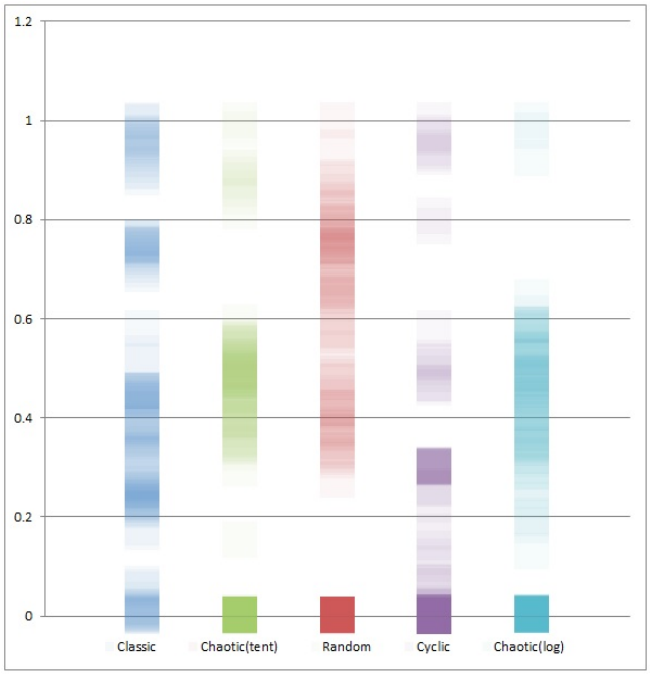


Fig. 3: Distribution of s_j for each affordability method, normalized between zero and one, against the x^2 Data Set

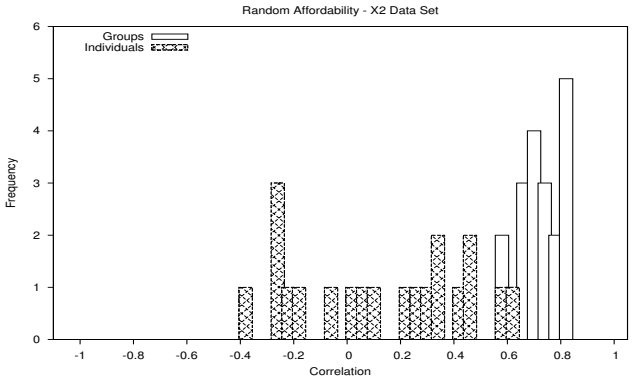


Fig. 4: Histogram based on the x^2 data set between frequency of random selection, c_j and c_p, against the correlations, $\rho(s_j, c_j)$ and $\rho(h_p, c_p)$, for individuals and unique selection groups, respectively.

5. Discussion

Once a network is trained, the relationship between s_j and the networks total error, in relation to affordability selection, is that not selecting highly salient neurons will lead to higher error rates. In other words, AfNN can be viewed as a short-term selective pruning of neurons. It follows that the neurons being selected through an affordability method are, preferably, those with the highest saliency. Further, a meaningful structural redundancy of neurons will ensure that neurons with high values for s_j are selected because there are simply more of them.

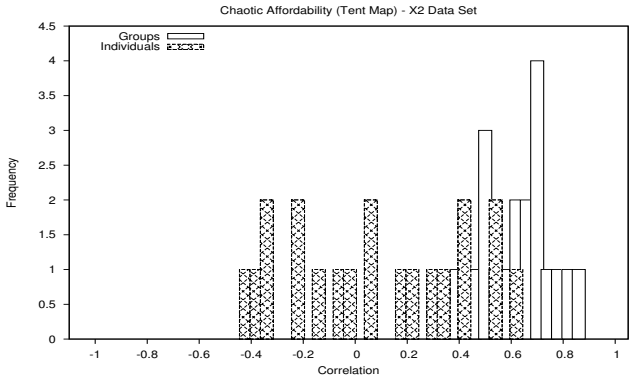


Fig. 5: Histogram based on the x^2 data set between frequency of chaotic selection, c_j and c_p, against the correlations, $\rho(s_j, c_j)$ and $\rho(h_p, c_p)$, for individuals and unique selection groups, respectively.

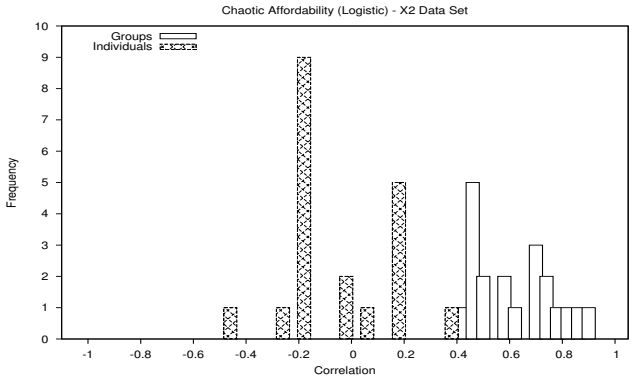


Fig. 6: Histogram based on the x^2 data set between frequency of chaotic selection, c_j and c_p, against the correlations, $\rho(s_j, c_j)$ and $\rho(h_p, c_p)$, for individuals and unique selection groups, respectively.

In terms of total error, the cyclic affordability method achieved the lowest error of any trial when compared to the baseline static structure (classic MLP). This method was also the only variant which satisfactorily classified all data points in the IDS. The highest error of any trial was incurred using random affordability which portrays erratic training and lowest average error in both data sets. As the cyclic affordability method was designed specifically to ensure consistent group selection it follows that a lower error rate herein supports the claim that consistent group selection leads to lower average error after training.

If this claim is to hold true then, in the first instance, the correlation between group selection and group saliency, $\rho(h_p, c_p)$, would need to highlight a dependent relationship. Figures 4 through 11 provide this evidence. The group correlation values, $\rho(h_p, c_p)$, across all variants average above 0.5 whilst similar calculations against individual neuron reinforcement, $\rho(s_j, c_j)$, stipulate no relationship between

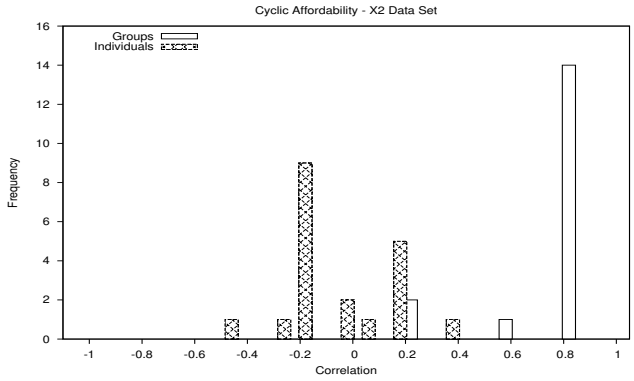


Fig. 7: Histogram based on the x^2 data set between frequency of cyclic selection, c_j and c_p, against the correlations, $\rho(s_j, c_j)$ and $\rho(h_p, c_p)$, for individuals and unique selection groups, respectively.

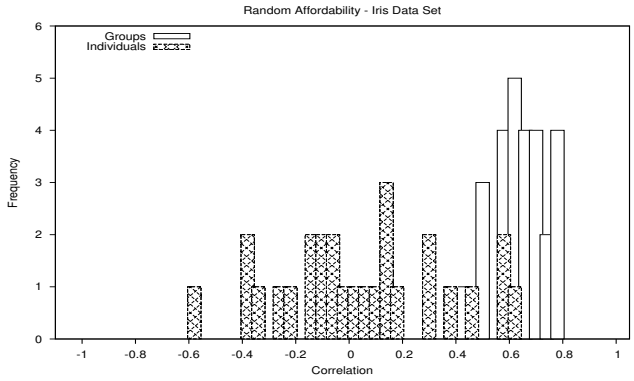


Fig. 8: Histogram based on the Iris data set between frequency of random selection, c_j and c_p, against the correlations, $\rho(s_j, c_j)$ and $\rho(h_p, c_p)$, for individuals and unique selection groups, respectively.

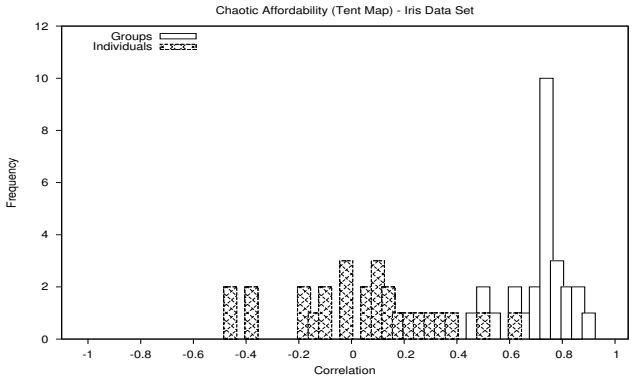


Fig. 9: Histogram based on the Iris data set between frequency of chaotic selection, c_j and c_p, against the correlations, $\rho(s_j, c_j)$ and $\rho(h_p, c_p)$, for individuals and unique selection groups, respectively.

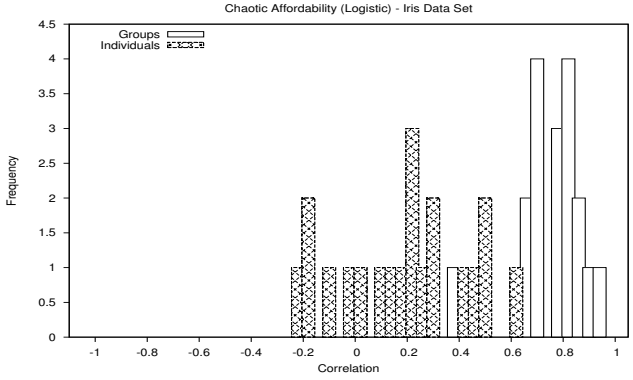


Fig. 10: Histogram based on the Iris data set between frequency of chaotic selection, c_j and c_p, against the correlations, $\rho(s_j, c_j)$ and $\rho(h_p, c_p)$, for individuals and unique selection groups, respectively.

how often a particular neuron is selected c_j and how salient it is towards the overall network error as measured by s_j.
Given that unique group selection reinforcement leads to groups with high saliency, and given the inherent relationship between group saliency h_p and the saliency of neurons within that group, it follows that the AfNN method which consistently reinforces neuron groups across all neurons in the hidden layer will produce meaningful structural redundancy and, therefore, lower error rates.
Along these lines, the distribution of individual s_j values in figures 2 and 3 provide evidence as to the existence of structural redundancy. Looking at the saliency distributions of the classic MLP against the IDS in figure 2 it can be discerned that three main groupings of saliency emerge. They can be viewed as the "near zero", above 0.2, and above 0.5 saliency clusters within the distribution. In this respect, the cyclic selection method most effectively exhibits a healthy saliency distribution in that it is not as heavily weighted near zero as the other variants are. Also, in terms of convergence, as mentioned earlier, the cyclic selection method performed best of out the AfNN variants (aside from a classic MLP approach which loses any benefit from structural redundancy). The random and chaotic variants do not contain as many highly salient neurons and, therefore, provide the least resistance to damage against the most important units of the network. For the x^2 data set the classic MLP distribution shows a larger amount of highly salient neurons than any of the affordability methods, with the random and cyclic methods being the closest.

6. Conclusion

In this study, data presented by Uwate and Nishio are revisited in order to more accurately measure the structural redundancy provided by and validate the claims regarding convergence rates of the AfNN method.
The findings herein suggest that weight value comparisons

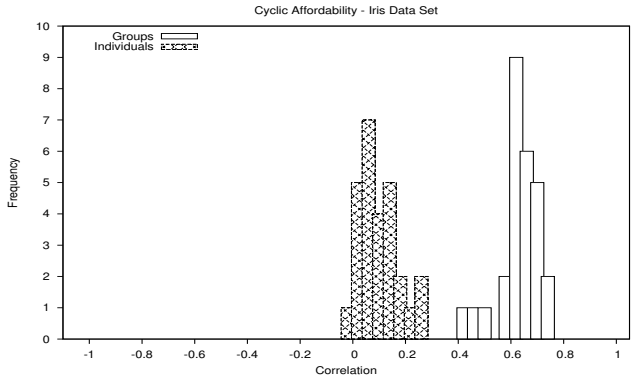


Fig. 11: Histogram based on the Iris data set between frequency of cyclic selection, c_j and c_p, against the correlations, $\rho(s_j, c_j)$ and $\rho(h_p, c_p)$, for individuals and unique selection groups, respectively.

do not present an accurate depiction of said redundancy and, instead, presents a modified version of saliency calculation derived from [8].

The data presented verifies the claim that all affordability variants can learn to classify to at least ninety percent accuracy and proves that they all provide a level of structural redundancy using a relevant measure of saliency. Further, it is clear that when training an AfNN the way in which groups are selected affects how the network as a whole performs and how relevant its neurons become. Conversely, how often individual neurons are selected has no bearing on how salient they become within the AfNN model.

However, the findings presented by this study are limited to the data sets which were tested and need to be validated in this respect. Also, further investigations into the performance impact of training an AfNN versus the loss of accuracy holds merit.

The implications of the findings presented in this study are relevant to studies in ANN damage recovery. Not only in providing structural redundancy in an effort to sustain damage whilst maintaining accuracy but, also, in retraining networks which use a form of neuronal pool selection.

Therefore, follow on work will be aimed at how this revelation can be utilized in re-training damage or lost neurons within the AfNN model and to what level of accuracy a network can achieve therein.

References

[1] M. Clergue and P. Collard, "Genetic algorithm for artificial neurogenesis," in *Evolutionary Computation Proceedings, 1998. IEEE World Congress on Computational Intelligence., The 1998 IEEE International Conference on*, 1998, pp. 410–415.

[2] M. Mulligan, L. Lu, R. Overall, G. Kempermann, G. Rogers, M. Langston, and R. Williams, "Genetic analysis of bdnf expression cliques and adult neurogenesis in the hippocampus," in *Biomedical Sciences and Engineering Conference (BSEC), 2010*, 2010, pp. 1–4.

[3] O. Michel and P. Collard, "Artificial neurogenesis: an application to autonomous robotics," in *Tools with Artificial Intelligence, 1996., Proceedings Eighth IEEE International Conference on*, 1996, pp. 207–214.

[4] R. P. Stroemer, T. A. Kent, and C. E. Hulsebosch, "Neocortical neural sprouting, synaptogenesis, and behavioral recovery after neocortical infarction in rats," *Stroke*, vol. 26, no. 11, pp. 2135–2144, 1995. [Online]. Available: http://stroke.ahajournals.org/content/26/11/2135.abstract

[5] Y. Uwate and Y. Nishio, "Back propagation learning of neural networks with chaotically-selected affordable neurons," in *Circuits and Systems, 2005. ISCAS 2005. IEEE International Symposium on*, may 2005, pp. 1481 – 1484 Vol. 2.

[6] Y. Uwate, Y. Nishio, and R. Stoop, "Scale-rule selection of affordable neural network for chaotic time series learning," in *Circuit Theory and Design, 2007. ECCTD 2007. 18th European Conference on*, 2007, pp. 811–814.

[7] Y. Uwate and Y. Nishio, "Learning process of affordable neural network for backpropagation algorithm," in *Neural Networks (IJCNN), The 2010 International Joint Conference on*, 2010, pp. 1–7.

[8] L. Cun, Denker, and Solla, "Optimal brain damage," *AT&T Bell Laboratories*, 1990. [Online]. Available: http://www.cnbc.cmu.edu/ plaut/IntroPDP/papers/

[9] R. Rivera and B. Dawes, "Boost c++ libraries," 2014. [Online]. Available: http://www.boost.org

[10] K. Bache and M. Lichman, "UCI machine learning repository," 2013. [Online]. Available: http://archive.ics.uci.edu/ml

Deep Neural Network Based Feature Representation for Weather Forecasting

James N.K. Liu[1], **Yanxing Hu**[1], **Jane Jia You**[1] , **and Pak Wai Chan** [2]
[1]Department of Computing, The Hong Kong Polytechnic University, Hong Kong
[2]Hong Kong Observatory, 134A Nathan Road, Kowloon, Hong kong

Abstract— *This paper concentrated on a new application of Deep Neural Network (DNN) approach. The DNN, also widely known as Deep Learning(DL), has been the most popular topic in research community recently. Through the DNN, the original data set can be represented in a new feature space with machine learning algorithms, and intelligence models may have the chance to obtain a better performance in the "learned" feature space. Scientists have achieved encouraging results by employing DNN in some research fields, including Computer Vision, Speech Recognition, Natural Linguistic Programming and Bioinformation Processing. However, as an approach mainly functioned for learning features, DNN is reasonably believed to be a more universal approach: it may have the potential in other data domains and provide better feature spaces for other type of problems. In this paper, we present some initial investigations on applying DNN to deal with the time series problem in meteorology field. In our research, we apply DNN to process the massive weather data involving millions of atmosphere records provided by The Hong Kong Observatory (HKO)*[1]*. The obtained features are employed to predict the weather change in the next 24 hours. The results show that the DNN is able to provide a better feature space for weather data sets, and DNN is also a potential tool for the feature fusion of time series problems.*

Keywords: Deep Neural Network, Stacked Auto-Encoder, Weather Forecasting, Feature Representation

1. Introduction

Deep Neural Networks (DNNs) or Deep Learning is the general term of a series of multi-layer architecture neural networks that are trained with the greedy layer-wise unsupervised pre-training algorithms[1], [2], [3]. Albeit controversial, DNNs have won great success in some fields including Computer Vision, Speech Recognition, Natural Linguistic Programming and Bioinformation processing. By applying the greedy layer-wise unsupervised pre-training mechanism, DNNs can reconstruct the original raw data set, in other words, DNNs can "learn" features with Neural Networks(NNs) instead of selecting features manually that we did traditionally[4]. And the intelligence models, like classifiers or regressors usually can obtain higher accuracy and better generalization with the learned features.

As its name suggested, DNN is a kind of NNs that structured by multiple layers. The word "Deep" indicates that such NN contains more layers than the "shallow" ones, which mainly includes the most widely used three-layer Feed Forward NNs in the past 30 years. Actually, multi-layer NN is not a new conception, some earlier studies have been conducted since 1990s[5], [6], but the successful implementation of multi-layer NNs was not realized until the provision of the novel Layer-wise unsupervised Pre-training mechanism by Hinton in 2006 that is employed to solve the training difficulties efficiently [1].

Although theoretically, a shallow NN with three layers trained with Back-Propagation(BP) has been proved that can approximate any nonlinear functions with arbitrary precision [7], when the number of hidden neurons is limited, the learning ability of shallow NNs may not be enough and poor generalization may be expected when using an insufficiently deep architecture for representing some functions. The significance of "deep" is that compared with shallow models, NN with deep architecture can provide a higher learning ability: functions that can be compactly represented by a deep architecture might be required to handle an exponential number of computational elements (parameters) to be represented by a deep architecture. More precisely, functions that can be compactly represented by a depth k architecture might require an exponential number of computational elements to be represented in a depth $k-1$ architecture [3].

Based on the Layer-wise unsupervised Pre-training mechanism, DNN can map the raw data set from the original feature space into a learned feature space layer by layer in the training process. In each layer, the unsupervised training may provide a kind of regularization to the data set and minimize the variance in each layer. Therefore, in the finally obtained feature spaces, classifiers or regressors have chances to obtain higher accuracy and better generalization. In some research areas, including computer vision [1], Speech recognition [8], Natural Linguistic Programming [9], and Bioinformation, DNNs have been reported achieving great success in the past five years.

The objective of our investigations is to explore the potential of DNN in other research domains. In previous research,

[1]http://www.hko.gov.hk/contente.htm

we note that for time series problem, a good representation of original feature space may be helpful for the applied model to get better performance [10]. Meanwhile, in time series problem, the correlations among features are obviously but not easy to be identified. If we can analyze the correlations and represent the features properly, the prediction accuracy is expected to be improved, and DNNs could be a reasonable and suitable tool to analyze the time series features. In this investigation, we apply a DNN model to predict the weather change in the next 24-hour period with a big data set. The massive data involving millions of weather records is provided by The Hong Kong Observatory (HKO). Our training method is to use the latest proposed greedy layer-wise unsupervised pre-training algorithm followed by a supervised fine-tuning. In detail, we choose a revised Auto-Encoder algorithm to build the network [11], the DNN is used to learn the features from the larger volume of raw data, and we evaluate the learned features according to the prediction accuracy. The contribution and significance of our investigation demonstrate that: compared with the classical models, NNs with Deep architectures can improve the prediction accuracy in weather forecasting domain, and our initial study gives the results that can show the potential of DNN on time series problems.

2. Weather Prediction Problem

The changes of climate that could impact people's daily life, therefore people never cease their efforts on predicting the trend of weather changes. Unlike data sets in other domain, weather data has some particularities. Specifically, there is season-to-season, and year-to-year variability in the trend of weather data. The cycle could be multi-month, multi-season or multi-year, and the main difficulty of investigations is to capture all the possible cycles.

Many significant research efforts are utilized to develop weather forecasting methods including artificial intelligence technologies that have been accepted as appropriate means for weather forecasting and reported encouraging results since 1980s [12], [13], [14]. Among many different intelligence models, single variable time series regression is the most fundamental and most widely applied one in weather forecasting, especially short-term predictions. Since our initial investigation is an exploration on the application of DNN in the area of weather forecasting, in this paper, we mainly concentrate on employing DNN to represent the feature space for single variable time series regression problem.

Generally speaking, for a certain variable, the objective of single variable time series regression is to find the relationship between its status in a certain future time point and its status in a series of past time points, and estimate its future status via:

$$v_t = f(v_{t-1}, v_{t-2}, \ldots, v_{t-n}) \quad (1)$$

The function f, can be obtained by employing different intelligence models such as Linear Regression, Generalized Linear Model, Autoregressive Integrated Moving Average Mode, etc.

In our investigation, we focus on forecasting four kinds of weather information, including temperature, dew points, Mean Sea Level Pressures(MSLP) and wind speed, in the next few hours. We will input the raw data sets into our DNN model, the input n-dimensional vector is composed of the status in $(t-1)th$, $(t-2)th$, $\ldots$, $(t-n)th$ time points, then we use the DNN to represent these status, and employ a regressor to estimate the status in tth time point. We hope the seasonal cycles can be captured via massive volume of data by the superior learning ability of the multi-layer structured NN.

3. Greedy Layer-wise Unsupervised Pre-training and Aotuencoder DNN

The essential challenge in training deep architectures is to deal with the strong dependencies that exist during training between the parameters across layers [15]. Multi-layer NNs have more parameters than shallow NNs. Moreover, in a multi-layer NN, due to the non-convexity of the complex model, the optimization with traditional BP training approach may fall in a local minimum rather than global minimum. This may bring poor generalization to the model.

This problem isn't well solved until Hinton et al. introduced Deep Belief Network (DBN) that greedily trained up one layer with a Restricted Boltzmann Machine (RBM) at a time in 2006 [1]. Shortly after, strategies for building deep architectures from related variants were proposed by Bengio [16] and Ranzato[17]. They solved the training problem of deep NN in two phases: in the first phase, unsupervised pre-training, all layers are initialized using this layer-wise unsupervised learning signal; in the second phase, fine-tuning, a global training criterion (a prediction error, using labels in the case of a supervised task) is minimized. Such training approach is called the Greedy Layer-wise Unsupervised Pre-training. Fig.1 [15] shows the comparison among different training methods for NNs with deep architectures. There is a family of training models categorized into the family of Greedy Layer-wise Unsupervised Pre-training approaches, In our investigation, with the consideration of the attribute type of the weather data, i.e., the collected data are all real numbers, we choose the Stacked Auto-Encoder to build the deep architecture of our NN model.

The Stacked Auto-Encoder, as its name suggested, is a stacked architecture NN that applies Auto-Encoder in each layer. In NN, a single "neuron" is a computational unit that taken as input vector $X = x_1, x_2, \ldots, x_n$ (and a $+1$ intercept term), and outputs $h_{W,b}(x) = f(W^T x) = f(\sum_{i=1}^{3} W_i x_i + b)$ with a nonlinear function $f : \Re \mapsto \Re$. W is the weight matrix that stands for the connection among different

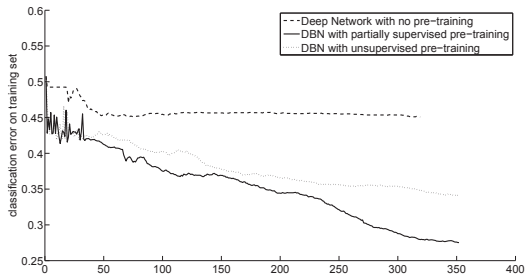


Fig. 1: Training classification error vs training iteration on DNNs, which shows the optimization difficulty for DNNs and the advantage of pre-training methods.

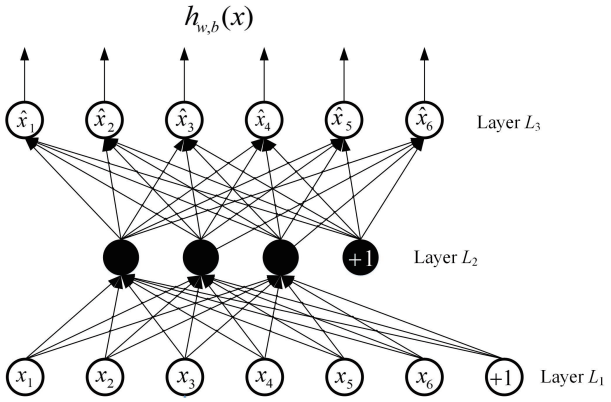


Fig. 2: An illustration of Auto-Encoder Algorithms. Layer L_1 is the input layer, and L_3 is the output layer. Via hidden layer L_2, we hope to represent the information x in layer L_1, so that the output $\hat{x}$ in L_3 can approximate the raw data x.

neurons in the network. In most of cases, sigmoid function $f(z) = \frac{1}{1+\exp(-z)}$ is employed. A typical Auto-Encoder tries to learn a function $h_{w,b}(x) \approx x$. In other words, it is trying to learn an approximation to the identity function, so as to output $\hat{x}$ that is similar to x . The identity function seems a typically trivial function trying to learn; but by placing constraints on the network, such as by limiting the number of hidden units, we can discover interesting structure about the data [17], e.g., for a data set, suppose that the original samples are collected from a 100-dimensional feature space, i.e. $x \in \Re^{100}$, set that there are 50 hidden units in the hidden layer, based on the requirement $h_{w,b}(x) \approx x$, the network is forced to learn a compressed representation of the input. That is, given only the vector of hidden unit activations $a^{(2)} \in \Re^{50}$, it must try to reconstruct the 100-dimensional input x. An illustration of Auto-Encoder is shown in Fig.2. If the inputs were completely random, each x_i comes from an I.I.D. Gaussian independent of the other features, then this compression task would be very difficult. But if there is a certain structure hidden in the data, for example, if some of the input features are correlated, such as in the feature space of time series analysis, then this algorithm will be able to discover some of those correlations.

The loss function of Auto-Encoder is:

$$J(W,b) = \left[\frac{1}{m}\sum_{i=1}^{m}\left(\frac{1}{2}\left\|h_{W,b}(x^{(i)}) - x^{(i)}\right\|^2\right)\right] + \frac{\lambda}{2}\sum_{l=1}^{n_l-1}\sum_{i=1}^{s_l}\sum_{j=1}^{s_{l+1}}\left(W_{ji}^{(l)}\right)^2 \quad (2)$$

where m is the number of training samples. The objective of the Auto-Encoder is to minimize Eq.(2) in order to make sure that the output $h_{W,b}(x^{(i)})$ can approximate the raw data $x^{(i)}$ as far as possible. The second term in Eq.(2) is a regularization term (also called a weight decay term) controled by the weight decay parameter λ that tends to decrease the magnitude of the weights, and helps prevent overfitting. We can minimize Eq.(2) by *gradient descent* to compute the configuration of the network.

Consequently, we need to combine Auto-Encoders layer by layer with a stacked structure to build the DNN. For each layer, we use an Auto-encoder to train the parameters in this layer, and then have these layers combined together. Specifically, in the training process of each layer, as shown in Fig.1, the input vectors have to pass through three layers, and the vectors in hidden layers (layer L_2, and for simplicity, we call the vectors in layer L_2 as the transformed vectors of the initially input vectors) are representations of the input vectors and can be used to reconstruct the input vectors. Thus, in every layer of the deep NN, the input of the current layer is the output of the previous layer, then we train the input data via an Auto-Encoder, and use the transformed vectors as the output of the current layer. Fig.4 shows the detailed mechanism of stacked Auto-Encoder based DNN. We can see that through a DNN, the raw data can be represented in new feature spaces layer by layer. In other words, DNN can learn features from the original data sets. Consequently, we apply any proper intelligence models with the learned features.

4. Experimental Results and Analysis

In our experiments, we concentrate on the evaluation of the learned features: four types of weather data sets, are employed and simulated, and the results are compared between models using raw features and models using represented features.

4.1 Weather Data Collection and Pre-processing

The HKO has provided great support to our investigation. Based on our collaboration with HKO, a massive volume of high quality real weather data could be applied in our experiment. Historical weather data sets, including the temperature, dew points, MSLP and wind speed data are employed in

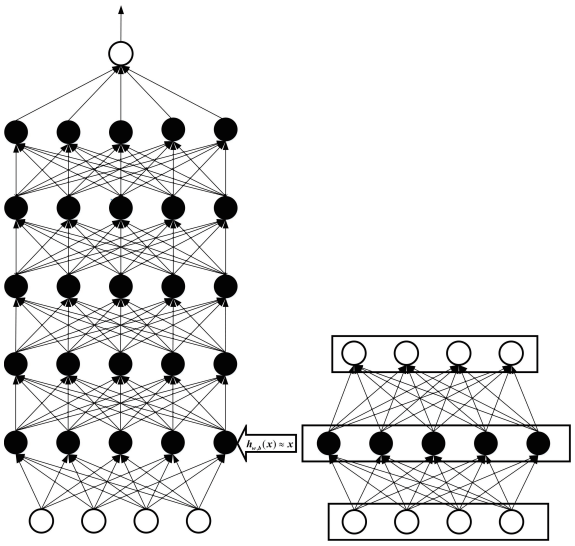


Fig. 3: A 5-hidden-layer DNN with Stacked Auto-Encoder method, by which each layer is greedily pre-trained with an unsupervised Auto-Encoder to learn a nonlinear transformation of its input (the output of the previous layer) that captures the main variations in its input, i.e. $h_{W,b}(x) \approx x$.

our model. The time range of the data sets is almost 30-year long, which covers the period from January 1, 1983 to December 31, 2012. In detail, the number of records in each data set is more than 260,000 respectively.

Unlike the temperature (measured with degree Celsius), dew points (measured in degree Celsius)and MSLP (measured in hectopascal(hPa)) data which have only one dimension , the wind speed data has two dimensions: the polar coordinate for the wind direction (measured with degree angle) and the speed (measured with meters per second). Moreover, for a certain time points, the direction of the air motion is not stable, i.e. the wind direction at that time point is not fixed. Such condition is denoted as "variable" in the raw data. Therefore, according to the requirement of our algorithm, we have to do some pre-processing on the data sets.

Different from the temperature (as Fig.4), dew points (as Fig.5) and MSLP (as Fig.6) data which are one-dimensional scalar quantities, the wind speed data (in a fixed horizontal plane) is a vector quantity that has two dimensions in the polar coordinate (as Fig.7), i.e. the angle to show its direction and the speed to measure the magnitude in this direction[18]. However, since our model is focused on single variable time series problems, we have to transform the data set to satisfy the model's requirement. According to the physical significance of the two dimensions, we denote the angle as θ and the speed as v to obtain:

$$v^0 = cos\theta \cdot v \tag{3}$$

where v^0 is the vector components of the wind speed in

Fig. 4: The distribution of temperature data in the last week of the data set

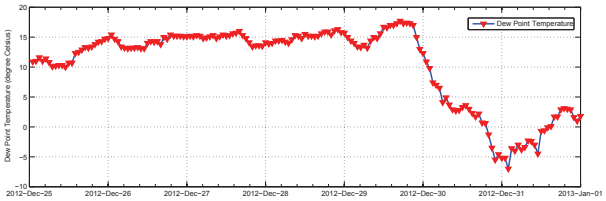


Fig. 5: The distribution of dew points temperature data in the last week of the data set

0 degree angle direction (as Fig.8). Thus, what we actually simulate is the time series of the speed components of the air motion in 0 degree angle direction. Moreover, there are about 3% wind speed data with the direction valued as "variable", for such condition, we consider it as a missing value in the data set and use the average value of the wind direction in its previous time point and its next time point to replace the value "variable".

4.2 Experiment on Temperature/Dew Points Temperature Forecasting

In our first experiment, we use a 4-layer DNN model to predict the temperature and dew points temperature in the next time point. The temperature data set records the real-time temperature, and the dew points temperature is a more complex quantity: the dew point is the temperature at which the water vapor in air at constant barometric pressure condenses into liquid water at the same rate at which it evaporates [19]. At temperatures below the dew point, water will leave the air. From Fig.4 and Fig.5, we can see that both of the two data sets show a smooth changing trend and are not as unstable as other two data sets. We may also observe

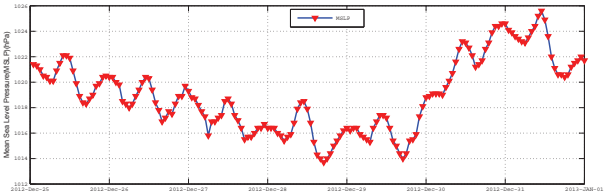


Fig. 6: The distribution of MSLP data in the last week of the data set

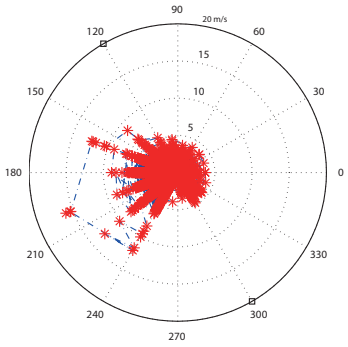


Fig. 7: The distribution of wind speed data in polar coordinate

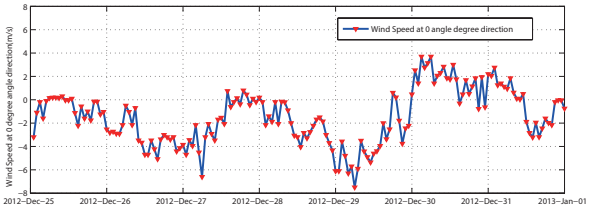


Fig. 8: The distribution of wind speed at a fixed diection

that the temperature data set and the dew points data set show very similar curves, which indicated that there may be a high relevance between these two data sets. However, the dew points curve is a little more fluctuating than the temperature cure, which imply that other factors may have effect on its values.

The objective of our experiments is to try to use the 7-day hourly records to forecast these two temperature related data in the next 24-hour period respectively, The whole network is with an input layer, two stack-organized Auto-Encoder layers, and the top layer which uses Support Vector Regression (SVR) to output the prediction results [20], [21]. Table I shows the parameter configuration of our experiment.

In the experiments, the training set has more than 230,000 records, and about 26,000 samples are selected as testing set. the ratio between training set and testing set is 9:1. The result is compared with Classical SVR. Note that the parameters in the Classical SVR are set as same as in the top SVR

Table 1: The parameter configuration of the experiments

Parameter	Value
Number of neurons in hidden layer 1	82
Number of neurons in hidden layer 2	40
λ	0.05
Learning rate	0.01
Max Iteration	400
Parameters in SVR	Default as LibSVM [22]

Table 2: The comparison of temperature prediction by SVR and DNN

Model	NMSE	DS	R^2
Classical SVR	2.179e-2	0.75	0.872
DNN with SVR in Top Layer	8.117e-3	0.82	0.915

Table 3: The comparison of dew points prediction by SVR and DNN

Model	NMSE	DS	R^2
Classical SVR	2.132e-2	0.75	0.856
DNN with SVR in Top Layer	9.552e-3	0.80	0.901

layer of the DNN model. In our experiment, we apply three criteria including Normalised Mean Square Error (NMSE), Directional symmetry (DS) and R^2 to evaluate the prediction results. Table II and Fig.9 give the result of temperature forecasting, and the results of dew points forecasting is given as Table III and Fig.10.

From Fig.9 and Fig.10 we can observe that both SVR and DNN can simulate the two temperature related real data sets very well after training with a massive volume of data. For the temperature data, the predicted results almost coincide with the real data, the similar predict results are also obtained in the dew points data. We can see that after training with a massive volume of data set, the SVR can efficiently simulate the temperature related data sets. Especially, the DNN isn't troubled with the overfitting problem when it is trained with the large data set.

Results shown in Table II and Table III demonstrate the positive role of the DNN in the simulation tasks. As we

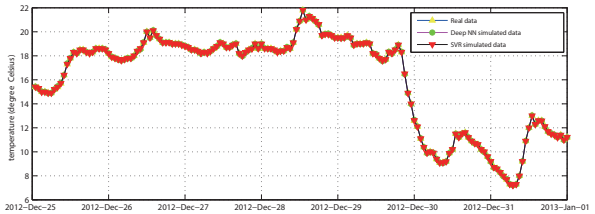


Fig. 9: The results of temperature prediction for the date in the last week

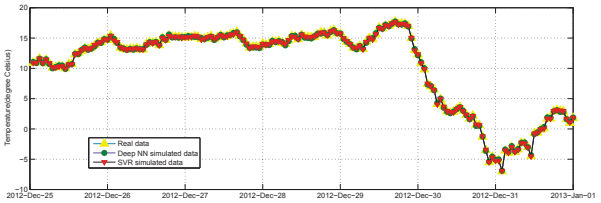


Fig. 10: The results of dew points temperature prediction for the date in the last week

Table 4: The comparison of MSLP prediction by SVR and DNN

Model	NMSE	DS	R^2
Classical SVR	6.26	0.70	0.851
DNN with SVR in Top Layer	2.13	0.72	0.924

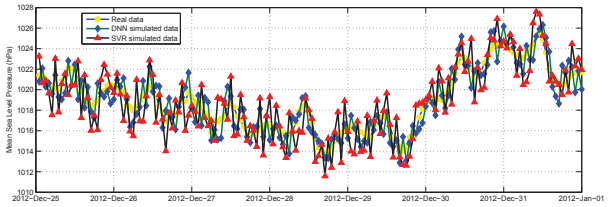


Fig. 11: The results of temperature prediction for the date in the last week

Table 5: The comparison of wind speed prediction by SVR and DNN

Model	NMSE	DS	R^2
Classical SVR	0.3721	0.72	0.831
DNN With SVR in Top Layer	0.2522	0.83	0.891

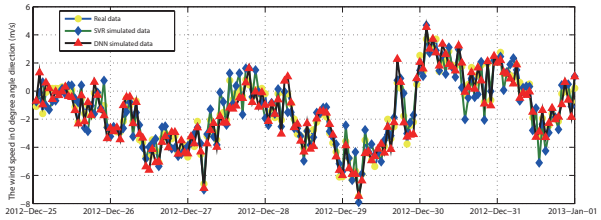


Fig. 12: The results of temperature prediction for the date in the last week

discussed in above, both of the two models have the same configuration in SVR part, the only difference is that in the DNN, we represent the raw features, and use the new features to train the SVR in the top layer. We can see that the DNN model greatly reduces the NMSE(the NMSE in SVR model has already been very small, but after the feature representation/granulation, the NMSE becomes even smaller). Also, the higher R^2 values are obtained. These results demonstrate that, with the represented features via a deep NN, the SVR model in the top layer can learn the raw data much better.

4.3 Experiment on MSLP Forecasting

In real applications, temperature simulation (in short term) is not a challenging task since that the change of temperature in short term is relatively stable. Therefore, although the DNN can work very well on simulation of temperature related data sets as shown in the previous experiments, the academic and practical significance is weak, experiments on more unstable data set is necessary to confirm the result.

In the following experiment, we apply the MSLP data to verify our model. The MSLP is the atmospheric pressure at sea level or (when measured at a given elevation on land) the station pressure reduced to sea level assuming that the temperature falls at a lapse rate of 6.5 K/km in the fictive layer of air between the station and sea level [23]. Observing Fig.6, we may find that the curve of MSLP data is more fluctuating than that of the temperature data. So it is reasonable to apply MSLP data to verify our model further. We adopt the parameter configuration of the previous experiments. Table V and Fig.11 give the results.

From the results shown in Table IV and Fig.11, we shall observe that there are obvious error values between the real data and the predicted data, which is different from the condition in previous experiments that the predicted curves almost coincide with the actual curves. However, the high R^2 value shows that both of the SVR and DNN model can explain more than 85% of the errors in the simulation, and the DNN model can avoid overfitting when training with the large data set.

Again, compared with the SVR model, the DNN model shows its advantages in this experiment. The NMSE is reduced and the R^2 value is obviously improved, such experimental results demonstrate that with the learned feature, the SVR model can provide higher accuracy in MSLP prediction.

4.4 Experiment on Wind Speed Forecasting

As shown in Fig.4, Fig.5, Fig.6 and Fig.8, the change of wind speed data is much more stochastic than other three data sets. Accordingly, the simulation of wind speed data is more difficult and has more academic and practical significance. To improve the learning ability of our model, we have made some modifications on the configuration of the DNN, a hidden layer with more neurons is added in the DNN, and the sparsity setting is given in this added layer. The results are shown in Table V and Fig.12.

Fig.12 shows the simulation results of the wind speed data in the last week of the data set. From Fig.12, we can observe that after training with a big volume of wind speed data, the model can capture the main trend of changes, and the DNN can give a better performance than simply using SVR. Inspecting the criteria in Table V, DNN can return a lower NMSE and higher R^2 value.

There are two special points which should be noticed in this experiment. (1) From the distribution of the MSLP data set and the wind speed data set, we know that it is more difficult to predict the wind speed than to predict MSLP; however, in our simulation, the latter experiments can obtain lower NMSE value. This fact implies that the added hidden layer is able to improve the learning ability of the whole network.(2) Note that a much higher DS value is obtained in DNN, this maybe caused by the fact that

features generated via DNN may have the largest possible variation, and such fact shows that the principle of DNN may be considered as an advanced form of Principal Component Analysis (PCA)[24].

Our experiments only make comparison between classical SVR and Stacked Auto-Encoder DNN with SVR in the top layer. Actually, some other models can also be applied to deal with weather data related time series problem. However, the main objective of our investigation is to attest the models' performance with the new represented features. The results demonstrate that compared with the raw features, the obtained features can explain the principle of the raw data set better. Moreover, the DNN can be combined with many other models, and the learned features can be employed to improve the performances of most models in computational intelligence field.

5. Conclusion, Limitation and Future Work

In our investigation, we explore an approach that using a novel computational intelligence technology to process massive volume of weather data. The proposed DNN model may represent the features of the raw weather data layer by layer , and experimental results show that the obtained features can improve the performances of classical computational intelligence models. The contribution of our investigation is significant: we give an approach that using computational intelligence method to learn features for weather forecasting, and our experiments demonstrate that the DNN algorithm also has the potential on time series problem.

Our initial investigation is to explore the use of DNN to deal with the large volume of weather data. Therefore, limitations also exist. The relation between the number of hidden layers and the output accuracy should be quantified, also, the number of hidden neurons should be optimized.

The main future work of our investigation is that, we will try to employ our model on more difficult weather data, such as rain fall data set; and moreover, we will continue exploring the theoretical principle of computational intelligence, especially, we will try to give the mathematical explanation of the DNN.

Acknowledgment

The authors would like to acknowledge the partial support of the CRG grants G-YL14 and G-YM07 of The Hong Kong Polytechnic University.

References

[1] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.

[2] H. Lee, R. Grosse, R. Ranganath, and A. Y. Ng, "Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations," in *Proceedings of the 26th Annual International Conference on Machine Learning*. ACM, 2009, pp. 609–616.

[3] Y. Bengio, *Learning deep architectures for AI*. Now Publishers Inc., 2009, vol. 2, no. 1.

[4] X. Wang and Q. He, "Enhancing generalization capability of svm classifiers with feature weight adjustment," in *Knowledge-Based Intelligent Information and Engineering Systems*. Springer, 2004, pp. 1037–1043.

[5] J. Schmidhuber, "Curious model-building control systems," in *Neural Networks, 1991. 1991 IEEE International Joint Conference on*. IEEE, 1991, pp. 1458–1463.

[6] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[7] G. Cybenko, "Approximation by superpositions of a sigmoidal function," *Mathematics of Control, Signals and Systems*, vol. 2, no. 4, pp. 303–314, 1989.

[8] A.-r. Mohamed, G. E. Dahl, and G. Hinton, "Acoustic modeling using deep belief networks," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 20, no. 1, pp. 14–22, 2012.

[9] Y. Bengio, "Deep learning of representations for unsupervised and transfer learning." *Journal of Machine Learning Research-Proceedings Track*, vol. 27, pp. 17–36, 2012.

[10] J. N. Liu and Y. Hu, "Application of feature-weighted support vector regression using grey correlation degree to stock price forecasting," *Neural Computing and Applications*, pp. 1–10, 2013.

[11] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *The Journal of Machine Learning Research*, vol. 9999, pp. 3371–3408, 2010.

[12] S.-M. Chen and J.-R. Hwang, "Temperature prediction using fuzzy time series," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 30, no. 2, pp. 263–275, 2000.

[13] K. Kwong, M. H. Wong, J. N. Liu, and P. Chan, "An artificial neural network with chaotic oscillator for wind shear alerting," *Journal of Atmospheric and Oceanic Technology*, vol. 29, no. 10, pp. 1518–1531, 2012.

[14] W.-C. Hong, "Rainfall forecasting by technological machine learning models," *Applied Mathematics and Computation*, vol. 200, no. 1, pp. 41–57, 2008.

[15] D. Erhan, Y. Bengio, A. Courville, P.-A. Manzagol, P. Vincent, and S. Bengio, "Why does unsupervised pre-training help deep learning?" *The Journal of Machine Learning Research*, vol. 11, pp. 625–660, 2010.

[16] S. M. Miller, Y. Geng, R. Z. Zheng, and A. Dewald, "Presentation of complex medical information: Interaction between concept maps and spatial ability on deep learning," *International Journal of Cyber Behavior, Psychology and Learning (IJCBPL)*, vol. 2, no. 1, pp. 42–53, 2012.

[17] M. Ranzato, F. J. Huang, Y.-L. Boureau, and Y. Lecun, "Unsupervised learning of invariant feature hierarchies with applications to object recognition," in *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*. IEEE, 2007, pp. 1–8.

[18] R. A. Pielke, *Mesoscale meteorological modeling*. Academic Press, 2002.

[19] J. Ancsin, "Dew points, boiling points and triple points of," *Metrologia*, vol. 9, no. 1, p. 26, 1973.

[20] V. Vapnik, *The nature of statistical learning theory*. Springer, 2000.

[21] A. J. Smola and B. Schölkopf, "A tutorial on support vector regression," *Statistics and Computing*, vol. 14, no. 3, pp. 199–222, 2004.

[22] C.-C. Chang and C.-J. Lin, "Libsvm: a library for support vector machines," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 2, no. 3, p. 27, 2011.

[23] T. Moses, G. Kiladis, H. Diaz, and R. Barry, "Characteristics and frequency of reversals in mean sea level pressure in the north atlantic sector and their relationship to long-term temperature trends," *Journal of Climatology*, vol. 7, no. 1, pp. 13–30, 1987.

[24] K. Yang and C. Shahabi, "A pca-based similarity measure for multivariate time series," in *Proceedings of the 2nd ACM International Workshop on Multimedia Databases*. ACM, 2004, pp. 65–74.

A Hybrid Artificial Neural Network and VEPSO based on Day-ahead Price Forecasting of Electricity Markets

O. Abedinia
Electrical Engineering Department
Semnan University
Semnan,
Iran

N. Amjady
Electrical Engineering Department
Semnan University
Semnan,
Iran

H. A. Shayanfar *
Department of Elec. Engineering
College of Tech. and Engineering
South Tehran Branch
Islamic Azad University
Tehran, Iran

oveis.abedinia@gmail.com, n_amjady@yahoo.com, hashayanfar@yahoo.com,

Abstract— *In this paper a new Hybrid technique of Artificial Neural Network (ANN) and Vector Evaluated Particle Swarm Optimization (VEPSO) is presented as a forecasting strategy for day-ahead price of electricity market. The proposed technique the proposed intelligent technique is applied to weights and bias of ANN to improve the learning capability through the minimum error. A comprehensive comparative analysis with other soft computing and hybrid models shows a significant improvement in forecast error, through the application of a proposed hybrid model. The proposed technique is tested on PJM electricity market through comparison with other strategies for this problem.*

Keywords*:* Price Forecasting, VEPSO, ANN, Feature Selection.

I. Introduction

Since the beginning of floating electricity prices, electricity price forecasting has become one of the main endeavors for researchers and practitioners in energy markets. In most merchandise markets, the effects of production or supply chain on prices are dampened by surplus storage [1]. By contrast, the electricity market lacks storage for feasible purposes, which is an intrinsic source of volatility. The volatility in currency markets makes electricity price forecasting a difficult yet challenging task. To forecast accurately these prices is critical for producers, consumers and retailers. In fact, they should set up bids for the spot market in the short term and define contract policies in the medium term, and in addition, they must define their expansion plans in the long term [2]. For these reasons, all the determinations that each market player must take are strongly affected by price forecasts. Several of these problems can be modeled as mathematical programs.

Several possible methods have been proposed by researchers for the day-ahead electricity price forecasting in literature as; the stationary time series models such as Auto-Regressive (AR) [3], dynamic regression and transfer function [4], Auto-Regressive Integrated Moving Average (ARIMA) [5] and nonstationary time series models like Generalized Auto-Regressive Conditional Heteroskedastic (GARCH) [6] have been proposed for this purpose. But, according to the linear prediction behavior of time series, Neural Network (NN) Fuzzy Neural Networks (FNN) and Hidden Markov Models [7] have been proposed to solve this problem which is generally a nonlinear function.

These techniques are simple, powerful, and flexible tools for forecasting, providing better solutions to model complex nonlinear relationships than the traditional linear models. But, they have weaknesses in the determination of network architecture, network parameters and the capability of modeling the nonlinear input/output mapping functions. However, electricity price is a time variant signal and its functional relationships rapidly varies with time [8].

This paper proposes a new hybrid strategy based on Artificial Neural Network (ANN) and Vector Evaluated Particle Swarm Optimization (VEPSO) and two-stage feature selection technique (MIMI) for price forecasting in PJM power market.

The remaining parts of the paper are organized as follows. In the second section, the proposed price

* Correspanding Author. E-Mail Address: hashayanfar@yahoo.com (H. A. Shayanfar)

forecast strategy is described. The proposed intelligent method is presented in section three. Section four presents the numerical results and section five concludes the paper.

II. Proposed Strategy

A. The proposed data model

The Short Time Fourier Transform (STFT) provides the time information by computing different FTs for consecutive time intervals, and putting them together. The 2006 hourly Ontario energy price (HOEP) series as shown in Fig. 1 is characterized with random and chaotic changes where we can observe negative prices as well as price spikes. Consecutive time intervals of the signal are obtained by truncating the signal using a sliding windowing function [9-10].

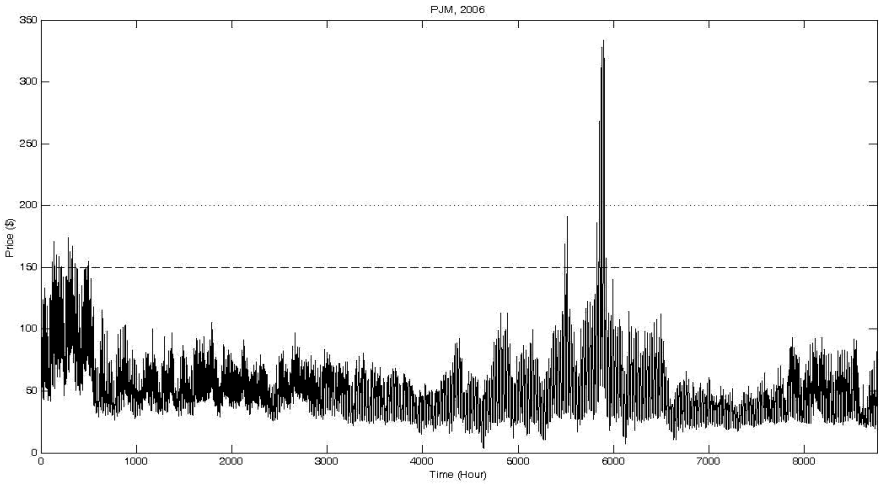


Figure 1. Hourly Ontario energy price in 2006

The increasing prominence of the computers has led to a new way of looking at the world. Artificial Neural Networks (ANN) and VEPSO are considered as the so called soft computing methods are now a days becoming predominant tools in the area of artificial intelligence linked application oriented methods. The Neural network theory was first adopted in 1940 where the starting point was the learning law presented by ITEBB in 1949, where demonstrated how neurons could exhibit learning behavior [11]. The application further waxed and waned away because of the lack of powerful technological advancement. The resurgence occurred recently due to the new methods that are emerging as well as the computational power suitable for simulation of interconnected neural networks [12]. Further to the technological advancement in the field of ANN, researchers were attracted on their important applications where logical and relational thinking is required. Among the major applications viz., robotics, analysis, optimal control, database, learning, signal processing, semiconductors, Power system related applications became a useful tool for the online researchers in this field.

ANN is biologically inspired and represented as a major extension of computation. They embody computational paradigms, based on biological metaphor, to mimic the computations of the brain [12]. The improved understanding of the functioning of neuron and the pattern of its interconnection has enabled researchers to produce the necessary mathematical model for testing their theories and developing practical applications. Main applications of the ANN's can be divided into two principal streams. First stream among this is concerned with modeling the brain and thereby explains its cognitive behavior. The primary aim of researchers in the second stream is to construct useful 'computers' for real world problems of classification or Pattern Recognition by drawing on these principles.

B. Definition of the Neural Network

Neural networks are systems that typically consist of a large number of simples processing units, called Neurons. A neuron has generally a high-dimensional Input vector and one single output signal. This output signal is usually a non-linear function of the input vector and a weight vector. The function to be performed on the Input vector is hence defined by the non-linear function and the weight vector of the neuron. The weight vector is adjusted in a training phase by using a large set of examples and the learning rate. The learning rule adapts the weight of all neurons in networks in order to learn an underlying relation in the training example.

C. Artificial Neural Network Fundamentals

Elementary processing unit of ANN's is neuron. Generally it contains several inputs but has only one output. The main differences between various existing models of ANN are mainly in their architectures or the way their basic processing elements (neurons) are interconnected. As basic element the neurons are not powerful but their interconnections allow encoding relationship between variables of the problems to which it is applied and providing very powerful processing capabilities. General model of the processing unit of ANN can be considered to have the following three elements [13]. Fig. 2 shows the schematic diagram of the neuron.

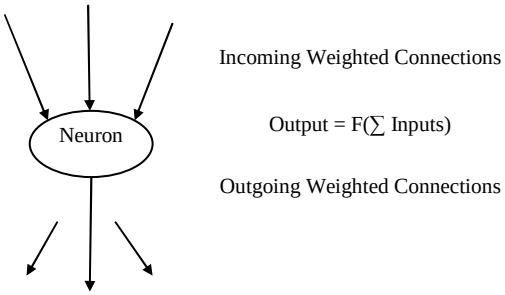


Figure 2. Schematic Diagram of the Neuron

In this paper the presented weights are optimized by VEPSO. So the descriptions of weights are presented in below as**;**

D. Weighted Summing Unit

The weighted summing unit consists of external or internal inputs (X_i (x_1, x_2, x_3 … x_n)) times the corresponding weights W_{ij} = (w_{i1}, w_{i2} … w_{in}). The fixed weighted inputs may be either from the previous layers of ANN or from the output of neurons. If these inputs are derived from neuron outputs, it forms the feedback architecture it has feedforward architecture [14].

In this paper a multi layer feedforward neural network is applied. Major application of feedforward neural network is in large-scale systems that contain a large number of variable and complex systems where little analytical knowledge is available. Fig 3 shows the three layer of this application.

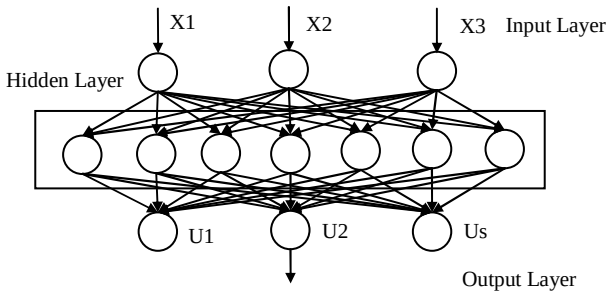


Figure 3. Three Layer Feedforward Neural Network

E. *Feature Selection*

The set of candidate inputs is so large and cannot be directly used for any forecaster. Besides, irrelevant and redundant candidate inputs may exist in it, which can be misleading for the forecast engine. Additionally, with more input features, forecasters usually require more historical data to extract the mapping function between the inputs and output, where it can be create some limitation in forecasting problem.

Relevancy: Individual relevance of the candidate inputs with the target variable is an important factor for feature selection. Mutual Information (MI) is a good criterion to measure relevancy between two random variables. The MI between two random variables x and y can be interpreted as the information about y that we obtain by studying x and vice versa. The entropy $H(x)$ of a continuous random variable x with probability distribution $P(x)$ is defined as follows [15]:

$$H(x) = -\int P(x)\log_2\left(P(x)\right)dx \tag{1}$$

The joint entropy $H(x,y)$ of two discrete random variables x and y with a joint probability distribution $P(x,y)$ is defined by;

$$H(x,y) = -\sum_{i=1}^{n}\sum_{j=1}^{m} P(x_i,y_j)\log_2\left(P(x_i,y_j)\right) \tag{2}$$

Where H(x,y) indicates the total entropy of random variables x and y [15]. The MI of continuous variables x and y, denoted by $MI(x;y)$, is defined based on their joint probability distribution $P(x,y)$ and the respective individual probability distributions $P(x)$ and $P(y)$ as follows [15]:

$$MI(x;y) = \iint P(x,y)\log_2\left(\frac{P(x,y)}{P(x)P(y)}\right) \tag{3}$$

MI between two discrete random variables x and y with n and m discrete values, respectively, becomes as follows:

$$MI(x;y)=\sum_{i=1}^{n}\sum_{j=1}^{m}P(x_i,y_j)\log_2\left(\frac{P(x_i,y_j)}{P(x_i)P(y_j)}\right) \quad (4)$$

If the mutual information between two random variables is large, the two variables are closely related and vice versa. If the mutual information becomes zero, the two random variables are totally unrelated or the two variables are independent. Entropy and MI are often illustrated in the form of a Venn diagram as shown in Fig. 4. *MI*(*x*;*y*) measures dependence between *X* and *Y*. If *MI*(*x*;*y*) is large (small), it means *x* and *y* are closely (not closely) related. If *x* and *y* are independent then *P*(*x*; *y)=P(x)P(y*), and so *MI*(*x*;*y*) = 0.

To enhance the redundancy filter if the $S_1 = \{x_1, x_2, \ldots, x_n\}$ is a set of candidate inputs including redundant features. The MI between each pair of features of S_1 can be calculated as:

$MI(x_i, x_j)$, $\quad 1 < i,j < n \quad (5)$

Higher value of $MI(x_i, x_j)$ means more common information between the candidate inputs which means the high level of redundancy. So, for each feature $x_i \in S_1$, its maximum redundancy in the set S_1 is computed:

$Max(MI(x_i, x_j))$, $\quad 1 < j \leq n \quad (6)$

By combining the feature selection technique and the proposed redundancy filter, we reach the following two-stage feature selection as:

$Max(MI(x_k, x_j))= MI(x_k, x_m)>TH2$, $\quad 1 < m \leq n \quad (7)$

I. Particle Swarm Optimization

A. Standard PSO

The standard of the PSO are best describe as sociologically inspired, since the original algorithm was based on the sociological behavior associated with bird flocking [16,17]. PSO is simple in concept, few in parameters, and easy in implementation, besides it has an excellent optimization performance. At first, PSO was introduced for continuous search spaces and because of the aforementioned features; it has been widely applied to many optimization problems soon after its introduction [18].

To explain how PSO algorithm works, an optimization problem which requires optimization of N variables simultaneously is considered here. PSO is initialized with a population of solutions, called "particles". At first, a random position and velocity is assigned to each particle. The position of each particle corresponds to a possible solution for the optimization problem. A fitness number is assigned to each particle which shows how good its position is. During the optimization process, each particle moves through the N-dimensional search space with a velocity that is dynamically adjusted according to its own and its companion's previous behavior. Updating the particle velocity is based on three terms, namely the "social," the "cognitive," and the "inertia" terms. The "social" part is the term guiding the particle to the best position achieved by the whole swarm of particles so far (gbest), the "cognitive" part guides it to the best position achieved by itself so far (pbest), and the "inertia" part is the memory of its previous velocity (ω.vn). The following formulae demonstrate the updating process of a particle position (xn) and its velocity (vn) in the nth dimension in an N-dimensional optimization space [17]:

$$v_i^{k+1} = w\, v_i^k + c_1.R_1.\left(pbest_i^k - x_i^k\right) + c_2.R_2.\left(gbest^k - x_i^k\right) \quad (8)$$

$$x_i^{k+1} = x_i^k + v_i^{k+1} \quad (9)$$

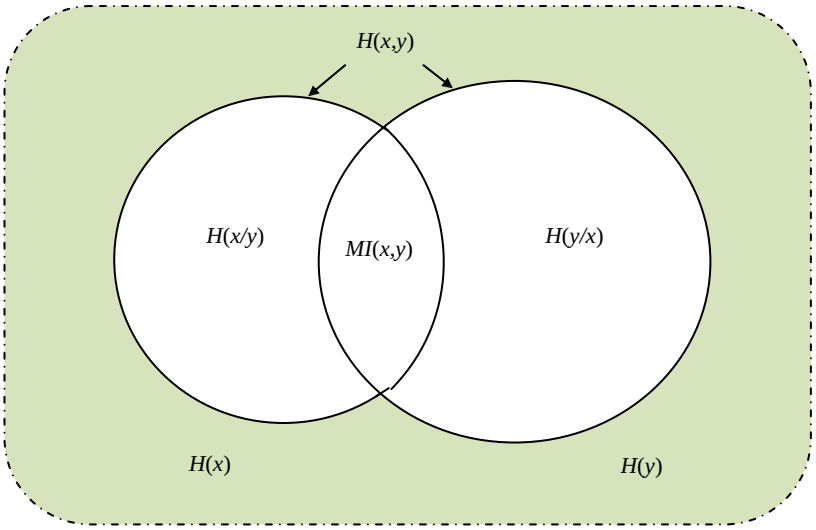


Figure 4. Representation of relationships between the entropies of two random variables X and Y and their mutual information.

In equation 11, R1 and R2 are random numbers uniformly distributed between 0 and 1. c1 and c2 are acceleration constants and ω is the inertia weight. These three parameters determine the tendency of the particles to the related terms. Moreover, another parameter is used to limit the maximum velocity of a particle (Vmax). All these parameters directly affect the optimization behavior; for example, the inertia weight controls the exploration ability of the process while the acceleration constants and maximum velocity are parameters for controlling the convergence rate [16, 17]. The iterative procedure of updating the velocities and positions of particles continues until the best position achieved by the whole swarm of particles (gbest) does not change over several iteration. Figure 5 shown this process of PSO method to CM problem.

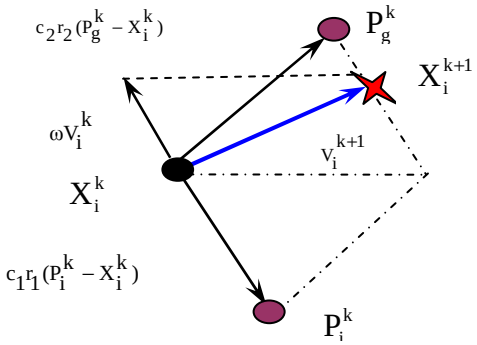


Figure 5. Velocity and location of particle updating process.

B. VEPSO

The vector evaluated approach can be classified as a criterion-based multi-objective strategy, where different stages of the optimization process consider different objectives [15].

The actual implementation involves assigning each objective function to one of multiple populations for optimization. Information with respect to the different populations are exchanged in an algorithm-dependent fashion resulting in the simultaneous optimization of the various objective functions. As previously stated, the advantage of this approach lies in reduced computational complexity, which is a desirable property when solving a complex combinatorial problem where the fitness function evaluations are in them computationally expensive.

The basic concept of VEPSO algorithm is illustrated in Figure 6. As an example, for the case of two objective functions, *X1* and *X2* are swarm 1 and swarm 2, respectively, while ***gbest1*** and ***gbest2*** are the *gbest* for swarm 1 and swarm 2, respectively. As usual, ***v1***, ***v2***, ***s1***, and ***s2*** are the velocities and positions of each swarm. *X1* evaluates the objective function *f1* and *X2* evaluates the objective function *f2*. There is no necessity for a complicated information migration scheme between the swarms as only two swarms are employed. Each swarm is exclusively evaluated according to the respective objective function. The *gbest* of the second swarm (*X2*) is used for the calculation of the new velocities of the first swarm's (*X1*) particles and accordingly, *gbest* of the first swarm (*X1*) is used for the calculation of the new velocities of the second swarm (*X2*).

The VEPSO assumes that the search behavior of a swarm is affected by a neighboring swarm. The procedure of exchanging information among swarms can be clearly viewed as a *migration* scheme in a parallel computation framework. The flow chart is given as figure 6.

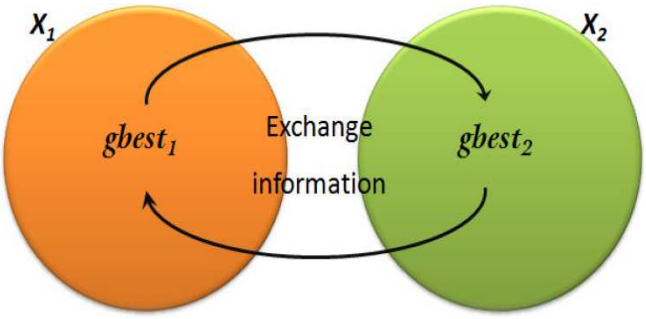


Figure 6. The basic concept of VEPSO.

II. Numerical Results

The proposed method is examined on the day ahead electricity markets of PJM, commonly used as the test case in many price forecasting papers [14]. In the USA, the PJM competitive market, one of the Regional Transmission Organizations (RTO), plays a vital role in the U.S. electric system. PJM ensures the reliability of the largest centrally dispatched control area in North America by coordinating the movement of electricity in all or parts of Delaware, Illinois, Indiana, Kentucky, Maryland, Michigan, New Jersey, North Carolina, Ohio, Pennsylvania, Tennessee, Virginia, West Virginia and the District of Columbia [4]. Data of the PJM electricity markets have been obtained from [5].

The examination of this paper has been performed by means of real data of the PJM electricity market in year 2006. Obtained results for four weeks corresponding to the four seasons of year 2006 are presented in Tables 1 and 2 [5]. Error values shown in Table 1 are in terms of Weekly Mean Error (WME):

$$WME = \frac{1}{168}\sum_{i=1}^{168}\frac{|P_{iACT} - P_{iFOR}|}{P_{iACT}} \quad (10)$$

where P_{iACT} and P_{iFOR} are actual and forecasted price of hour i, respectively.

Additional to the mean error, stability of results is another important factor for comparison of forecast methods. Different statistical measures such peak, variance, and standard deviation of error have been proposed as the index of uncertainty [5]. In Table 2, weekly peak error (WPE) of the examined methods for the test weeks are shown:

$$WPE = \underset{1\le i\le 168}{Max}\left(\frac{|P_{iACT} - P_{iFOR}|}{P_{iACT}}\right) \quad (11)$$

As seen again the proposed method has the least values of WME among all examined methods in [19]. Obtained results from the proposed method has for the Fall's test week (with the highest WME) are shown in Fig. 5. It can be seen that even for the worst test case, the proposed method has has acceptable accuracy.

TABLE I. WME FOR 4 WEEKS OF PJM ELECTRICITY MARKET IN YEAR 2006 (%) [19]

Test Week	ARIMA	LM	BFGS	BR	Proposed
Winter	11.21	9.82	12.90	13.22	7.8
Spring	15.30	8.87	10.12	12.92	7.7
Summer	13.56	10.43	11.46	11.98	6.8
Fall	12.93	9.54	9.83	12.24	7.4
Average	13.25	9.67	11.08	12.59	7.5

TABLE II. WPE FOR 4 WEEKS OF PJM ELECTRICITY MARKET IN YEAR 2006 (%) [19]

Test Week	ARIMA	LM	BFGS	BR	Proposed
Winter	45.10	22.31	25.42	23.63	15.4
Spring	34.32	18.45	21.39	20.13	13.4
Summer	44.48	27.34	38.12	30.22	24.2
Fall	55.81	32.12	29.23	33.11	23.2
Average	44.93	25.05	28.54	26.77	18.5

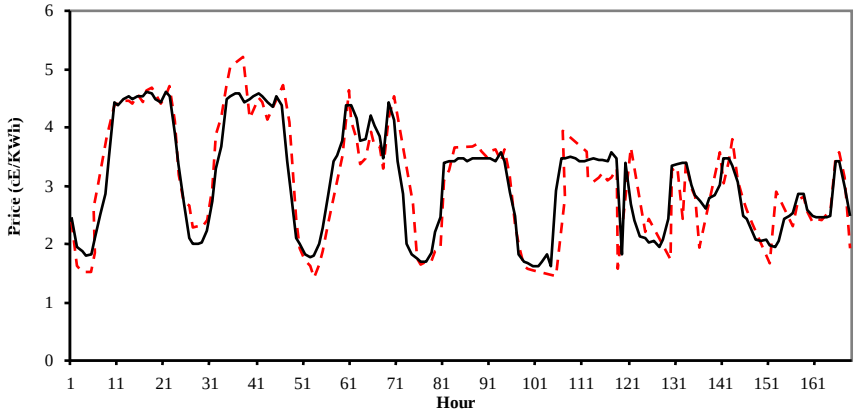


Figure 7. Hourly prices (dark), price forecasts (grey), and absolute value of forecast errors (red) for the summer's test week

III. Conclusion

In this paper a new Hybrid technique of ANN and VEPSO is presented as a forecasting strategy for day-ahead price of electricity market. Companies that trade in electricity markets make extensive use of market clearing price prediction techniques either to bid or hedge against volatility. However market clearing price prediction has its own complexities. It is a nonstationary, volatile, and nonlinear signal owning multiple periodicity, high frequency components, significant outliers and especially time dependent characteristic. The proposed technique the proposed intelligent technique is applied to weights and bias of ANN to improve the learning capability through the minimum error. A comprehensive comparative analysis with other soft computing and hybrid models shows a significant improvement in forecast error, through the application of a proposed hybrid model. The proposed strategy has been examined on the PJM electricity market and compared with other technique. These comparisons reveal the forecast capability of proposed method.

References

[1] N. Amjady, M. Hemmati, Energy price forecasting - problems and proposals for such predictions, IEEE Power & Energy Magazine, 4(2) (2006) 20-29.

[2] N. Amjady, Day-ahead price forecasting of electricity markets by a new fuzzy neural network, IEEE Trans. Power Syst., 21(2) (2006) 887-896.

[3] S. Ng, T.J. Vogelsang, Forecasting autoregressive time series in the presence of deterministic components, Econometrics Journal (2002), volume 5, pp. 196–224.

[4] N. Amjady, F. Keynia, Day ahead price forecasting of electricity markets by a mixed data model and hybrid forecast method, Int. J. Electric Power and Energy Systems, 30(9) (2008) 533-546.

[5] J Contreras, R. Espinola, F. J. Nogales, and A. J. Conejo, "ARIMA Models to Predict Next-Day Electricity Prices," IEEE Trans. On Power systems, Vol. 18, No. 3, August 2003, pp. 1014-1020.

[6] R. C. Garcia, J. Contreras, M. V. Akkeren, and J. B. C. Garcia, "A GARCH Forecasting Model to Predict Day-Ahead Electricity

Prices," IEEE Trans. On Power systems, Vol. 20, No. 2, May 2005, pp. 867-874.

[7] Md. Rafiul and B. Nath , StockMarket, Forecasting Using Hidden Markov Model: A New Approach, Proceedings of the 2005 5th International Conference on Intelligent Systems Design and Applications (ISDA'05).

[8] N. Amjady, " Day-Ahead Price Forecasting of Electricity Markets by a New Fuzzy Neural Network," IEEE Trans. on Power Systems, Vol. 21, No. 2, May 2006, pp. 887-896.

[9] M. A. Plazas, A. J. Conejo, F. J. Prieto, " Multimarket Optimal Bidding for a Power Producer," IEEE Trans. on Power Systems, Vol. 20, No. 4, Nov. 2005, pp. 2041-2050.

[10] N. Amjady, M. Hemmati, "Energy Price Forecasting - Problems and Proposals for such Predictions" IEEE Power and Energy Magazine, Vol. 4, No. 2, March/April 2006, pp. 20-29.

[11] M. P. Garcia, and D. S. Kirschen, "Forecasting System imbalance Volumes in Competitive Electricity Markets," IEEE Trans. On Power systems, Vol. 21, No. 1, Feb. 2006, pp. 240-248.

[12] L. Zhang, P. B. Luh, K. Kasiviswanathan, "Energy Clearing Price Prediction and Confidence Interval Estimation With Cascaded Neural Network," IEEE Trans. On Power systems, Vol. 18, No. 1, Feb. 2003, pp. 99-105.

[13] O. B. Fosso, A. Gjelsvik, A. Haugstad, M. Birger, and I. Wangensteen, "Generation scheduling in a deregulated system. The Norwegian case," IEEE Trans. Power on Power Systems, vol. 14, No. 1, Feb. 1999, pp. 75–81.

[14] F. J. Nogales, J. Contreras, A. J. Conejo, and R. Espinola, "Forecasting Next-Day Electricity Prices by Time Series Models," IEEE Trans. on Power Systems, Vol. 17, No. 2, May 2002, pp. 342-348.

[15] J. J. Guo, and P. B. Luh, "Improving Market Clearing Price Prediction by Using a Committee Machine of Neural Networks," IEEE Trans. on Power Systems, Vol. 19, No. 4, November 2004, pp. 1867-1876.

[16] B. Panida, B. Chanwit, O. Weerakorn, "Optimal congestion management in an electricity market using PSO with time-varying acceleration coefficients", Computers and Mathematics with Applications, Vol. 3, pp. 1-10, 2010.

[17] N.M. Tabatabaei, Gh. Aghajani, N.S. Boushehri, S. Shoarinejad, " Optimal location of facts devices using adaptive particle swarm optimization mixed with simulated annealing". International Journal on "Technical and Physical Problems of Engineering", Issue. 7, Vol. 3, No. 2, pp: 60-70, Jun2011.

[18] R. Eberhart, Y. Shi, "Comparison between genetic algorithms and particle swarm optimization", Proceedings of the Seventh Annual Conference on Evolutionary Programming IEEE Press 1998.

[19] N. Amjady, F. Keynia, Day-Ahead Price Forecasting of Electricity Markets by Mutual Information Technique and Cascaded Neuro-Evolutionary Algorithm, IEEE Trans. Power Syst., 24(1) (2009) 306-318.

BIOGRAPHIES

Oveis Abedinia received the B.S. and M.Sc. degrees in Electrical Engineering in 2005 and 2009, respectively. Currently, he is a Ph. D. student in Electrical Eng. Department, Semnan University, Semnan, Iran. His areas of interest in research are Application of Artificial Intelligence to Power System and Control Design, Load and Price Forecasting, Restructuring in Power Systems, Heuristic Optimization Methods. He has two industrial patents, authored of one book in Engineering area in Farsi and more than 70 papers in international journals and conference proceedings. Also, he is a member of Iranian Association of Electrical and Electronic Engineers (IAEEE) and IEEE.

Nima Amjady (SM'10) was born in Tehran, Iran, on February 24, 1971. He received the B.Sc., M.Sc., and Ph.D. degrees in electrical engineering from Sharif University of Technology, Tehran, Iran, in 1992, 1994, and 1997, respectively. At present, he is a Professor with the Electrical Engineering Department, Semnan University, Semnan, Iran. He is also a Consultant with the National Dispatching Department of Iran. His research interests include security assessment of power systems, reliability of power networks, load and price forecasting, and artificial intelligence and its applications to the problems of power systems.

Heidarali Shayanfar received the B.S. and M.S.E. degrees in Electrical Engineering in 1973 and 1979, respectively. He received his Ph. D. degree in Electrical Engineering from Michigan State University, U.S.A., in 1981. Currently, he is a Full Professor in Electrical Engineering Department of Iran University of Science and Technology, Tehran, Iran. His research interests are in the Application of Artificial Intelligence to Power System Control Design, Dynamic Load Modeling, Power System Observability Studies, Voltage Collapse, Congestion Management in a Restructured Power System, Reliability Improvement in Distribution Systems and Reactive Pricing in Deregulated Power Systems. He has published more than 490 technical papers in the International Journals and Conferences proceedings. He is a member of Iranian Association of Electrical and Electronic Engineers and IEEE.

A study on Violence in Major Metropolises using a variables treatment via neural networks - Self Organizing Maps

Isnard T. Martins Author[1] **and Edgard T. Martins Co-author**[2]
[1]Professor Dr. Universidade Estácio de Sá, Rio de Janeiro, R.J, Brazil
[2]Professor Dr. Universidade Federal de Pernambuco, Recife, PE, Brasil

Abstract - *This paper provides a classification of violence in Brazilian cities, treating homicide rates related to population, grouping processed using neural networks results. The source data include indicators of homicides per 100,000 inhabitants, referring to the Brazilian capitals, edited by the Ministry of Health (2003) and demographic indexes disclosed by the IBGE (2010). The ultimate vision of the study will be the result of treatment of the variables processed via neural network, using SOM (Self Organized Method) Competitive Learning architecture, gathering the capitals in four clusters. A original method to extract organizational samples plotted on trigonometric space was used.*

Keywords: Neural Networks, Competitive learning, Operations Research

1 Introduction

A method for classifying subgroups known as as self-organizing maps is used to study organizational information settings, applying unsupervised classification techniques of data through a tool that use principles of neural networks. This method use unorganized data entry, not knowing possible classes or organizational standards relating to selected data. One of these methods is called SOM - Self Organized Maps applying algorithms auto unsupervised learning, where neurons compete for the discovery of input patterns in a method known as Competitive Learning, a form of learning where the set of input patterns is partitioned into specific groups constituting a simplified form called "Winner takes all".
The objective of this algorithm, geared towards self-organized learning is to discover significant patterns or features in the input data, developing discoveries without the aid of a teacher. To foster learning, the algorithm has a set of rules of a local nature, which enable learning to calculate an input-output mapping with specific desirable properties. The word "local" means that the synaptic weight of a neuron is limited to the immediate neighborhood of that neuron. The learning process goes repeatedly modifying the synaptic weights of a neural network connections in response to activation patterns, meeting the pre-established rules, to develop a final configuration for the selected sample.

2 The Method

One method used in neural networks, known as self organized maps is used to classify input information, which are unknown classes or organizational patterns inherent to the selected data. One of these methods is called SOM - Self Organized Maps, applying self learning algorithms (unsupervised) where neurons compete for the discovery of input patterns in a method known as Competitive Learning, a form of learning that partitions the set of input patterns specific groups of data [1].
VisualSom is a software developed by the author [5] for processing standardized variables, collected data base of criminal information were processed for SOM analysis. To effect from test data of the main Brazilian cities in 2010 [6], which normalized after, they become treated vectors that can take ranges of maximum and minimum values, which can vary from +1 to -1 were collected. Successive interactions train the neural network, constructing the desired numbers of clusters for testing. The results of this test were grouped into four clusters, representing the data as the ratio of crimes per 100,000 inhabitants in the cities analyzed.

3 Competitive Learning

Neural networks have been extensively used for multiple purposes which are required mathematical principles to finding input patterns inherent in the data groups. Various applications using neural networks are reported in the literature. Hauck cites the application of neural networks for the extraction of entities from texts by Tucson Police, Texas [7]. Xu and Chen [8] mentioned the use of neural networks to identify shortest paths in graphs, applying a Hopfield network of two layers to research the problem of shortest path and Hollmen et al cite applications for troubleshooting diagnosis, image pattern recognition, process control, analysis and monitoring.
Neural networks are able to generalize their knowledge from previous examples, with the ability to handle noise and distortion, responding properly to new patterns [1]. Neural networks find applications in many fields such as data modeling, predictive time series analysis, pattern recognition, due to an important feature: the ability to learn from data entry with or without supervisor.
According to Haykin [2], the goal of an algorithm for self organized learning is to discover significant patterns or

features in the input data, making this discovery without a teacher. To foster learning, the algorithm has a set of rules of a local nature, which enable learning to calculate an input-output mapping with specific and desirable properties. The word "local" means that the application to the synaptic weight of a neuron is limited to the immediate neighborhood of that neuron. The learning process repeatedly modifies the synaptic weights of a neural network connections in response to activation patterns, meeting the pre-established rules, to develop a final configuration.

The main goal of a self-organizing map (SOM) is to transform an input pattern of arbitrary dimension into a single or two-dimensional map, converting the inputs adaptively in a topologically ordered map. There are three essential processes involved in the formation of a self-organizing map:

- **Competition** - the neuron with the highest value of the discriminant function, which provides the basis for competition is declared the winner.
- **Cooperation** - the winning neuron determines the spatial location of a topological neighborhood of excited neurons, providing the basis for cooperation between the neighboring neurons.
- **Synaptic Adaptation** - allows the excited neurons increase their individual values of the discriminant function with respect to the input pattern by adjusting its synaptic weights.

3.1 Characteristics of Competitive Learning

Assuming that all input vectors and weight vectors are normalized - there are all vectors within a circle of radius 1.
Classification techniques have been used to identify groupings of homogeneous features, also known as clusters. Assuming the preliminary unawareness of existing classes or relationships between entities in a group, Kohonen cites the problem called unsupervised classification, where are employed data clustering methods to identify subgroups [3].

Hayken mentions that in Competitive Learning, the output neurons of a neural network compete to become active, thus only a single output neuron will be active at any given moment [2]. This feature is critical in this learning model to allow classification of a set of input patterns. According to this method, the learning mechanism allows only one winner neuron will be active ("on"), the neuron "takes all" [9].

3.2 The VisualSom Tools

A method for extracting models of organizational intelligence map was developed by the author [11]. This application, implemented in Visual Basic 6, fitted for a classification algorithm of organizational structures that uses models of neural networks with unsupervised learning. This application called VisualSOM, enables unsupervised classification of variables and provides results in a self-organized data map.

This application has the following key functions:

- Demonstrate graphically the results promoted by a competition between processors in a self-organized neural network.
- Demonstrate and monitor in real time the competition of neurons, showing in graphic form the movements of the positions taken by the interactions during competition.
- Develop tests for a method called Competitive Learning, providing as a result, Clusters and clustering of standardized data.

This study features settings of subgroups extracted from information databases, which can reveal important results to identify patterns and groups, through a system of unsupervised learning.

Different formats of connectivity arise in settings of subgroups, which can be classified into classes in order to identify if perhaps operational similarities may exist.
A test using this model was applied to generate clusters, using a criminal activities information database of in Brazilian capitals.

The Table 1 presents the original data received and procedures for neural competition.Table 2 presents the data used by the SOM algorithm to identify patterns.

Table 1 - List of data used in the test for organizing Clusters

Capitals	Homicide / 100 mil	Population
Aracaju	33,6	571.149
Belém	21,9	1.393.399
Belo Horizonte	28,2	2.375.329
Boa Vista	46,4	284.313
Brasília	33,5	2.570.160
Campo Grande	37,2	786.797
Cuiabá	65,6	551.098
Curitiba	21,1	1.751.907
Florianópolis	11,1	421.240
Fortaleza	24,3	2.452.185
Goiânia	22,2	1.302.001
João Pessoa	31,9	723.515
Macapá	43,4	398.204
Maceió	37,9	932.748
Manaus	32,4	1.802.014
Natal	6,7	803.739
Palmas	21,8	208.165
Porto Alegre	30	1.409.351
Porto Velho	60,7	373.917
Recife	67,4	1.537.704
Rio Branco	35,2	336.038
Salvador	11,8	2.675.656
São Luís	14,9	1.014.837
Teresina	20,3	814.230
Vitória	54,4	327.801

Table 2 - List of normalized data used in the test for organizing Clusters

Normalized Data Homicide	Normalized Data Population
-0,003	-0,63
-0,350	0,05
-0,163	0,78
0,377	-0,82
-0,006	0,75
0,104	-0,44
0,947	-0,60
-0,374	0,32
-0,671	-0,70
-0,279	0,78
-0,341	-0,10
-0,053	-0,51
0,288	-0,73
0,125	-0,32
-0,039	0,23
-0,801	-0,42
-0,353	-0,84
-0,110	0,07
0,801	-0,72
1,000	0,12
0,045	-0,77
-0,650	1,00
-0,558	-0,27
-0,398	-0,41
0,614	-0,77

The SOM function is generated from the centroids entries where the entries are grouped, resulting in the targeting of the inputs and in similar organizational structures.
Assuming that all input vectors and weight vectors are normalized - there are all vectors within a circle of radius 1.

The example illustrated in figure 1 shows partial input vectors distributed in a competition by the search for patterns of organizational settings. The line called W4 (neuron) is a synaptic weight that will compete with existing weights for seeking pre-existing input patterns.

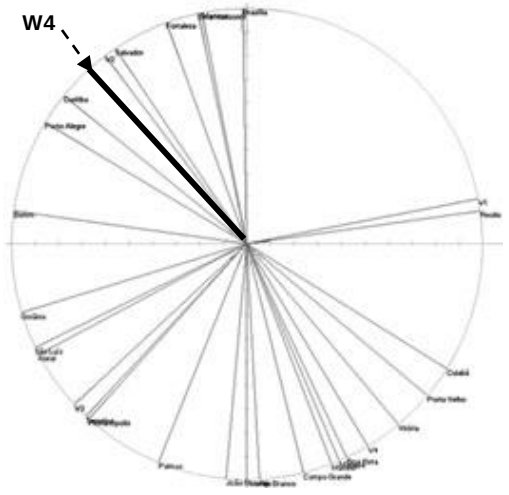


Figure 1 - Partial view of a pre-competition SOM

3.3 Treatment of information input for competition

The variables are called "Inputs" and concentrators neurons (clusters) are called "weights". This information remain temporarily residing in a table located in a working panel of the program (Figure 2). In basic SOM algorithm, the topological relationships and the number of neurons used is determined from the beginning. The scale selected for the sample may affect the accuracy and generalizability of the model. The number of neurons determine the level or density of the resulting model [9]. If the number of clusters is unknown, it is recommended a division of hyperspace, between two to √n clusters, where n is the number of samples in the data set [10].

After reading the inputs, the system distributes VisualSOM variables in hyperspace in a circle of radius 1, which will start a contest of self learning. The input data fed to competition are organized into a table and converted to trigonometric references. Unlike other approaches of neural networks, the SOM network performs unsupervised training, thus during the learning process, the network processing units adjusts its weights based, mostly, in neighboring connections [12].

The variables are subject to the following treatment steps:

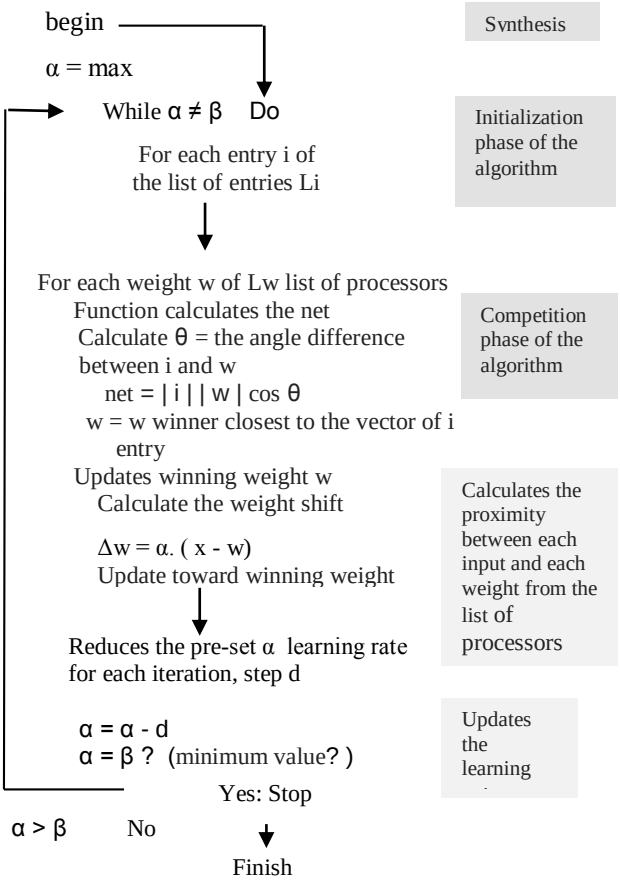
- Identification of potential organizational similarities.
- Extraction of organizational settings selected.
- Export of the structure containing the list of organizational settings to an environment where self learning principles of neural networks for pattern classification are applied.
- Normalization the data for processing by the SOM algorithm.
- Groups are assigned according to which it is desired to data clusters.
- Activation of the net function of self-organization (competition) resulting in representative groups of entries concentrators, into similar organizational structures.

4 Algorithm processed by VisualSom System

Consider the database M, representing homicides in Brazilian capitals during the annual period of 2003.
Consider Li as a list of iinputs which containing the standard settings related to (i1, i2, i3 ... in) representing population ratios and homicide, extracted from database M.
The entries correspond to organized groups on which it is desired to obtain patterns and similarities in their settings.
Consider Lw a list of weights, containing selected processors to clusters identification (w1, w2, w3 ... wn)
α contains the learning rate for the evolution of the algorithm.

The learning rate is initialized to the maximum value for α developments planned for the algorithm.
d contains the value of the Predetermined step to reduce α, each iteration of the algorithm
β is the minimum value of α for completion of the learning cycle

VisualSom algorithm for unsupervised classification [11]



The competition adjusts the weights so that similar input vectors activates the same neuron. Only the weight of the winner neuron is updated, the winner will be the one that best represents the input pattern of the group to which it belongs.
The update of the weights of the winner processor approximates the vector W every step further from the current input vector. The competition uses a learning rate (α), typically assuming adaptive value between 0.5 and 0.1, which decreases mono tonically with learning.
Kohonen mentions that the data must be normalized before applying self organization algorithm [3]. This procedure is not required, but recommended, given of better numerical precision results and handling limits of the input data

5 Additional information about the Competition

Figure 2 illustrates the graphical display system VisualSom where entries are distributed for classification.

A key feature of the system is VisualSom display dynamically the evolution of the competition, allowing follow the trail of displacement of each neuron during the competition.

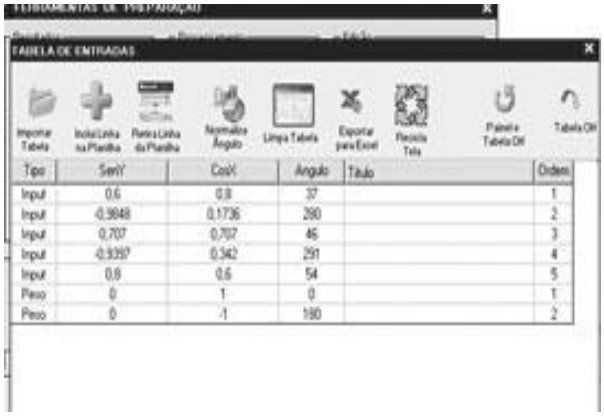


Figure 2-Input Table of information [11]

Inputs can optionally be displayed in the graph pane, labeled with mnemonic codes or unique entry titles (original names of the inputs). Different colored lines show the course of action of each neuron during the competition. Each stage of the competition uses a learning rate that is used by the SOM algorithm, which decays mono tonically in successive iterations, up to a predetermined limit, ending the cycle of learning. The command for operation are activated from the Control Panel of VisualSom (Figure 3)

Figure 3 – Functions and Control Panel VisualSom

The competition of the processors is opened by pressing [F11] or directly on the System panel.
With this operation, the system performs 20 complete cycles, using the parameters set in the Function panel (Alpha value and number of cycles of the competition).

The next phase is the refinement operation, decreasing the learning rate (alpha value) in successive iterations. Figure 4 presents a Monitor Panel. The progress bar indicates the progress of the processing. The lighter panel on top shows the current rate of learning in use and LEDs that indicate whether the network is learning or is paralyzed in the current cycle. The signalman also indicates learning, presenting in the Green, if stage is enabled or red if it is paralyzed. If you want to stop processing just a click of the mouse on the sign. This small LED next to the switch on the panel is activated by errors originating in the operation of the program.

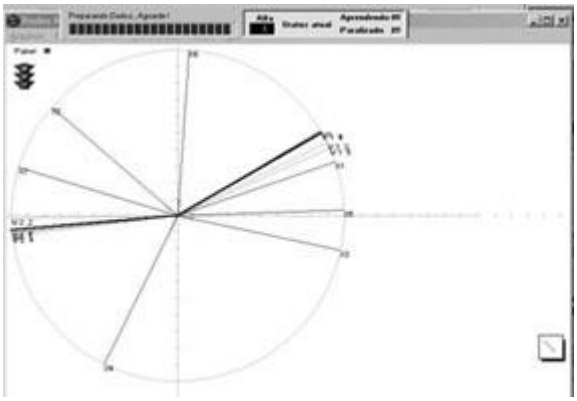


Figure 4 - Competition Monitoring Panel [5]

The system uses VisualSom principles of competitive learning [4], aiming to search similar organizational patterns in the input data . Processing was carried out from a temporary file exported from Excel, containing standardized data where was organized a survey of organizational patterns.

Table 3 presents the main characteristics of the SOM neural function, used as follows:

Table 3 - SOM Neural Functions

Rule Propagation	$net_j = \Sigma x_j . w_{ji} = x.w_j$ (1)
Activation function	Step (for the winner neuron only)
Topology	A single layer of processors
Learning algorithm	Unsupervised $\Delta w_{ki} = \alpha . (x_i - w_{ki})$ (2)
Values of input / output	Binary / continuous

Figure 5 illustrates the Learning Process:

$$\Delta w_{ki} = \alpha . (x_i - w_{ki}) \qquad (3)$$

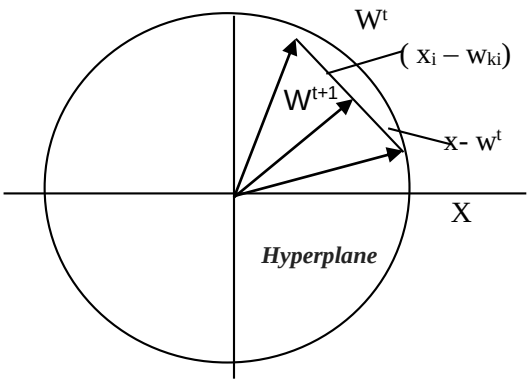


Figure 5 - Competition Hiperplane

Only the weight of the winner neuron is updated because this neuron is the one that best represents that input pattern of the group to which it belongs.
The update of the weights of the winner processor ever closer the W vector from the current input. In the case of normalized vectors, the winner is the input processor closest of vector W. The winner is the processor which has the largest of net function value [4].

$$net_1 = \Sigma x_j . w_{ji} = x.w_j = ||x_2|| \, ||w_1|| \cos \theta = \cos \theta \qquad (4)$$

The propagation rule (weighted sum) indicates the proximity of the two vectors.
We use $0 < \alpha < 1$ because we want the vector W represents a class or group of patterns and not a specific pattern.
Thus W seeks the center of the group.
The value of α should be adaptive, decreasing with time.

7 Conclusions and Results

The completion of this study presents a result of treated variables processed by neural networks using a SOM (Self Organized Method) Competitive Learning architecture, gathering the homicides of brasilian cities in four main clusters.

After the first results were observed in this study distortions caused by vectors of great magnitude within the sample, were the representatives of Rio de Janeiro and São Paulo. In this case it is recommended extract such values and again normalize the vectors, and after, reprocess again the data set.

This distortion causes a concentration in a cluster by high magnitude data and in others, can remaining overpopulated clusters. Figure 4 illustrates classification where two vectors (X22 and X25) which have exceptionally high values. These extremes Vectors were extracted from input sample data and thus were reprocessed the final results.

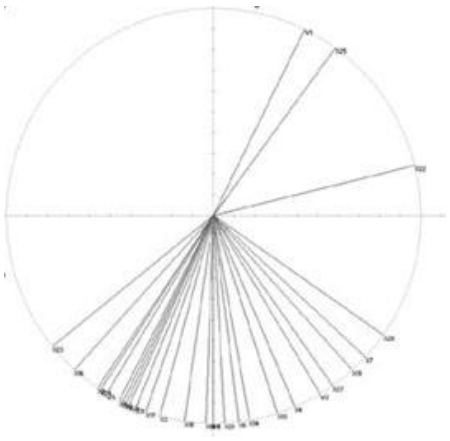


Figure 6- Rating with distortion

At this preliminary stage of the analysis, the input vectors (numeric values) were plotted on the unit circle, with the distribution shown in Figure 5. We can observe, in the first quadrant, the reference vectors of Rio de Janeiro and São Paulo, which, due to magnitude, caused a distortion in the

other vectors of the selected set for the test, therefore were extracted.

7.1 Final Results

- The vector associated with the population, positioned above the horizontal axis represents a relatively high value;

- The vector associated with the population, positioned below the horizontal axis represents a relatively low value;

- The vector associated with Homicide Index, positioned to the right of the vertical axis represent a relatively high indicator;

- The vectors associated with Rio de Janeiro and São Paulo were extracted of neural classification, being significant for final analysis.

Capitals	Homicide / 100 mil	Population
Rio de Janeiro	49,50	6.320.446
São Paulo	58,50	11.253.503

The processed results were divided into four quadrants, divided into four groups, combining the trigonometric reference:

magnitude	*population index*
Sine > 0,5 :	large
0 < Sine < 0,5 :	Medium/large
0 > Sine < -0,5 :	Medium/small
Sine < -0,5 :	small

magnitude	*Homicide index*
Cosine > 05 :	large
0 < Cosine < 0,5 :	Medium/large
0 > Cosine < -0,5 :	Medium/small
Cosine < -0,5 :	small

The hyperspace was arranged as shown in Figure 6 and the final distribution of the vectors after the competition, as shown in Figure 7.

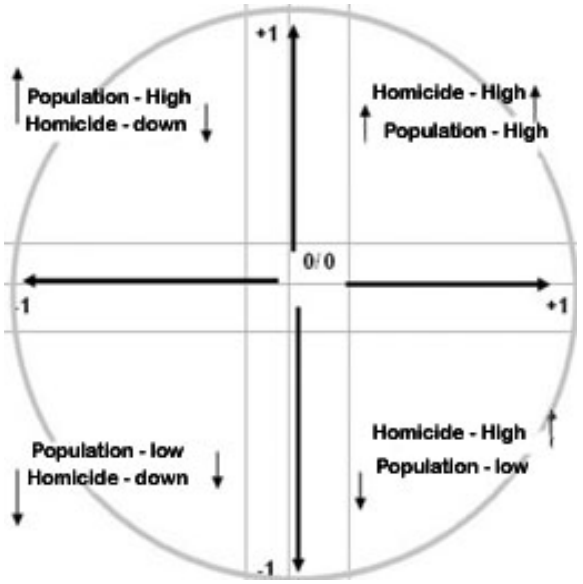


Figure 6 - Final configuration of distribution of vectors

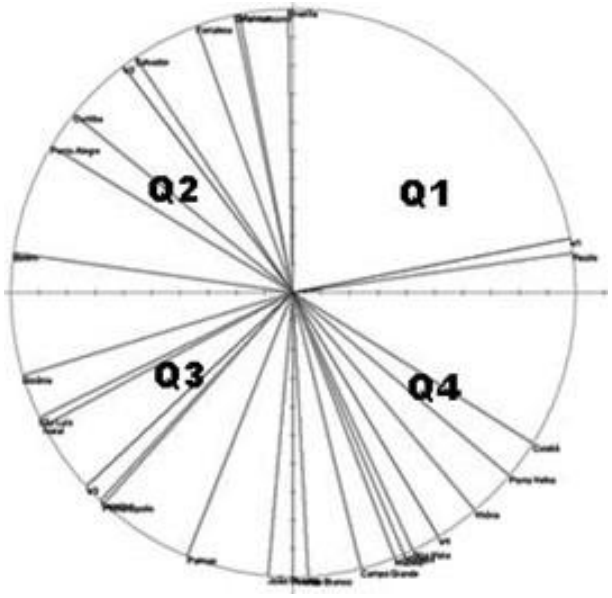


Figure 7 - Results of the competition after treatment unsupervised

Using a trigonometric circle as a classification environment (hyperspace), the variables were distributed in the quadrants of the hyperspace where, after processing of the neural algorithm, centroids (neurons) best representative of capitals in each respective quadrant were generated. Quantities identified as "Large", "Small", "High" or "Low" are references without intermediate variations or sub-classifications, without restrictions imposed only for the of this essay purposes.

Table 4 presents the final results of this study, as follows (references shown in the final results are limited to the analyzed model)

Table 4 - Summary of final results

Quadrant 1 Populational - large (+) Homicides - large (+)	Quadrant 2 Populational - large (+) Homicides - few (-)
Recife (extracted) Rio de Janeiro São Paulo	Belém Belo Horizonte Brasília Curitiba Fortaleza Manaus Porto Alegre Salvador
Typical City of Centroid 1	Typical City of Centroid 2
Recife	Salvador
Quadrant 3 **Populational - few (-)** **Homicides - few (-)**	**Quadrant 4** **Populational - few (-)** **Homicides - large (+)**
Florianópolis Goiânia Natal Palmas São Luís Teresina	Aracaju Boa Vista Campo Grande Cuiabá João Pessoa Macapá Maceió Porto Velho Rio Branco Vitória
Typical City of Centroid 3	Typical City of Centroid 4
Teresina	Boa Vista

As a result of this study, was offered a classification of violence in Brazilian cities, treating related homicide rates in the population, gathering processed through neural networks results. The source data were represented by indicators of homicides per 100,000 inhabitants, referring to the Brazilian capitals, edited by the Ministry of Health and demographic indexes disclosed by the IBGE [6].

8 References

[1] VELLASCO Marley, *Redes Neurais - Notas*, ICA Núcleo de Pesquisa em Inteligência Computacional Aplicada, PUC-Rio, 2005

[2] HAYKIN S. *Redes Neurais, Princípios e Prática*. (2001) Bookman, 2a, Edição. Porto Alegre, 2001

[3] KOHONEN, T. *"The Basic SOM." Self-organizing maps*. Springer Berlin Heidelberg, 2001. 105-176.

[4] KOHONEN, T., *Self-Organization Maps*, Springer-Verlag, Berlin. 1995.

[5] ISNARD Martins. Software Visual SOM – Competitive Learning Redes Neurais. DEI , PUC-Rio, 2005/2006/2009

[6] IBGE. Instituto Brasileiro de Geografia e Estatística. Anuário Censo Demográfico, 2010.

[7] HAUCK R.V, H. Atabakhsh, P. Ongvasith, H. Gupta, H. Chen, *Using coplink to analyze criminal-justice data*, IEEE Computer 35 (3) 30– 37, 2002.

[8] XU Jennifer, Chen H, *Fighting organized crimes: using shortest-path algorithms to identify associations in criminal networks*. Decision Support System 38 (2004) 473-487

[9] HOLLMEN J, Vesanto J, Simiula 0, Alhoniemi E. *Process Modeling Using the Self-Organizing Map*. Integrated Computer-Aided Engineering Journal IOS Press. 1999

[10] VESANTO J, Alhoniemi E. *Clustering of the self-organizing map*. IEEE Trans. Neural Networks 11. (2000) 586–600

[11] ISNARD Martins. *Descoberta de Conhecimento em Históricos Criminais: Algoritmos e Sistemas. Tese de Doutorado*. PUC-Rio, Engenharia Industrial, 2003

[12] KIANG Melody. *Extending the Kohonen self-organizing map networks for clustering analysis*. Computational Statistics & Data Analysis 38 (2001) 161–180

Information-Theoretic Supervised SOM with Multiple-Winners

Ryotaro Kamimura
IT Education Center and School of Science and Technology, Tokai University
1117 Kitakaname Hiratsuka Kanagawa 259-1292, Japan
ryo@keyaki.cc.u-tokai.ac.jp

Abstract— *In this paper, we propose a new information-theoretic method to clarify class structure and to improve prediction performance. The method is composed of unsupervised and supervised phase. The unsupervised phase is governed by the information-theoretic SOM. To clarify class structure, we introduce multiple winners. Then, with this knowledge created by the information-theoretic SOM, supervised learning is applied. We conducted a preliminary experiment to examine the effect of multiple winners and how knowledge obtained by the information-theoretic method can affect the performance of supervised learning. Experimental results showed that multiple winners could be used to clarify class structure. When the number of winning neurons increased, clearer class boundaries appeared. In addition, knowledge obtained by this unsupervised learning could greatly accelerate supervised learning and the final testing errors were smaller than those obtained without this knowledge.*

Keywords: multi-layered neural networks, SOM, competitive learning, multiple winners, supervised, unsupervised, information-theoretic, unlabeled

1. Introduction

In the present paper, we propose a new type of information-theoretic method to clarify class structure and to improve prediction performance. The clarification of class structure has been one of the most important objectives of the self-organizing map (SOM) [1], [2], [3]. Though the SOM has been widely used to visualize class structure, we have had severe difficulty in clarifying class structure, because SOM does not necessarily create interpretable knowledge from its self-organizing processes. Thus, there have been many attempts to improve the clarification of class structure [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16]. To improve visualization performance, we have so far introduced the information-theoretic SOM [17]. By controlling mutual information or free energy, final representations can be easily changed and we can obtain very interpretable representations. In addition, we have found that the number of winners in the self-organizing processes has much influence on the clarification of class structure. In the present paper, we try to confirm the use of the multiple winners for the clarification of class structure.

Then, we examine whether this knowledge obtained by the information-theoretic method can be used to improve prediction performance. We have had attempted to extend the SOM to supervised learning [18] because we expect that SOM knowledge can be used to facilitate supervise learning. However, we have not yet succeeded in incorporating fully the knowledge by the information-theoretic SOM. In the present paper, knowledge obtained by the information-theoretic SOM is directly given to supervised learning. Thus, we can evaluate the prediction performance more exactly.

2. Theory and Computational Methods

2.1 Information-Theoretic Supervised SOM

In the first place, we explain how to train neural networks by using the information-theoretic supervised SOM. As shown in Figure 1(a), we suppose that there are a number of unlabeled data, but the number of labeled data is very limited. To utilize fully the knowledge contained in abundant unlabeled data, we train a neural network in an unsupervised way. This unsupervised learning is performed by the information-theoretic method where principally learning lies in increasing information contained in the neural network. In learning, neurons compete and cooperate with each other. The cooperation between neurons is based on relations between the limited number of winning neurons. Figure 1(b) shows an unsupervised phase where a neuron at the center is influenced by the first and the second winner in Figure 1(b2).

Then, by using the knowledge obtained by this unsupervised learning, we train the neural network in a supervised way with a limited number of labeled data in Figure 1(a) [19]. As shown in Figure 1(c1), connection weights obtained by the unsupervised learning are directly used for supervised learning.

2.2 Firing Probabilities by Multiple Winners

Let us explain how to compute firing probabilities. Now, the sth input pattern can be represented by $\mathbf{x}^s = [x_1^s, x_2^s, \cdots, x_L^s]^T,\ s = 1, 2, \cdots, S$. Connection weights into the jth competitive neuron are denoted by $\mathbf{w}_j = [w_{1j}, w_{2j}, \cdots, w_{Lj}]^T,\ j = 1, 2, \ldots, M$. The output from a output neuron is computed by

$$v_j^s = \exp\left(-\frac{\| \mathbf{x}^s - \mathbf{w}_j \|^2}{2\sigma^2}\right), \tag{1}$$

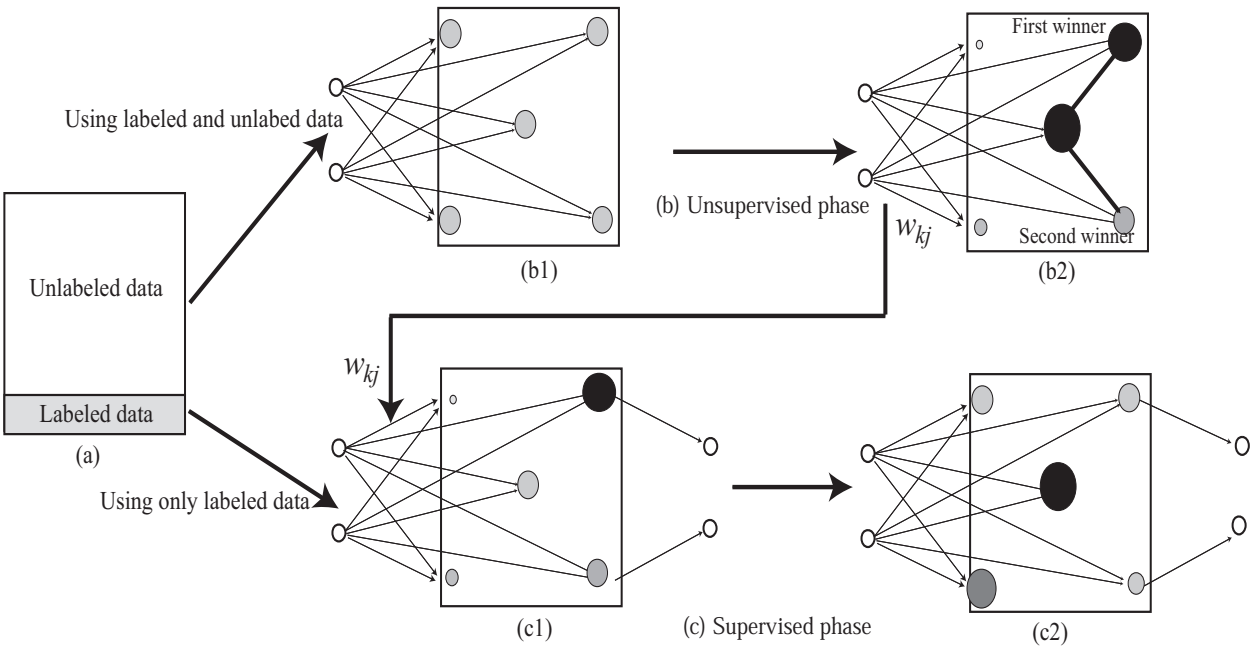


Fig. 1: Learning processes for the information-theoretic supervised SOM.

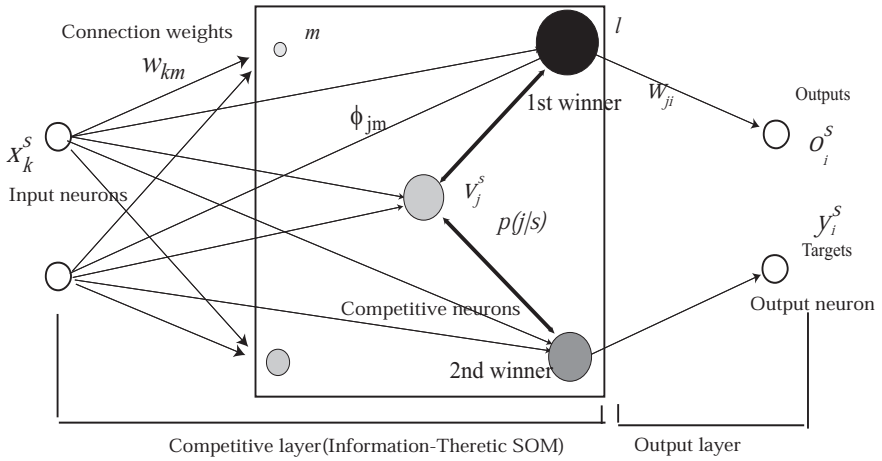


Fig. 2: Network architecture for the information-theoretic supervised SOM.

where σ denotes a spread parameter. By normalizing the output, we have the firing probability

$$p(j \mid s) = \frac{v_j^s}{\sum_{m=1}^{M} v_m^s}. \tag{2}$$

In addition to this firing probability, we consider the output by multiple neurons or winners to realize cooperation between neurons as done in SOM. Now, suppose that the neurons c_1, c_2 are the first and the second winner and so on. then the corresponding outputs can be ranked as follows:

$$v_{c_1} > v_{c_2} > \ldots > v_{c_M}. \tag{3}$$

Following the formulation of SOM, we compute distance between the winner and the other neuron by

$$\phi_{jc_1} = \exp\left(-\frac{\| \mathbf{r}_j - \mathbf{r}_{c_1} \|^2}{2\sigma_{ngh}^2}\right), \tag{4}$$

where $\mathbf{r}_j$ denotes the position of the jth neuron on the output map and σ_{ngh} is the spread parameter. The jth neuron's output is the weighted sum of R winners' outputs and computed by

$$z_j^s(R) = \sum_{m=1}^{R} \phi_{jc_m} v_{c_m}. \tag{5}$$

The firing probability by the multiple winners is defined by

$$q(j \mid s; R) = \frac{z_j^s(R)}{\sum_{j=1}^{M} z_m^s(R)}. \quad (6)$$

2.3 Unsupervised Phase

In the unsupervised phase, the individual neurons try to imitate the outputs by multiple winners to realize self-organizing. Learning should be performed to reduce difference between these outputs. We represent this difference by using the Kullback-Leibler divergence

$$KL = \sum_{s=1}^{S} p(s) \sum_{j=1}^{M} p(j \mid s) \log \frac{p(j \mid s)}{q(j \mid s; R)}. \quad (7)$$

In addition to this KL divergence, we have the other errors which must be minimized, namely quantization errors between connection weights and input patterns

$$Q = \sum_{s=1}^{S} p(s) \sum_{j=1}^{M} p(j \mid s) \|\mathbf{x}^s - \mathbf{w}_j\|^2. \quad (8)$$

Fixing this quantization errors and minimizing the KL-divergence, we have the optimal firing rates

$$p^*(j \mid s) = \frac{q(j \mid s; R) \exp\left(-\frac{\|\mathbf{x}^s - \mathbf{w}_j\|^2}{2\sigma^2}\right)}{\sum_{m=1}^{M} q(m \mid s; R) \exp\left(-\frac{\|\mathbf{x}^s - \mathbf{w}_m\|^2}{2\sigma^2}\right)}. \quad (9)$$

In addition, for connection weights, we can obtain the re-estimation formula [17]

$$\mathbf{w}_j = \frac{\sum_{s=1}^{S} p^*(j \mid s) \mathbf{x}^s}{\sum_{s=1}^{S} p^*(j \mid s)}. \quad (10)$$

2.4 Supervised Fine Tuning

In the output layer, the output from the output neuron is computed by

$$o_i^s = f\left(\sum_{j=1}^{M} W_{ji} v_j^s\right), \quad (11)$$

where f denotes a differentiable function and W_{ji} are connection weights from the competitive neurons to the output ones. The error is computed by

$$E = \frac{1}{2} \sum_{s=1}^{S} \sum_{i=1}^{N} (y_i^s - o_i^s)^2, \quad (12)$$

where y is the target and N is the number of output neurons. We can differentiate this error function with respect to connection weights in the competitive and output layer. We here show the update formula for the competitive layer

$$\Delta w_{kj} = \frac{\eta}{\sigma^2 S} \sum_{s=1}^{S} \delta_j^s (x_k^s - w_{kj}), \quad (13)$$

where δ is the error signal sent from the output layer.

3. Results and Discussion

3.1 Experimental Outline

We conducted a preliminary experiment to examine the effect of multiple winners and the effect of knowledge by unsupervised learning on prediction performance. We used the wine data from the well-known machine learning database[1] and chose the first 1000 patterns (500 white and 500 red wines). In the unsupervised phase, we used all 1000 patterns to produce connection weights for supervised learning. For supervised learning, we tried to reduce the number of training patterns as much as possible to see an effect of unsupervised learning clearly. Then, the number of training pater was only set to 50, randomly chosen from 1000 patterns, because with this training, we could have satisfactory good prediction performance. The remaining 400 patterns were for the validation and 500 patterns were for the testing data.

3.2 Effects of Multiple Winners

First, we examined the effect of multiple-winners. Figure 3(a) shows the U-matrix by the conventional SOM. The U-matrix is the most popular method to visualize connection weights and represents distances between neurons. If the distances become larger, the values of U-matrix become larger. Usually, the class boundary is located in the places where the U-matrix shows larger values. As shown in Figure 3(a), by the SOM, class boundaries in warmer color were detected on both sides of the matrix. However, they were not explicit boundaries separating red and white wines. When the number of winners was one in Figure 3(b), the obtained U-matrix was similar to that by SOM in Figure 3(a). When the number of winners increased from 10 in Figure 3(c) to 40 in Figure 3(f), class boundaries in warmer color became more explicit. In particular, a boundary in brown in the middle of the U-matrix was clear. When the number of winners increased from 50 in Figure 3(g) to 70 3(i), those class boundaries in warmer color became slightly weaker. When the number of winners was 110 and 130 in Figure 3(j) and (k), two class boundaries appeared on the upper side and lower side of the U-matrix. Finally, when the number of winners was almost maximum (150), class boundaries were pushed toward the corners of the U-matrix in Figure 3(l).

To check the validity of the boundaries, we plotted the U-matrix as well as the corresponding labels in Figure 4. When the number of winners was 40 in Figure 4(a), white and red wines are classified diagonally, and the class boundary in brown seems to represent this boundary. In addition, the slightly weaker boundary on the upper side of the U-matrix suggests the division of white wine into two groups. When the number of winners increased to 110, in Figure 4(b), two class boundaries in warmer color divided wines into two groups separately.

[1] http://archive.ics.uci.edu/ml/

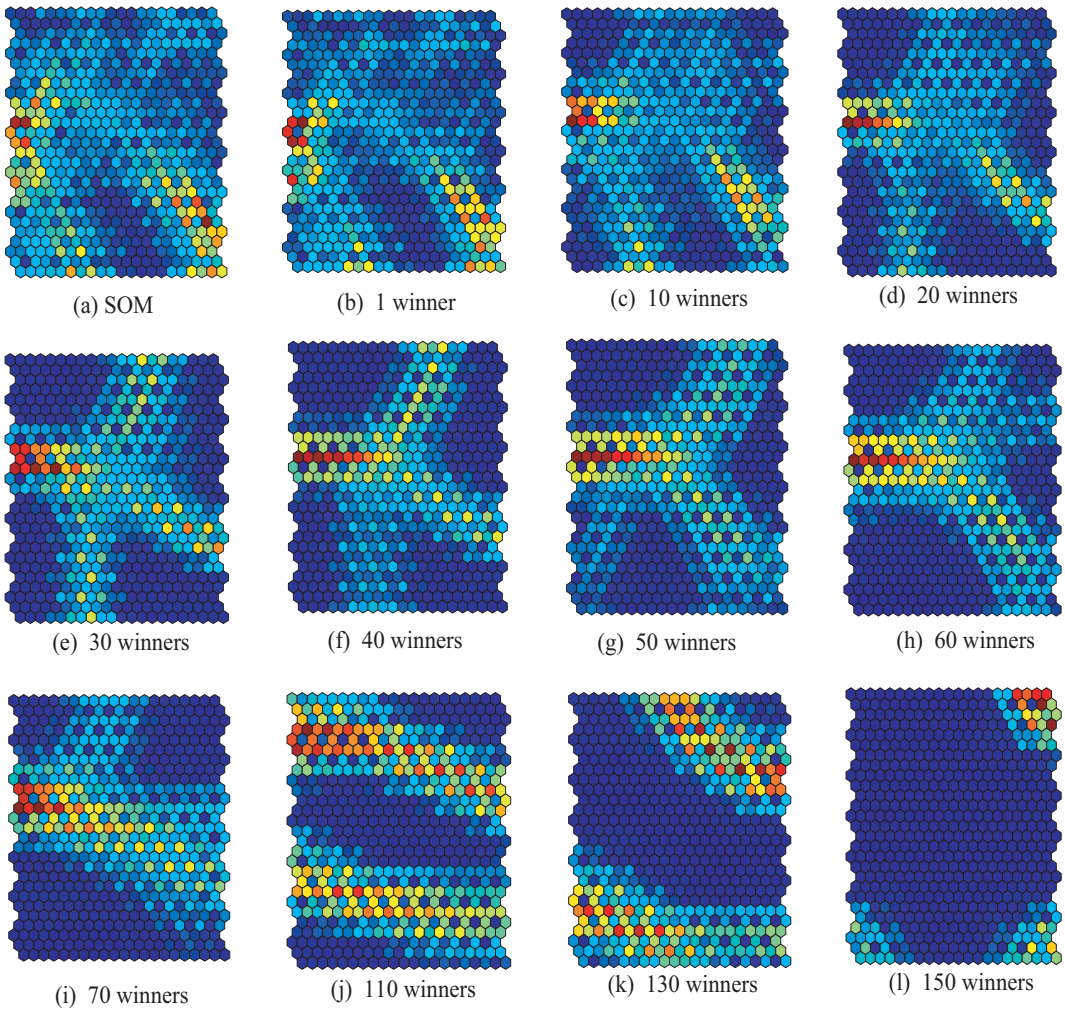


Fig. 3: U-matrices by SOM (a) and the information-theoretic SOM when the number of winners increased from 1 (b) to 150 (l).

Experimental results show that increase in the number of winners can enhance class boundaries.

3.3 Effects on Prediction Performance

Then, we examined whether knowledge obtained by unsupervised learning, affects the prediction performance. Figure 5 shows training (blue), validation (green) and testing (red) error rates by the method without knowledge of the unsupervised learning (a) and with knowledge by the unsupervised SOM (b), when the number of winners was 40 in Figure 4(a). Without knowledge, the lowest validation error was obtained with 136 epochs in Figure 5(a). On the other hand with knowledge by the information-theoretic SOM, the validation error reached its minimum value with only six epochs in Figure 5(b). Thus, we can say that the knowledge by unsupervised learning greatly accelerated the learning.

Figure 6 (a) shows the number of epochs to reach the minimum value of the validation error. The number of epochs by the supervised SOM was very small, ranging between two and six. On the other hand, without knowledge, the number of epochs was 136. Figure 6 (b) shows testing errors when the the validation error were minimum. Without the knowledge, the testing error was 0.044. On the other hand, the testing errors were around 0.02 for any number of winners. For example, when the number of winners was 40, the testing error was 0.018. Experimental results show that knowledge by unsupervised learning could make learning much faster and the prediction performance much higher.

4. Conclusion

In the present paper, we have proposed a new type of information-theoretic method. In the method, unsupervised information-theoretic SOM was first used with multiple winners to obtain knowledge from input patterns. Then, with this knowledge, supervised training was applied. We conducted a preliminary experiment to see whether the method can

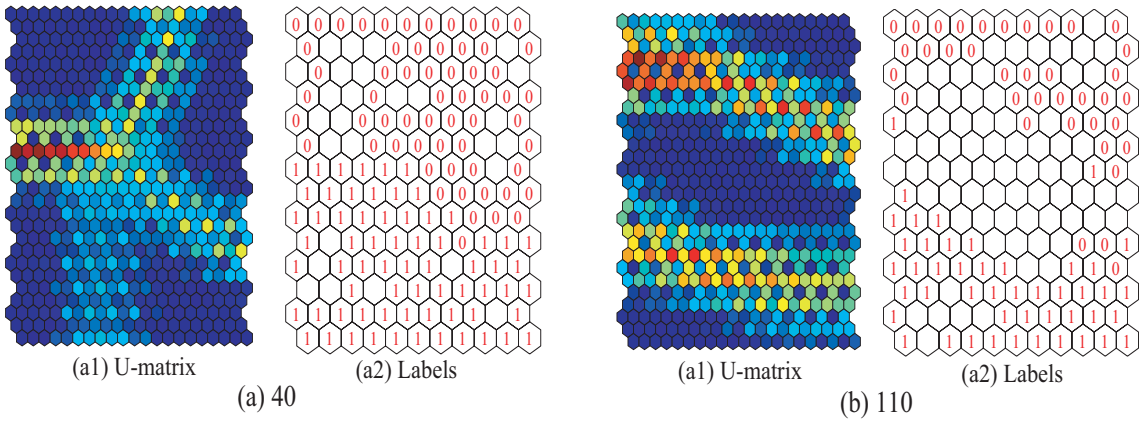


Fig. 4: U-matrices and labels when the number of winners was 40 (a) and 110 (b).

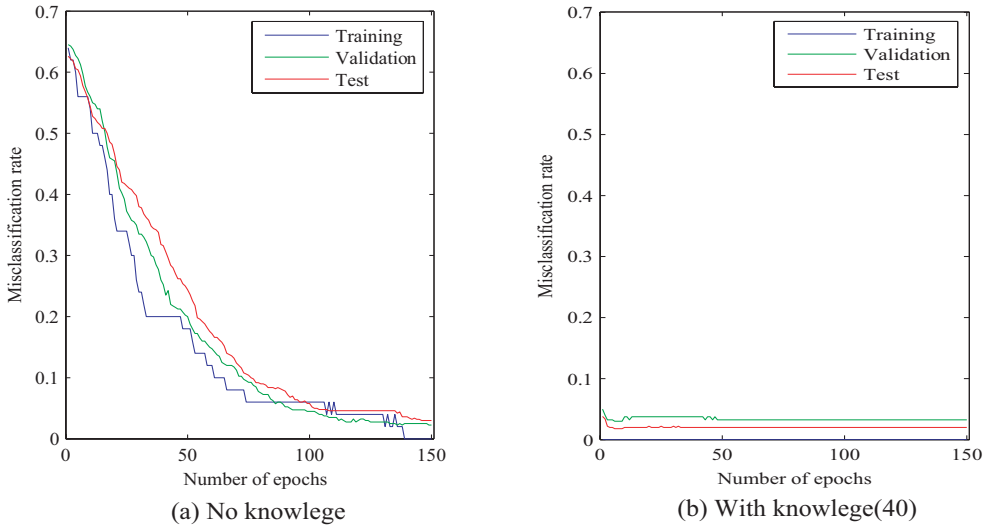


Fig. 5: Training (blue), validation (green) and testing (red) error rates by the method without knowledge by unsupervised learning (a) and with the knowledge (b).

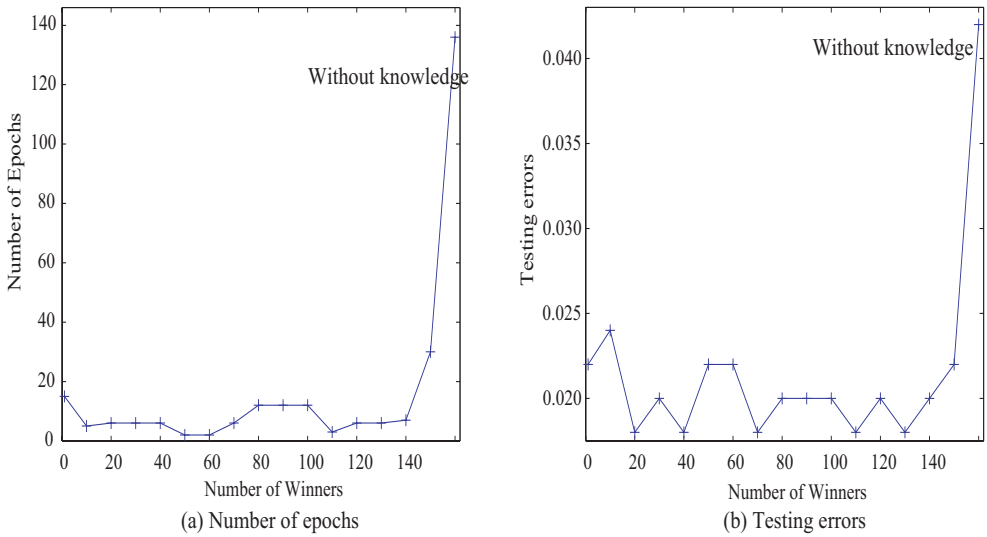


Fig. 6: The number of epochs for reaching the minimum validation errors (a) and testing errors (b) when the validation errors were minimum.

produce clearer class structure and prediction performance increases with this knowledge. Experimental results confirmed that class structure could be clarified by increasing the number of winners and the perdition performance was increased greatly by using the knowledge by the supervised SOM.

The method can be used to improve the performance of neural networks for many problems, because unsupervised and supervised learning are linked seamlessly in the same architecture.

References

[1] T. Kohonen, *Self-Organization and Associative Memory*. New York: Springer-Verlag, 1988.

[2] T. Kohonen, *Self-Organizing Maps*. Springer-Verlag, 1995.

[3] T. Kohonen, "The self-organization map," *Proceedings of the IEEE*, vol. 78, no. 9, pp. 1464–1480, 1990.

[4] E. Merényi, K. Tasdemir, and L. Zhang, "Learning highly structured manifolds: harnessing the power of soms," in *Similarity-Based Clustering*, pp. 138–168, Springer, 2009.

[5] K. Tasdemir and E. Merényi, "Exploiting data topology in visualization and clustering of self-organizing maps," *Neural Networks, IEEE Transactions on*, vol. 20, no. 4, pp. 549–562, 2009.

[6] J. Vesanto, "SOM-based data visualization methods," *Intelligent Data Analysis*, vol. 3, pp. 111–126, 1999.

[7] S. Kaski, J. Nikkila, and T. Kohonen, "Methods for interpreting a self-organized map in data analysis," in *Proceedings of European Symposium on Artificial Neural Networks*, (Bruges, Belgium), 1998.

[8] I. Mao and A. K. Jain, "Artificial neural networks for feature extraction and multivariate data projection," *IEEE Transactions on Neural Networks*, vol. 6, no. 2, pp. 296–317, 1995.

[9] C. De Runz, E. Desjardin, and M. Herbin, "Unsupervised visual data mining using self-organizing maps and a data-driven color mapping," in *Information Visualisation (IV), 2012 16th International Conference on*, pp. 241–245, IEEE, 2012.

[10] S. Shieh and I. Liao, "A new approach for data clustering and visualization using self-organizing maps," *Expert Systems with Applications*, 2012.

[11] H. Yin, "ViSOM-a novel method for multivariate data projection and structure visualization," *IEEE Transactions on Neural Networks*, vol. 13, no. 1, pp. 237–243, 2002.

[12] M.-C. Su and H.-T. Chang, "A new model of self-organizing neural networks and its application in data projection," *IEEE Transactions on Neural Networks*, vol. 123, no. 1, pp. 153–158, 2001.

[13] S. Wu and T. Chow, "Prsom: A new visualization method by hybridizing multidimensional scaling and self-organizing map," *Neural Networks, IEEE Transactions on*, vol. 16, no. 6, pp. 1362–1380, 2005.

[14] L. Xu, Y. Xu, and T. W. Chow, "PolSOM-a new method for multidimentional data visualization," *Pattern Recognition*, vol. 43, pp. 1668–1675, 2010.

[15] Y. Xu, L. Xu, and T. Chow, "Pposom: A new variant of polsom by using probabilistic assignment for multidimensional data visualization," *Neurocomputing*, vol. 74, no. 11, pp. 2018–2027, 2011.

[16] L. Xu and T. Chow, "Multivariate data classification using polsom," in *Prognostics and System Health Management Conference (PHM-Shenzhen), 2011*, pp. 1–4, IEEE, 2011.

[17] R. Kamimura, "Self-enhancement learning: target-creating learning and its application to self-organizing maps," *Biological cybernetics*, pp. 1–34, 2011.

[18] R. Kamimura, "Contradiction resolution between self and outer evaluation for supervised multi-layered neural networks," *Journal of Advanced Research in Artificial Intelligence*, vol. 2, no. 7, pp. 29–38, 2013.

[19] Y. Bengio, "Learning deep architectures for ai," *Foundations and trends® in Machine Learning*, vol. 2, no. 1, pp. 1–127, 2009.

EEG signal classification by improved MLPs with new target vectors

José Ricardo Gonçalves Manzan[1], **Shigueo Nomura**[1], **and João Batista Destro Filho**[1]
[1]Faculty of Electrical Engineering, Federal University of Uberlândia, Uberlândia, Minas Gerais, Brasil

Abstract - *This paper proposes the use of new target vectors for MLP learning in EEG signal classification. A large Euclidean distance provided by orthogonal bipolar vectors as new target ones is explored to improve the learning and generalization abilities of MLPs. The data set consisted of EEG signals captured from normal individuals and individuals under brain-death protocol. Experimental results are related to MLP performance comparison by training the networks with three types of target vectors (conventional, orthogonal bipolar and non-orthogonal ones). We have concluded that the use of orthogonal bipolar vectors as target ones has contributed to improve the MLP performance on tasks for EEG signal classification.*

Keywords: EEG signal brain-death, pattern classification, multilayer perceptron, target vectors, orthogonal bipolar vectors.

1 Introduction

Artificial neural networks (ANN) is a science field that has emerged as a set of powerful tools capable of solving engineering problems that previously could not be solved. One of those tools is multilayer perceptron (MLP) that has been applied to function approximation, time series prediction, pattern recognition [1], sound signal processing [2], and biomedical signal processing [3] problems.

Related works are presented in Section 2. Section 3 presents a motivation for the work. Hypothesis to be solved by this work is described in Section 4. The different types of target vectors are defined in Section 5. In Section 6, we can verify the experimental procedure. Section 7 presents experimental results. Section 8 performs the discussion and Section 9 presents the conclusion.

2 Related Works

Researches on pattern recognition mainly description and classification are fundamental to engineering and science. Several techniques such as statistical approach, theoretical decision and syntactic approach have been adopted [11]. In recent years, ANN techniques represented by MLPs have been widely used because of promising results. One of the advantages of using ANN is the ability for training in a supervised or unsupervised form.

The traditional approaches on artificial intelligence use the sequential processing. On the other hand, MLP models use a learning mode with parallel and distributed processing. His principles, architectures and training have inspired in biological neurons and training takes place by means of examples. Trial and error strategies contribute to the ability to differentiate patterns. MLP has a similar behavior when a large number of neurons send excitatory or inhibitory signals to other neurons composing the network.

Several researchers [4] [5] [6] have focused on improving ANN performances. Some proposed strategies are regarded to input pattern improvement, MLP architecture optimization, and learning algorithm enhancement.

However, strategies related to the use of different target vectors rather than conventional ones are not so usual. For instance, Dietterich has published a work based on error-correcting output codes [7].

EEG signal analysis is very important for public health applications in the context of Brazilian health system, for which costless, non-invasive and simple diagnostic examinations such as electroencephalography (EEG) are needed [8]. In this case, brain death protocols (BDP) require complementary examinations for medicines taking a decision about the therapeutic procedure to be performed on patients at the Intensive Care Unit [9]. Very few works of the literature are devoted to the application of EEG to BDP [10], but none of them, to our knowledge, make use of MLP.

3 Motivation

The biological cognition has abilities to recognize and distinguish patterns, even if they have a high degree of degradation on their features [11] [12] [13]. In the case of ANN, the appropriate choice of parameters can provide good training. In this case the network can get an excellent power of generalization. This is good for constructing a model with high flexibility to properly recognize too degraded patterns. Moreover the choice of parameters may be inappropriate and it can cause the overtraining problem of networks. With the

overtraining, the network loses the generalization ability to correctly classify those degraded patterns.

Several proposals in order to improve the ability to recognize degraded patterns have been carried out. In most cases they have focused on how to treat input vectors [14]. The previous work [15] shows satisfactory results in using improved target vectors for degraded pattern recognition. This work shows effects of adopting orthogonal bipolar vectors as targets on improving the MLP performance to classify the EEG signals.

In addition, for the application discussed in this paper, neural nets are very important, since noise disturbs recorded data, and since too many clinical details influences the standard visual analysis of EEG by neurologists. Generally, decisions on BDP are taken based just on this visual analysis [16].

4 Hypothesis

In case of conventional bipolar vectors (CBV), the inner product between two of them is not null. On the other hand, orthogonal bipolar vectors (OBV) always have null inner product between them. Also, the similarity between two OBVs is lower than that corresponding similarity between two CBVs. Furthermore, the orthogonality between two OBVs leads to the largest Euclidean distance as well as possible. The hypothesis is that larger Euclidean distance and lower similarity of OBVs can affect on the MLP performance improvement to recognize degraded patterns.

This paper presents experimental results of use the OBVs and CBVs in recognition of EEG signals of normal patients and individuals under the brain-death protocol. We have realized that the results are consistent with the hypothesis.

5 Representation of Vectors

5.1 Orthogonal Bipolar Vectors

Equations (1) and (2) represent two possible target vectors, the equation (3) represents the inner product and equation (4) the Euclidean distance.

$$\vec{V} = (v_1, v_2, ..., v_n) \quad (1)$$

$$\vec{W} = (w_1, w_2, ..., w_n) \quad (2)$$

$$\vec{V} \bullet \vec{W}^T = v_1 \cdot w_1 + v_2 \cdot w_2 + ... + v_n \cdot w_n \quad (3)$$

$$d_{V,W} = \sqrt{(w_1 - v_1)^2 + (w_2 - v_2)^2 + ... + (w_n - v_n)^2} \quad (4)$$

Consider the case where $\vec{V}$ and $\vec{W}$ are orthogonal with length *n*. There will be *n*/2 components whose product is positive and *n*/2 components whose product is negative. Positive product of components is related to the ones which the terms have the same signal. These terms do not affect on the result of the Euclidean distance given by equation (4). On the other hand, for the terms with opposite signals, the square of their difference is 4. The squares of differences contribute into the Euclidean distance resolution. Therefore, if we have larger number (*n*) of components then we have larger Euclidean distances. Equations (5) and (6) represent examples of OBVs. The inner product of those OBVs is given by equation (7). The OBVs can be generated by implementing the algorithm as described in [17].

$$\vec{V} \overset{def}{=} (1,1,1,1,-1,-1,-1,-1) \quad (5)$$

$$\vec{W} \overset{def}{=} (1,1,-1,-1,1,1,-1,-1) \quad (6)$$

$$\vec{V} \bullet \vec{W} = 1 \cdot 1 + 1 \cdot 1 + 1 \cdot (-1) + 1 \cdot (-1) + (-1) \cdot 1 + (-1) \cdot 1 + (-1) \cdot (-1) + (-1) \cdot (-1) = 0 \quad (7)$$

5.2 Conventional Bipolar Vector (CBV)

In case of conventional bipolar vector (CBV), the component corresponding to the pattern *i* represented by vector $\vec{U}$ at the position *i* values 1. All the other components value (–1) as represented by equation (8).

$$\vec{U} \overset{def}{=} (-1,-1,...,1,...,-1) \quad (8)$$

If $\vec{U_1}$ and $\vec{U_2}$ are conventional then the terms of equation (4) are null except for two terms corresponding to the positive component of the vector $\vec{U}$ given by equation (8). So, the Euclidean distance for CBVs is smaller than the distance for OBVs. Consider the example of two CBVs, $\vec{U_1}$ and $\vec{U_2}$ that represent two patterns defined by equations (9) and (10) from a set of four patterns to be classified.

$$\vec{U_1} = (1,-1,-1,-1) \quad (9)$$

$$\vec{U_2} = (-1,1,-1,-1) \quad (10)$$

The Euclidean distance between $\vec{U_1}$ and $\vec{U_2}$ is calculated by equations (11) and (12). Those equations show

that only two terms are non nulls. Therefore, the Euclidean distance value between two CBVs is always $2\sqrt{2}$.

$$d_{\overline{U_1},\overline{U_2}} = \left(\begin{array}{l} \left[1-(-1)\right]^2 + \left[(-1)-1\right]^2 \\ +\left[(-1)-(-1)\right]^2 + \left[(-1)-(-1)\right]^2 \end{array} \right)^{\frac{1}{2}} \quad (11)$$

$$d_{\overline{U_1},\overline{U_2}} = \sqrt{\left[-2\right]^2 + \left[-2\right]^2 + 0^2 + 0^2} = 2\sqrt{2} \quad (12)$$

5.3 Non-Orthogonal Bipolar Vector (NOV)

The non-orthogonal bipolar vectors (NOVs) are similar to CBVs. However, they have larger lengths and they are equal to OBVs. They consist of a unitary component at the position "i" to represent the i[th] pattern and other components are equal to "– 1" as given by equation (13).

$$\overrightarrow{T} \overset{def}{=} \left(-1,-1,...,1,...,-1,-1\right) \quad (13)$$

Let $\overrightarrow{T_1}$ and $\overrightarrow{T_2}$ as two NOVs with *n* components represented by equations (14) e (15). The Euclidean distance between $\overrightarrow{T_1}$ and $\overrightarrow{T_2}$ is calculated by equations (16) and (17). In a similar way to CBVs, the Euclidean distance between NOVs is equal to $2\sqrt{2}$.

$$\overrightarrow{T_1} = \underbrace{\left(1,-1,-1,...,-1\right)}_{n \text{ components}} \quad (14)$$

$$\overrightarrow{T_2} = \underbrace{\left(-1,1,-1,...,-1\right)}_{n \text{ components}} \quad (15)$$

$$d_{\overline{T_1},\overline{T_2}} = \left(\begin{array}{l} \left[1-(-1)\right]^2 + \left[(-1)-1\right]^2 \\ +\left[(-1)-(-1)\right]^2 \\ \underbrace{+...+\left[(-1)-(-1)\right]^2}_{n \text{ terms}} \end{array} \right)^{\frac{1}{2}} \quad (16)$$

$$d_{\overline{T_1},\overline{T_2}} = \sqrt{\underbrace{\left[-2\right]^2 + \left[-2\right]^2 + ... + 0^2 + 0^2}_{n \text{ terms}}} = 2\sqrt{2} \quad (17)$$

NOVs can be defined as an extended version of CBVs. For this reason, NOVs can have arbitrary lengths. Then, one can generate NOVs and OBVs with the same lengths. This allows a fair comparison of MLP performances, since OBVs have different lengths compared to CBVs.

6 Experimental Procedure

6.1 Experimental Data

Adult normal individuals and patients were both recorded at Uberlandia University Hospital (HCU), by using a *Linx BNT* EEG amplificator, and the standard 10-20 electrode placement. For each person, a 20-minute recording was performed, then a neurologist randomly selected 15 segments without any kind of artifact, by means of visual analysis. Notice that one of these segments is composed of data of all 20 channels or electrodes during a one-second duration window. All these procedures have been fully authorized by the local Ethics Committee, through protocol 369/11.

"Normal individuals" translates into people aged 18-60 years old, men or women, without any previous neuropathology and not taking absolutely no neurological drug during lifetime. Adult patients under BDP at our local Intensive Care Unit have been selected, so that no effects of analgesics were involved. For patients, a retrospective data collection was performed, considering the period February 2009 - February 2012. Based on this, five patients have been randomly chosen for the MLP processing.

Input data to the MLP are prepared as follows. First, for each normal individual or patient, consider the 15 segments extracted from the EEG recording by the neurologist. These 15 segments were normalized according to (20), which takes into account the average (18) and the standard deviation (19). Then we used Fast Fourier Transform (21) for the estimation of the twenty median frequencies of each segment, wherein each median frequency is tied to one single channel.

$$\overline{x} = \frac{1}{n}\sum_{i=1}^{n} x_i \quad (18)$$

$$s = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(x_i - \overline{x}\right)^2} \quad (19)$$

$$X_i = \frac{\left(x_i - \overline{x}\right)}{s} \quad (20)$$

$$F(u,v) = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y) e^{-j2\textcircled{R}\left(\frac{ux}{M}+\frac{vy}{N}\right)} \quad (21)$$

Wherein:

- *n* is a number of samples.

- x_i is the ith sample value.
- $\bar{x}$ is the mean of data.
- s is standart deviation of data.
- X_i is ith sample normalized.
- M is number of samples of variable 1;
- N is number of samples of variable 2;
- $f(x,y)$ is discretized continuous function in M samples along the x-axis and N samples along the y-axis;
- $F(u,v)$ is value of fourier discrete transform.

Finally, we calculated the mean and standard deviation of the median frequencies of each channel, considering all the fifteen segments. The feature vector for the MLP neural network training finally consisted of 40 values. They were ordered according to the order of the channels, the first of which is the average and the second is the standard deviation.

The current amount of experimental data is small due to the serious difficulties for collecting and registering this kind of data in hospital. Furthermore, various strict conditions for validating the data have limited the amount of experimental data.

Given this fact, sample combinations with the available data set were performed. In each combination were used 2 normal individuals and 2 patients under BDP for training and 3 normal individuals and 2 patients under BDP to test the performance of MLPs. Therefore, six combinations according to Table 1 were obtained.

Table 1: Combinations Used In The Experiment

Combination	Training data		Test data	
	Patients under BDP	Normal individuals	Patients under BDP	Normal individuals
1	1 and 2	1 and 2	3 and 4	3, 4 and 5
2	1 and 3	1 and 3	2 and 4	2, 4 and 5
3	1 and 4	1 and 5	2 and 3	2, 3 and 4
4	2 and 3	2 and 3	1 and 4	1, 4 and 5
5	2 and 4	2 and 5	1 and 3	1, 3 and 4
6	3 and 4	3 and 5	1 and 2	1, 2 and 4

6.2 MLP topologies

Five topologies for MLP training according to the target vector length were adopted. They have been defined by the combination (number of input neurons x number of hidden neurons x number of output neurons) as follows: 40 x 50 x 2; 40 x 50 x 4; 40 x 50 x 8; 40 x 50 x 16 and 40 x 50 x 32.

The number of 2 units in the output layer was defined by number of diagnoses, being normal and brain death. In case of using OBVs as target vectors, since OBVs have been generated with the lengths as 2, 4, 8, 16, 32, 64, 128 and so on, OBVs with lengths of 4, 8, 16 and 32 units were chosen in the output layer. In parallel to the OBVs, NOVs with lengths of 4, 8, 16, and 32 units have been adopted in the experiments.

6.3 MLP training stage

The experimental simulations were performed using the traingdx toolbox of Matlab software version R2008. The traingdx toolbox adopts the momentum and adaptive learning rate for MLP learning. For this reason, we can get a more rapid convergence assuring consistent results. Furthermore, in order to get fair experimental results, we have adopted MLP topologies that allow the use of same initial synaptic weights for training. This procedure was done to assure a fair comparison between different target vectors.

The adopted initial learning rate was 0.1, since traingdx works with adaptive learning rates. For each training epoch is calculated an error that is compared to the tolerance (stopping criterion for training).

The equations for calculating the mean squared error to be used in the stopping criterion are as follows:

$$E_p = \frac{1}{2}\sum_{j=1}^{Ns}\left(d_j - y_i\right)^2 \quad (22)$$

$$E_m = \frac{1}{N}\sum_{p=1}^{N} E_p \quad (23)$$

Where:

- E_p is the squared of a pattern p;
- N_s is the number of output neurons;
- d_j is the desired output for neuron j;
- y_j is the net output for neuron j;
- E_m is the mean squared error of all patterns for each epoch;
- N is the number of patterns;

The training is finalized when the stopping condition $E_m < £$ is satisfied. The value of $£$ is the tolerance for error during the training process.

The simulations were performed on a computer with INTEL(R) CORE i5-2410TM, 2.30GHz processor and 4 GB RAM.

6.4 Experimental Results

The graph of Figure 1 compare the performance of CBV(2), OBV(4) and NOV(4). Figure 2 show graph that compare ther performance of CBV(2), OBV(8) and NOV(8). The comparison of CBV(2), OBV(16) and NOV(16) is represented in the graph of Figure 3. Finally the Figure 4 shows the graph the comparison of CBV(2), OBV(32) and NOV(32).

The best performance is 93.33% with the use of OBV(4) and OBV(8) and 86.7% with the use of CBV(2). In all the cases, the use of OBVs has lead to more consistent results. Furthermore, the results with the use of OBVs show good performances at large values of tolerance for error in training. The performance of CBV(2) and NOVs only achieves good performance with small tolerance values. The overtraining problem has been detected with few epochs of training by using CBVs and NOVs as targets.

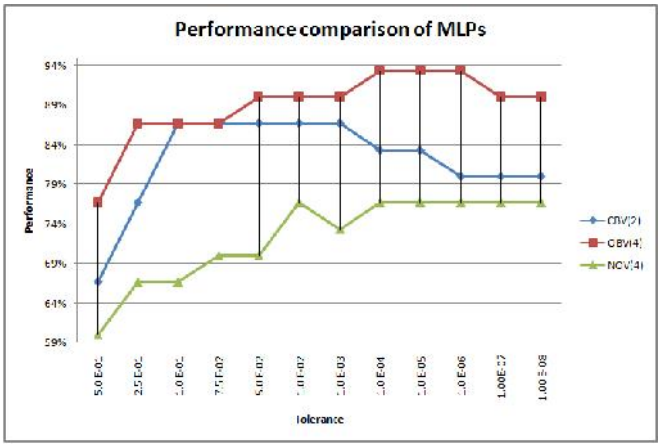


Fig. 1. Results for performance comparison of MLPs trained with CBV(2), OBV(4), and NOV(4) as target vectors

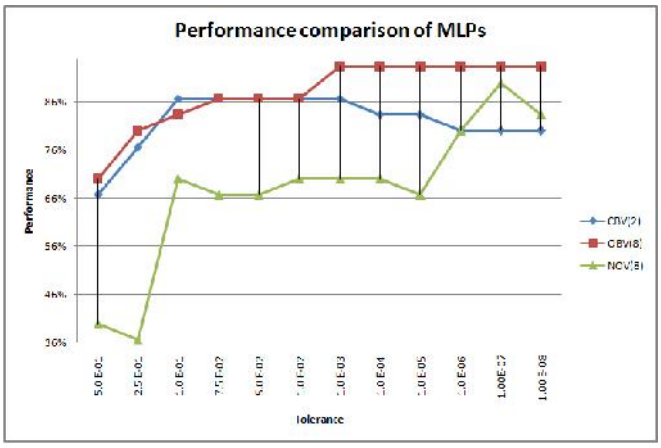


Fig. 2. Results for performance comparison of MLPs trained with CBV(2), OBV(8), and NOV(8) as target vectors

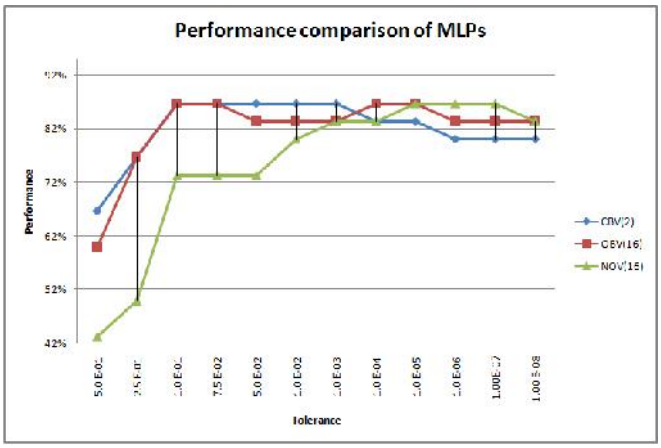


Fig. 3. Results for performance comparison of MLPs trained with CBV(2), OBV(16), and NOV(16) as target vectors

Previously, the OBV experimental results for application to the digits extracted from license plate degraded images [15] have presented an increase of 8% on the MLP performance. All these results have strengthened the viability of using OBVs as target vectors for MLP in pattern recognition.

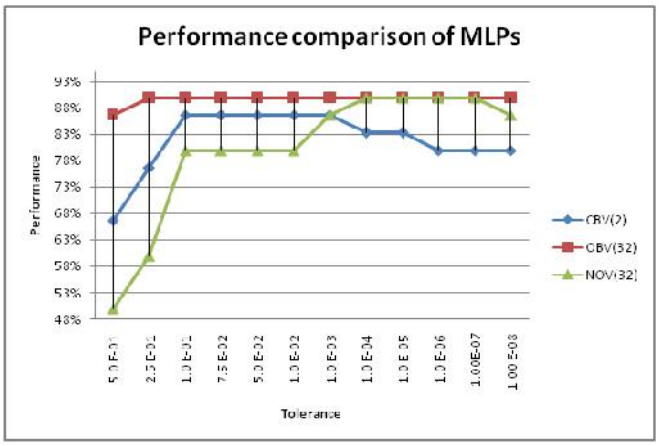


Fig. 4. Results for performance comparison of MLPs trained with CBV(2), OBV(32), and NOV(32) as target vectors

7 Discussion

The results have shown that the use of OBVs as targets improves the performance on recognizing biomedical patterns represented by EEG signals in all the cases.

Besides obtaining better recognition rates, the use of OBVs has contributed into reducing the computational load. With the use of OBVs, the number of epochs for training the ANN is smaller than the case with the use of other vectors.

This is quite evident when comparing CBV(2) and NOV (32) with OBV (32). For a tolerance of 2.5E-01, the recognition rate for OBV (32) is 90% while for CBV(2) is 76.67% and for NOV (32) is 60%.

8 Conclusion

This work presented the experimental results by using MLP models for classifying EEG signals of normal individuals and individuals under brain-death protocol.

The experimental results using real input data from a hospital have shown the advantages of adopting orthogonal bipolar vectors as targets for MLP learning improvement.

We also have concluded that the use of orthogonal bipolar vectors provides a better separation of pattern features due to larger Euclidean distance between these vectors.

So, the results can confirm the hypothesis of our work suggesting orthogonal bipolar vectors as new expectation values for an improved MLP learning in EEG signal classification.

9 References

[1] N. C. L. Abreu, Extraction of speech signals in noisy environments by decomposition os statistically indepent basis functions (in Portuguese), Dissertation – **Federal University Maranhão** (2003).

[2] A. M. Santos, J. M. Seixas, B. B. Pereira, R. A. Medronho, Using Artificial Neural Networks and Logistic Regression in the Prediction of Hepatitis A (in Portuguese), **Brazilian Journal of Epidemiology** (2005); 8(2): 117-26.

[3] X. Wang, C. Chang, F. Du, Achieving a More Robust Neural Network Model for Control of a MR Damper by Signal Sensitivity Analysis, **Neural Computing & Applications**, 13(2002), 330-338.

[4] M. A. Costa, A. P. Braga, B. R. Menezes, Improving Neural Networks Generalization With New Constructive and Pruning Methods**, Journal Of Intelligent & Fuzzy Systems**, 153(2006), 805-809.

[5] C. M. Lee, S. S. Yang, C. L. Ho, Modified back-propagation algorithm applied to decision-feedback equalization, **Iee Proceedings – Vision, Image & Signal Processing**, 153(2006), 805-809.

[6] T. G. Dietterich, G. Bakiri. Solving Multiclass Learning Problems via Error-Correcting Output Codes, **Journal of Artificial Intelligence Research**, Vol. 2, pp.263-286 (1995).

[7] M. A. Montenegro. **Clinical Pratice of EEG**. Lemos Editors, Brazil, 303 p., 2001. (In Portuguese).

[8] E. Niedermeyer, F.L. Silva. **Electroencephalography – Basic principles, clinical applications and related Field**. 5ª ed. Lippincott Williams & Wilkins, USA, 1277 p., 2005.

[9] D. P. Subha, P. Joseph, R. U. Acharya, et al. **EEG signal analysis: a survey**. J. Med. Syst, Vol. 34, pp. 195 – 212 (2010).

[10] T. Kohonen, Associative Memory: a System-Theoretical Aproach, **Springer-Verlag** (1977).

[11] K. S. Fu, Syntatic Methods in Pattern Recognition, Academic Press (1974).

[12] L. N. Cooper, A Possible Organization of Animal Memory and Learning, Proc. of the Nobel Symposium on Collective Propertiers of Physical Systems, B. Lundquist and S. Lundquist (1973).

[13] R. O. Duda, P. E. Hart, Pattern Classification and Scene Analysis, **Wiley** (1973).

[14] S. Nomura, J. R. G. Manzan, K. Yamanaka. An Experimentation with improved target vectors for MLP in classifying degraded patterns, **Journal of the Brazilian Neural Network Society, Vol. 8, Iss. 4, pp.240 – 252** (2010).

[15] J. S. Ebersole, T. A. Pedley. **Current practice of clinical electroencephalography.** 3ª ed. Lippincott Williams & Wilkins, USA, 974 p. (2003).

[16] L. Fausset, Fundamentals os Neural Networks: Architecture, Algorithms, and Applications, **Prentice-Hall**, (2003).

APPLICATION OF ARTIFICIAL NEURAL NETWORK MODEL IN FORECASTING ZARIA RAINFALL

FOLORRUNSHO J. O. (Ph.D)

GEOGRAPHY DEPARTMENT,

KADUNA STATE UNIVERSITY, KADUNA, NIGERIA

Abstract- *Several models have been developed in hydrological forecasting yet, the Artificial Neural Network (ANN) model provides a quick and flexible means of creating reasonable output. It has also been shown a higher performance level when compared with conventional methods. This study present a method of rainfall prediction by developing an ANN- based model using major weather variables such as temperature, relative humidity, wind speed and sunshine hours as inputs while the rainfall as the target output. The data used for the study were obtained from the Nigerian Meteorological Agency, Zaria and Meteorological Unit, Soil Science Department, Ahmadu Bello University, Zaria on a monthly average basis over a twelve (12) years period (1990 – 2002). As part of the ANN model development procedures, the data sets of 84 raining months in the study area (April – October) was partitioned into two parts with 70% of the entire data sets used as the training data while the remaining used as the testing and the validation data. The results of this study revealed that the developed ANN – based model in the Predict Demo NeuralWare environment shows a correlation value of 81% when the original rainfall values of the study area were compared with the forecasted rainfall values.*

Keywords: ANN, forecasting, tropics

1 Introduction

Rainfall stands out as perhaps the single, most unique element of all the climatic elements such that it's total amount, intensity, duration, variability, reliability and its spatial and temporal distribution influence phenomenon especially in the tropical region where prevailing economic activity is simply agro-based (Oladipo, 1987, Hyuwa, 2005). Past geographical survey has established that most theoretical analysis of the rainfall characteristics in Nigeria has been based on a very delicate assumption. This is done with the view of rainfall normally in its distributive pattern, especially for an annual series. However, without care, similar assumption could be made for either the monthly and/or weekly rainfall analysis (Folorunsho, 1997).

On the basis of these rainfall attributes, a thorough statistical analysis of rainfall distribution over the study area will not only be imperative but useful to the agricultural, social, commercial and industrial sectors of the economy of the study area but at the same time be a stepping stone to sustainable development of the entire country. Thus, it is imperative therefore to find out if this condition is practically obtainable in Samaru considering the fact that agricultural practice in Zaria is also rain fed. Also, not just for no other thing to study, but because of the importance of such knowledge for all planning schemes for which rainfall is widely used. Further to this is the fact that such assumption about rainfall condition in the area may have serious and delicate implications on agricultural production.

In agricultural production especially in the tropics, rainfall is without doubt a critical climatic factor. It is known fact that one of the two major limiting factors to agricultural production next to soil fertility is nothing but insignificant water supply (Oladipo, 1993). Rainfall is the main source of soil moisture in any given environment. Thus an assessment of its distribution be it monthly, weekly, and especially daily distribution) is therefore of great importance in agricultural planning.

In Nigeria, the dominant feature of rainfall is its seasonal character. The large energy content of rainfall system is its variability from year to year which is mainly attributed to the fluctuation in the movement of the two different dominant air masses, the Inter Tropical Discontinuity (Ayoade, 1973; Ati, 1996). Despite the recent advances made in science and technology, farmers and their crops are still left at the mercy of rainfall especially in Sub-Saharan Africa, where the Northern region of Nigeria lies. Hence water supply for agricultural practices is highly dependent on precipitation. Moreover in areas where the climate is greatly influenced by drought and desertification, the condition of precipitation in relation to yield, the rate of evapo-transpiration and soil moisture content may help promote or hinder crop production. This is subject to availability of moisture at the evaporating surface and the ability of the atmosphere

to vaporize the water. When plants transpire at the maximum rate (i.e. when soil is completely saturated) the term potential evapo-transpiration is applied (Jackson, 1977; Oguntoyinbo, 1983). This condition has great implication on crop production.

In 1983, it was reported in the media that the Niger State government lost about 5 million naira worth in agricultural sector alone due to lost of crop input for that year. This was as a result of the inability of an effective planning in that; there are no proper and adequate knowledge of rain distribution in the area (NEST, 1991). Against this backdrop, this study seeks to develop and apply the artificial neural network (ANN) model in forecasting the rainfall of Samaru, Zaria within the period 1992 – 2002.

2 The Study Area

ANN has been applied in non-linear system modeling with enormous applicability in time series analyses of recent (Dunis and Williams, 2002). Due to possess some important and attractive characteristics, ANN can be chosen for rainfall forecasting. Firstly, ANN is a method which can be easily adapted to the types of data with few limiting hypothesis against classical or conventional methods. Secondly, for ANN, there exists a general functional structure that can be generalized (Hornik, Stinchcombe and White, 1989). ANN is a network of interconnected elements which are inspired from the studies of biological nervous systems. Producing an output pattern using an input pattern is the main function of an ANN. Connectionist models, such as ANN, are well suited for machine learning where connective weights are adjusted to improve the performance of network (Zhang and Hu, 1998; Hu *et al*, 1999; Franses and Griensven, 1998).

According to Hsu, *et.al*, (1995), ANN approaches to rainfall-runoff modeling are more efficient than the conventional flow forecasting models whenever explicit knowledge of the hydrological balance is not required and when the system may be treated as a black-box. Be it as it may, recent experiments have reported that ANN may offer a promising alternative for rainfall-runoff modeling (Smith and Eli, 1995; Tokar and Johnson, 1999), streamflow prediction (Sivakumar, *et. al*, 2002; Kisi, 2004, 2005; Antar, *et. al*, 2006), and reservoir inflow forecasting (Jain, *et. al*, 1999) to mention just a few.

The aim of this study is to apply the artificial neural network model in forecasting the rainfall of Samaru, Zaria, Kaduna State within the period 2000 – 2012 and this will be achieved by the following objectives which are to: develop a reliable forecasting model(s) to predict Samaru rainfall; examine the reliability of the model in predicting the rainfall of the study area and, validate the model.

The study area is Samaru in Zaria, Kaduna State located within latitude $11^0 09'$ – $11^0 10'$ North and longitude $7^0 38'$- $7^0 39'$ East. It is a small area that was formerly called "Gan'gu Huku" in Hausa land, meaning where stream flows having trees at the center. Samaru is one of the suburbs of Zaria which is made up of distinct loosely coordinated units. Samaru is about 765 meters above sea level. Samaru area is situated on the extension of urban Zaria to the northwest area along the Zaria - Sokoto road and it is located at about 83km north of Kaduna town along the Kaduna - Kano highway. Samaru is about 13 Kilometer from Zaria city along Sokoto road, 8 kilometer to Shika and 7 kilometer from Basawa (Mortimore, 1975). The study area is one of the important towns in Zaria particularly being the domain of the famous Ahmadu Bello University, Zaria and has become the nodal point in terms of road transport (Fig. 1.).

The climate of the study area exhibits a tropical continental type which also falls under A_W of the Koppens climatic classification scheme, with distinct wet and dry seasons (Hore, 1990). The rainfall is of the frontal and convectional types. The study area has only one rainfall season, maximum peak in August and annual rainfall at about 1200-1500mm (Adejuwon, 1988). Similarly, it was reported by Kowal (1972), that the late afternoon rains are common because of the convectional currents which are the major signs of air lift.

Samaru lies within the northern guinea savanna ecological zone with few tree species and grasses. A designation which implies a woodland vegetation type inter-spaced by grasses and vegetation is largely deciduous. The dominant tree species include *Isobalina Doka, Isobalina Tomentossa, Sativa spp, Vitex spp* and *Uapaca Togoensis* with a well developed grass layer of tuff led *Andropogon spp* (Mortimore, 1975). The trees are between 10 and 15 metres high with broad leaves and crown almost touching each other and they are deciduous in nature (Jackson, 1970). The drainage system of Samaru focuses on the Galma River which is a major tributary of River Kaduna and is characterized by high stream frequencies and drainage density.

The Galma River is fed by major tributaries like the Kubanni and Saye rivers with the formal draining the study area. Due to the nature of the relief, many of the streams shows evidence of high sediment load, meandering and braiding due to deposition in the channel and the valley floors. The Galma River is the only river with a perennial flow while others (Saye, Shika, Kubanni and Giwa rivers) are seasonal. Kubanni and Shika rivers respectively form a confluence at Galma in Zaria, which is a tributary of River Kaduna (Kaatyl, 1999). Thus the drainage patterns are generally dendritic (Folorunsho, 2004).

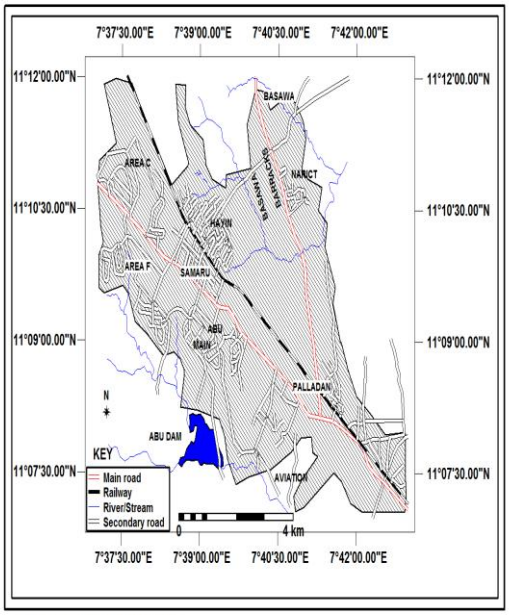


FIG. 1: SAMARU; STUDY AREA

The study area is developed on the crystalline metamorphic rock of the Nigeria basement complex. It is composed mainly of the fine grain gneisses and migmatite, with some coarse grained granitic outcrops in few places. The gneisses are moderately to weekly foliated, principally made up of the quartz and oligoclase. Weathering depth is irregular, but thorough depth range from 10m to deep pockets, occasionally extending to 60m (Folorunsho, 2004).

3 Materials and Method

3.1 Types and Sources of Data

The primary data used for this study include the monthly averages of the temperature, relative humidity, wind speed and sunshine hours of the study area which was used as the input data. These were obtained from the Nigerian College of Aviation Technology (N.C.A.T.) Zaria, while the monthly average rainfall data series for Samaru spanning 1990 to 2002 was obtained from the Meteorological Department, Institute of Agricultural Research, (I.A.R), Ahmadu Bello University, Samaru, Zaria. The secondary data were obtained from journals, books, internet materials and order research work directly or indirectly related to the study and the study area.

3.2 Data Analysis

The twelve (12) years mean monthly rainfall data used for this study considered only the raining months in the study area (April to October). This is one of the requirements of the ANN- based model development. For this study, the ANN – based model development using the NeuralWare Predict Demo interface was used to achieve the desired operations. Employing the six (6) basic procedures: launching of the model and loading the data sets partitioned into 70% of the data sets for training and 30% for the testing and validation; configuration of the model; training of the model using four (4) sets of variables for input; model performance testing; model validation for reliability for forecasting the rainfall and, graphical plots of the actual and forecasted rainfall output in order to certify the ability of the model to learn the trend, cycles and pattern of the rainfall of the study area.

4 Results and Discussion

The monthly mean data sets of all the input variables were loaded for the ANN model development which was achieved using the outlined procedures in the NeuralWare PredictDemo interface. These processes were used in order to achieve the training of the model with the results displayed in Figure 2.

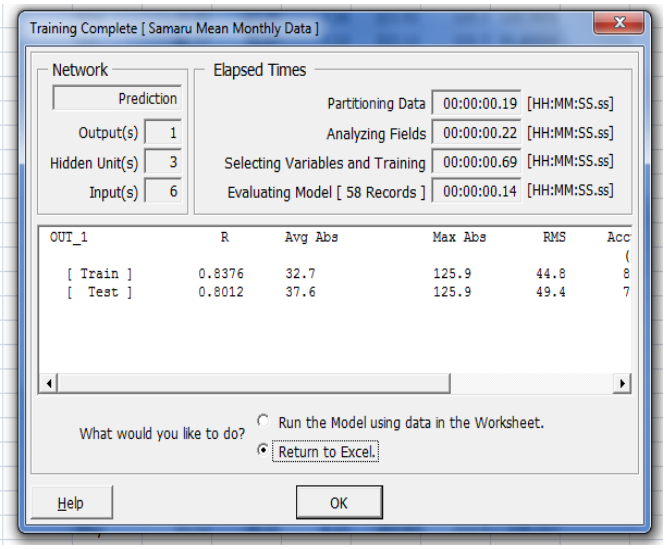


Figure 2: Training Complete Dialog Box

Having completed the training processes as shown in figure 2 above, the most obvious test of reliability of a good ANN- based model is that the training data set performance and the test dataset are fairly similar. The key indicator of such model is the performance of the model when subjected to the standard procedures and statistical tests outlined earlier. For this study, these values were actually found to be similar with the correlation value (R) between the actual and forecasted mean monthly data shown as 81%, root mean square error (RMS) is 52.5, accuracy of 96.4%, and the confidence interval of 107.7 at 0.05 significant level (see figure 3).

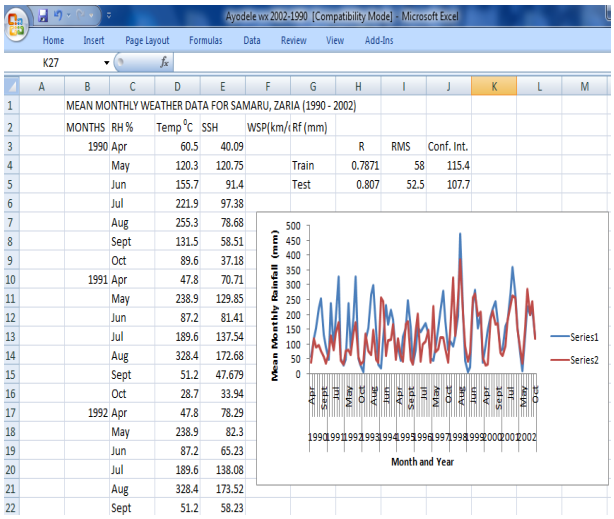


Figure 3: Test Command Dialog Box for the Validation of the Data

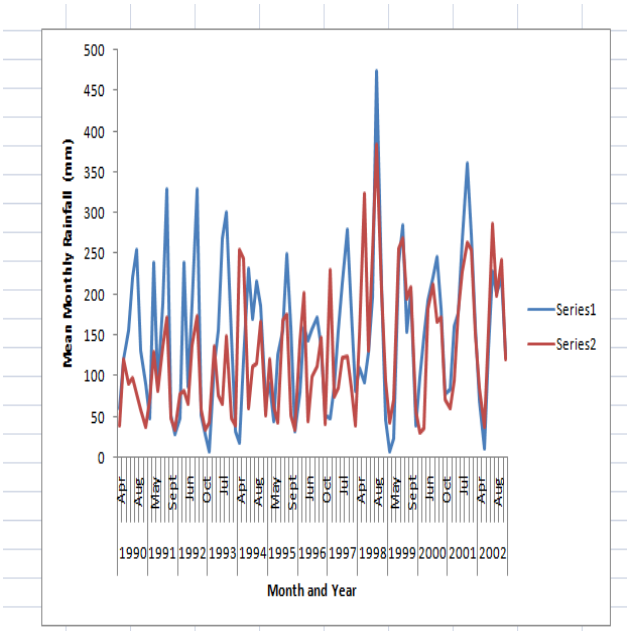


Figure 4: Actual and Forecasted Mean Monthly Rainfall Values

From the obtained test results generated by the ANN – based model shown in figure 3 above, these are indications that the model has learned both the trend and cycle of the actual data. This further confirms the reliability of the model for sustainable water resources development and management in the study area. However, in order to validate the ANN – based model developed, the performance of the developed model to forecast mean monthly rainfall from the actual mean monthly rainfall data was tested. From the test result, a comparison of the actual mean monthly rainfall output and the ANN – based model output was tested and the result shown in figure 4 below.

From figure 4, it is obviously clear that the developed model has a high performance level (with R-value of 81%) having learned both the trend and cycle of the Samaru mean monthly rainfall. This present a good effect on the reliability of the ANN model developed for forecasting rainfall series that can improve formulation of policies that can promote sustainable development of all the rain-fed activities in the study area.

5 Conclusion

The water resources planning, development and management is of paramount importance for the sustainable development of any region. Therefore, developing a model to effectively forecast the rainfall is quite apt considering the fact that rain-fed agricultural practice is dominant in the study area. In this research, an ANN – based model was developed to forecast the rainfall of Samaru depending on the available weather variables (temperature, relative humidity, wind speed, sunshine hours and rainfall). The ANN – based model was developed in the NeuralWare Predict Demo environment. As a result of this, considering the results obtained from this research, it can be concluded that the ANN is a reliable modeling tool for forecasting rainfall of the study area (and any other region) given the strong, high and positive Correlation Coefficient value of 81% displayed. From the foregoing, it will be apt to conclude this research that the ANN – based model is a veritable tool for overcoming the rainfall record paucity, inconsistency, unreliability hampering sustainable water resources development study area.

References

[1] Adejuwon, S.A ''An Assessment of the Patterns of Rainfall fluctuations between 1922 and 1985 in Nigeria'', Ph.D thesis, Department of Geography, Obafemi Awolowo University, Ile-Ife, 1988.

[2] Antar, M.A., Elassiouti, I. and Allam, M.N. ''Rainfall-Runoff Modelling Using Artificial Neural Network Techniques: A Blue Nile Catchment Case Study'', Hydrological Processess, 20, pp. 1201-1216, 2006.

[3] Ati, O.F. ''A Comparison of Methods to Detect the Onset of Growing Season and its Trends for some Stations in the Sudan Savanna in Northern Nigeria'', M.Sc, Thesis, Department of Geography, Ahmadu Bello University, Zaria, 1996.

[4] Ayoade, J.O. ''Annual Rainfall Trends and Periodicities in Nigeria'', Nigerian Geographical Journal Vol. 16, pp. 167-172, 1973.

[5] Dunis, C. and Williams, M. ''Modelling and Trading the EUR/USD Exchange Rate: Do Neural Network Models Perform Better?'' Use Trad. Regul., 8(3), 211 – 239, 2002.

[6] Folorunsho, J.O. ''Normality of Monthly and Weekly Rainfall Series in Kano'', Kano State, B.Sc, Dissertation, Geography Dept., A.B.U., Zaria, 1997.

[7] Folorunsho, J.O. ''An Examination of Some Streamflow Characteristics of River Kaduna State'', Unpublished M.Sc. Thesis, Geography Dept., A.B.U., Zaria, 2004.

[8] Franses, P. H.Griensven, K. H. ''Forecasting Exchange Rates Using Neural Networks for Technical Trading Rules'', Stud. Nonlinear Dynamics, Econometrics, 2(4): 109-114, 1998.

[9] Hore, P.N. ''Weather and Climate'', In: M.J. Mortimore (ed) Zaria and Its Regions, Occassional Paper No. 4, Dept. of Geog. A.B.U. Zaria, pp. 41-54, 1970.

[10] Hornik, K. Stinchcombe, M. and White, H.'' Multilayer Feedforward Networks are Universal Approximators, Neural Network, 2: 359-366, 1989.

[11] Hu, M, Y. Zhang, G. Jiang, and Patuwo, B. E. '' A Cross-Validation Analysis of Neural Network Out-of-Sample Performance in Exchange Rate Forecasting, Decis. Sci., 30(1), 197-216, 1999.

[12] Hsu, K., Gupta, H.V. and Sorooshian, S. ''Artificial Neural Network Modelling of the Rainfall-Runoff Process'', Journal of Water Resources, 31, pp. 2517-2530, 1995.

[13] Hyuwa, G.N. ''Statistical Analysis of Daily Rainfall Characteristics at Jos (1930-2003)'', B.Sc. Dissertation, Department of Geography, Ahmadu Bello University, Zaria, 2005.

[14] Jackson, G. ''Vegetation around the City and the nearby Villages in Zaria'', In: Mortimore (ed), Zaria and its Region, Op. Cit. pp. 61-71, 1970.

[15] Kisi, O. ''River Flow Modelling using Artificial Neural Networks'', ASCE Journal of Hydrologic Engineering, Vol. 9**,** No.1, pp 60-63, 2004.

[16] Kisi, O. ''Daily River flow Forecasting using Artificial Neural Networks'', Turkey Journal of Engineering, and Environ. Sci., Vol., 29, pp. 9-20, 2005.

[17] Kowal, J.M. and Adeoye, K.B. ''An assessment of Aridity and the Severity of the 1973 Drought in Northern Nigeria and Neighbouring Countries'', Savanna, 2(2**)**: 145-158, 1973.

[18] Mortimore, M. J. ''Zaria and its Region'', Occasional Paper No. 4, Department of Geography, Ahmadu Bello University, Zaria, Published for the 14th Annual Conference of the Nigerian Geographical Association, Zaria, pp. 15-129, 1975.

[19] Nigerian Threatened Environment: National Profile, N.E.S.T., Intec Printers Ltd, Ibadan, 1991.

[20] Oguntoyinbo, J.S, ''Climate'', In: Oguntoyinbo, J.S., Areola, O.O., and Filani, M. (eds), Geography of Nigerian Development, Heinemann Publisher, Ibadan, pp. 57-60, 1983.

[21] Oladipo, E.O. ''Some Features of Growing Season Precipitation Fluctuations in the Interior Plains of North America'', Journal of Climatology, Vol.7, pp.531-540, 1987.

[22] Smith, J and Eli, R.N. ''Neural Network Models of Rainfall-Runoff Process'', Journal of Water Resources Planning and Management, pp. 499-508, 1995.

[23] Sivakumar, B., Jayawardena, A.W., and Fernando, T.M.K.G., ''River flow Forecasting: Use of Phase Space Reconstruction and Artificial Neural Networks Approaches'', Journal of Hydrology, 265, pp. 225-245, 2002.

[24] Tokar, A.S.and Johnson, P.A. ''Rainfall-Runoff Modelling Using Artificial Neural Networks'', Jour. Hydrol. Eng. ASCE, pp. 232-239, 1999.

[25] Zhang, Q. and Stanley, S.J. ‘’Foercasting Rain-Water Quality Parameters for the North Saskatchewan River by Neural Network Modelling’’, Journal of Water Resources, 31, pp. 2340 – 2350, 1997.

Use of Artificial Neural Networks for Completing Data Series Streamflow Hydrologic

Sérgio Luis Yoneda; Rogério Andrade Flauzino; Lucas Assis de Moraes; Marcel Ayres de Araújo
Department of Electrical and Computing Engineering, University of São Paulo – USP, São Carlos, Brazil

Abstract—*Knowing the hydrological behavior of hydroelectric reservoirs affluent rivers is one of the main tools for managing the production of electricity. To identify the streamflow values of a river is extremely important for planning the operation and expansion of hydroelectric systems. However, in many cases there are insufficient or incomplete data for the specific location of a particular river. Concerning this situation, the research exposed in this paper applies an Artificial Neural Network to complete the missing data of a hydrological streamflow time series. Through comparison between the results obtained by correlational techniques and the proposed methodology, it is concluded that the later can complete the series in a more realistic way and with great precision.*

Keywords—Artificial neural networks. Fluviometric stations. Hydropower. Streamflow time series.

1 INTRODUCTION

The Brazilian energy matrix is composed predominantly by hydropower, which makes managing the amount of water in the reservoirs of dams an important aspect for planning the generation of electricity. However, one of the confronted problems is the natural variability of the hydrologic regime due to geography, climate change and rain precipitation present in the vast Brazilian territory. In this context, knowing the streamflow of the affluent rivers that feed these dams is vital for an economic and reliable operation [1]-[3].

Given the large Brazilian hydroelectric potential, of which only about 30 % has been used, and considering the competitiveness of the energy generated by water sources compared with other sources, the Brazilian government created several tax incentives for investments in hydroelectric centrals. As a result, there was a large increase in the number of these hydroelectric central projects, overloading the review and approval responsible agents. Thus, studies that aim to ease in terms of time and accuracy the estimation of the possible electrical potential of rivers and affluents have been developed [3]-[6].

In hydroelectricity works, seasonal aspects, characteristics of streamflows and particularities related to the uses of the water determine the effective potential for hydroelectric power systems. Through multiple daily measurements of water level, data are obtained from fluviometric stations in order to determine a function called Key-Curve, which models the phenomenon and generates the series of flows. It is required a lot of information with numerous records for the same river, but in the operations of a station, interruptions may occur while gathering data, which generates failures in observing and filling the series of flows. Such situations make streamflow forecasting a stochastic nonlinear problem, which implies considerable uncertainty and difficulty of obtaining and building prediction models [5], [6].

At this juncture, there is extensive literature related with time series forecasting in different areas of knowledge, for different applications. These series require temporal dependency relationships among their data to identify the generating point of the series with the purpose of extracting their periodicity, describing their behavior and to predicting them. Furthermore, most of the techniques used in these models are based on the Box-Jenkins method, which considers linear relationships between the system variables [7].

Nevertheless, in [8] time series are used to forecast streamflow and manage water resources. The hydrological model adjusted for streamflow forecasting was used in Iraí, in the Uruguay River, retrospectively for the period of 1995-2001. The results were compared to the observed discharge and the predictions obtained using the traditional statistical method, and it was observed that the global climate model underestimates rainfall throughout most of the basin, especially in winter, but the forecasts represent relatively well the interannual variability of rainfall in the region.

In [9], time series models are used in order to observe trends of annual precipitation and streamflow in the São Mateus River Basin, in the period of 1966-2008. For that, data were collected from three fluviometric stations, and evaluations of rainfall and streamflow data were considered for the semester, for the trimester and for the wettest month of the year. Through this research, it can be noted that there is a wide variation in precipitation: winter has low rainfall index while summer has a high index. With respect to flow, it is noted that only the first decade showed a lower flow compared to the rest of the period, and after 1970 the flow was stable.

However, the time series with actual sequence of values are not linear and generally have noises that distort their

The University of São Paulo, under FAPESP and CAPES Research Grants, supported this paper.

features and can compromise their predictions. To avoid these problems, many studies have been developed by applying Artificial Neural Networks (ANN) and Fuzzy Inference Systems (FIS), whose operation is described in [10]-[12], in time series forecasts to predict future values based on past values. This is due to their generalization, association and learning abilities, supervised training and parallel computing [10].

These qualities make them capable of performing more complex relationships between the terms of a time series, representing better the reality of the problem, thereby achieving lower estimation errors. Also, these intelligent systems can identify and assimilate the characteristics of the series, such as periodicity and seasonality, often camouflaged by noise, without requiring a complex theoretical formulation, as in statistical procedures [10].

The use of ANN for forecasting climatological streamflow time series has been performed by several researchers in many studies, among which [13]-[15] show the feasibility of using these networks and their success in this application.

However, to perform a reliable forecast of a time series, it is recommended, and sometimes necessary and mandatory, that it is complete, that means, there must not be any corrupted or missing data [16]-[18]. Thus, the research presented in this paper seeks to complete a time series missing data, in order to aid the agents interested in forecasting time series.

The hydrological streamflow time series studied belongs to the Formoso River, in the Paraná River basin. Data from a fluviometric station near the region will be used, but this station has not kept the flows for every month throughout the years. So, an ANN will be applied to recover the missing data.

This paper is organized into five sections developed after the introduction. In section II, the main aspects related to the scenario studied are summarized. In section III, the methodology used in this research is presented, while section IV exposes the computational results obtained. In sequence, the results of computer simulations are widely analyzed in section V, and finally, section VI reports the conclusions of this research.

2 STUDIED SCENARIO

Knowing that the full knowledge of a historical series of streamflows is of great importance for the planning and operation of hydroelectric power plants, and also knowing that it is common to find incomplete streamflow series for certain rivers and regions, it is formed a problem which investigation is needed.

This work exemplifies this problem by presenting the case of Formoso River, an affluent by the right bank of the river Sepotuba located in the sub-basin 66, belonging to the Paraná river basin, in Mato Grosso, as illustrated by Fig. 1 [19]. The Formoso river, until now, did not have officially any monitoring of its water supply. Therefore, it was sought the availability of fluviometric stations near the region, which are listed in Table 1.

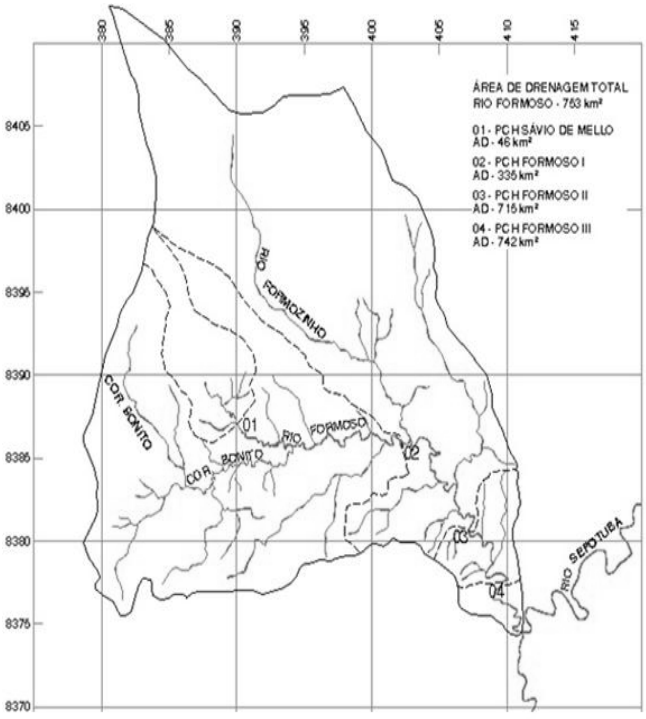


Figure 1. Formoso river basin [19].

TABLE I. FLUVIOMETRIC STATIONS NEAR FORMOSO RIVER REGION.

Nº	Station	Code	Source	River	LTAF (m³/s)	Drainin g (km²)	Period Start	Period End
A	Cachoeira	66040000	ANA	Sepotuba	115	4	07/1979	01/1986
A	Tapirapuã	66050000	ANA	Sepotuba	167	5	08/1971	12/2002
A	S. J. do Sepotuba	66055000	ANA	Sepotuba	227	9	10/1969	10/2004
A	Fazenda Salu	66071300	ANA	Jauru	72.3	2	02/1984	12/1994
A	UHE Jauru	66055000	ENGECON	Jauru	88.8	2	01/1931	12/1999
A	Água Suja	66071400	ANA	Jauru	96.5	3	07/1979	07/2004
A	Nortelândia	66006000	ANA	Santana	38.2	2	12/1967	09/2004
B	Porto Esperidião	66072000	ANA	Jauru	102	6	12/1965	01/2003
B	MT 125	66065000	ANA	Cabaçai	74.6	5	09/1969	10/2003
B	Barra do Bugres	66010000	ANA	Paraguai	168	10	01/1966	08/2004
B	Porta Estrela	66015000	ANA	Paraguai	192	12	06/1971	11/2003
B	Jauquara	66008000	ANA	Jauquara	22.8	1	12/1967	08/2005

Thus, the station Tapirapuã was elected to have its nonexistent data filled by the method developed in this work. The streamflow time series from such station is exposed in Table 2, and the unknown data are highlighted in gray.

3 METHODOLOGY

To fill in the unknown data of Table 2, a technique using an ANN was developed. The ANN used presents a dynamic behavior, and is able to memorize relationships and to store information. Thus, these characteristics of this network qualify it to be used in time-varying systems. The ANN was trained a monotone crescent activation function, so that its dynamic behavior is stable. [10].

TABLE II. INCOMPLETE STREAMFLOW SERIES FOR THE TAPIRAPUÃ STATION.

Year	Jan	Feb	Mar	Apr	May	June	July	Aug	Sept	Oct	Nov	Dec
1971								76	76.3	89.9	101	104
1972	134	239	181	183	129	102	96.2	88.6	88.3	91.3	106	150
1973	154	161	162	124	113	93.7	83.1	76.3	72.5	73.1	98.7	123
1974	216	218	244	221	159	125	104	89.9	81.6	86.4	88	123
1975	160	172	196	188	140	110	100	91.7	83.1	87.7	123	167
1976	222	329	250	217	172	139	116	98.5	96.1	88.6	118	162
1977	214	234	191	177	170	141	115	95.8	102	111	121	167
1978	235	205	202	203	185	162	121	105	102	114	125	189
1979	344	321	304	251	179	152	132	114	127	111	119	160
1980	226	301	320	244	191	147	126	115	114	109	116	148
1981	258	250	249	201	163	134	114	106	95	102	125	135
1982	208	291	386	248	175	157	129	123	142	159	180	259
1983	270	287	282	265	202	170	137	122	112	118	187	259
1984	253	213	248	259	190	147	125	119	115	117	146	235
1985	290	238	397	310	204	158	142	123	119	133	132	122
1986	215	275	259	224	193	146	124	123	119	132	123	158
1987	221	216	255	180	150	136	111	98.5	94.2	105	105	208
1988			373	298	212	169	141	120	109	112	121	197
1989	237	299	298	247	204	152	138	132	117	112	119	126
1990	201	226	212				116	103	115	140	129	144
1991	202	276										155
1992												
1993	267	290	285	235	194	164	139					
1994	219	258	258	237	172	150	133	122	112	118	112	147
1995	244	374	256	253	176	147	129	118	108	112	128	152
1996	210	225	312	227	174	145	129	118	112	116	198	217
1997	237	313	303	231	179	160	134	122	121	123	127	
1998						127	111	110	106	106	138	209
1999	202	218	329	262	159	137	122	107	105	102	115	131
2000	148	199	280	174	149	127	114	103	98.4	99.4	129	197
2001	213	201	224	188	144	133	110	94.8	102	115	127	196
2002				196	155	127	113	105	105	103	113	122

However, before performing the ANN training, an analysis of the number of known and unknown neighbors for each cell of Table 2 was made. Each cell has up to eight neighbors, as shown by Fig. 2, which shows the distribution of the amount of unknown neighbors.

Once the topology of the network is defined, a study was made of the possibility of recovering a missing data in Table 2 based on the number of known neighbors. Fig. 3 shows it is necessary that a cell have at least three known neighbors to make it possible to retrieve its data.

Thus, it is clear that the ANN must have at least three inputs, which are the known neighbors of an unknown cell, for the information of such cell is recovered.

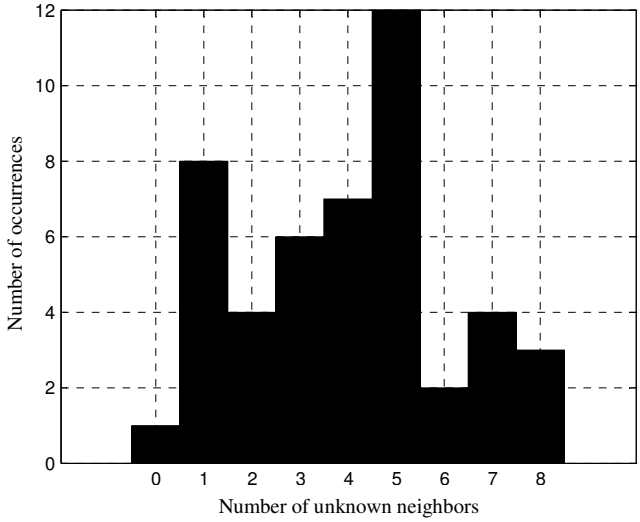


Figure 2. Number of unknown neighbors for the cells on Table 2.

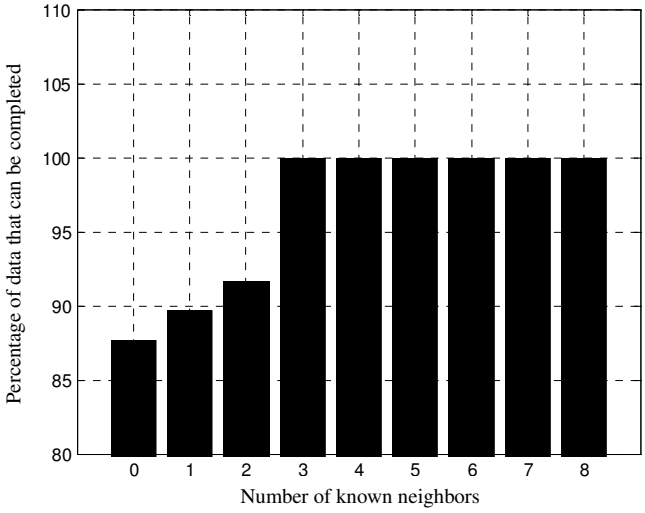


Figure 3. Percentage of data able to be completed due to the unknown number of neighbors considered by the ANN.

4 COMPUTATIONAL RESULTS

Knowing the minimum number of required inputs (three), it was possible to implement the method to solve the problem of the case studied. Hence, the performed simulations with the ANN using the data presented above were able to fill the Tapirapuã station streamflow series, as shown in Table 3.

TABLE III. TAPIRAPUÃ STATION STREAMFLOW SERIES (FILLED BY THE PROPOSED METHOD).

Year	Jan	Feb	Mar	Apr	May	June	July	Aug	Sept	Oct	Nov	Dec
1971	0	0	0	0	0	0	0	76	76	90	101	104
1972	134	239	181	183	129	102	96	89	88	91	106	150
1973	154	161	162	124	113	94	83	76	73	73	99	123
1974	216	218	244	221	159	125	104	90	82	86	88	123
1975	160	172	196	188	140	110	100	92	83	88	123	111
1976	222	329	250	217	172	139	116	99	96	89	118	162
1977	120	234	191	177	170	141	115	96	102	111	121	167
1978	235	205	202	203	185	162	121	105	102	114	125	189
1979	344	321	304	251	179	152	132	114	127	111	119	160
1980	226	301	320	244	191	147	126	115	114	109	116	148
1981	258	250	249	201	163	134	114	106	95	102	125	135
1982	208	422	386	248	175	157	129	123	142	159	180	259
1983	270	197	282	265	202	170	137	122	112	118	187	259
1984	253	213	248	259	190	147	125	119	115	117	146	235
1985	290	238	397	310	204	158	142	123	119	133	132	122
1986	215	275	156	224	193	146	124	123	119	132	123	158
1987	221	216	255	180	150	136	111	99	94	130	129	129
1988	131	321	373	298	212	169	141	120	109	112	121	197
1989	237	299	298	247	204	152	138	132	117	112	119	126
1990	201	226	212	136	199	201	116	103	115	140	129	144
1991	202	276	119	102	120	140	171	141	126	125	143	155
1992	133	188	148	103	104	85	128	71	43	152	129	162
1993	267	290	285	235	194	164	139	89	86	132	146	152
1994	219	258	258	237	172	150	133	122	112	118	112	147
1995	244	374	256	253	176	147	129	118	108	112	128	152
1996	210	225	312	227	174	145	129	118	112	116	198	217
1997	237	163	303	231	179	160	134	122	121	123	127	129
1998	150	92	175	134	150	127	111	110	106	106	138	209
1999	202	218	196	262	159	137	122	107	105	102	115	131
2000	148	199	280	174	149	127	114	103	98	99	129	197
2001	213	201	224	188	144	133	110	95	102	115	127	196
2002	0	0	0	196	155	127	113	105	105	103	113	122

5 RESULTS ANALYSIS

There are three main points regarding the results that should be highlighted. The first is the iterative process. As it can be seen in Table 2 and in Fig. 2, there are some data with less than three known neighbors, and these data cannot be completed using only three inputs in the ANN. In these cases, it is necessary to retrieve some of the data first, which reduces the number of unknown neighbors between cells, and thus it becomes possible to recover iteratively all the data of the series. Fig. 4 details how many iterations it takes for the process to converge depending on the number of known neighbors of a cell. Note that for zero, one, or two known neighbors, the method converges, but it completes the series with untruthful data.

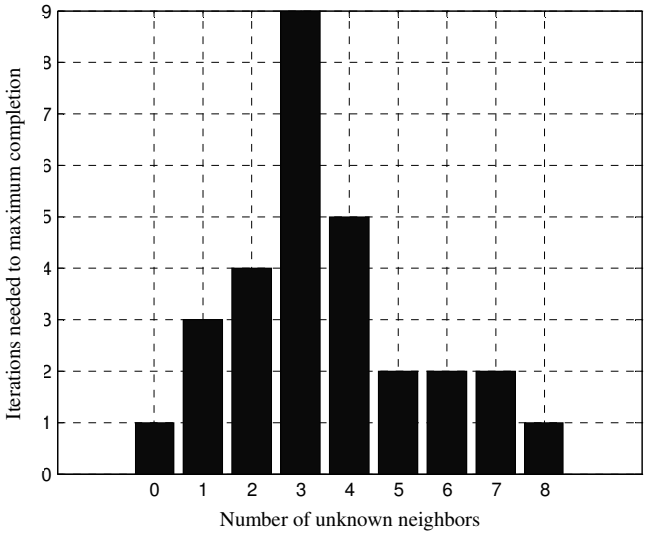


Figure 4. Iterations needed to maximum completion of the streamflow unknown data.

Another point to be discussed is that some cells in Table 3 were completed with a null value. This happened because there are no known data in the previous year or the year after these cells, what means that they do not have a total of eight neighbors, but three or five, and this undermines the process of data recovery.

Finally, the third point that needs to be highlighted is the quality of data recovery. It is important to analyze the performance of the proposed methodology in simulations concerning hydroelectric power systems to verify the generation of electricity provided by the series.

The studied streamflow series had already been completed and extended before, through correlations with data from Cachoeira, São José de Sepotuba, Porto Estrela and Barra do Bugres stations. This series, presented in Table 4, serves as a reference for this work.

There are three main differences between the complete streamflow series shown. The first is that there is no null data in Table 4. On the other hand, the second difference is that the completed data in Table 3 are more varied than those in Table 4, which gives a highest likelihood with reality. Finally, the third difference is the average streamflow, which is the most important factor for energy calculations. The series completed with the proposed method provides an average of 160.53 m^3/s, while the series completed through correlations with other stations has an average of 164 m^3/s. This shows that there is an error of only 2.16% between the methods.

TABLE IV. TAPIRAPUÃ STATION STREAMFLOW SERIES (FILLED BY THE CORRELATIONS METHOD).

Year	Jan	Feb	Mar	Apr	May	June	July	Aug	Sept	Oct	Nov	Dec
1969	213,3	249,2	259,7	217,4	166,8	137,8	118,6	106,5	104,0	76,8	85,8	97,4
1970	102,2	171,5	172,9	129,0	117,3	91,9	84,4	74,8	74,1	78,9	76,2	78,9
1971	108,4	123,5	138,6	126,9	109,8	92,6	83,0	76,0	76,3	89,9	101,0	104,0
1972	134,0	239,0	181,0	183,0	129,0	102,0	96,2	88,6	88,3	91,3	106,0	150,0
1973	154,0	161,0	162,0	124,0	113,0	93,7	83,1	76,3	72,5	73,1	98,7	123,0
1974	216,0	218,0	244,0	221,0	159,0	125,0	104,0	89,9	81,6	86,4	88,0	123,0
1975	160,0	172,0	196,0	188,0	140,0	110,0	100,0	91,7	83,1	87,7	123,0	167,0
1976	222,0	329,0	250,0	217,0	172,0	139,0	116,0	98,5	96,1	88,6	118,0	162,0
1977	214,0	234,0	191,0	177,0	170,0	141,0	115,0	95,8	102,0	111,0	121,0	167,0
1978	235,0	205,0	202,0	203,0	185,0	162,0	121,0	105,0	102,0	114,0	125,0	189,0
1979	344,0	321,0	304,0	251,0	179,0	152,0	132,0	114,0	127,0	111,0	119,0	160,0
1980	226,0	301,0	320,0	244,0	191,0	147,0	126,0	115,0	114,0	109,0	116,0	148,0
1981	258,0	250,0	249,0	201,0	163,0	134,0	114,0	106,0	95,0	102,0	125,0	135,0
1982	208,0	291,0	386,0	248,0	175,0	157,0	129,0	123,0	142,0	159,0	180,0	259,0
1983	270,0	287,0	282,0	265,0	202,0	170,0	137,0	122,0	112,0	118,0	187,0	259,0
1984	253,0	213,0	248,0	259,0	190,0	147,0	125,0	119,0	115,0	117,0	146,0	235,0
1985	290,0	238,0	397,0	310,0	204,0	158,0	142,0	123,0	119,0	133,0	132,0	122,0
1986	215,0	275,0	259,0	224,0	193,0	146,0	124,0	123,0	119,0	132,0	123,0	158,0
1987	221,0	216,0	255,0	180,0	150,0	136,0	111,0	98,5	94,2	105,0	105,0	208,0
1988	213,3	249,2	373,0	298,0	212,0	169,0	141,0	120,0	109,0	112,0	121,0	197,0
1989	237,0	299,0	298,0	247,0	204,0	152,0	138,0	132,0	117,0	112,0	119,0	126,0
1990	201,0	226,0	212,0	217,4	166,8	137,8	116,0	103,0	115,0	140,0	129,0	144,0
1991	202,0	276,0	259,7	217,4	166,8	137,8	118,6	106,5	104,0	108,3	124,8	155,0
1992	213,3	249,2	259,7	217,4	166,8	137,8	118,6	106,5	104,0	108,3	124,8	164,0
1993	267,0	290,0	285,0	235,0	194,0	164,0	139,0	106,5	104,0	108,3	124,8	164,0
1994	219,0	258,0	258,0	237,0	172,0	150,0	133,0	122,0	112,0	118,0	112,0	147,0
1995	244,0	374,0	256,0	253,0	176,0	147,0	129,0	118,0	108,0	112,0	128,0	152,0
1996	210,0	225,0	312,0	227,0	174,0	145,0	129,0	118,0	112,0	116,0	198,0	217,0
1997	237,0	313,0	303,0	231,0	179,0	160,0	134,0	122,0	121,0	123,0	127,0	190,0
1998	174,2	249,2	259,7	217,4	150,9	127,0	111,0	110,0	106,0	106,0	138,0	209,0
1999	202,0	218,0	329,0	262,0	159,0	137,0	122,0	107,0	105,0	102,0	115,0	131,0
2000	148,0	199,0	280,0	174,0	149,0	127,0	114,0	103,0	98,4	99,4	129,0	197,0
2001	213,0	201,0	224,0	188,0	144,0	133,0	110,0	94,8	102,0	115,0	127,0	196,0
2002	209,9	274,4	252,4	196,0	155,0	127,0	113,0	105,0	105,0	103,0	113,0	122,0
2003	204,4	290,2	286,7	232,5	157,1	128,3	111,8	105,6	104,3	126,2	160,5	184,5
2004	239,4	286,0	202,4	207,8	170,1	137,8	129,7	108,4	101,5	106,3	124,8	164,0

6 CONCLUSIONS

The proposed method can be an interesting option to simplify data recovery from average streamflow series required to estimate the electrical potential of the rivers, since in inventory studies, particularly in the north of Brazil, there is a lack of data due to the small amount of fluviometric stations in operation.

Although the proposed method has worked very well for the data center of the table, it is worth noting that the recovery of data that does not have eight neighbors becomes complicated through this method. For future experiments, it would be interesting to test other strategies to try to achieve better accuracy for the marginal data.

7 REFERENCES

[1] R. G. Otsuki e D. S. Reis Junior. "Análise Comparativa de Metodologias de Estimativa de Séries de Vazões Médias Mensais Aplicadas a Estudos Energéticos de Aproveitamentos Hidrelétricos", *XIX Simpósio Brasileiro de Recursos Hídricos*, Maceió, AL, 2011.

[2] E. W. M. Lucas, F. A. S. Souza, F. D. S. Silva, P. S. Lucio. "Modelagem hidrológica determinística e estocástica aplicada à região hidrográfica do Xingu – Pará." *Revista Brasileira de Meteorologia*, v.24, n.3, 308-322, 2009.

[3] Matriz Energética Nacional 2030. Brasília, Ministério das Minas e Energia (MME), Empresa de Pesquisa Energética (EPE), 2007.

[4] Ministério de Ciência e Tecnologia / Centro de Estudos e Gestão Estratégica (MCT/CGE). *Diretrizes estratégicas para o Fundo de Recursos Hídricos de Desenvolvimento Científico e Tecnológico*. Brasília, 2001.

[5] C. E. M. Tucci. *Hidrologia: Ciência e Aplicação*. UFRGS. 2ª ed. Porto Alegre. ABRH, 2001.

[6] Centrais Elétricas Brasileiras. *Potencial hidrelétrico brasileiro*. Portal de Minas e Energia do Governo Federal, 2007. Disponível em:

http://www.eletrobras.com.br/EM_Atuacao_SIPOT/sipot.asp. Acessado em: 18/10/2013.

[7] G. Box, G. Jenkins, G. Reinsel. *Time series analysis: forecasting and control*. 3ª Ed. New Jersey: Prentice Hall, 1994.

[8] W. Collischonn, C. E. M. Tucci, R. T. Clarke, P. L. S. Dias, G. S. Oliveira. "Previsão Sazonal de Vazão na Bacia do Rio Uruguai 2: Previsão Climática- Hidrológica." *Revista Brasileira de Recursos Hídricos*. v.10, n.4. pp. 61-72, 2005.

[9] R. A. P. Freitas, D. D. Lindermann, L. S. Souza, H. Faria, M. Santos, A. Elesbom, F. Casagrande. "Análise de Séries Temporais de Vazão e Precipitação na Bacia Hidrográfica do Rio São Mateus." *In Anais do XVI Congresso Brasileiro de Meteorologia*, Belém, PA, 2010.

[10] I. N. d. Silva, D. H. Spatti e R. A. Flauzino, *Redes Neurais Artificiais para Engenharia e Ciências Aplicadas - Curso Prático*, São Paulo: Artliber, 2010.

[11] T. Masters. *Neural, Novel and Hybrid Algorithms for Time Series Prediction*. New York: John Wiley and Sons, 1995.

[12] M. Brown e C. Harris. *Neurofuzzy Adaptive Modelling and Control*. London, Prentice Hall, 1994.

[13] M. Campolo, P. Andreussi, A. Soldati. "River flow forecasting with a neural network model." *Water Resources Research*, Washington, v. 35, n. 34, p. 1191–1198, 1999.

[14] J. Anmala, B. Zhang, R. S. Govindaraju. "Comparison of artificial neural networks and empirical approaches for predicting watershed runoff." *Journal of Water Resources Planning and Management*, New York, v. 126, n. 3, p. 156–166, 2000.

[15] R. Sacchi, M. C. Ozturk, J. C. Príncipe, A. A. F. M. Carneiro, I. N. d. Silva. "Water Inflow Forecasting Using the Echo State Network: a Brazilian Case Study", *IEEE Proceedings of International Joint Conference on Neural Network*, Orlando FL, USA, 2007.

[16] A. Garnier, J. Eynard, M. Caussanel, S. Grieu, "Missing Data Estimation for Energy Resources Management in Tertiary Buildings", *2nd International Conference on Communications, Computing and Control Applications (CCCA)*, Marseilles, France, 2012.

[17] V. P. Oikonomou, J. Spilka, C. Stylios, L. Lhostka, "An Adaptive Method for the Recovery of Missing Samples from FHR Time Series", *2013 IEEE 26th International Symposium on Computer-Based Medical Systems (CBMS)*, Porto, Portugal, 2013.

[18] H.-m. Li, P. Wang, L.-y. Fang, J.-w. Liu, "An Algorithm Based on Time Series Similarity Measurement for Missing Data Filling", *2012 24th Chinese Control and Decision Conference (CCDC)*, Taiyuan, China, 2012

[19] Agência Nacional de Energia Elétrica (ANEEL), *Estudo de Inventário Hidrelétrico do rio Formoso*, 2007.

SESSION

KNOWLEDGE ENGINEERING, EVOLUTIONARY STRATEGIES, GENETIC ALGORITHMS AND APPLICATIONS

Chair(s)

TBA

Psychological Constructs for AI Systems: The Information Continuum

James A. Crowder
Raytheon Intelligence, Information, and Services
16800 E. Centretech Parkway, Aurora, Colorado 80011

Abstract - *Research for the development of credible solutions within the Information Continuum has been a 17 year journey that began in the mid 1990's when the we were designing new ways to perform data capture, processing, analysis, and dissemination of high volume, high data rate, streams of information (what today would be called a "big data" problem). Hence, data analysis and lack of quality user interaction within that process are not a new problem. Users have continued to be challenged with keeping up with the vast volumes and multiple streams of data that have had to be analyzed. By the time a user had grabbed a time-slice of data, plotted it, and analyzed it, 100's of Gigabytes of data had passed through the system. In order to provide some semblance of rational attack against the onslaught of data we created what could be likened to a virtual window into the system that allowed the analysts to "walk into the middle" of the data and look at it as it flowed through the system. Analysts could reach out and grab a set of data, rotate it through its axes, and perform automated analysis on the data while remaining within the system data flow. This way analysts could intelligently and rapidly hop between portions of data within multiple data streams to gain pattern and association awareness.*

Keywords: Information Continuum, Big Data, Artificial Intelligence, Knowledge Density, Knowledge Relativity.

1. Introduction

Our work in data representation and visualization resulted in a realization that each point in time within the rapid data flow was an independent and discrete information continuum with specific and qualitative state. Subsequently, analogous thoughts began to emerge from research in artificial intelligence and artificially cognitive system theory [6]. Envisioned was a virtual view within a portion of the human brain where one could view a given neural node, or a given neuron, and subsequently view data flow as data/information traveled in and out of the neuron [4]. Once gathered, a hypothesis emerged that the analysis of brain locale, data, and study of brain processes through this type of virtual environment, could lead to important understanding of learning, inferring, storing, and retrieving (reconstruction) and/or all aspects of human neural processing [1]. This led to the possibility of a theoretical Neural Information Continuum (NIC) [5].

2. Information Flow within a Synthetic Continuum

One of the first areas that must be investigated when considering an Artificially Intelligent System (AIS) is the flow of information. Humans take in ~200,000 pieces of sensory information each and every second of every day of our lives. Our senses (see, hear, smell , touch, etc.) are constantly receiving and processing information, correlating it, reasoning about it, assimilating it with what we already know, and finally leading to decision making based upon what was learned. For a system to become dynamic, self-evolving and ultimately autonomous, we propose to provide these same abilities; although the sensors and sensory perception systems may be synthetic and different, sensing a variety of information types that humans can't sense (e.g., infrared or RF information), the processes for autonomy, which correlate, learn, infer and make decisions, are the same. Besides receiving information from a variety of sources and types (e.g., auditory, visual, textual, etc.), another important aspect of information, is that the content is received at different times and at a variety of latencies (temporal differences between information). Additional characteristics include, a variety of

associations between the information received and information the system may have already learned, or information about subjects never encountered. Therefore, these information characteristics and the challenging real-time processing required for proper humanistic assimilation, help us form the theory of the Autonomic Information Continuum (AIC). One of the first steps in developing our theory of synthetic autonomic hypotheses is observing/understanding the information continuum and the associated characteristics and operational relationships within the human brain. Hence, as we develop understanding of information flows into and out of neural nodes, types of information, processing mechanisms, distributions of information, enable us to establish foundational mathematical representations of these characteristics and relationships.

Processing, fusing, interpreting, and ultimately learning about and from received information requires taking into account a host of factors related to each piece or fragment of information. These include [7]:

- Information Types
- Information Latencies
- Information Associations e.g.:
  - Time, State, Strength, Relationship Type, Source, Format etc.
- Information Value
- Information Context

Mathematically modeling the information continuum field surrounding a node within our synthetic AIC, is accomplished via inclusion of each discrete association for any node *u*, takes the form shown in the following equation:

$$C\frac{du(x,y,t)}{dt} = -\frac{1}{R}u(x,y,t) + \int_x\int_y w(x,y)z(x,y,t)dxdy + I(x,y,t)$$

where:
u *represents the unit node of the system,*
x *represents the preprocessed input to node u,*
y *represents the output from node u,*
w represents the relative contextual information threads and association weight of *u* with its surrounding nodes, including a decay factor for each relative information thread that describes the relative contextual decay over time, where:

$$w = \sum_{j=1}^{M} {}^{1}\!/_{r_j} T_j KD_j W_j$$

where:

T *represents the Contextual Information Thread j derived from Fuzzy, Self-Organizing Contextual Topical Maps*
KD *represents Knowledge Density j of Information Thread T*
W *represents Weighting for Contextual Thread j, and:*

$$\sum_j W_j = 1$$

I represents the processing activity for node *u*,
z represents the learning functionality for node *u*,
1/R represents the decay rate for node u[1], and
C represents the capacity of node *u*.

This information field continuum equation allows us to analyze the equilibrium of nodal states within the AIC and to continuously assess the interactions and growth of independent information fragments within the system. Even in the most dense, most complex, cluttered information environments, each fragment of information and each action within the AIC is entropically captured explicitly and implicitly within Information Continuum Equation (ICE). This equation is the entropic engine which

[1] In this case, the decay represents the information's relative value over time.

provides the ongoing analysis and virtual view into a synthetic AIC. Equation 2-1 enables us to assess the performance and quality of processing and to understand the capacities, information flows, associations, and interactions of knowledge and memories within the system, as well as, supporting analysis and inherent understanding of real-time system behavior. The variables in ICE can be interpreted as the average values in a heterogeneous assembly of information nodes, where ICE describes the behaviors of the interactions among ***n*** node assemblies within a synthetic AIC processing system. The objective is to have the ability to measure, monitor, and assess multi-level states and behaviors, and how and what kinds of associative patterns are generated relative to the external inputs received by an AIC system. ICE provides the analysis needed to understand the AIS's ability for processing external content within an AIC. Hence, real-time assessment and monitoring, and subsequent appropriate control, are expected to allow us to avoid developing a rogue AIC, much to the chagrin of Hollywood script writers.

3. Information Processing Models

Establishing a hierarchy of information flow within an AIC is a key objective for development of synthetic autonomic characteristics (e.g. cognition, thinking, reasoning, and learning). An AIC will need to be able to ingest and process a variety of inputs from many diverse information sources, dissect the information into its individual information fragments, fuse the information, and then turn this information into a formation which can be used to determine action-actionable intelligence. An AIC system must be able to assess situations previously not encountered, and then decide on a course of actions, based on its goals, missions, and prior foundational collected knowledge pedigree.

The underlying issues and challenges facing Artificially Intelligent systems today are not new. Information processing and dissemination within these types of systems have generally been expensive to create, operate and maintain. Other artificially intelligent system challenges involve information flow throughout the system. If flow is not designed carefully and purposefully, the flood of information via messages within these systems and between their software and hardware components can cause delays in information transfer, delaying or stunting of the learning process which can result in incorrect or catastrophic decisions.

Therefore, real-time decision making processes must be supported by sensory information and knowledge continuously derived from all cognitive processes within the system simultaneously, in a collectively uniform and cooperative model. Additionally, transformation from information to knowledge within an AIC system requires new, revolutionary changes to the way information is represented, fused, refined, presented, and disseminated. Like the human brain, the cognitive processes within an AIC must form a cognitive ecosystem that allows self-learning, self-assessment, self-healing and sharing of information across its cognitive sub-processes, such that information is robustly learned and rapidly reusable.. This AIC ecosystem involves inductive, deductive, experimental, and abductive thinking in order to provide a complete Data-to-Information-to-Knowledge process explained in detail throughout the rest of the book. At a high-level, we are applying the theory of AIC and applying the constructs to the development of humanistic analogous AIS. The AIS human brain analogy provides two-main layers of processing, a ***Deductive Process*** and an ***Investigative Process***. The ***Deductive Process*** is utilized for assembling information that has been previously learned and stored in memories (deductive and inductive logic), whereas the ***Investigative Process*** looks for patterns and associations that have not been seen before (abductive and experimental logic) [2]. Figure 1 illustrates the differences between

deductive, inductive, abductive, and experimental inferences [8].

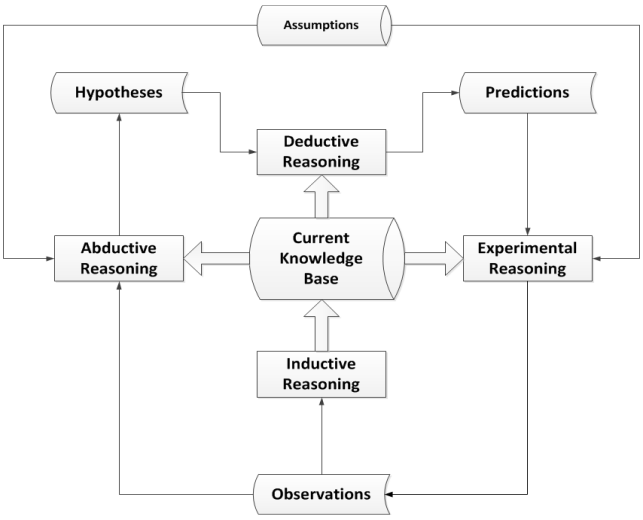


Figure 1 - Differences between Logical Inference Systems

Inductive Reasoning: Inductive reasoning involves concluding after evaluating facts; reasoning from specific facts to a general conclusion and allowing for inferencing. It also requires human experience to validate conclusions. An example might be: Zebras at the zoo have stripes; therefore all zebras have stripes [10].

Deductive Reasoning: Deductive reasoning is just the opposite. Deductive reasoning moves from general principle to specificity. This type of reasoning is based upon accepted truths. An example of deductive reasoning might be: I know that all zebras have stripes therefore when I go to the zoo, if I see a zebra, it will have stripes.

Abductive Reasoning: Abductive reasoning allows for explanatory hypothesis generation or generating ideas outside of the given facts to explain something that has no immediate satisfactory explanation.

There are a number of ways in which people reason, but most human reasoning follows one of these three reasoning systems. Other ways that humans' reason includes cause and effect reasoning where causes and after effects are considered. Analogical reasoning is a way of relating things to other novel situations. Comparative reasoning as it implies involves comparing things. Still another reasoning method is conditional or if/then reasoning. Many of us have used the pros and cons methods of reasoning as well. Systemic reasoning involves thinking that the whole is greater than the sum of its parts, and finally reasoning by using examples or analogies. Hence, there are numerous logical ways in which people reason about events and situations. An autonomous artificially intelligent system must be able to employ these same reasoning methods in order to interact with a random and often chaotic world.

4. Discussion

If we desire to create an Artificial Cognitive Architecture that encompasses the AIC discussed above, in order to create a system that can truly think, reason, learn, utilizing the inferences shown in Figure 1, we must consider the overall implications of such a system, including the psychological impacts and considerations both for humans and for the system itself [9]. Further research is needed to understand the psychological effects of not only real human-AI interaction, but also the effect of human interaction on AIC learning and self-evolving [3]. Sometimes learning from humans is a dangerous thing.

7. References

1. Ackley, D. and Litman, M. 2002. Interactions between Learning and Evolution. Artificial Life XII. Addison-Wesley.
2. Ade. H. and Denecker, M. 1995. Abductive Inductive Logic Programming.IJCAI-95.
3. Anderson, J. 2004. Cognitive psychology and its implications: John r. Anderson: Worth Pub.
4. Ashcraft, M. 1994. Human Memory and Cognition. Harpercollins College Division, New York, NY.

5. Crowder, J. 2010. Flexible Object Architectures for Hybrid Neural Processing Systems. Proceedings of the 12th annual International Conference on Artificial Intelligence, Las Vegas, NV.
6. Crowder, J. 2012. Cognitive System Management: The Polymorphic, Evolutionary, Neural Learning and Processing Environment (PENLPE). Proceedings for the AIAA Infotech@Aerospace 2012 Conference, Garden Grove, CA.
7. Crowder, J. 2012. The Artificial Cognitive Neural Framework. Proceedings for the AIAA Infotech@Aerospace 2012 Conference, Garden Grove, CA.
8. Crowder, J. and Carbone, J. 2011. The Great Migration: Information to Knowledge using Cognition-Based Frameworks. Springer Science, New York, NY.
9. Crowder, J. and Friess, S. 2012. Artificial Psychology: The Psychology of AI. Proceedings of the 3rd Annual International Multi-Conference on Informatics and Cybernetics. Orlando, FL.
10. DeRaedt, L. 1992. Interactive Theory Revision: An Inductive Logic Programming Approach. Academic Press

Implementation of an Evolutionary Algorithm Using the Node-Depth-Degree Encoding in a Nios II processor in FPGA

Gustavo Siqueira Vinhal,
Anderson da Silva Soares,
Telma Woerle de Lima and
Gustavo Teodoro Laureano
Intitute of Informatics
Federal University of Goiás
Goiânia, GO, Brazil

Clarimar José Coelho
Computer Science Departament
Pontifical University Catholic of Goiás
Goiânia, GO, Brazil

Marcilyanne Moreira Gois and
Alexandre Cláudio Botazzo Delbem
Instituto de Ciências Matemáticas e
de Computação,
Universidade de São Paulo,
São Carlos, SP, Brasil

***Abstract*—Many problems relevant to NP-Hard class are present in the real world. Among them we can mention the problems of network design (PNDs) that involve electricity distribution, vehicle traffic, and others. Do not exist efficient algorithms that provide an exact solution for these types of problems. Over the years research has been developed used evolutionary algorithms (EAs) to provide an efficient solution for these problems. In addition, appropriate data structures may further improve the performance of EAs to PNDs. The node-depth-degree encoding (NDDE) shows significant results for PNDs. The application of EAs in hardware can improve the performance of the algorithm. In this sense, this work presents the implementation of an EA in Nios II processor of a FPGA board to solving the PND minimum spanning tree with degree constraint. The results demonstrate that the implementation of EAs in hardware brings significant results with better performance, due to the power of parallelism present in the FPGA.**

Keywords: Network Design, Minimum Spanning Tree with Degree Constraint, Evolutionary Algorithms, Node-Depth-Degree Encoding, FPGA.

I. Introduction

Network Design Issues (NDIs) are present in many real-world applications in fields of engineering and science. Such applications involve problems such as electricity distribution, vehicle routing, and computer networks, among others. In real-world scale, large amount of elements (vertices) and connections (edges) restricts the algorithms used for these problems. The traveling salesman problem [1] is an example for which there is no deterministic algorithm of polynomial complexity to solve it when the graph has large dimension.

This type of problem is classified as NP-Hard in which there is no deterministic algorithm that runs in polynomial time to solve it. To solve these problems are used as approximation algorithms such as metaheuristics and heuristics. These algorithms do not provide an exact solution to the problem, but a rough one. Among these techniques are highlighted the Evolutionary Algorithms (EAs, Evolutionary Algorithms).

EAs have the simulation of the natural evolution of species as main point. Given a set (population) of individuals (possible solutions), at each iteration (generation) operators are applied for best individuals being found. However, EA with standard encoding may not be efficient when applied to a large network, i.e. with several vertices. In general, these EAs applied to large networks require processing time and memory for validation or correction of infeasible solutions.

To improve the efficiency of an EA applied to a NDI with high complexity scale, dynamic data structures (encoding) have been researched more appropriate [4]. Such encodings increase the quality of solutions generated by the EA, as it enables better exploiting the search space. Furthermore, the use of specific encodings minimizes the need for remediation processes and validation of solutions.

Among various encodings in the literature, the Node-Depth-Degree Encoding (NDDE, Node-Depth-Degree Encoding) stands out due to complexity $O(\sqrt{n})$ while others require time $O(n)$, where n and the number of vertices of the graph [9, 24]. Thus, a suitable encoding for NDIs for large scale.

In this sense, the implementation of EAs that use a representation NDDE in FPGA (Field Programmable Gate Array) provides significant results in performance, allowing its use for NDIs with thousands and even millions of vertices. This result is possible due to the intrinsic parallelism of the FPGA that fits the EA processes in order to reduce the computational time. Furthermore, the implementation in FPGA allows the computation time becomes limited by a constant for graphs with different numbers of vertices.

This work aims at developing an EA for the minimum spanning tree problem with restricted degree in in FPGA in its Nios II processor. The proposed EA will use the NDDE representation implemented in FPGA. The main goal with the development of the proposed EA is to analyze its ability to optimization with the maintenance of constant processing time.

II. Representation Node-Depth-Degree

The choice of representation is important to the performance of evolutionary algorithms (EAs). The choice of an inadequate representation can affect the convergence time of the algorithm, besides beholding the process similar to the methods of random [6].

Network design issues (NDIs) are generally represented by spanning trees of a graph. The tree encoding, by means of a string for the NDI, results in large difference between the genotype and phenotype in the tree structure. This difference can be seen in the results that represents an unfeasible network in practice. Thus, conventional representations in EA for NDIs have generated satisfactory results, since the computation time for generating appropriate networks is high for large scale graphs [3]. Thus, several studies have been performed in order to find more appropriate representations for EAs applied to NDIs [6].

The representation NDDE is based on concepts of paths, depth and degree of the vertex in a tree [15]. It consists of triple (de_i, v_i, deg_i) stored in vector form, where de_i represents the depth and deg_i the degree apex v_i ($i \in \{1, 2, ..., n\}$ | n is the number of nodes of the tree). The tree root will always have depth 0. The order of the remaining vertices in the vector is determined by using any search method in graphs, such as depth search (DFS, Depth-First Search) [16]. Figure 1 illustrates an example of a tree. Table I has the NDDE tree representation presented in Figure 1.

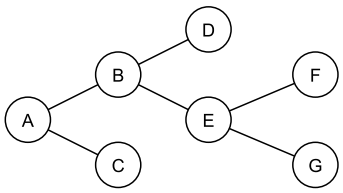


Fig. 1. Tree with seven vertices.

TABLE I. NDDE WHICH REPRESENTS THE TREE OF FIGURE 1.

i	1	2	3	4	5	6	7
de	0	1	2	3	3	2	1
v	A	B	E	G	F	D	C
deg	2	3	3	1	1	1	1

From the tree encoding in NDDE, it can be applied mutation operators on this representation for new solutions. In particular, PAO and CAO operators can be used.

III. PAO AND CAO OPERATORS

In order to obtain new feasible solutions, the PAO and CAO mutation operators are applied in the trees represented by NDDE. Basically, both operators prune a tree T_{ori} at a given vertex and transfer the pruned subtree to another tree T_{des}.

The PAO needs two vertices, one origin and one target. The origin vertex *p* indicates the root of the subtree pruned from the origin tree T_{ori}. The target vertex *a* in T_{des} is the vertex to which the subtree will be connected.

The CAO and similar to the PAO operator with the difference that now there is a third vertex *r* representing the new root of the subtree.

A. PAO - Preserve Ancestor Operator

For PAO application, randomly chooses a vertex *p* in the origin tree T_{ori} and a vertex *a* in the target tree T_{des}. After choosing the vertex *p*, the transfer process occurs according to the following steps:

1) Determine the interval ($v_p - v_l$) which represents the pruned subtree in the origin tree T_{ori}. As the apex *p* is known, it is just to find the vertex *l* according to NDDE. This vertex is the last of the pruned subtree. The interval ($v_p - v_l$) is composed of v_i vertices where $v_i > v_p$ and $de_i > de_p$.
2) Enter the interval ($v_p - v_l$) in the target tree T_{des} at position v_a + 1.
3) Edit the values of the degree and depth of the vertices.

The operator PAO causes few changes in the tree. Figure 2 illustrates an example of the PAO application in the tree of Figure 1. The representation NDDE of T_{des} is shown in Table II. The selected vertices are *a* = C and *p* = E.

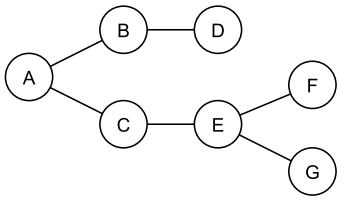


Fig. 2. Tree of Figure 1 with application of PAO operator.

TABLE II. NDDE WHICH REPRESENTS THE TREE OF FIGURE 2.

i	1	2	3	4	5	6	7
de	0	1	2	1	2	3	3
v	A	B	D	C	E	F	G
deg	2	2	1	2	3	1	1

B. CAO - Change Ancestor Operator

In the CAO operator the subtree that will be transferred receives a new root represented by the vertex *r*. The vertex *r* is chosen randomly in the range ($v_p - v_l$) and must be different from the vertex *p*. The order of the vertices of the subtree is changed. Instead of copying the subtree rooted at *p*, firstly, the subtree rooted at vertex *r* is stored in a temporary vector. Subsequently, it stores the first subtree rooted in the previous vertex to *r*, ($v_r - 1$), which is present in the range of ($v_p - v_r$) and that does not contain a subtree already present in the temporary vector. The temporary vector is filled until the vertex *p* is reached.

CAO operates according to the following steps:

1) Choose the vertex *p*, and within the range ($v_p - v_l$), the new root represented by the vertex *r*.
2) Copy the subtree rooted at the vertex *r* to the temporary vector.
3) Copy the subtree rooted at each previous vertex *r* in the interval ($v_p - v_r$) to the temporary vector. Each subtree cannot contain vertices already present in the temporary vector.
4) Insert the vertices of the temporary vector in the target tree T_{des} at position v_a + 1.
5) Edit the values of degree and depth of the vertices.

The operator CAO causes significant changes in the tree. Figure 3 illustrates an example of the CAO application in the tree of Figure 1. The NDDE representation is shown in Table III Nodes chosen are a = C, p = B and r = G.

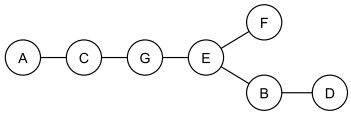


Fig. 3. Tree of Figure 1 with application of CAO operator.

TABLE III. NDDE WHICH REPRESENTS THE TREE OF FIGURE 3.

i	1	2	3	4	5	6	7
de	0	1	2	3	4	4	5
v	A	C	G	E	F	B	D
deg	1	2	2	3	1	2	1

IV. RECONFIGURABLE COMPUTING

The reconfigurable computing has as its basic principle of operation the reprogrammable logic devices. These devices allow to program and reprogram several times directly in hardware. This feature makes the application flexible and with high performance.

Devices of hardware widely used in reconfigurable computing are FPGAs (Field Programmable Gate Array). These devices have speed and features similar to those of ASICs (Application Specific Integrated Circuit), while they are flexible as general purpose computers.

FPGAs are devices with high degree of parallelism [21], being widely used in more complex applications, both in specific and general purpose. These devices allow the user to reconfigure the gates, making the FPGA work according to the desired need. Thus, FPGAs offer flexibility in the development of complex solutions of high performance with low design cost.

The FPGA is composed of three components:

- Logic Blocks (CLB, Configuration Logical Blocks): circuits consisting of flip-flops which can be configured independently;
- Input and Output Blocks (IOB, Input/Output Block): interface of the FPGA pin connections with the outside world;
- Interconnection keys: trails that connect CLBs and IOBs.

V. IMPLEMENTED ARCHITECTURE

The architecture of hardware used is illustrated by Figure 4 and it consists of the following components:

- Microprocessor softcore - Nios II;
- External memory, system memory hardware responsible for the storage of NDDEs, the weights of edges of the graph G and GA implemented code to be run on the Nios II microprocessor;
- Avalon Bus, responsible for communication between system internal devices in hardware;
- JTGA Interface responsible for system communication in hardware with external devices (e.g. PC);
- P-PAO-S Module (Parallel Circuit for PAO) that performs the PAO mutation operator.

According to Figure 4 the external computer (PC) generates random seeds to set the initial state of the LFSR (Linear Feedback Shift Register) pseudorandom number generator. This generator is used to obtain the vertices p and a used in the PAO operator. The seeds are transferred from the computer to the system in hardware through the JTGA interface.

The Nios II microprocessor performs the execution of the entire evolutionary process of GA, besides being responsible for reading and transferring NDDEs of external memory for the P-PAO-S and the copy to the external memory of the best NDDEs found by P-PAO-S. All this communication is performed by Avalon bus.

In the module P-PAO-S, the Working Controller is responsible for communication between the Nios II microprocessor and external memory in order to send and receive T_{ori} and T_{des} for Workers. Each T_{ori} and T_{des} obtained in the external memory is replicated to all workers. Each worker performs the operator PAO with vertices p and a different from each other. After execution of the Workers, the Comparer of Fitness performs a comparison of fitness of trees generated. The best pair of trees found is then compared with the old trees and the best pair found is stored in a Buffer. Finally, the Working Controller accesses the Buffer and copies the best pair of trees to the external memory.

A. Representation Node-Depth-Degree in FPGA

The NDDE parallelization in FPGA was carried out taking into account that each NDDE is represented by a set of integer vectors that can be implemented in the form of memory blocks, whose number of bits is proportional to the vector size.

Each vector is handled in parallel in the FPGA and has average size of $O(\sqrt{n})$, where n is the number of vertices of the graph. The handling strategy of NDDEs parallel in hardware was named HP-RNP (Hardware-Parallelized Depth-Node Representation). Equation 1 shows the processing time of the HP-RNP.

$$\frac{\textit{number of modified forests}}{\textit{processing time}} = \frac{\sqrt{n}}{O(\sqrt{n})} = O(1). \qquad (1)$$

According to Equation 1, the parallelization process of HP-RNP performs $\sqrt{n}$ changes in the spanning forest. As new forests are generated in the average time of $O(\sqrt{n})$, the processing time to generate a new forest is limited by a constant $O(1)$.

B. Implementation of Genetic Algorithm

The GA implementation in the Nios II processor is aimed at reducing the burden of NDDEs through the evolutionary

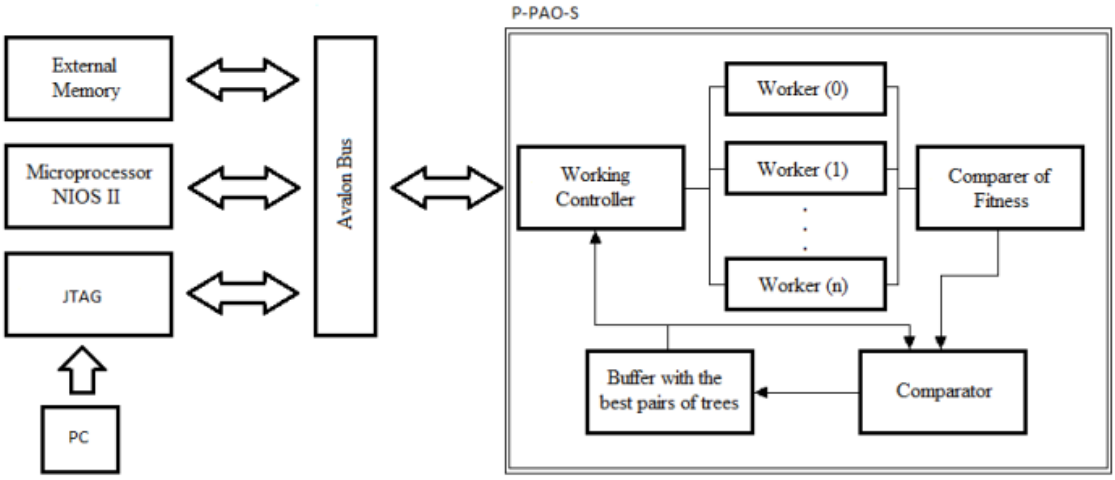


Fig. 4. Architecture used.

process with a lower running time compared with the implementation perfomed in software. This time reduction is due to the power of parallelism of the FPGA.

Before the GA implementation by the Nios II processor, some steps are performed: reading of weights matrix, generation of initial forests and storage of forests generated in the external memory. The modified algorithm of Kruskal [22] was used for generation of initial forests. The modification consists of including the degree of each vertex on the data structure that forest stores. The degree is used by the PAO operator and its inclusion is checked in order to avoid generating infeasible initial solutions. The algorithm uses an array of random weights to generate the initial forests. After generation of forests, the weights of edges are updated with the values of the data sets used (see Section VI). Each forest generated has only one tree, according to the formulation of the minimum spanning tree problem with degree constraint.

Once generated the initial forests, the GA is run. At each GA iteration, the evolution process performs the forest selection through the tournament of 3. Three forests are randomly selected to compete among themselves. The best fitness (reduced weight) is then selected and sent to the PAO operator. The number of times the PAO operator is executed in a single GA generation is given by Equation 2.

$$\text{Running} = \frac{2.400.000}{\text{No. workers} * \text{No. Generatons}} \quad (2)$$

According to Equation 2 the amount of running depends on the number of workers allocated and the number of GA generations. The greater the number of workers, the greater number of running of operators in a single generation. Therefore, the total number of running of the operator is reduced when there is increase of workers.

The implementation of the PAO operator works with the entry of two trees, one source and one target. To represent the dc-MST's NDDEs, each forest has only one tree. To circumvent this restriction, tree of forest is divided into two. Figure 5 illustrates how the process of splitting and joining of a forest occurs.

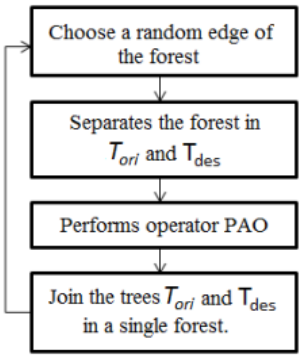


Fig. 5. Splitting and joining the forest.

According to Figure 5, the forest division occurs randomly by removing an edge that connects two vertices i and j of the tree. The source tree T_{ori} will be the pruned subtree with root j. The target tree T_{des} will be the original tree without the pruned subtree. Shortly after the running of the PAO operator, the rest of the tree T_{ori} is connected to T_{des} by the edge removed. The randomly choice of edge before each running of PAO operator makes any effects arising from this process is alleviated.

The splitting and joining the forest occur before and after running of each operator PAO, respectively. Therefore, it is performed the same number of times that the PAO is executed (as Equation 2). As the forest division occurs before the operator PAO bein run, edge removed and being the same for all workers.

At the end of each GA generation the best individual obtained by the PAO application in the Workers is inserted into the population, if it is better than the worst forest population. If it is worse than all the forests that new individual is discarded. This process of inserting the new individual obtained in the population is considered fully elitist.

Besides the FPGA implementation, it was also applied to the test cases a version in software of a genetic algorithm with node-depth-degree encoding called EANDD. The aim of

the application in software was the comparison of the results depending on runtime.

VI. Tools Used

The Cyclone-II family FPGA, EP2C70F896C6 model, was used. This model works at a frequency of 500MHz, has SDRAM memory of 1MB, display LCD, PCI express and USB connections for communication with external devices. It has 68416 programmable logic cells. Moreover, it contains the Nios II processor and supports the interface JTGA (Joint Test Action Group).

For programming the FPGA device was used software Quartus software version 10.1. This software owner is produced by Altera, the same company that developed the FPGA board. The Quartus is an EDA tool (Eletronic Design Automation) in which one can perform synthesis, simulation, analysis and programming in FPGAs devices. Through it the time for projects development in hardware configurable is reduced due to the automation of its tasks. Implement a circuit design involves the following steps: project description, compilation, simulation and programming of FPGA device.

Besides Quartus, the SOPC Builder (System on a Programmable Chip) tool was used for generating a complete SOPC. This tool allows the automation of connections between the components of hardware, thus creating self-performance systems. The SOPC Builder has a large library of components such as memory controllers, interfaces, peripheral, processors (such as Nios II) and co-processors. The interconnections between components are performed by using the Avalon bus.

The GA developed here called H-GA (Hardware Genetic Algorithm) to be run on the Nios II processor was implemented using the C language. The descriptive language of hardware used was BSV.

The EANDD was performed using a computer with Intel Core i7 and 32GB of RAM.

The data used for this study were obtained directly from the authors, and these are cases benckmark in the literature for evaluating GAs applied to dc-MSTP. These data are arranged in two sets of graphs called M-Graphs.

The set M-Graphs consists of complete graphs for 50, 100, 200, 300, 400 and 500 vertices. For graphs with 50, 100 and 200 vertices, the test set has three adjacency matrices with different weights, which are differentiated by the code N1, N2, N3. The construction process for weights of edges of these graphs was proposed by [18].

VII. Results

The H-GA was used to solve the minimum spanning tree problem with degree constraint (dc-MSTP). Among the tests performed, it was decided by the parameters shown in Table IV for the run the algorithm and obtaining the results that will be presented and discussed in this section.

The population size was determined by means of empirical testing. With a population of less than 6 individuals, the H-GA presented premature convergence to optimal sites. With a population greater than 6 individuals, the processing time had increased, but the values of fitness showed no improvement.

TABLE IV. Parameters of H-GA.

Number of Generations	300
Population Size	6
Selection Operator	Tournament of 3
Mutation Operator	PAO

The best individual obtained from the various applications of the mutation operator in workers is inserted into the population only if it has a value of fitness better than the worst individual in the population. That is, if the new individual A have a fitness better than the individual B, the worst fitness in the population, A is inserted into the population instead of B. Otherwise, A is discarded. This process is known as total elitism, i.e. all individuals are preserved, being discarded only if they are worse than a new individual.

The H-GA was carried out with each of the graphs of both test sets, with the degree constraints $d = 3, 4$and5. Due to the stochastic nature of GAs, each test was replicated 10 times and its average will be displayed in the results.

Below are the results obtained using the graphs set M-graphs. After running the H-GA were obtained values of fitness of final populations. Table V shows these values, expressed by the mean of the best solutions obtained in each run and the mean of the population.

Table V also shows the improvement achieved between the mean values of fitness of the best individuals of the final and initial populations. The H-GA obtained, on average, improvement of 60% of the values of fitness. The greatest improvement occurred for case M-Graph 200 (N3) with constraint degree of 3, with a gain of 74.57%. The gain was worse for the case M-Graph 100 (N1) with constraint degree 5 of 46.56%. Again the results show that the H-GA has optimization capacity managing to optimize the fitness solutions of the initial population, and thus having better solutions in the final population.

The EANDD was applied in graphs M-Graphs with the same settings as the H-GA. Table VI shows the comparison between the means of the best fitness and running times obtained by EANDD and H-GA. In addition, the optimal solutions obtained in [23] are presented. The results presented in [23] are obtained by an exact algorithm of branch and cut. In [23] was presented only the results for the degree constraint equal to 5. Therefore, Table VI only shows the results for this constraint.

As shown in Table VI once again the results obtained via software were better than those obtained via hardware for values of fitness. However, the runtime used by H-GA is notably lower than EANDD. This result illustrates the benefits of FPGA parallelism.

VIII. Conclusion

This paper proposes the development of a genetic algorithm (GA) in hardware with the representation node-depth-degree to problems for network design issues (NDIs). It was first introduced a review of the minimum spanning tree with degree constraint problem. Subsequently, it was presented a concept of EAs, in particular the GA as a method proposed to be used

TABLE V. FITNESS OF THE FINAL POPULATIONS.

Vertices	Degree	Best Individuals Mean	Mean of Average Population	Improvement (%)
M-Graph 50 (N1)	3	14,150350	14,594200	58,35
	4	15,345000	15,967891	55,21
	5	15,743500	16,257724	55,15
M-Graph 50 (N2)	3	14,274801	14,733051	58,44
	4	15,121450	15,700350	56,59
	5	15,280350	15,964050	56,22
M-Graph 50 (N3)	3	11,147400	11,582700	61,23
	4	13,059150	13,454791	55,29
	5	13,672650	14,153225	53,01
M-Graph 100 (N1)	3	23,261798	23,924072	64,30
	4	29,086349	29,900223	55,82
	5	34,191200	36,771515	46,56
M-Graph 100 (N2)	3	25,870801	26,497435	62,68
	4	30,821202	31,472759	55,92
	5	32,100001	32,884784	53,75
M-Graph 100 (N3)	3	25,269051	25,889934	63,23
	4	30,259102	31,251935	55,99
	5	32,141703	33,043402	53,66
M-Graph 200 (N1)	3	34,720905	35,165537	73,44
	4	49,394902	50,398387	62,41
	5	53,735303	54,405376	59,44
M-Graph 200 (N2)	3	41,054257	41,470089	70,13
	4	53,776660	54,759043	61,10
	5	58,157404	59,203274	58,28
M-Graph 200 (N3)	3	31,452245	31,754544	74,57
	4	45,365646	46,139745	63,49
	5	50,372746	51,372021	59,33
M-Graph 300	3	60,105370	60,273181	69,33
	4	73,083044	73,660283	62,45
	5	77,020954	78,126577	60,05
M-Graph 400	3	93,564608	93,756509	65,94
	4	106,776948	107,431808	60,91
	5	115,085364	115,990491	57,97
M-Graph 500	3	128,783529	128,880621	62,85
	4	137,424813	137,956787	60,58
	5	146,288269	146,835473	58,12

TABLE VI. COMPARISON OF FITNESS ACHIEVED FROM THE IMPLEMENTATION IN SOFTWARE AND HARDWARE.

Vertices	Optimum	H-GA		EANDD	
		Fitness	Time (s)	Fitness	Time (s)
M-Graph 50 (N1)	6,60	15,74	0,00008877	8,92	0,3456
M-Graph 50 (N2)	5,78	15,28	0,00008873	8,22	0,3796
M-Graph 50 (N3)	5,50	13,67	0,00008882	6,44	0,4924
M-Graph 100 (N1)	11,08	34,19	0,0001368	14,13	1,0826
M-Graph 100 (N2)	11,33	32,10	0,0001366	16,01	1,1544
M-Graph 100 (N3)	10,19	32,14	0,0001365	15,02	1,0302
M-Graph 200 (N1)	18,33	53,73	0,0001286	23,06	2,9034
M-Graph 200 (N2)	19,16	58,15	0,0001283	26,01	3,1752
M-Graph 200 (N3)	16,13	50,37	0,0001281	17,83	2,4270
M-Graph 300	40,69	77,02	0,0001346	50,83	4,1314
M-Graph 400	54,69	115,08	0,0001574	78,27	6,5264
M-Graph 500	79,34	146,28	0,0001624	108,27	10,0430

in solving the NDI of the minimum spanning tree with degree constraint (dc-MSTP).

The GA developed used the NDDE representation aiming at reducing the time complexity to $O(\sqrt{n})$. To get workable solutions, the PAO was used as the mutation operator. The GA was implemented in the architecture of hardware using the Nios II processor in a FPGA board. For a better understanding of the benefits of using this architecture were introduced concepts of reconfigurable hardware. The test set used is composed of two types of graphs representing forests Benckmark. The results obtained with the GA application were presented and discussed. For both types of graphs was observed improvement in the fitness. This improvement was shown to reduce the weights used for forestry problem. However, this improvement was not good enough compared to the EANDD. These inferior results can be improved with some parameter settings. For example, one can change the way of choosing vertices p and a used in PAO operator. Another example is the change in module HP-RNP for acceptance of forest with one tree, thus the PAO operator would apply more appropriately to NDIs of benckmark that have this feature. Such modification may improve the GA performance since each application of PAO operator will not be needed to division and subsequently,

the forest connection. Since, this edge can be exactly what needs to be changed in the solution to obtain improvement of the individual and it will not be allowed due to the splitting process. Such changes do not interfere with the H-GA performance, since the algorithm is able to solve the dc-MSTP at a limited processing time by a constant $O(1)$. Thus, the use of AEs in hardware is validated for NDIs solutions.

REFERENCES

[1] Jonathan L. Gross and Jay Yellen (2004) 'Handbook of Graph Theory', *CRC Press*

[2] D. S. Johnson and J. K. Lenstra and A. H. G. Rinnooy Kan (1978) 'The complexity of the network design problem', *Networks*, 9, 279-285

[3] Telma Woerle de Lima (2009) 'Estruturas de Dados Eficientes para Algoritmos Evolutivos Aplicados a Projeto de Redes', *Universidade de SÃ£o Paulo - USP SÃ£o Carlos*, Abril, SÃ£o Carlos, SÃ£o Paulo, PHD Thesis

[4] Telma Woerle de Lima and Franz Rothlauf and Alexandre C. B. Delbem (2008) 'The Node-Depth Encoding: Analysis and Application to the Bounded-Diameter Minimum Spanning Tree Problem', *ACM*, July, Atlanta, Georgia, EUA, pp.969-976

[5] Gunther R. Raidl (2000) 'An efficient evolutionary algorithm for the degree-constrained minimum spanning tree problem.', *IEEE*, Evolutionary Computation, 2000. Proceedings of the 2000 Congress on, n.1, 104-111

[6] Franz Rothlauf (2006) 'Representations for genetic and evolutionary algorithms.', *Springer*

[7] Franz Rothlauf and David E. Goldberg and Armin Heinzl (2002) 'Network random keys: A tree representation scheme for genetic and evolutionary algorithms', *MIT Press*, Evolutionary Computation, v.10, n.1, 75-97

[8] A.C.B. Delbem and Andre de Carvalho and Claudio A. Policastro and Adriano K.O. Pinto and Karen Honda and Anderson C. Garcia 2004 'Node-depth encoding for evolutionary algorithms applied to network design', *Springer*, Genetic and Evolutionary Computation–GECCO 2004, 678-687

[9] Alexandre C.B. Delbem and Telma Woerle de Lima and Guilherme P. Telles 2012 'Efficient forest data structure for evolutionary algorithms applied to network design', *IEEE*

[10] Gengui Zhou and Mitsuo Gen 1997 'A note on genetic algorithm for degree-constrained spanning tree problems.', *Networks (USA)*, v.30, n.2, 91-95

[11] John H Holland 1975 'Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence.', *U Michigan Press*

[12] Kenneth A De Jong 2006 'Evolutionary computation: a unified approach.', *MIT press Cambridge*, v.262041944

[13] M Ridley 2001 'Evolution.', *Pheonix*

[14] Ricardo Liden 2008 'Algoritmos Geneticos, Uma importante ferramenta da InteligÃªncia Computacional', *Brasport*, 2Âª Edicao

[15] Alexandre C. B. Delbem and A. de Carvalho 2003 'A Forest Encoding for Evolutionary Algorithms Applied to Design Problems', *Lecture Notes in Computer Science*, v. 2723, pp.634-635, Genetic Algorithm and Evolutionary Computation Conference

[16] Thomas H. Cormen and Charles E. Leiserson and Ronald L. Rivest and Clifford Stein 2002 'Algoritmos Teoria e Pratica - Traducao da 2 Edicao Americana', Elsevier, 2Âª Edicao

[17] Jason Eisner 1997 'State-of-the-art algorithms for minimum spanning trees-a tutorial discussion', *Citeseer*

[18] Joshua Knowles and David Corne and Martin Oates 2000 'A New Evolutionary Approach to the Degree Constrained Minimum Spanning Tree Problem', *IEEE Trans. Evolutionary Computation*, v. 4, n. 2, pp.125-134

[19] Giorgio Ausiello 1999 'Complexity and approximation: Combinatorial optimization problems and their approximability properties', *Springer*

[20] Mohan Krishnamoorthy and Andreas T Ernst and Yazid M Sharaiha 2001 'Comparison of algorithms for the degree constrained minimum spanning tree', *Journal of heuristics, Springer*, v. 7, n. 6, pp.587-611

[21] Katherine Compton and Scott Hauck 2000 'An introduction to reconfigurable computing', *IEEE Computer*, April

[22] Joseph B. Kruskal 1956 'On the Shortest Spanning Subtree of a Graph and the Traveling Salesman Problem.', *Proceedings of the American Mathematical Society*, v.7, n.1, 48-50, February

[23] Guenther R. Raidl and Bryant A. Julstrom 2003 'Edge sets: an effective evolutionary coding of spanning trees.', *IEEE Transactions on*, Evolutionary Computation, v.7, n.3, 225-239

[24] Gois, M. M., Matias, P., Perina, A. B., Bonato, V. and Delbem, A. C. B. (2014). 'A Parallel Hardware Architecture based on Node-Depth Encoding to Solve Network Design Problems.' *International Journal of Natural Computing Research (IJNCR)*, v. 4, n.1, 54-75.

Eliminating Cognitive Ambiguity with Knowledge Relativity Threads

James A. Crowder, John N. Carbone
Raytheon Intelligence, Information, and Services
16800 E. Centretech Parkway, Aurora, Colorado 80011

Abstract - *Today, systems and humans continue to struggle with satisfying the desire to obtain the true essence of, and actionable knowledge from, an ever increasing and inherently duplicative, non-context specific, multi-disciplinary information content. Continuous improvement via increased automation has been the engineering norm for centuries. Now, automation has evolved to autonomous systems research and is our immediate future. Humans are expanding exploration within ever challenging environments generally unfriendly to the physical human condition. Simultaneously, the volume, velocity, variety, and complexity of systems and data continue to increase rapidly. However, development of valuable readily consumable knowledge and context quality continues to improve more slowly and incrementally. New concepts, mechanisms, and implements are required to facilitate the development and competency of complex systems to be capable of discovering the essence of ambiguities during autonomous operation, self-healing, and critical management of internal knowledge economies. They require ever-increasing fidelity of self-awareness of their real-time internal and external operational environments. Presented here are proposed solutions for defining, enhancing, sharing and communicating the essence of inherently ambiguous n-dimensionally related information content and new infrastructure to drive systems towards self-production, presentation, and representation of reliable context-specific knowledge development.*

Keywords: Knowledge Essence, Artificial Reliability, Self-Evolvable Systems, Knowledge Relativity Threads, Physical Mechanics, Metacognition.

1. Introduction

Improving decision quality autonomy of artificially intelligent systems requires reliable information discovery, decomposition, reduction, normalization, and context specific knowledge recall (Carbone 2010). Hence, capturing the essence of any given set of information content is paramount. When describing how science integrates with information theory, Brillouin (Brillouin 2004) defined knowledge as resulting from exercising thought. Knowledge was mere information without value, until a choice was made based upon thought. Additionally, Brillouin concluded that a hundred random sentences from a newspaper, or a line of Shakespeare, or even a theorem of Einstein have exactly the same information value. He concluded that information content had "no value" until it had been thought about and turned into knowledge. Artificially infused robotic systems must be able to integrate information into their Cognitive Conceptual Ontology (Crowder, Taylor et al. 2012) in order to be able to "think" about, correlate and integrate information. Humans think to determine what to do or how to act. It is this decision-making that can be of great concern when processing ambiguity because of the sometimes serious ramifications which occur when erroneous inferences are made. Often there can be severe consequences when actions are taken based upon incorrect recommendations. Inaccurate inferences can influence decision-making before they can be detected or corrected. Therefore, the underlying challenge of our day is to reliably understand the essence of a situation, action, activity, and to significantly increase capability and capacity to make critical decisions from a complex mass of real-time information content. Harnessing actionable knowledge from these vast environments of exponentially growing structured and unstructured sources of rich interrelated cross-domain data is imperative, (Llinas, Bowman et al. 2004) and a major challenge for autonomous systems that must wrestle with context ambiguity without the advantage of human intervention.(Crowder, Carbone et al. 2014) The paper corpus comprises transdisciplinary focused information theory and discusses combining application of cognitive psychology, computational linguistics, calculus, computer science, and mature physical geometric mechanics for defining new systems with vital adaptive presentation and storage representation capabilities along with analytics for enhancing understanding of ambiguous characteristics using Knowledge Relativity Threads (KRT) (Carbone 2010).

We describe qualitative cognitive creativity value(s) in terms of essence quality of capture rooted in mature physical calculus; weight, state, and time via Knowledge Relativity Threads (KRT). Our contention is that these are missing from today's systems and imperative to facilitate reliable real-time cognition-based information content discovery to knowledge construction, for ultimately providing, not just enhanced, but credible decision making for human-system collaboration and autonomous self-evolvable systems.

2. The Essence of Meaning

Dourish (Dourish 2004) expressed two notions of context, technical, for conceptualizing human action relationships between the action and the system, and social science, where he reported that "ideas needed to be understood in the intellectual frames that give them meaning." Efforts deriving frames of reference have historically been focused on understanding the essence of relationships between objects in physical environments. Standard space/time, strong/weak, and quantum mechanics analysis use mature calculus to describe processes, weighted relations and physical states as they relate to time. (Hibbeler 2009) Similarly, humans continuously develop frames of reference, comparisons, and prototypical associations as they apply cognitive processes to solve problems and interact and interpret their daily environment.(Gärdenfors 2004) These processes are well researched. Notably, in 1957, Newell et al. (Newell, Shaw et al. 1957) and Simon (Simon 1961) together developed models of human mental processes and produced General Problem Solver (GPS) to perform "means-end analysis" to solve problems by successively reducing the difference between a present condition and end goal. GPS organized knowledge into symbolic objects with related contextual information which were systematically compared and stored. Almost a decade later human cognition researcher Sternberg (Sternberg 1966) described a now well-known paradigm where, participant observations were taken during focused experiments to determine how quickly a human participant could compare and respond to answers based upon experimentally varying the size and ambiguity of given knowledge. Sternberg Paradigm is known for (1) organizing knowledge and modifying context while using a common process for describing the nature of human information processing and (2) human adaptation based upon changes in context. More recently, Rowley and Hartley (Rowley and Hartley 2008) described the development of knowledge as the organization and processing required to convey understanding, accumulated learning, and experience. Polyn and Kahana's (Polyn and Kahana 2008) research contends that recall of a known item representation is driven by an internally maintained context representation. They described how neural investigations had shown that the recall of an item represented in the mind is driven by an internally maintained context representation that integrated information with a time scale. Therefore, we will describe generation of practical humanistic based structural representations of meaning in the form of time, context, prototypical objects and relations.

3. Physical Representation of Meaning

Research shows that the community of disciplines researching how humans generate knowledge has traditionally focused upon how humans derive meaning from interactions and observations within their daily environments, driving out ambiguity to obtain thresholds of understanding. With similar goals, Information Theory, and Complexity Theory focus more closely on the actual information content. Zadeh pioneered the study of mechanisms for reducing ambiguity in information content, informing us about concepts in "fuzzy logic" and the importance of granular representations of information content,(Zadeh 2004) and Suh focused upon driving out information complexity via the use of Axiomatic design principles.(Suh 2005) Hence, a vast corpus of cognitive related research continually prescribes one common denominator, ***representation*** of how information content, knowledge and knowledge acquisition should be modeled. Gordenfors (Gärdenfors 2004)acknowledges that this is the central problem of cognitive science and describes three levels of representation: *symbolic* - Turing machine like computational approach, *associationism* – different types of content relationships which carry the burden of representation, and thirdly, ***geometric*** – *structures which he believes best convey similarity relations as multi-dimensional concept formation in a natural way*; learning concepts via similarity analysis has proven dimensionally problematic for the first

two and is also partially to blame for the continuing difficulties when attempting to derive actionable intelligence as content becomes increasingly distended, vague, and complex.

Historically, there are many examples and domains, which employ concepts of conceptual representation of meaning as geometric structures. (e.g. cognitive psychology (Suppes et al. 1989), cognitive linguistics (Langacker, 1987)(Lakoff, 1987),(Talmy, 1988), (Taylor & Raskin, 2008), transdisciplinary engineering(Carbone 2010), knowledge storage(Crowder and Carbone 2011), computer science(entity relationship, sequence, state transition, and digital logic diagrams), Markov chains, neural nets and many others. It should be noted here that there is not one unique correct way of representing a concept. Additionally, concepts have different degrees of granular resolution as Zadeh (Zadeh 2004) describes in fuzzy logic theory. However, geometric representations can achieve high levels of scaling and resolution (Gärdenfors 2004) especially for n-dimensional relations, generally difficult if not impossible to visualize above the 4th dimension. However, high dimensionality can be mathematically represented within systems in a number of ways. Hence, mature mathematics within the physical domain allows this freedom. Therefore, we show the overlay of physics based mathematical characteristics to enhance relational context and develop a unifying underlying knowledge structure within Information Theory. We employ Knowledge Relativity Threads (KRT)(Carbone 2010) to minimize ambiguity by developing detailed context and for conveying knowledge essence simply and robustly. The next section describes presentation formation, representation, and the process of organization of n-dimensional contextual relationships for humanistic prototypical object data types with application of the common denominators: time, state, and context.

4. Knowledge Relativity (KR)

The premise of Knowledge Relativity Threads (KRT) (Carbone 2010) primarily originates from computational physics concepts as an analogy to Hibeller [51] where the concept of relating the motion of two particles is as a frame of reference and is measured differently by different observers. Different observers measure and relate what they behold, to a context of what has been learned before and what is being learned presently. The reference frame of each knowledge building action contains the common denominators of time, state, and context; the point in time and all the minutia of detailed characteristics surrounding and pronouncing the current captured state of all related context. Historically, organization, presentation, representation of knowledge and context has been researched across many disciplines (.e.g. Psychology, Computer Science, Biology, Linguistics etc.) because of the primal need to survive, understand, and make sense of a domain. However, most systems we engineer today are increasingly incapable of processing, understanding, presenting, and structurally representing the volume, velocity, variety, and complexity of content because first, they are not built to learn, only to store,(Ejigu, Scuturici et al. 2008) and second, the content systems store and filter are what is generally or explicitly known to be true, not the more valuable and higher fidelity tacit knowledge that is context specific to each frame of reference or situation. (Nonaka 2002)

Therefore, we build KRTs upon the concept of "Occam Learning" (Kearns and Vazirani 1994) to construct continually renegotiable systems (Gruber 2008) with the simplest portrayal (e.g. present, represent) capable of encapsulating complex n-dimensional causal structures, within and between the complex data generated from the observed/captured behavior.(Crowder and Carbone 2011)

4.1 Knowledge Organization

System core components historically have included Organization, Presentation, and Representation for collecting, processing, visualizing, and storing information content. To develop more dynamic systems it is not enough to simply collect, filter and store data as most systems do today. Additionally, engineers frequently develop systems for a user base with little conceptual insights into actual operations. Therefore, for decades system organization has consistently required significant effort, subject matter expertise, and highly creative engineers who have the right combination of artistic and scientific backgrounds to successfully bridge the ambiguities and the domains. We contend that simple changes in perspective could cause a paradigm shift in how valuable and complimentary systems could be for the human condition. (e.g. synchronous-asynchronous,

serial-recursive). We propose the following existing perspectives:

1) Humans cognitively process thousands of pieces of sensory information simultaneously
2) Cognition helps humans dynamically adapt to change
3) Increasing the amount of work that systems perform, maximizes time
4) Increasing the amount of work that systems perform that humans CAN'T, minimizes human risk
5) The apex of system performance benefits for humans are in degrees of autonomy

We propose three simple new perspectives which could govern organization of a valuable autonomic systems cognitive approach:

1) Increasing the autonomic nature of a system is directly proportional to the cognitive versatility of a systems internal humanistic architecture
2) Increasing the amount of cognitive recursion is directly proportional to inference potential
3) Increasing a cognitive based system's asynchronous operation maximizes performance

Therefore, the key ingredients for improving system Organization are: Human Cognition based architecture, recursion, and higher levels of asynchronous operations.

4.2 Knowledge Presentation

Presented here are proposed solutions for enhancing sharing and communicating the essence of inherently ambiguous n-dimensionally related information content and new infrastructure to drive systems towards production, presentation, and representation of reliable context-specific knowledge development. We describe qualitative cognitive creativity value in terms of essence quality of capture rooted in mature physical calculus; weight, state, and time via Knowledge Relativity Threads (KRT).

5. Knowledge Domain Essence Thresholds

One aspect of Human Cognition and Knowledge Organization that must be folded into any qualitative artificial architecture is the notion of inference potential. Internally, humans, usually without realizing it, utilize the concepts of knowledge relativity, knowledge density (how much they know about a subject or group of subjects), and analytical competency (do I understand the domain well enough), to derive a sense of inference potential about a subject or situation they are currently dealing with (i.e., can I effectively form coherent thoughts and come to actionable decisions) (Crowder, Carbone, and Friess 2013).

In order to mimic real-time human decision making processes, an effective cognitive processing systems must be supported by information derived from the fusion process and must operate in a uniform and cooperative model, fusing data into information and knowledge so the system can make informed decisions (Crowder and Friess 2013). One such construct that aids in this assessment is the measure of a system's ability to provide quality information and/or inference about a particular subject or question.

5.1 Knowledge Density: a Pathway to Artificial Metacognition

In order for an artificial cognitive system to be able to provide Qualitative Cognitive Creativity (QCC) we must provide the system with the ability to understand its own limitations and capabilities and to reason about them in light of the duties or missions it is given. In humans, we call this ability "Metacognition." Metacognition in humans refers to higher order thinking which involves active control over a SELF's cognitive processes engaged in learning and performing. Activities such as planning how to approach a given task, monitoring comprehension, and evaluating progress toward the completion of a task are metacognitive in nature [Crowder and Friess 2012, Crowder and Carbone 2012).

Qualitative Cognitive Creativity (QCC) requires different kinds of metacognitive awareness: declarative, procedural, and conditional knowledge:

- **Declarative Knowledge**: refers to knowing "about" things,
- **Procedural Knowledge**: refers to knowing "how" to do things, and
- **Conditional Knowledge**: refers to knowing the "why" and "when" aspects of cognition.

Here we classify QCC into three components:

- **Metacognitive Knowledge**: (also called metacognitive awareness) is what the system knows about itself as a cognitive processor (Crowder and Friess 2011)].
- **Metacognitive Regulation**: is the regulation of cognition and learning experiences through a set of activities that help the system control its learning (Crowder and Friess 2012). This may be based on its understanding of its own "knowledge gaps."
- **Metacognitive Experiences**: are those experiences that have something to do with the current, on-going cognitive endeavors (current mission).

In order to achieve the QCC metacognitive abilities, the system must have the ability to measure its own knowledge about a particular topic or subject. As explained earlier, this is a measure of topical or subject knowledge, and involves measuring the "***density***" of knowledge the system possesses about this subject or topic in question. This ***Knowledge Density*** measure is based on the number of separate information fragments relative to the taxonomy of the topic or subject. Figure 1 provides the Knowledge Density Measure, based on separable topical information fragments (Crowder 2013).

We use knowledge fragment measurements to ensure that we only store information relative to a topic or subject once. Information that is taken in is parsed and information fragments that have not been stored before are pulled out and stored in a cognitive map for that topic.

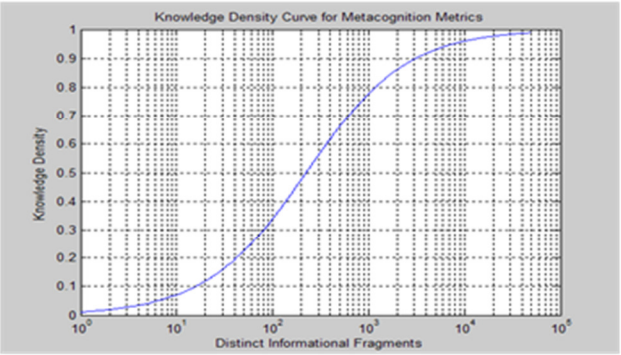


Figure 1 - Knowledge Density Computation

Renyi's entropy measurement is utilized to separate information into topical information fragments. Renyi's entropy measurement is defined as [Crowder, Carbone, and Friess 2013):

$$H_R(Y) = -\log \int_y p(y)^2 \, dy \qquad (1)$$

Computationally, this is difficult, however, Renyi's measure, combined with the Parzen Density estimation method provides a computational model. We start by looking at the information densities, *p(y)*, as a sum of related topical cognitive maps, each centered at y_i, we get:

$$p(y) = \frac{1}{N} \sum_{i=1}^{N} G(y - y_i, \sigma I) \qquad (2)$$

Therefore, Renyi's entropy can be computed as the sum of local information interactions (separate information fragments) over all pairs of informational entities. We utilize Fuzzy Topical maps to compute the components of the Knowledge Density measure. The use of Fuzzy Topical Maps to compute the Knowledge Density is done because:

- It's robust in the presence of inexact information.
- It utilizes conditional possibilistics
  - Mutual Information measurement
  - Joint Informational membership rather than joint probabilities
- Excellent at showing qualitative relationships not attainable with Bayesian methods
  - Excellent at showing qualitative relationships not attainable with Bayesian methods
  - Creates decisions with conditional possibilistic attributes
- More useful with general questions about a subject domain

6. Conclusions

Much more research is needed to validate this work and produce an automated way to compute Knowledge Density, Inference Potential, and, in the end, Qualitative Cognitive Creativity. This work is also dependent on further research work on subjects such as Artificial Neural Emotions, Metcognitive and Metamemory constructs, as well as further work on the overall cognitive processing structures. The purpose of this work is to provide a framework as research continues, for cognitive systems to provide meaningful knowledge disambiguate and knowledge self-assessment.

7. References

1. Brillouin, L. 2004. Science and information theory. *Dover*.
2. Newell, A. 2003. Unified Theories of Cognition. *Cambridge MA: Harvard University Press*.
3. Crowder, J. A. 2010a. The Continuously Recombinant Genetic, Neural Fiber Network. *Proceedings of the AIAA Infotech@Aerospace-2010*, Atlanta, GA.
4. Crowder, J. A. 2010b. Anti-Terrorism Learning Advisory System (ATLAS): Operative Intelligent Information Agents for Intelligence Processing. *Proceedings of the AIAA Infotech@Aerospace-2010*, Atlanta, GA.
5. Crowder, J. A., 2010c. Flexible Object Architectures for Hybrid Neural Processing Systems. *Proceedings of the 11th International Conference on Artificial Intelligence*, Las Vegas, NV.
6. Crowder, J. A., and Carbone, J. 2011a. Recombinant Knowledge Relativity Threads for Contextual Knowledge Storage. *Proceedings of the 12th International Conference on Artificial Intelligence*, Las Vegas, NV.
7. Crowder, J. A., and Carbone, J. 2011b. Transdisciplinary Synthesis and Cognition Frameworks. *Proceedings of the Society for Design and Process Science Conference 2011*, Jeju Island, South Korea.
8. Crowder, J. and Friess S. 2012. Artificial Psychology: The Psychology of AI. *Proceedings of the 3rd International Multi-Conference on Complexity, Informatics, and Cybernetics*, Orlando, FL
9. Crowder, J. 2012a. Cognitive System Management: The Polymorphic, Evolutionary, Neural Learning and Processing Environment (PENLPE). *Proceedings for the AIAA Infotech@Aerospace 2012 Conference*, Garden Grove, CA.
10. Crowder, J. 2012b. The Artificial Cognitive Neural Framework. *Proceedings for the AIAA Infotech@Aerospace 2012 Conference*, Garden Grove, CA.
11. Crowder, J., Raskin, V., and Taylor, J. 2012. Autonomous Creation and Detection of Procedural Memory Scripts. *Proceedings of the 13th Annual International Conference on Artificial Intelligence*, Las Vegas, NV.
12. Carbone, J. (2010). A Framework for Enhancing Transdisciplinary Research Knowledge, Texas Tech University Press.
13. Crowder, J. and J. Carbone (2011). Occam learning through pattern discovery: Computational Mechanics in AI Systems. Proceedings of the 13 th annual International Conference on Artificial Intelligence.
14. Crowder, J. and J. Carbone (2011). Recombinant Knowledge Relativity Threads for Contextual Knowledge Storage. Proceedings of the 13 th annual International Conference on Artificial Intelligence.
15. Crowder, J., J. N. Carbone, et al. (2014). Artificial Cognition Architectures, Springer.
16. Crowder, J. A., J. M. Taylor, et al. (2012). Autonomous Creation and Detection of Procedural Memory Scripts. Proceedings of the 13th Annual International Conference on Artificial Intelligence.
17. Raskin, V., Taylor, J. M., & Hempelmann, C. F. 2010. Ontological semantic technology for detecting insider threat and social engineering. *New Security Paradigms Workshop*, Concord, MA.
18. Rosenblatt F. 1962. Principles of Neurodynamics. *Spartan Books*.
19. Scally, L., Bonato M., and Crowder, J. 2011. Learning agents for Autonomous Space Asset Management. *Proceedings of the Advanced Maui Optical and Space Surveillance Technologies Conference*, Maui, HI.
20. Taylor, J. M., & Raskin , V. 2011. Understanding the unknown: Unattested input processing in natural language, *FUZZ-IEEE Conference*, Taipei, Taiwan.
21. Dourish, P. 2004a. Where the action is: The foundations of embodied interaction. *The MIT Press*.
22. Dourish, P. 2004b. What we talk about when we talk about context. *Personal and ubiquitous computing, vol. 8, pp. 19-30*.
23. Torralba, A. 2003. Contextual priming for object detection. *International Journal of Computer Vision, vol. 53, pp. 169-191*.
24. Dey, A. 2001. Understanding and using context. *Personal and ubiquitous computing, vol. 5, pp. 4-7*.
25. Coutaz, J., Crowley, J., Dobson, S., and Garlan, D. 2005. Context is key. *Communications of the ACM, vol. 48, pp. 53*.
26. Winograd, T. 2001. Architectures for context. *Human-Computer Interaction, vol. 16, pp. 401-419*.
27. Hong, J. and Landay, J. 2001. An infrastructure approach to context-aware computing. *Human-Computer Interaction, vol. 16, pp. 287-303*.
28. Howard, N. and Qusaibaty, A. 2004. Network-centric information policy. *Proceedings of the Second International Conference on Informatics and Systems*.
29. Ejigu, D., Scuturici, M., and Brunie, L. 2008. Hybrid approach to collaborative context-aware service platform for pervasive computing. *Journal of Computers, vol. 3, pp. 40*.

Genetic Algorithm Based Optimization for Energy-aware Hybrid Flow Shop Scheduling

Jihong Yan, Jing Wen, Lin Li
School of Mechatronics Engineering, Harbin Institute of Technology, Harbin 150001, PR China

Abstract - *Based on MATLAB/SimEvents simulation platform and genetic algorithm, the simulation and optimization of energy consumption in hybrid flow shop is studied. Considering workshop layout and workpiece path, machining process simulation in hybrid flow shop is realized. With the processing time, processing power and stand-by power introduced into simulation model, the energy consumption simulation in hybrid flow shop is achieved. Genetic algorithm based weighted optimization method is employed to optimize the energy consumption simulation model, which takes the minimum makespan and minimum energy consumption as optimization objectives. Finally, a case study is performed to verify the effectiveness of the simulation model and optimization method, and analysis on optimization results is carried out.*

Keywords: genetic algorithm; hybrid flow shop; SimEvents; energy consumption; simulation; optimization

1 Introduction

In recent years, the proportion of energy consumption in manufacturing is up to 30-50% of the total energy consumption all over the world [1]. Wasting energy in workshop leads directly to increasing production costs and severe environmental pollution. Simulation and optimization of energy consumption in hybrid flow shop can enable the workshop to improve the energy efficiency of machine tools and show benign to environments.

Researches on workshop energy consumption are focused on mathematical modeling and scheduling optimization [2]. Dai et al. [3] established a mathematical model of energy consumption in flexible manufacturing system, and an improved genetic-simulated annealing algorithm is adopted to make a significant trade-off between the makespan and the total energy consumption to implement a feasible scheduling. He et al. [4] optimize the energy consumption and makespan by locating the optimal or near optimal solutions of the bi-objective model based on the Tabu search mechanism. Most of the existing researches, which concerning the issue of optimizing energy consumption for the workshop scheduling problem, are implemented on the basis of mathematical model. Compared to simulation model, mathematical model is a simplified model that far from reality. Therefore, it is necessary to propose a simulation model which can both realize the energy consumption simulation in workshop and interact with artificial intelligence.

Workshop is a typical Discrete Event Dynamic System(DEDS), whose simulation model can be built on platforms such as ProModel、Flexsim、eM-Plant、WITNESS [5,6]. There are many researchers use Stateflow simulation module integrated in MATLAB environment for DEDS simulation [7,8]. Stateflow is a simulation platform based on finite state machine theory, aiming to display the transfer between different states without data transmission. SimEvents is a simulation toolbox provided by MATLAB for discrete event simulation. Compared to Stateflow, SimEvents is more suitable for hybrid flow shop simulation because it can simulate the entity flow process in system. In addition, the interaction among SimEvents, Simulink and Stateflow can be achieved in MATLAB environment [9]. With the excellent numerical processing functions in MATLAB, analysis and optimization based on the SimEvents simulation model is efficient.

In this paper, SimEvents based energy consumption simulation and optimization method is studied Genetic algorithm is employed to optimize both makespan and energy consumption in the simulation environment. In addition, an optimization example is conducted to validate the effectiveness of the simulation model and optimization method.

2 Hybrid flow shop simulation

2.1 Hybrid flow shop layout simulation

Hybrid flow shop is more complex than flow shop because of its parallel machine tools in some processing stages. SimEvents based physical layout model of hybrid flow shop is shown in Fig. 1. Each *Entity Generator* block produces a kind of workpiece sharing the same processing route and parameters. *FIFO Queue* blocks and *Server* blocks represent buffers and machine tools in workshop respectively. As shown in Fig.1, W kinds of workpieces wait to be processed and all the workpieces pass the P processing stages in sequence. All the workpieces are expected to be processed by one of the M_n ($M_n \geq 1$) machine tools in the nth ($n=1,2,\ldots,P$) stage

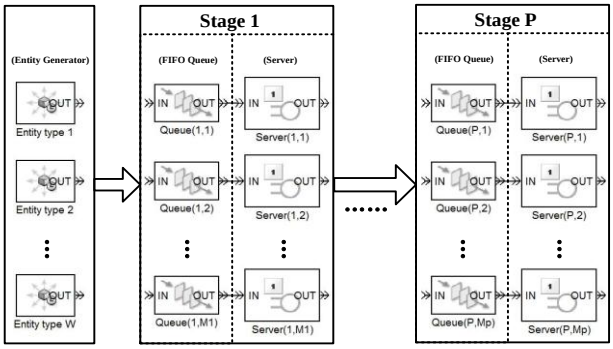


Figure 1: Simulation model of hybrid flow shop layout

2.2 Logic control simulation modeling

Different workpieces have different processing routes. Processing time in a machine tool varies with workpieces type. Therefore, simulation in logic control level aims to realize processing route control of entities and service time control of servers.

By setting attributes for entities can the logic control and service time control be achieved. Fig.2 shows an overview of the block diagram. *Set Attribute* block accepts an entity, assigns data to it, and then outputs it. Assigned data is stored in attributes of the entity, where each attribute has a name and a value. Take the *Entity type* 1 block in Fig.2 as an example, *Set Attribute*_1 block is placed behind of it with attributes named *type*, *P*1 and *PT*1 which represent workpiece type, the number of the machine and the service time in the first stage. *Path Combiner* is a block designed to merge multiple paths into a single path. The entities then follow a merged path but might be treated differently later based on their individual attribute values. The *Output Switch* block selects one among a number of entity output ports on the basis of attribute *P*1. Service time of *Server* block depends on attribute *PT*1.

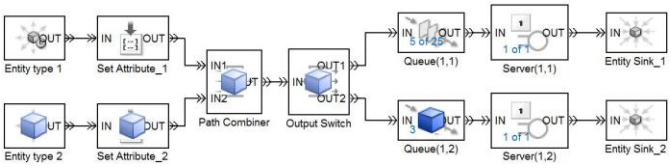


Figure 2: Logic control in hybrid flow shop

2.3 Energy consumption simulation modeling

Hybrid flow shop is mainly composed of machine tools, its energy consumption simulation model can be built by simulating energy consumption of each machine tool . The simulation methods change with the operating states of machine tools which include waiting state and processing state. Assume that power of machine tools is a constant that equals to stand-by power during waiting state. Waiting time is unknown in advance and decided by operating states of machine tools. Simulation model of processing state can be built with average processing power and processing time which varies with different workpieces. To sum up, energy consumption simulation model of hybrid flow shop is based on the stand-by power, average processing power and processing time. Stand-by power can be obtained by measurement. Average processing power and processing time are obtained based on preprocessing of power profile data.

As an example, the power consumption of workpiece *i* and workpiece *j* (*i*, *j*=1,2,⋯,*N*,*i*≠*j*) when they pass through Server(*n*,*k*) is represented graphically in Fig. 3. The processing time are presented as *PT*(*n*,*k*,*i*) and *PT*(*n*,*k*,*j*).

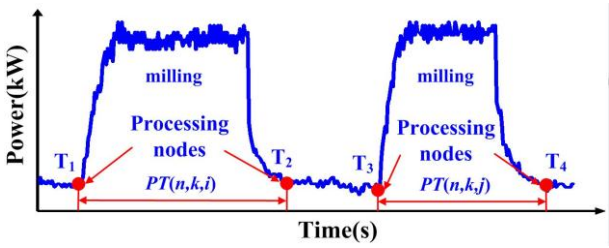


Figure 3: Power profile of a milling process

Eq.(1) shows the average processing power of workpiece *i*.

$$PP(n,k,i) = \int_{T_1}^{T_2} p(t)dt \,/\, PT(n,k,i) \qquad (1)$$

Take *Server*(1,1) block as an example, energy consumption simulation method is presented. With a *MATLAB Function* block, a MATLAB function is available in a Simulink model and energy consumption simulation can be achieved. Set parameters for *MATLAB function* block as indicated in Fig.4. If the *state* port input 0, server is in waiting state and block outputs stand-by power. When *state* port input 1, which means the server is in working state, check the *type* port and output average processing power of corresponding workpiece type. The *Integrator* block outputs the integral of power at the current time. Power information and energy consumption are shown in real time in *Scope* 1 and *Scope* 2 respectively.

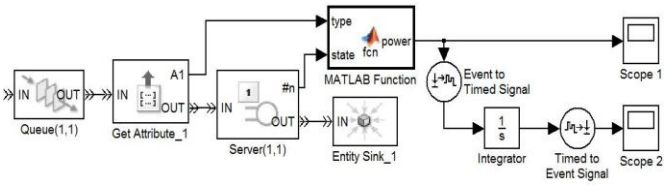


Figure 4: Energy consumption simulation model of machine tools

3 Genetic algorithm based optimization

3.1 Decision variable

There are *N* jobs waiting to be processed, and each job must experience *P* stages. A reasonable scheduling scheme is shown as matrix $A_{N\times P}$. Take the second row as example, the second workpiece will be processed in machine M_{21}, M_{22}, … , M_{2P} successively.

$$A_{N\times P}=\begin{bmatrix} M_{11} & M_{12} & \cdots & M_{1P} \\ M_{21} & M_{22} & \cdots & M_{2P} \\ \vdots & \vdots & \ddots & \vdots \\ M_{N1} & M_{N2} & \cdots & M_{NP} \end{bmatrix} \quad (2)$$

A chromosome is composed of N segments, and each segment is a row in the matrix contains P genes. A chromosome can be expressed as:

$$chromosome=[M_{11}\,M_{12}\cdots M_{1P}\,M_{21}\,M_{22}\cdots M_{2P}\cdots\cdots M_{N1}\,M_{N2}\cdots M_{NP}]$$

3.2 Optimization function

Makespan and energy consumption are two optimization objectives with different significance and dimensions, which makes bi-objective optimization difficult. Normalized weighted method is an effective way to solve multi-objective problem by assigning each optimization object with fixed weight. Optimization function for searching the minimum value of makespan and energy consumption weighted sumation is shown as Eq.(2).

$$\begin{aligned} &opt. \\ &\min(SumCost)=\min((1-\alpha)\,f(SumTime) \\ &\qquad\qquad +\alpha\,f(SumEnergy)) \end{aligned} \quad (3)$$

Where α donates the weighted factor of energy consumption. *f*(*SumTime*) and *f*(*SumEnergy*) are normalized function which can be calculated by Eq.(3) and Eq.(4).

$$f(SumTime)=\frac{SumTime-minTime}{maxTime-minTime} \quad (4)$$

$$f(SumEnergy)=\frac{SumEnergy-minEnergy}{maxEnergy-minEnergy} \quad (5)$$

3.3 Optimization procedure

Genetic algorithm is a heuristic searching algorithm, imitating genetic choice and natural elimination to search individual in solution space. The major procedures of genetic algorithm are population initialization, parent population selection, crossover, and mutation. The process of evolution stops as soon as the termination clause is met.

(1) Coding matrices are randomly generated, representing different scheduling schemes. Chromosomes are obtained based on the coding matrices and the initial populations are created.

(2) The criterion for parent population selection is fitness value. Fitness value of each parent population can be computed by using Eq.(5).

$$FitValue=\exp(-scale\cdot SumCost) \quad (6)$$

Where *scale* is scale factor. Elite section is adopted to protect fittest individual, and roulette wheel selection is adopted to create new population.

(3) Two-point crossover method is employed in this paper. The crossover individuals and points are selected randomly. Crossover probability is set to determine whether this pair of individuals will implement crossover operation.

(4) The key operation in mutation procedure is confirming mutant genes. Array T is designed to have the same length with chromosome in [0,1]. If any of the array element is less than mutation probability, its corresponding gene will be replaced by another reasonable gene.

4 Optimization example

4.1 Experimental design

There are 4 different workpieces waiting for being processed, and every workpiece must experience 3 stages at the same direction. The number of machines at each stage is 2, 2 and 3. Energy consumption data can be obtained by collecting and analyzing power profile of all the machine tools. Processing time, average processing power and stand-by power are shown in Table 1 and Table 2 respectively.

Table 1: Processing time of optimization example

Time(s)	Stage 1		Stage 2		Stage 3		
	M1	M2	M3	M4	M5	M6	M7
Job1	96	120	133	107	144	144	115
Job2	96	120	112	90	204	204	163
Job3	96	120	90	72	353	353	282
Job4	96	120	122	98	403	403	323

Table 2: Power of optimization example

Power	Stage 1		Stage 2		Stage 3		
(kW)	M1	M2	M3	M4	M5	M6	M7
Job1	3.12	2.17	4.02	5.19	1.49	1.65	2.31
Job2	3.12	2.17	4.02	5.19	1.49	1.65	2.31
Job3	3.12	2.17	4.02	5.19	1.34	1.48	2.08
Job4	3.12	2.17	4.02	5.19	1.41	1.56	2.19
Wait	0.41	0.74	0.42	0.30	0.41	0.42	1.02

4.2 Energy consumption simulation

Energy consumption simulation model of the optimization example, as shown in Fig.5, is mainly composed of *Entity Generator* subsystem and *Machine* subsystems. Worpiece flow is controlled by *Set Attribute* block in *Entity Generator* subsystem and attribute value of each workpiece is determined by the scheduling scheme from genetic algorithm. Each machine tool corresponds to a *Machine* subsystem whose simulation model details are shown in Fig.4.

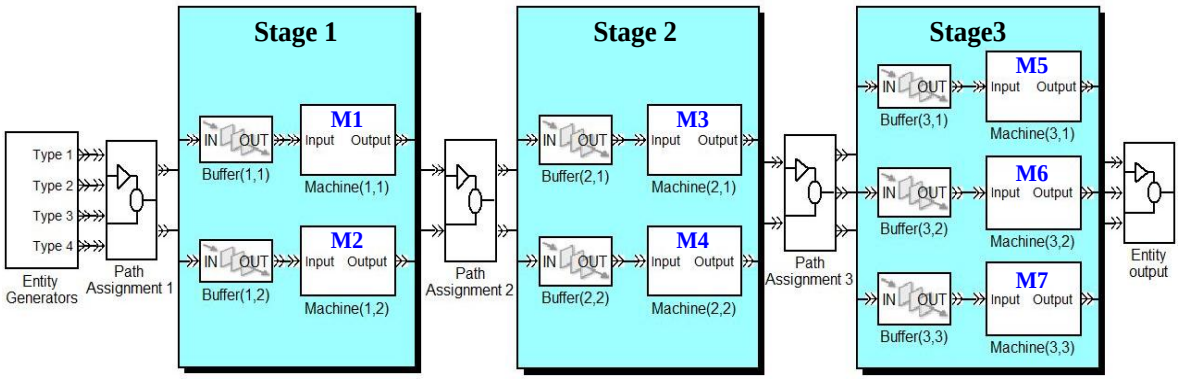


Figure 5: Energy consumption simulation model

4.3 Genetic operators

(1) Coding and decoding. In the case of 4 jobs on 3 stages, each chromosome consists of 4×3 genes. A scheduling scheme is shown as *chromosome*1:

*chromosome*1:[1 3 6 2 4 6 1 4 5 2 4 7]

The first three gene '1 3 6' donates the first workpiece is processed in M1, M3, M6 in different stages.

(2) Crossover. Take *chromosome*1 and *chromosome*2 as example, two-point crossover method is employed to obtain chromosome1*'* and chromosome2*'*.

chromosome:[1 3 6‖2 4 6 1 4‖5 2 4 7]

chromosome:[2 3 7‖2 3 7 1 3‖5 1 4 6]

*chromosome*1′:[1 3 6‖2 3 7 1 3‖5 2 4 7]

*chromosome*2′:[2 3 7‖2 4 6 1 4‖5 1 4 6]

(3) Mutation. Mutation value varies from stage to stage. In stage 1 mutation value is set to 1 or 2, while in stage 3 the mutation value is randomly geneated among 5, 6 and 7. Take *chromosome*1*'* as example, genes underlined are selected to have mutation operation , then, *chromosome*1*''* is obtained.

*chromosome*1′:[1 3 $\underline{6}$‖2 $\underline{3}$ 7 1 3‖5 $\underline{2}$ 4 7]

*chromosome*1″:[1 3 $\underline{5}$‖2 $\underline{4}$ 7 1 3‖5 $\underline{1}$ 4 7]

4.4 Optimization results and discussion

The values of parameters in genetic algorithm are as follows:

Size of population: 20;

Number of evolutional generation: 100;

Scale factor: 0.01;

Crossover probability: 0.8;

Mutation probability: 0.05.

The optimization results, as shown in Table 3, are obtained based on SimEvents simulation model (Fig.5) and genetic algorithm. E, PP, WP, T represent total energy consumption, processing energy consumption, waiting energy consumption and makespan respectively. Convergence curve (Fig.6) indicates that optiaml solution can be obtained in the 40th generation when the optimization objective is makespan.

Table3: Optimization results of genetic algorithm

Weight	E (kJ)	PP(kJ)	WP(kJ)	T (s)
0	4758.59	4751.39	7.20	612.48
0.5	4609.35	4591.34	18.01	975.38
1	4445.93	4435.12	10.53	1356.78

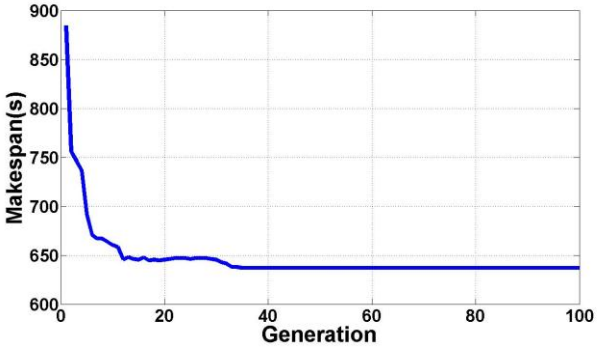


Figure 6: Convergence curve (optimization objective is makespan)

Fig.7 indicates the optimization results when weight of energy consumption is 1, which means the optimization objective is energy consumption and all the four workpieces are processed on M2, M3 and M5. Take machine tools in stage 3 as example, the processing power of M5 is much less than M6 and M7 as shown in Table 2, which means all the workpieces are prefer to be sent to M5 in stage 3 to obtain minimum energy consumption. If setting makespan as optimization object where the weight of energy consumption is zero, then the final optimal scheduling scheme can be shown in Fig.8, that presents a balanced distribution of four

workpieces in each stages without regard to energy efficiency of machine tools.

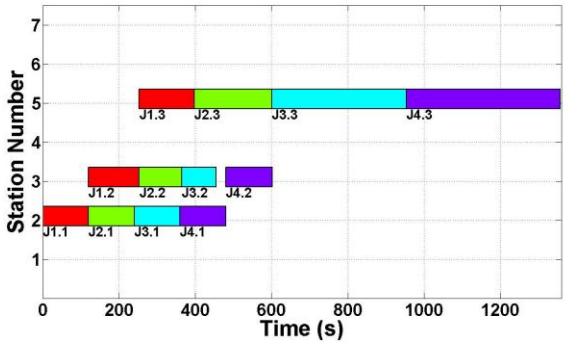


Figure 7: Optimization results of minimum energy consumption

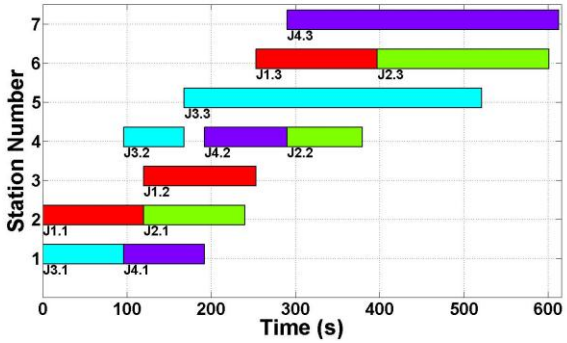


Figure 8: Optimization results of minimum makespan

Optimization results shows that the simulation model can simulate workpiece paths and power of all machine tools accurately. Simulation model can feedback results to genetic algorithm to optimize the scheduling schemes by taking the minimum makespan and minimum energy consumption as optimization objectives.

5 Conclusions

Based on SimEvents simulation platform, simulation models of facility layout, logic control and energy consumption information are established. Hybrid flow shop layout simulation adopts *Entity Generator*, *FIFO Queue* and *Server* blocks to simulate workpiece, buffer and machine tool respectively. Logic control simulation is achieved by setting attributes for entities. With the processing time, stand-by power and cutting power incorporated into simulation model, the energy consumption simulation in workshop is achieved. Genetic algorithm based weighted optimization method is employed to optimize the energy consumption simulation model. Finally, a case study is performed to verify the effectiveness of the simulation model and the optimization method.

Simulation model based optimization is practical but inefficient because of the simulation run-time. Future work will be conducted to improve operation efficiency of genetic algorithm by reducing the frequency of calling simulation model.

Acknowledgments

This research is funded by the National Natural Science Foundation of China (#71271068).

References

[1] Park C W, Kwon K S, Kim W B. Energy consumption reduction technology in manufacturing — A selective review of policies, standards, and research[J]. International Journal of Precision Engineering and Manufacturing, 2009, 10(5): 151-173.

[2] Salonitis K, Ball P. Energy efficient manufacturing from machine tools to manufacturing systems[J]. Procedia CIRP, 2013, 7:634-639.

[3] Dai M, Tang D B, Giret A, et al. Energy-efficient scheduling for a flexible flow shop using an improved genetic-simulated annealing algorithm[J]. Robotics and Computer-Integrated Manufacturing, 2013, 29(5):418-429.

[4] He Y, Liu F, Cao H J, et al. A bi-objective model for jobshop scheduling problem to minimize both energy consumption and makespan[J]. J. Cent. South Univ. Technol, 2005,12(2):167-171.

[5] Chen H F, Lu J X, Li Y D, et al. Simulation decision-making study of buffer area allocation in workshop layout of a discrete manufacturing enterprise[J]. Journal of Zhejiang University of Technology, 2010,03:246-250+256.

[6] Chen C L, Wang J X. Research on simulation optimization method of steam turbine blade workshop based on Witness[J]. Light industry machinery, 2010,04:116-118+122.

[7] Huang X M. Modeling and analysis for hybrid dynamic system of production line in integrating matlab environment[J]. Modular Machine Tool & Automatic Manufacturing Technique, 2013,05:9-11+15.

[8] Cheng S, Zhang H, Zhu Z H. Simulation and analysis of production line hybrid system under MATLAB[J]. Journal of system simulation, 2006,06:1649.

[9] A. Apte. Integrating Matlab, Simulink and Stateflow components to SimEvents models. http://www.mathworks.cn. 2012-04-15.

SESSION

BAYESIAN BASED METHODS AND APPLICATIONS

Chair(s)

TBA

A Bayesian Network Analysis of Eyewitness Reliability: Part 1

Jack K. Horner
PO Box 266
Los Alamos NM 87544
jhorner@cybermesa.com

ICAI 2014

Abstract

In practice, many things can affect the verdict in a trial, including the testimony of eyewitnesses. Eyewitnesses are generally regarded as questionable sources of information in a trial setting: cases that turn on the testimony of a single eyewitness almost never result in a guilty verdict. Multiple eyewitnesses can, under some circumstances, collectively exhibit more robust behavior than any witness individually does. But how reliable, exactly, are multiple eyewitnesses? The legal literature on the subject tends to be qualitative. Here I describe a highly idealized Bayesian network model of the relation between eyewitness behavior and trial verdict. In a companion paper, I describe a more refined Bayesian model of the same setting. It turns out that the highly idealized model provides nearly as much information as the more refined one does.

Keywords: eyewitness, Bayesian network

1.0 Introduction

In practice, many things can affect the verdict in a trial -- procedural conventions, material evidence, the psychology of the jurors, the persuasive power of the attorneys, and often, the testimony of eyewitnesses. Eyewitnesses are generally regarded as questionable sources of information in a trial setting ([3]): cases that turn on the testimony of a single eyewitness almost never result in a guilty verdict and are rarely brought to trial.

Multiple eyewitnesses can, under some circumstances, collectively exhibit more robust behavior than any witness individually does. But how reliable, exactly, are multiple eyewitnesses? The outcome of the recent trial of George Zimmerman, accused of second-degree murder or manslaughter of a teenager, rested heavily on the answer to this question ([10]). The legal literature on the subject tends to be qualitative (see, for example, [3]). A quantitative model is required.

Throughout, I will use the term *correct verdict* to mean a verdict that agrees with what actually happened, independently of the trial. I will use the term *verdict-determining-event* (VDE) to mean an event that could be witnessed by an eyewitness or that could contribute to a verdict.

2.0 A highly idealized Bayesian model

There is some correlation between whether a correct verdict is reached and whether witnesses correctly observe a VDE. If the witnesses accurately observe the VDE, the probability of a correct verdict is generally higher than if they don't observe the VDE accurately. Various estimates of the ratios of the probabilities in these two cases range from 2:1 to 3:1 ([3]).

A distinguishing feature of the relationship between the probability of a correct verdict and the accuracy of observation of the witnesses is that the probability of a correct verdict depends on the accuracy of the observation of the VDE.

This tells us that *conditionality* ([4], p. 23) is in play. In probability theory, conditionality can be captured as a *conditional probability*. A *conditional probability*, P(X|Y), "the probability of X, given Y" is a probability measure defined by ([4], Section 9.1)

$$P(X|Y) = P(X \cap Y) / P(Y)$$
Eq. 1

where

X and Y are sets
$\cap$ is set intersection.

We can model this problem in a simple Bayesian network (BN, [2]). The heart of a BN is Bayes Theorem ([4], p. 320), derivable from the probability axioms ([4], p. 23). In its simplest form, Bayes Theorem is

$$P(A|B) = P(B|A) \times P(A) / P(B)$$
Eq. 2

where

P(A) is the probability of A
P(B) is the probability of B
P(A|B) is the probability of A, given B
P(B|A) is the probability of B, given A

Bayesians view probability as *degree of belief*, and regard Eq. 1 as a statement about the relationship between the degree of belief in a hypothesis (A) and the degree of belief in evidence (B). In this view, the left-hand-side of Eq. 1 is the degree of belief in hypothesis A, given evidence B. P(B|A), to Bayesians, denotes the degree of belief in the evidence B, given the hypothesis and call P(B|A) in Eq. 1 a "prior probability", or more briefly, a "prior". I will use the term "prior" to denote the uninterpreted term P(B|A) in Eq. 1, but remaining agnostic about the Bayesian claim that probability is a measure of degree of belief.

We need only any three of the quantities in Eq. 1 to determine the remainder. For example, if we have values for each of

P(A) = unconditional probability of a correct verdict
P(B) = unconditional probability of an accurate observation
P(B|A) = the probability of an accurate observation, given a correct verdict

we can calculate P(A|B), the probability of a correct verdict, given an accurate observation.

A BN is a system of conditional probabilities conforming to Eq. 2, mapped onto a directed graph ([5]) of whose nodes represent random variables ([4], Section 3.1). A possible value of a random variable X is called a *state* of X. An edge between a node A and a node B means that the distribution of possible values (states) of B depends probabilistically on the distribution of possible values (states) of A.

For the sake of discussion, let's assume a highly idealized model of the relationship between eyewitness and verdict in which:

(SC)

- the verdict is determined solely by the testimony of three independent eyewitnesses who observe a VDE

- in the case of a correct verdict, each eyewitness has a probability of only 0.75 of correctly observing the VDE
- in the case of an incorrect verdict, each eyewitness has a probability of 0.25 of correctly observing the VDE
- in the absence of any observation, the probability of a correct verdict is 0.5

Figure 1 shows a graphical user-view of a BN, *SimpleThreeWitness* (*STW*), that satisfies (SC) and is implemented in [1].

Each box in Figure 1 represents a random variable of a system. An arrow from a box A to a box B signifies that the distribution of the values of B depends probabilistically depends on the distribution of the values of A (and by Eq. 2, conversely). Thus, for example, in Figure 1 the probability that Witness3 correctly observed a VDE depends probabilistically on whether a correct verdict was delivered.

The prior probabilities of *STW* are defined in tables (not shown) to be the probabilities in the (SC).

Each box in Figure 1 has three regions, delimited by horizontal borders.

The top region of a box contains the name of a (random) variable of interest, e.g., *Correct Verdict.*

The middle region of a box consists of three elements (read horizontally):

i. a textual value-range for the random variable named in the top region of the box. For example, the top box in Figure 1 represents the random variable *Correct Verdict*.

ii. to the right of (i), a numerical literal (expressed as a percentage) indicating the probability that the variable of interest has a value lying in the value-range. For example, in Figure 1, the variable *Correct Verdict* has a probability of 96.4% of being true.

iii. to the right of (ii) a (segment of a) a histogram representation of the probability that the variable of interest has a value lying in the value-range denoted by (ii). Taken as a whole, the histogram spanning the middle region of the box represents the probability distribution for the variable named in (i), conditional on the variables at the tails of the arrows whose heads touch the box.

In Figure 1, the "Correct Verdict?" box has a pink background; the bottom row of boxes, a grey background. A box with a grey background means the variable corresponding to that box is intended as an "input" (also called an "asserted-value" or "finding") variable. Input variables represent information that is posited as given. A box with a pink background means the variable corresponding to that box is intended as an "output" (also called a "calculated") variable.

In *STW*, a variable can be toggled between a finding and a calculated value by a mouse-click.

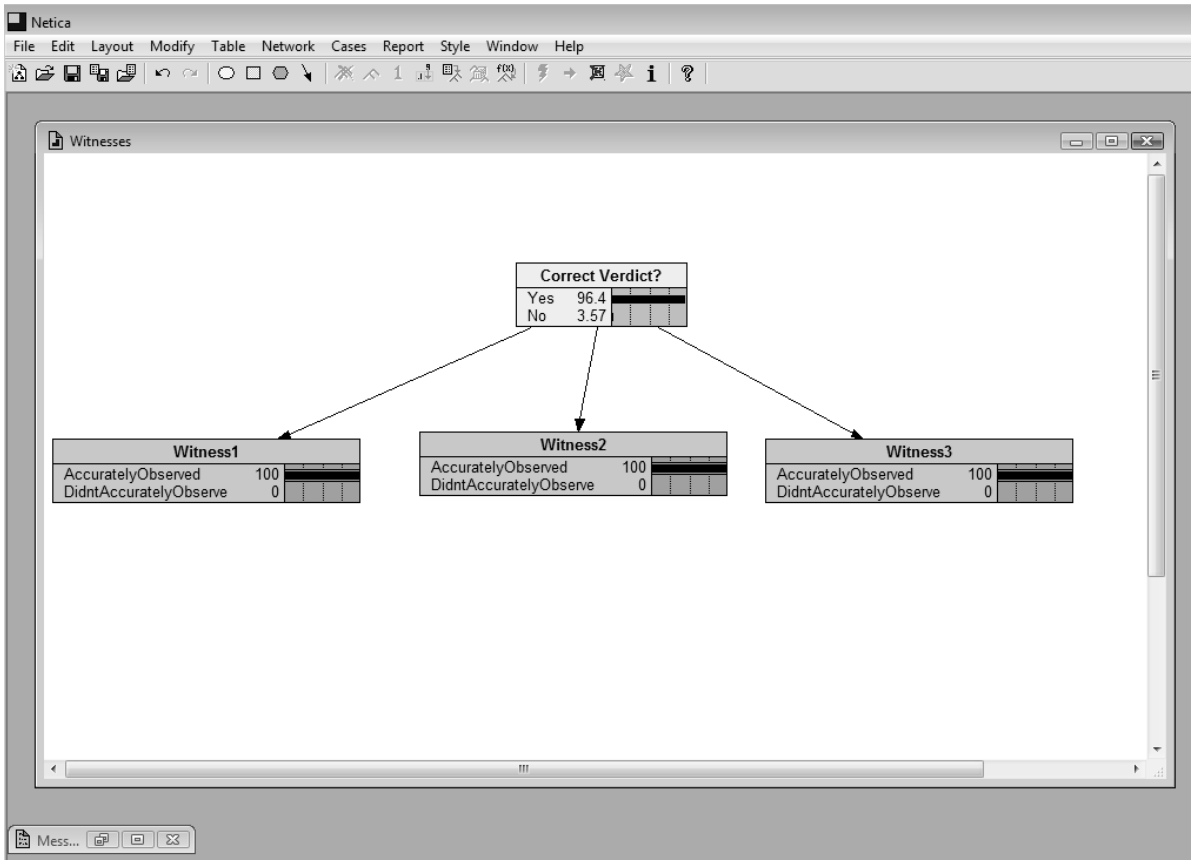


Figure 1. User-view of *STW*, assuming all three witnesses correctly observe a VDE. In the configuration, the probability of a correct verdict is 0.96.

In this example, if all three witnesses correctly observe the VDE, then the probability of a correct verdict is 0.96, even though the prior probability of any single eyewitness correctly observing the VDE is no greater than 0.5.

What happens in *STW* if only two of the three witnesses correctly observe the VDE? Figure 2 shows that the probability of a correct verdict is 0.75 -- even though the probability that any particular witness correctly observes the VDE ranges from 0.25 - 0.6.

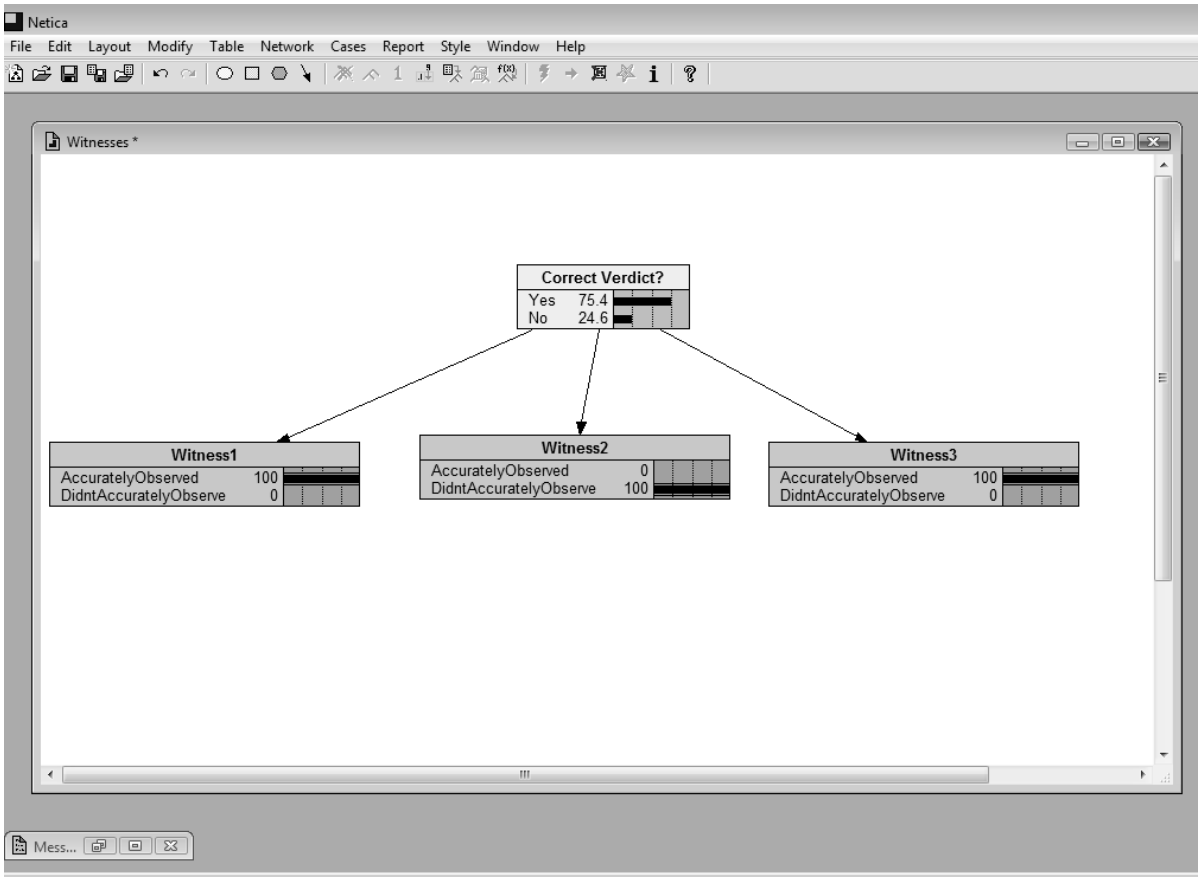


Figure 2. User-view of *STW* when only two witnesses observe the VDE correctly. The probability of a correct verdict is 0.75.

How sensitive is *STW* to the particular choice of priors? In general, answering this question requires analyzing a large set of cases. The case in which the probability of accurate observation is the same whether a correct verdict is achieved turns out to be especially interesting. In particular, let's assume the conditions of *STW*, but assume that probability of a witness accurately observing the VDE is 0.75 regardless of whether the verdict is correct. In such a case, we expect the probability of a correct verdict, given accurate observations by the witnesses, to be the same.

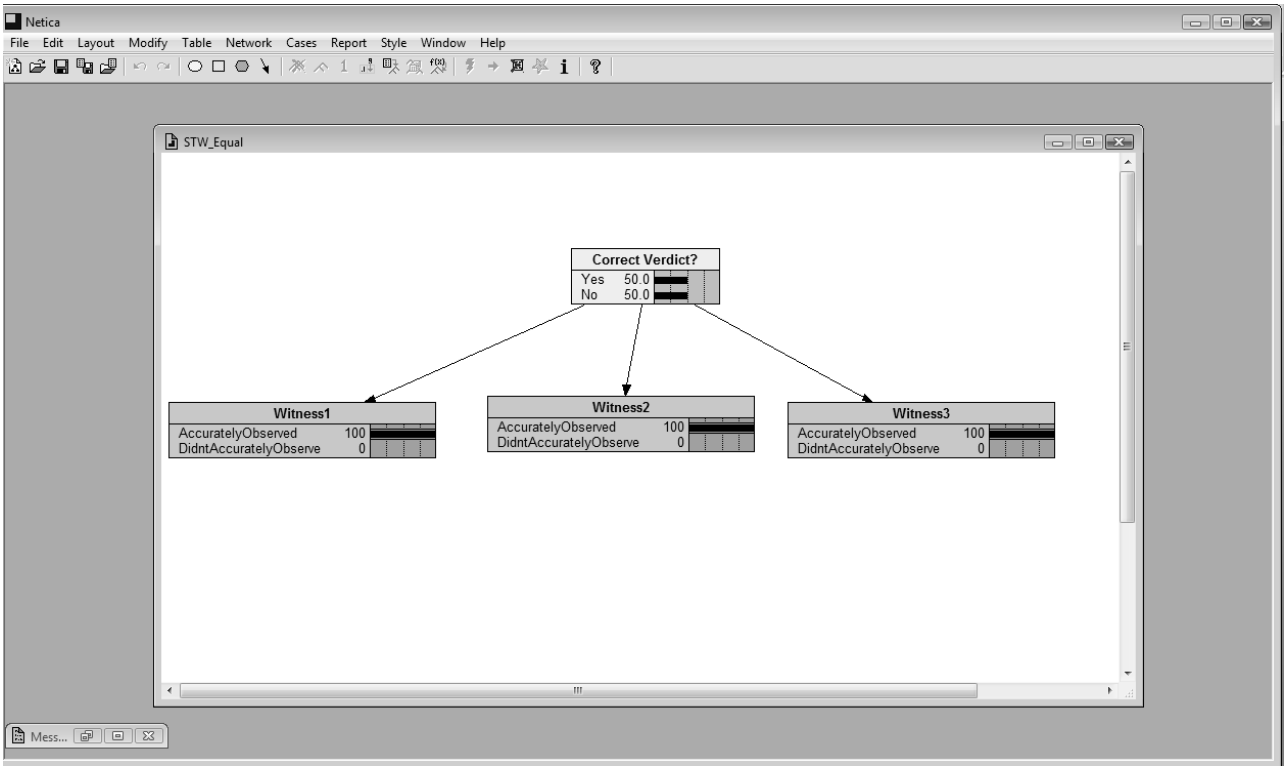


Figure 3. User-view of *STW*, assuming all witnesses correctly observed the VDE.

As expected, *STW_Equal* predicts the same distribution of probabilities for *Correct Verdict* if none of the witnesses correctly observe the VDE (Figure 3).

3.0 Discussion

The analysis above motivates several observations:

1. The technique shown here can be extend to an arbitrary number of witnesses, although the effect of more than three correctly observing witnesses contributes little.

2. The effect of one inaccurate witness is significantly mitigated by at least two accurate witnesses of a VDE. Adding the testimony of more than two accurate witnesses has decreasing returns. In addition, adding witnesses always runs the risk of introducing a witness whose testimony could raise doubt about the testimony of the rest. From the prosecution's point of view, this risk may not be negligible.

3. A companion paper describes a more refined model that shows the predictions of the model described in this paper are surprisingly informative.

4.0 Acknowledgements

This work benefited from discussions with Tom Rudkin, Brent Boerlage, Tony Pawlicki, Ron Giere, and Wesley Salmon. For any errors that remain, I am solely responsible.

5.0 References

[1] Norsys Software, Inc. *Netica* v4.08. 2008. http://www.norsys.com.

[2] Pearl J. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Revised Second Printing. Morgan Kaufmann. 1988.

[3] Steblay NK and Loftus EF. Eyewitness identification and the legal system. In Shafir E, ed. *The Behavioral Foundations of Public Policy*. Princeton. 2013.

[4] Chung KL. *A Course in Probability Theory*. Third Edition. Academic Press. 2001.

[5] Diestel R. *Graph Theory*. Springer. 1997.

[6] Jensen FV. *Bayesian Networks and Decision Graphs*. Springer. 2001.

[7] Pearl J. *Causality: Models, Reasoning, and Inference*. Second Edition. Cambridge. 2009.

[8] Jensen FV. *An Introduction to Bayesian Networks*. Springer. 1996.

[10] State of Florida v Zimmerman, 2012-CF-001083-A.

[12] Salmon WC. *The Foundations of Scientific Inference*. Pittsburgh. 1967.

[13] Chang CC and Keisler HJ. *Model Theory*. North Holland. 1990.

[14] Halmos PR. *Measure Theory*. D. Van Nostrand Reinhold. 1950.

[15] Bernays P. *Axiomatic Set Theory*. North-Holland, 1968. Dover reprint, 1991.

A Bayesian Network Analysis of Eyewitness Reliability: Part 2

Jack K. Horner
PO Box 266
Los Alamos NM 87544
jhorner@cybermesa.com

ICAI 2014

Abstract

In practice, many things can affect the verdict in a trial, including the testimony of eyewitnesses. Eyewitnesses are generally regarded as questionable sources of information in a trial setting: cases that turn on the testimony of a single eyewitness almost never result in a guilty verdict. Multiple eyewitnesses can, under some circumstances, collectively exhibit more robust behavior than any witness individually does. But how reliable, exactly, are multiple eyewitnesses? The legal literature on the subject tends to be qualitative. In a companion paper, I describe a highly idealized Bayesian network model of the relation between eyewitness behavior and trial verdict. In this paper, I describe a more refined Bayesian model of the same setting. It turns out that the highly idealized model provides nearly as much information as the more refined one does.

Keywords: eyewitness, Bayesian network

1.0 Introduction

In practice, many things can affect the verdict in a trial -- procedural conventions, material evidence, the psychology of the jurors, the persuasive power of the attorneys, and often, the testimony of eyewitnesses. Eyewitnesses are generally regarded as questionable sources of information in a trial setting ([3]): cases that turn on the testimony of a single eyewitness almost never result in a guilty verdict.

Multiple eyewitnesses can, under some circumstances, collectively exhibit more robust behavior than any witness individually does. But how reliable, exactly, are multiple eyewitnesses? The outcome of the recent trial of George Zimmerman, accused of second-degree murder or manslaughter of a teenager, rested heavily on the answer to this question ([10]). The legal literature on the subject tends to be qualitative (see, for example, [3]). A quantitative model is required.

Throughout, I will use the term *correct verdict* to mean a verdict that agrees with what actually happened, independently of the trial. I will use the term *verdict-determining-event* (VDE) to mean an event that could be witnessed by an eyewitness or that could contribute to a verdict.

2.0 A more refined Bayesian model

In a companion paper, I described a highly idealized model of a three-eyewitness domain. In many trials, witnesses observe only disjoint, or only partially overlapping, segments of an possible event timeline that is probabilistically related to a correct verdict. In the George Zimmerman trial ([10]), for example, witness saw or heard various, largely disjoint, parts of a hypothesized timeline of events.

To capture at least some of the more important features of cases of this kind, let's consider a model, *WitnessTimeline* (*WT*), in which:

(SC)

- There are three eyewitnesses
- There are three possible events which lie on a hypothesized timeline that occurred with high probability in the case of a correct a correct verdict. A possible event can occur or not occur.
- Each witness makes an observation of exactly one possible event. No two witnesses observed/didn't observe any one of the events. That is, there is a one-to-one onto mapping between witnesses and possible events. Any witness can be mistaken about whether he/she observed his/her respective event.
- If a possible event occurred, then the probability that the corresponding witness observed that possible event is 0.75. If a possible event didn't occur, then the probability that the corresponding witness observed that possible event is 0.25.
- If the hypothesized timeline occurred, then the probability that each of the possible events occurred is 0.9
- In the case of a correct verdict, the probability that the hypothesized timeline occurred is 0.9
- in the absence of any observation, the probability of a correct verdict is 0.5

Figure 1 shows a graphical user-view of more refined model, *Witness Timeline (WT*), that satisfies (SC) and is implemented in [1]. Each box in Figure 1 represents a random variable of a system. An arrow from a box A to a box B signifies that the distribution of the values of B depends probabilistically depends on the distribution of the values of A (and by Eq. 2, conversely). Thus, for example, in Figure 1 the probability that Witness3 correctly observed a VDE depends probabilistically on whether a correct verdict was delivered.

The prior probabilities of *STW* are defined in tables (not shown) to be the probabilities in the (SC).

Each box in Figure 1 has three regions, delimited by horizontal borders.

The top region of a box contains the name of a (random) variable of interest, e.g., *Correct Verdict.*

The middle region of a box consists of three elements (read horizontally):

i. a textual value-range for the random variable named in the top region of the box. For

example, the top box in Figure 1 represents the random variable *Correct Verdict*.

ii. to the right of (i), a numerical literal (expressed as a percentage) indicating the
probability that the variable of interest has a value lying in the value-range. For
example, in Figure 1, the variable *Correct Verdict* has a probability of 96.4% of
being true.

iii. to the right of (ii) a (segment of a) a histogram representation of
the probability that the variable of interest has a value lying in the
value-range denoted by (ii). Taken as a whole, the histogram spanning the middle
region of the box represents the probability distribution for the variable named in (i),
conditional on the variables at the tails of the arrows whose heads touch the box.

In Figure 1, the "Correct Verdict?" box has a pink background; the bottom row of boxes, a grey background. A box with a grey background means the variable corresponding to that box is intended as an "input" (also called an "asserted-value" or "finding") variable. Input variables represent information that is posited as given. A box with a pink background means the variable corresponding to that box is intended as an "output" (also called a "calculated") variable.

In *WT*, a variable can be toggled between a finding and a calculated value by a mouse-click.

Now suppose we knew nothing about what the witnesses observed. This situation is depicted Figure 5. As expected ([12], pp. 65-68), *WT* predicts that the probabilities of all possible states of the system are equal (0.5).

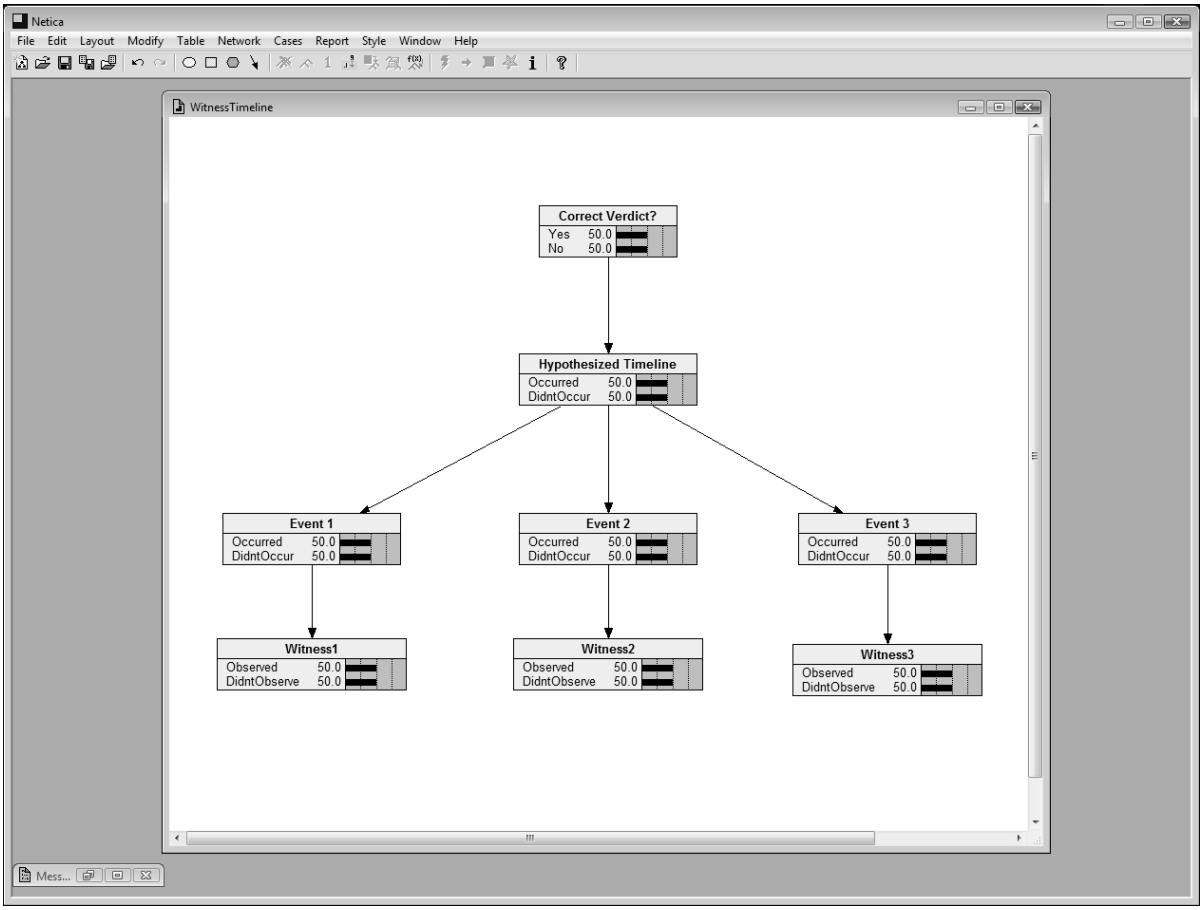


Figure 1. User-view of *WT*, given no data about the observations of the witnesses. In this case, all states of the network are equally probable.

Now let's suppose that all three witnesses observed their respective possible events and that the events in fact occurred. This situation is depicted in Figure 2. Then under the configuration described above, *WT* predicts that the probability of a correct verdict is ~0.84.

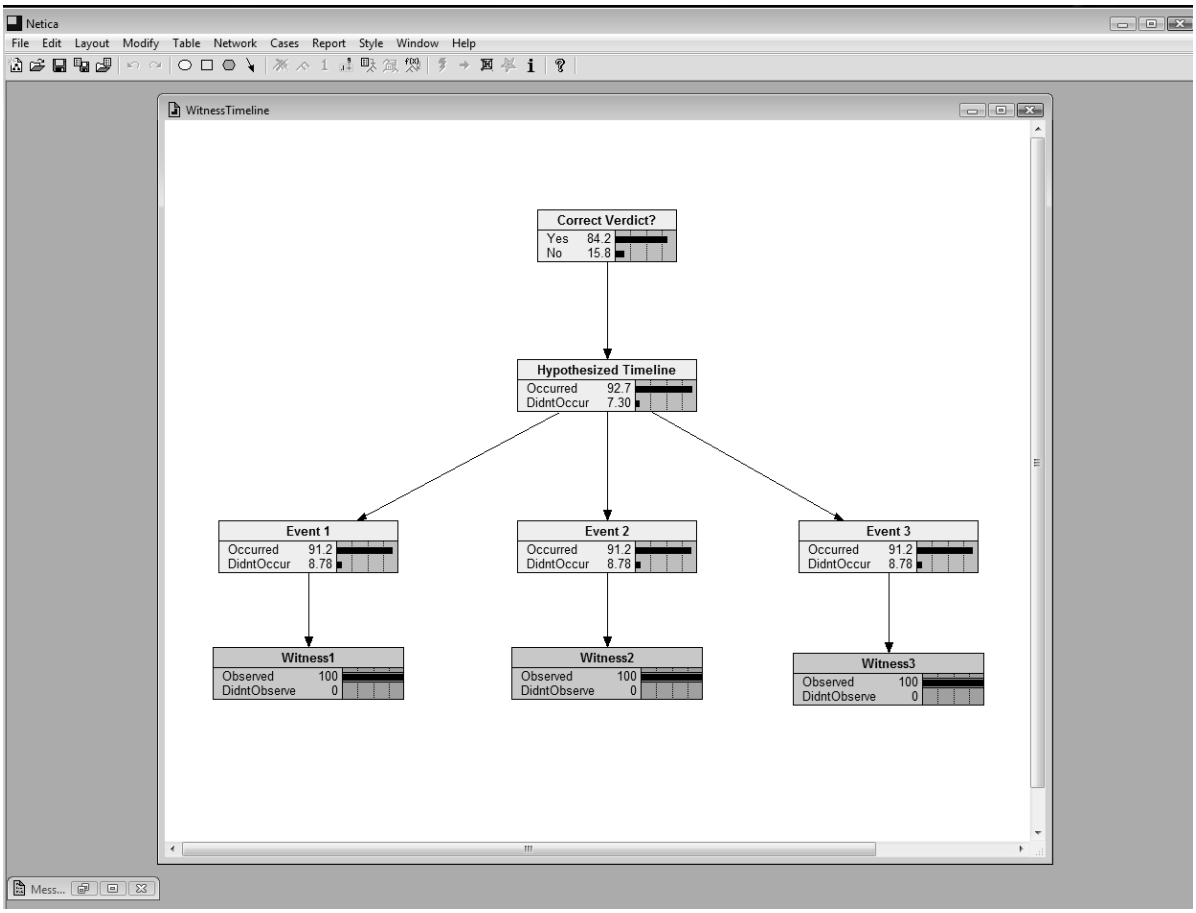


Figure 2. User-view of *WT*, a BN in which there is a one-to-one onto mapping between a set of three witnesses and a set of three events that lie on a hypothesized timeline. By assumption, the hypothesized timeline occurs with high probability in the case of correct verdict.

What happens to Correct Verdict in *WT* if one of the witnesses does not observed his/her event? This situation is depicted in Figure 7. *WT* predicts that, as expected, the probability of a correct verdict is lower than the configuration depicted in Figure 3.

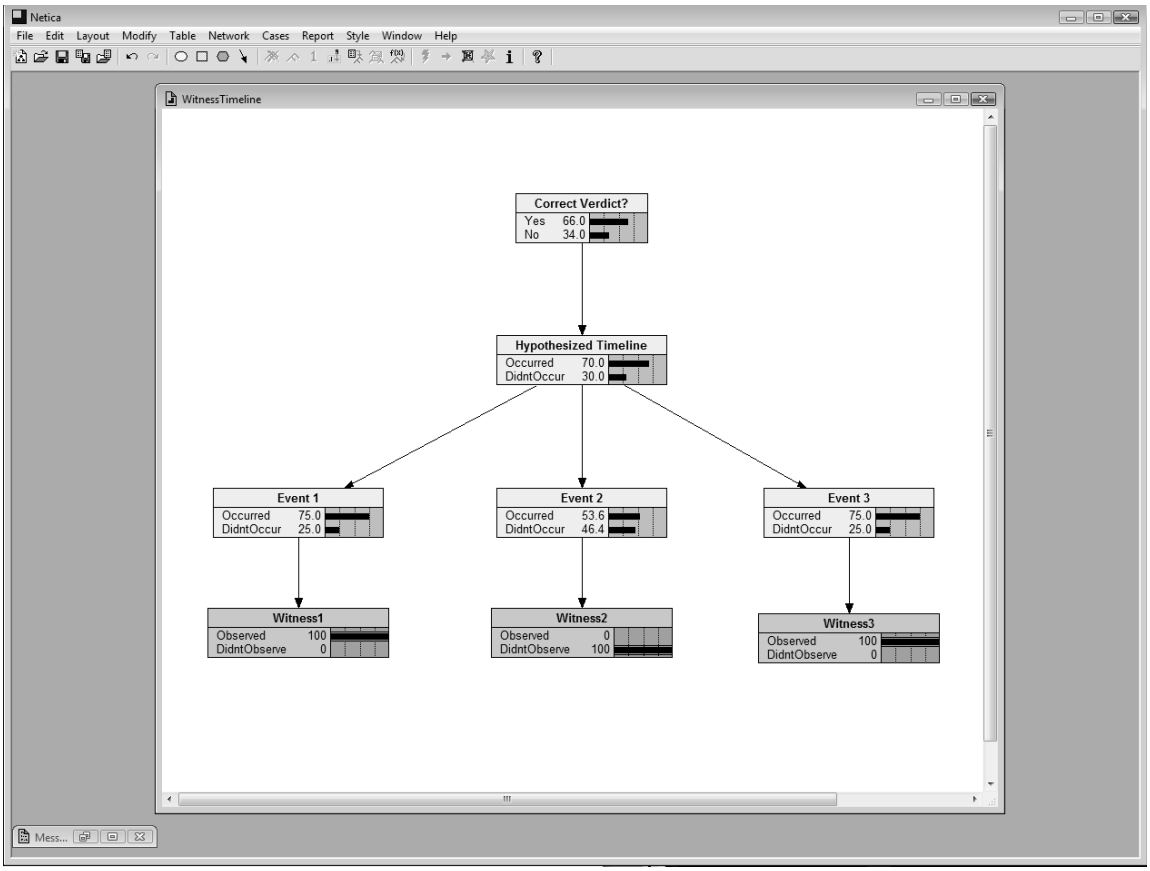


Figure 3. User-view *WT*, assuming two witnesses observe his/her possible event, but the third witness does not observe his/her event. The Correct Verdict occurs with probability 0.66.

3.0 Discussion

The analysis above motivates several observations:

1. The rubric shown here can be extend to an arbitrary number of witnesses, although the effect of more than three correctly witnesses, all other suffering being the same, contributes little.

2. The effect of one inaccurate witness is significantly mitigated by at least two accurate witnesses of a VDE. Adding the testimony of more than two accurate witnesses has decreasing returns. In addition, adding witnesses always runs the risk of introducing a witness whose testimony could raise doubt about the testimony of the rest. From the prosecution's point of view, this risk may not be negligible.

3. A companion paper describes a highly idealized model, *STW*, of the three-eyewitness domain. The current paper shows the predictions of *STW* are surprisingly informative. As expected, because *WT* has more probability distributions between the eyewitnesses and the verdict than *STW* does, whether a correct verdict is obtained depends less on the eyewitnesses in *WT* than in *STW*.

4.0 Acknowledgements

This work benefited from discussions with Tom Rudkin, Brent Boerlage, Tony Pawlicki, Ron Giere, and Wesley Salmon. For any errors that remain, I am solely responsible.

5.0 References

[1] Norsys Software, Inc. *Netica* v4.08. 2008. http://www.norsys.com.

[2] Pearl J. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Revised Second Printing. Morgan Kaufmann. 1988.

[3] Steblay NK and Loftus EF. Eyewitness identification and the legal system. In Shafir E, ed. *The Behavioral Foundations of Public Policy*. Princeton. 2013.

[4] Chung KL. *A Course in Probability Theory.* Third Edition. Academic Press. 2001.

[5] Diestel R. *Graph Theory.* Springer. 1997.

[6] Jensen FV. *Bayesian Networks and Decision Graphs.* Springer. 2001.

[7] Pearl J. *Causality: Models, Reasoning, and Inference.* Second Edition. Cambridge. 2009.

[8] Jensen FV. *An Introduction to Bayesian Networks*. Springer. 1996.

[9] Horner JK A Bayesian network analysis of eyewitness reliablility: Part 1. Submitted to the *Proceedings of the 2014 International Conference on Artificial Intelligence.* CSREA Press.

[10] State of Florida v Zimmerman, 2012-CF-001083-A.

[12] Salmon WC. *The Foundations of Scientific Inference.* Pittsburgh. 1967.

[13] Chang CC and Keisler HJ. *Model Theory*. North Holland. 1990.

[14] Halmos PR. *Measure Theory*. D. Van Nostrand Reinhold. 1950.

[15] Bernays P. *Axiomatic Set Theory*. North-Holland, 1968. Dover reprint, 1991.

SESSION

INFORMATION RETRIEVAL AND SEARCH METHODS + DECISION SUPPORT SYSTEMS, EXPERT SYSTEMS AND APPLICATIONS

Chair(s)

TBA

Improving Clinical Practice using Clinical Decision Support Systems with Medical Logic Modules

Ch. J. Schuh, W. Seeling, and J. S. de Bruin
Center for Medical Statistics, Informatics and Intelligent Systems (CeMSIIS),
Section for Medical Expert and Knowledge-Based Systems
Medical University of Vienna, Spitalgasse 23,
A-1090 Vienna, Austria

Abstract - *The provisioning of clinical decision support systems (CDSS) would enable the discovery of patterns in health data which might be important for the fight against nosocomial infections, incorrect diagnosis, and improper use of medication. Since the potential of medical decision making was first realized, hundreds of articles introducing decision support systems (DSS) have been published in the last three decades. But even today, only few systems are in clinical use, and their full potential for optimizing the healthcare system is far from realized. Clinician's acceptance and utilization of CDSS depends on its workflow-oriented context sensitive accessibility and availability at the point of care, and on the integration into a hospital information system (HIS).*
This paper outlines technical and medical aspects of a seamless integration of two CDSS into a HIS at the General Hospital of Vienna. The medical knowledge representation and reasoning of the CDSS is realized with Arden Syntax and Medical Logic Modules (MLM). Experiences gained by the clinical use of the systems are used to analyze the little use of CDSS in today's clinical practice.

Keywords: Hospital information systems, clinical decision support system, fuzzy based decision support, Arden syntax, and medical logic modules

1 Introduction

Over the past three decades medical treatment has made enormous progress. In modern health care environment, the amount of information available is very large, and in order to manage it computers are used in medicine in almost all areas. Clinicians and nurses are still performing time-consuming manual data analysis for making the most optimal medical decision for each individual patient [1-3]. They must choose from and interpret a huge variety of clinical data, while facing pressure to decrease uncertainty, risks to patients and costs. Computer technology can assist by generating case-specific advice for clinical decision making. The computer systems used are usually referred to as clinical decision support systems or CDSS [4, 5].
While electronic health records and databases help physicians manage this rising tide of information, patient-specific recommendations provided by clinical decision support systems can do even more by improving decision making and helping ensure patient safety. The provisioning of CDSS would enable the discovery of patterns in health data which might be important for the fight against nosocomial infections, incorrect diagnosis, unnecessary prescriptions, and improper use of medication. Current hospital information systems (HIS) are not offering an infrastructure for data-driven guidance, modeling of critical illness and infection surveillance.
Since the potential of medical decision making was first realized, hundreds of articles introducing CDSS have been published in the last three decades. But over the years' experience has shown that the expectations were not always fulfilled. Even today, only few systems, so asserted, are in clinical use. Even fewer are in use outside their site of origin, and their full potential for optimizing the healthcare system is far from realized.
The greatest barrier to routine use of decision support by clinicians has been inertia. Systems has been designed in the past for single problems that arise infrequently and have generally not been integrated into the routine data-management environment for the user [4], [14].
Clinicians' acceptance and utilization of CDSS depends on its workflow-oriented, context-sensitive accessibility and availability at the point of care, and ideally integrated into a HIS [11-13]. Commercially operated HIS often focus on administrative tasks and mostly do not provide additional knowledge based functionality. Their monolithic and closed software architecture encumbers integration of and interaction with external software modules [44 - 49].
This paper outlines technical and medical aspects of a seamless integration of two CDSS into a HIS at the General Hospital of Vienna. The medical knowledge representation and reasoning of the implemented CDSSs is realized with Arden Syntax and Medical Logic Modules (MLM).
The next section gives a brief history of CDSS with the Arden and Fuzzy Arden Syntax concept for MLMs that have been used in these applications. Further the conceptual architecture of CDSS and the integration of into the HIS will be introduced also. Finally, experiences gained by the clinical use of the implemented systems are used to analyze the little use of CDSS in today's clinical practice.

2 Decision Support Systems

A generic decision support system (DSS) receives a certain amount of data as input, processes it using a specific methodology and offers as a result some output that can help the decision-makers.
The concept of clinical decision support or DSS in general, is built on the paradigm of support. The term "decision support systems" (DSS) was coined the beginning of the 1970's to denote a computer program that could support a manager in making semi-structured or unstructured decisions.
It is not due to just data retrieval and numeric calculations either, which are the functions found in a traditional DSS. What is needed is a system which can process quantitative and qualitative data of varying levels of precision and, by reasoning, transform this data into opinions, judgments, evaluations and advice. These intelligent systems must be able to expect a tolerance for imprecision, uncertainty, and partial truth to achieve tractability, robustness, low solution cost, and better rapport with reality [15 - 17].

2.1 Clinical Decision Support Systems (CDSS)

CDSSs has been defined as ''any computer program designed to help health professionals make clinical decisions, deal with medical data about patients or with the knowledge of medicine necessary to interpret such data''.
In general computer applications should identify and reduce the rate of errors, inappropriate or inefficient actions, and adverse events. Therefore a CDSS can be broadly defined as DSS that integrate patient data with a knowledge-base and an inference mechanism to produce patient specific output in the form of care recommendations, assessments, alerts and reminders, to actively support practitioners and clinicians in clinical decision making [18, 19], [44]. A typical clinical therapeutic cycle in a simplified view is shown in Fig.1.

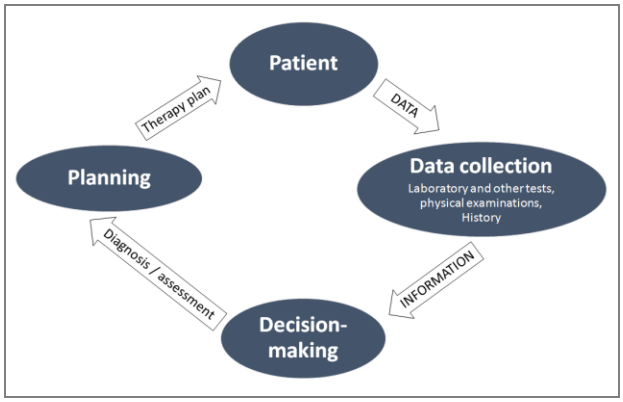


Fig. 1: The Diagnostic-Therapeutic Cycle (a simplified view)

Patient data can be input by digital entry queried from a HIS, patient data management systems (PDMS), or transmitted from other medical devices [30]. Patient data are compared against a knowledge-base and made sense of by an inference mechanism. The inference mechanism itself can be highly variable in sophistication ranging from simple 'if rules 'yes' or 'no' and 'if 'then', 'else' statements to Bayesian prediction techniques and/or with fuzzy logic [15 - 17].
Expert or knowledge-based systems are another type of CDSS capable of being programmed to perform decision making at the level of a domain expert [19]. These systems represent the most prevalent type of CDSS used in medical clinical practices today. Though CDSS can include different components, and though domain knowledge can be structured in a variety of ways, certain elements are common to all: a user interface, a knowledge base, a database, a knowledge acquisition facility, and an inference mechanism.
The two CDSSs presented in this paper [20, 21], interacts with the Arden and Fuzzy Arden Syntax concept, and the knowledge representation is realized with MLMs.

2.2 CDSS with Fuzzy Logic and Fuzzy Control

The concept of fuzzy set theory, which was developed by Zadeh (1965), makes it possible to define inexact medical entities as fuzzy sets. The Fuzzy set theory [15 - 17] derives from the fact that most natural classes and concepts are of fuzzy rather than crisp nature. By generalization of usual set theory an object cannot only be seen as an element of a set (membership value 1) or not an element of this set (membership value 0), but it can also have a membership value between 0 and 1 , (Fig. 2). Therefore fuzzy sets defined by their membership function μ which is allowed to assume any value in the interval [0, 1] instead of their characteristic function.

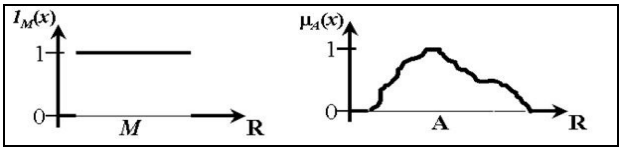


Figure 2: Characteristic function of a set *M* and membership function of a fuzzy set *A*.

A clinical fuzzy decision support system (CFDSS) is simply a DSS that is focused on using a knowledge management based on the fuzzy set theory in such a way to achieve clinical advice for patient care based on some number of items of patient data. A more far-reaching concept of modeling relationships was introduced by Sanchez 1979 [29]. Sanchez postulates the concept of "medical knowledge" based on a relationship between symptoms and diagnoses [35].
Using this composition formula as an inference rule, Assilian and Mamdani developed the concept of f*uzzy control* in the early 1970s [25].
Mamdani's development of fuzzy controllers in 1974 gave rise to the utilization of these fuzzy controllers in ever-expanding capacities [25, 26], [43]. Therefore these new intelligent CFDSS must be able to expect a tolerance for imprecision, uncertainty, and partial truth to achieve

tractability, robustness, low solution cost, and better rapport with reality.

3 Applications & Clinical Results

There are many different methodologies that can be used by a CDSS in order to provide support to the health care professional. In our case the inference mechanism, i.e. a set of rules derived from the physician's (experts) and evidence-based medicine, and the knowledge base itself, are implemented for both CDSSs through medical logic modules (MLMs) based on a language such as Arden syntax [22 - 24].

3.1 Technical Aspects

3.1.1 Arden Syntax decision support sharing efforts

The Arden Syntax for Medical Logic Modules or rather for CDSSs is a language for encoding medical knowledge bases that consist of independent modules. The Arden Syntax has been used to generate clinical alerts, diagnostic interpretations, management messages, and screening for research studies and quality assurance. An Arden Syntax knowledge base consists of rules called Medical Logic Modules (MLMs), which are stored as simple ASCII files that can be written on any text editor. An MLM is a hybrid between a production rule (i.e. an "if-then" rule) and a procedural formalism. Each MLM is invoked as if it were a single-step "if-then" rule, but then it executes serially as a sequence of instructions, including queries, calculations, logic statements and write statements. One of the earliest efforts at sharing clinical decision support content was the Arden Syntax Medical Logic Module (MLM) repository. Arden Syntax now is a standard for encoding event-driven rule based clinical knowledge for use in clinical decision support systems [22 - 25]. A first draft of the standard was prepared at a meeting at the Arden Homestead, New York, in 1989. Health Level 7 (HL7) published Arden Syntax version 2.0 in the year 1999 and has been hosting the development of all newer versions of the Arden Syntax standard ever since [22]. The present, most recent version of Arden Syntax is version 2.8. [26].

3.1.2 Hospital Information System (HIS)

The communication mechanism, the host system of the CDSSs will allow showing the results to the users, as well as have input into the system.

At the General Hospital of Vienna this fully featured HIS is i.s.h.med and represents the communication mechanism of the implemented CDSSs. Additionally the HIS i.s.h.med provides a clinical workplace, a parametric medical document (PMD) and an interface via Web Service. The Arden Syntax Server represents in this scenario the reasoning and inference mechanism with MLMs, which can be realized in fuzzy and crisp nature, and processes the patient data, which is received from the HIS i.s.h.med via web services (Fig. 3).

The clinical workplace (Host System CDSS) provides a work environment for the medical users (including patient and ward lists, access to scheduled appointments, etc.). Information received from the CDSS, e.g. alerts or reminders, can be displayed in the respective list(s) on the clinical workplace.

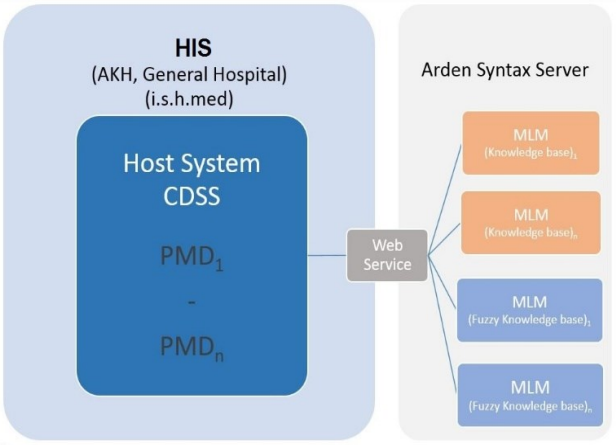


Fig. 3: CDSS integration into the HIS i.s.h.med at the General Hospital of Vienna

The PMD provides a framework for medical documentation which can be customized to the special medical needs of the respective clinical department and the PMDs are used as the user interface of the CDSSs.

The web service has two basic communication functions: First, it provides laboratory (e.g. tumor markers) and clinical data (see above) for sending to the Arden Syntax server in XML format. Second, it receives results (and explanation, if applicable) from Arden Syntax server and saves those into the i.s.h.med PMD (Fig 4).

By programming, the PMD can – on receiving results – trigger certain "events" in the software, e.g. put alerts or reminders on a patient list on the clinical workplace.

3.2 Prediction of Melanoma Metastasis Events

3.2.1 Clinical Background

Cutaneous Melanoma (CM), the most lethal form of skin cancer can be highly metastatic. The most common site of metastatic disease in melanoma is the regional lymph nodes indicating that metastatic spread usually occurs via the lymphatic system.

There is substantial evidence that cases of CM are still increasing worldwide. The increase of the incidence amounts to about 4-8% [32, 33]. According to Meves, a duplication of the incidence until 2020 is conceivable [34]. Today's incidence in Europe ranges between 12-15 /100,000 inhabitants [33]. The most widely used prognostic indicator for survival is Breslow thickness, however, this is still inaccurate for a significant number of patients [31]. CM is initially treated by surgical excision. After excision, tumors are classified according to the American Joint Committee on Cancer (AJCC) published TNM classification for CM, based on studies from Balch et al. [37, 38]. The AJCC classification [39] allows to classify CM into different categories, predicting the risk for widespread metastatic disease. The presence of metastatic disease correlates with the concentration of

several tumor markers. The already routinely established tumor markers for CM are S100β, MIA and LDH. These parameters were chosen for the predictive model [40-42], and in addition for the implementation of the CDSS knowledge base.

3.2.2 Predictive probability model

The patient's pretest probability assessment is based on predictive characteristics from the literature. These include the tumor thickness according to Breslow [31], mitotic rate and ulceration which can be used to make conclusions of the tumor behavior. The final version of the seventh edition of the AJCC melanoma staging and classification [37 - 39] includes the revised TNM classification for CM.

This classification is particularly well suited for rule-based programming languages because it's IF-THEN rule structure. The knowledge base developed in this particular CDSS calculates the present risk for metastasis in CM patients. Calculations are based on the pretest probability for metastasis in combination with the recent results from the tumor markers stated above and is shown in Fig. 4.

3.2.3 TSM-CDS Application

The knowledge base itself is a combination of multiple risk assessments.

In detail it's a rule-based interpretation of the TNM classification according AJCC. Further the interpretation of the tumor markers S100β, melanoma inhibitory activity (MIA) and lactate dehydrogenase (LDH). And final the risk assessment of survival function (present statistical mortality risk), based on the results of the AJCC. The front end of the system is the implementation of the PMD (Fig. 4).

Fig. 4: PMD screen of the TSM-CDS application.

The CDSS supports the physicians by calculating the tumor stage. Furthermore, it offers an interpretation whether a given pattern of tumor markers is suspicious for an underlying metastatic event. As mentioned above the CDSS integrates seamless into the workflow of the HIS i.s.h.med. Specifically, results from tumor markers are automatically fed into the CDSS out of the laboratory information system (LIS). Further the clinical data are extracted from patient's history and from the patient histopathological report. As a result these steps of data extraction feed the CDSS with all relevant data.

As mentioned the pre-test probability according to TNM was implemented in Arden Syntax, and the rules are grouped in MLMs. In this current version the knowledge base is crisp nature. In a future project there will be an extended version with fuzzy Arden Syntax and fuzzy MLMs.

The HIS- PMD itself has only a German user interface, thus the PMDs of the TNM-CDSS is also in German (Fig. 4). The values (LDH 100, MA 15, S100ß 0.5) in combination with a tumor thickness of 1 mm has as TNM result "T1aN0M0 (IA)".

Currently, the CDSS is in clinical evaluation and calculates the probability whether not a given pattern of tumor markers is suggested for metastatic disease, but will not display this result to the user. More than 260 clinical cases have been gathered up to now. The response system is received just in the background and not shown to the physician. Instead, the user is prompted to give his or her expert opinion whether or not the given pattern is suggestive for metastatic disease.

The results in general confirm the applicability of the application to represent medical knowledge, thus rendering the TNM process transparent and comprehend-sible [42].

The system appears to be well accepted by the clinical experts. This is mainly due to the fact that the CDSS is almost seamlessly integrated into the routine HIS i.s.h.med. Initial data show that the comparison of the physicians' decisions with the CDSS resulted in 106 (49.53%) complete matches, which implies that the CDSS and the physician completely agreed. In 48 (22.43%) cases, the system calculated a lower risk for the patient, whereby in 10 (4.67%) cases the calculations resulted in a higher risk, respectively. In 50 (23.36%) cases, no decision was neither possible for the CDSS nor for the physician, due to the lack of parameters. Parameters are automatically extracted from its data sources without any hassle for the physicians in charge. The performance of the system is still under investiga-tion.

3.3 Fuzzy monitoring systems: MONI-ICU

Since 2004, the Clinical institute of Hospital Hygiene at the Vienna General Hospital has used an electronic monitoring system called MONI-ICU (acronym for monitoring of nosocomial infections in intensive care units) to automatically detect nosocomial infections, otherwise known as healthcare-associated infections (HAIs). The newest installment of this system incorporates fuzzy sets and logic to represent abstract, linguistic clinical concepts.

HAIs are infections that result from a patient's treatment in a healthcare setting such as a hospital. Due to the relatively high presence of multidrug resistant pathogens in hospitals, HAIs can have severe consequences on a patient's health and recovery, and this risk only increases as more pathogens develop antibiotic resistances. In order

to asses and counter the threat of HAIs, the European Centre for Disease Prevention and Control has developed HAI surveillance programs [8]. While HAI incidence rates have gone down as a result of these programs, they also place a high burden on personnel and hospital resources [7, 8].

The MONI-ICU surveillance system [9, 10] has been created to decrease the burden of infection surveillance on hospitals, thereby allowing infection control specialists to concentrate on infection prevention. The system combines electronic patient data such as microbiology and biochemistry test results with clinical data from ICU patient data management systems and administrative patient data to detect a variety of HAI types such as blood stream infections, urinary tract infections, catheter-related infections and pneumonia.

As a knowledge base, the system uses a set of rules derived from the ECDC HAI surveillance definitions, which are part of the ECDC HAIICU surveillance protocol [6]. These surveillance definitions have been translated by clinical and infection control experts into a computer-readable Arden Syntax representation. Furthermore, fuzzy sets have been defined by infection control experts to represent many of the basic clinical and biochemical concepts whose values can be derived directly from raw data; fuzzy logic is then applied throughout the knowledge base to combine these basic concepts into more abstract, linguistic clinical terms, which combined with results from other data sources are used to deduce the presence or absence of HAIs, and to what extent.

Fig. 5 shows a graphical depiction of part of the knowledge base.

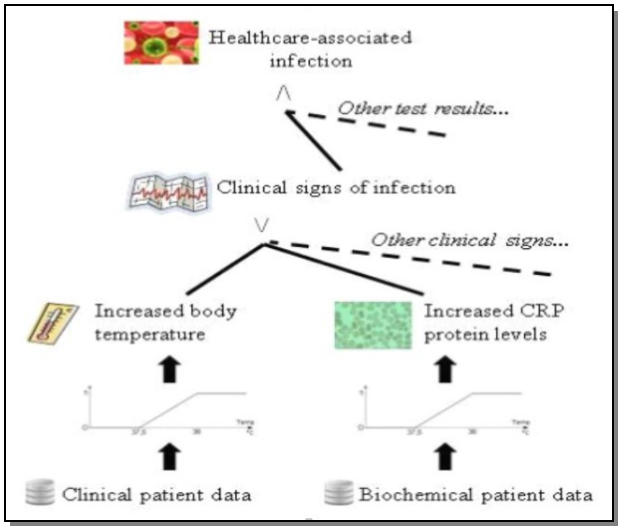


Fig. 5: The MONI-ICU knowledge base and info structure

In this figure, clinical patient data are filtered for information on body temperature, and using a fuzzy set, the raw data are transformed into a fuzzy value for the abstract clinical term increased body temperature. Similarly, biochemical data indicating a patient's C-reactive protein (CRP) value is transformed to a fuzzy value for the abstract term increased C-reactive protein levels.

Afterwards, fuzzy values for these and other clinical indications of infection are grouped into a clinical denominator called clinical signs of infection with a single fuzzy value using the standard (Gödel) triangular conorm. In turn, this fuzzy value is abstracted along with other test results (e.g. microbiology results and administrative data) using the standard (Gödel) triangular norm into a final denominator indicating the presence of HAIs. Currently, the system is in the clinical test phase, and the daily results are used by the Clinical institute of Hospital Hygiene as part of their daily afternoon briefing, and as a research platform.

Preliminary evaluation of the system indicates that it performs better than manual ward surveillance (sensitivity 87% vs. 40% for manual surveillance, specificity 99% vs. 94% for manual ward surveillance). Furthermore, the time spent on surveillance by infection control specialists has been reduced by almost 85% [9].

4 Conclusions and future work

A problem that occurs with any form of clinical knowledge representation is the need to interact with a clinical database in order to provide alerts and reminders. Database schemata, clinical vocabulary and data access methods vary widely so any encoding of clinical knowledge, such as a MLMs, must be adapted to the local institution in order to use the local clinical repository. This hinders knowledge sharing. Arden Syntax is the only standard for procedurally representing declarative clinical knowledge (contrast GLIF or PRO*forma*, for example, which are more declarative formats), so this problem is associated with Arden, but it is not unique to it. Based on the literature, current CDSS are limited in application. Roughly seventy known proprietary medical CDSS exists. Only ten out of seventy geared towards routinely use. Unfortunately there is no information available about a real daily average usage of these systems.

The concept of CDSS or DSS in general, is built on the paradigm of *support.* Again a well-designed CDSS should have the potential to assist physicians who can and do use it as often as possible in the daily routine work. In some situations physicians learns from using a CDSS about criteria, facts or process issues that need to be considered in a specific decision situation. CDSS encourage and promote "rationality" in decision making. CDSSs are intended to support not replace physicians, so the users need to consciously interact with a CDSS to use it effectively.

One large roadblock to acceptance is workflow integration. This is mainly also in our case due to the fact that the CDSS is almost seamlessly integrated into the routine HIS i.s.h.med. Often these systems are stand-alone applications, requiring the clinician to cease working on their current report system, switch to the CDSS, input the necessary data, and receive the information.

Further, a big issue is that the expectation needs to be created that the physicians are the ultimate authority and that the physicians can anytime "over rule" or choose to

ignore analyses and recommendations of the CDSS. This is a feature key of the PDM concept of the HIS i.s.h.med that is used at the General Hospital.

Anticipated limitations of CDSS are that an optimal physician's treatment requires that physicians be able to have the following information, in real time, if possible: What is happening right now? What will happen in the future? What do I need to create the future I want? To answer these questions effectively, physicians requires data that are factual, factual inferential (why type questions) and predictive (what if questions). To date, the best support that a CDSS has been able to provide is data that answer factual and maybe some forms of predictive questions [3]. As mentioned above one big argument of the rare utilization at this time is that most of the CDSS have not progressed beyond the prototype stage [46]. There are no standards or universally accepted evaluation or validation methodologies to ensure that the system's knowledge base is complete and correct.

With respect to the deployment and support of CDSS, it also appears a major barrier to progress is lack of appreciation of the difficulty of the problem. On the surface, for instance in our case also, most of the CDSS does not appear to be very complicated to implement, i.e. the MLS are not highly sophisticated and so on.

The point that is often overlooked, however is that robust sustainable use of CDSS is not at all simple, even with the if … then rules MLM concept or order sets, when one considers it not with respect to a single point in time but from a long –term maintenance and update perspective. The knowledge assets underlying CDSS are time consuming and expensive to generate, and subject to change and reuse them if once created, would be highly advantageous with Arden Syntax MLM concept. We think that the lack of such capabilities is one of the primary impediments to driving widespread CDSS adoption and use.

In our case both introduced systems are used consequently in the daily routine and fulfill the questions mention above. The absence of a well-defined or universal evaluation methodology makes these questions of course difficult to answer. To date, an examination of the literature indicates that there is virtually no information available related to the cost or cost effectiveness of CDSS. Most of the CDSS, ours equally, are university-based developments, and still in prototype stage. These costs regarding the initial investment of CDDS tend to be hidden and therefore difficult to access. This frightens or hinders the industry's interest in funding and encouraging the development of CDSS in health care in general.

The physicians at the General Hospital of Vienna, and many others have a real positive outlook on the potential for CDSSs, particularly in relating to practitioner performance. However, until the use of CDSS in general is a routine as the use of the blood pressure cuff, it is important to be sensitive to resistance to using these systems.

5 References

[1] B.C. Delaney, Can computerized decision support systems deliver improved quality in primary care, Br. Med. J. 319 (1999) 1281—1282.

[2] K. Kawamoto, C.A. Houlihan, E.A. Balas, D.F. Lobach, Improving clinical practice using clinical decision support systems: a systematic review of trials to identify features critical to success, BMJ 330 (7494) (2005) 765.

[3] A.X. Garg, N.K. Adhikari, H. McDonald, M.P. Rosas-Arellano,P.J. Devereaux, J. Beyene, et al., Effects of computerized clinical decision support systems on practitioner performance and patient outcomes: a systematic review ,JAMA 293 (10) (2005) 1223–1238.

[4] Musen M, Shahar Y, Shortliffe EH. Clinical decision support systems. In: Shortliffe EH, Perrault L, Wiederhold G, et al. (eds). Medical Informatics: Computer Applications in Health Care and Biomedicine. New York: Springer-Verlag, 2001:573-609.

[5] B. Kaplan, Evaluating informatics applications-clinical decision support systems literature review, Int. J. Med. Inform.64 (2001) 15-37.

[6] HAIICU Protocol v1.01 STANDARD and LIGHT. In 2010. [Accessed November 12, 2012] Available from: http://www. ecdc.europa.eu/en/aboutus/calls/Procurement20Related20Documents/5_ECDC_HAIICU_protocol_v1_1.pdf.

[7] Haley RW, Culver DH, White JW, Morgan WM, Emori TG, Munn VP and Hooton TM. The efficacy of infection surveillance and control programs in preventing nosocomial infections in US hospitals. Am J Epidemiol. 1985;121(2):182-205.

[8] Klompas M and Yokoe DS. Automated surveillance of health care-associated infections. Clin Infect Dis. 2009;48(9):1268-1275

[9] de Bruin JS, Adlassnig KP, Blacky A, Mandl H, Fehre K and Koller W. Effectiveness of an automated surveillance system for intensive care unit-acquired infections. J Am Med Inform Assoc. 2012;

[10] Koller W, Blacky A, Bauer C, Mandl H and Adlassnig KP. Electronic surveillance of healthcare-associated infect-ions with MONI-ICU a clinical breakthrough compared to conventional surveillance systems. Stud Health Technol Inform. 2010;160(Pt 1):432-436

[11] Hothorn T, Hornik K, Zeileis A. Unbiased Recursive Partitioning: A Conditional Inference Framework. Journal of Computational and Graphical Statistics. September 1, 2006; 15(3): 651-674

[12] Adlassnig KP, Rappelsberger A. Medical knowledge packages and their integration into health-care information systems and the World Wide Web. MIE 2008; S121-126.

[13] Mueller ML, Ganslandt T, Eich HP, Lang K, Ohmann C, Prokosch HU: Towards integration of clinical decision support in commercial hospital information systems using distributed, reusable software and knowledge components. Int J Med Inform 2001, 64(2-3):369-377.

[14] E.H. Shortliffe, Knowledge-Based Systems in Medicine MIE 1991 Proceedings, Springer, Berlin, 1991, pp. 5–9.

[15] Zadeh, L. A.: Fuzzy Sets. Information and Control, 8, 1965, pp. 338-353.

[16] Zadeh, L. A.: Biological Applications of the Theory of Fuzzy Sets and Systems. The Proceedings of an International Symposium on Biocybernetics of the Central Nervous System. Little, Brown and Company: Boston 1969, pp. 199-206.

[17] Zadeh, L. (1997). In Jang, J. S., Sun, C. T. and Mizutani, E. (Eds.) Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence. Upper Saddle River, NJ. Prentice-Hall.

[18] Eddy D. M., Clinical decision making from theory to practice: a collection of essays from the Journal of the American Medical Association. Sudbury, MA: Jones and Bartlett; 1996.

[19] Spiegelhalter D. J., Bayesian Analysis in Expert Systems. MRC Biostatistics Unit, Institute of Public Health, Cambridge, 1992.

[20] Hripcsak G. Arden syntax for medical logic modules. MD Comput 1991; 8 (2):76–8.

[21] G. Hripscak, Writing ARDEN Syntax Medical Logic Modules, Computers in Biology and Medicine, Vol. 24, 331–363, 1994.

[22] Health Level Seven, Arden Syntax for Medical Logic Systems, Version 2.1, Health Level Seven, Inc., 3300 Washtenaw Ave, Suite 227, Ann Arbor, MI 48104, 2002.

[23] Health Level 7. Arden Syntax for Medical Logic Systems Standard Version 2.6. Ann Arbor MI: Health Level 7; 2007

[24] Clinical decision support work group. Health Level Seven, Inc.: 2009. Available at: http://www hl7.org/. Accessed March 20, 2009.

[25] Samwald, Matthias; Fehre, Karsten; De Bruin, Jeroen; Adlassnig, Klaus-Peter (2012). "The Arden Syntax standard for clinical decision support: Experiences and directions". Journal of Biomedical Informatics 45 (4): 711–8. doi:10.1016/j.jbi.2012. 02.001. PMID 22342733.

[26] Arden_Syntax_http://www.hl7.org/special/Committees/arden/index.cfm [accessed 31.01.12].

[27] Assilian, S., Mamdani, E.: Learning Control Algorithms in Real Dynamic Systems. Proc.4th International IFAC/IFIP Conference on Digital Computer Applications to Process Control, Zürich, March 1974.

[28] Mamdani, E.: Application of Fuzzy Algorithms for Control of Simple Dynamics Plant. Proceedings of the IEEE, 121(12), 1974, pp. 1585-1888.

[29] Sanchez, E.: Medical Diagnosis and Composite Fuzzy Relations. Gupta, M. M.; Ragade, R. K.; Yager R. R. (Eds.): Advances in Fuzzy Set Theory and Applications. Amsterdam: North-Holland 1979, pp. 437-444.

[30] Osheroff JA, Teich JM, Middleton BF, et al. A roadmap for national action on clinical decision support. American Medical Informatics Association; 2006 June 13. Available at: http://www.amia.org/inside/initiatives/cds/. Accessed March 20, 2009.

[31] Breslow, A., Thickness, cross-sectional areas and depth of invasion in the prognosis of cutaneous melanoma. Ann Surg, 1970. 172(5): p. 902-8.

[32] Bosserhoff, A.K., et al., [MIA ("melanoma inhibitory activity"). Biological functions and clinical relevance in malignant melanoma]. Hautarzt, 1998. 49(10): p. 762-9.

[33] Hauschild, A., et al., Malignes Melanom, in Chirurgische Onkologie - Strategien und Standards für die Praxis. 2008, Springer Verlag: Wien. p. 449-465.

[34] Meves, A., Intensivkurs Dermatologie. 2006: Urban&Fischer Verlag.

[35] Apkon M, Mattera JA, Lin Z, et al. A randomized outpatient trial of a decision-support information technology tool. Arch Intern Med 2005 Nov; 165(20):2388-94.

[36] Bates DW, Kuperman GJ, Wang S, et al. Ten Commandments for effective clinical decision support: making the practice of evidence-based medicine a reality. J Am Med Inform Assoc 2003 Nov; 10(6):523-30.

[37] Balch, C.M., et al., Final version of the American Joint Committee on Cancer staging system for cutaneous melanoma. J Clin Oncol, 2001. 19(16): p. 3635-48.

[38] Balch, C.M., et al., Final version of 2009 AJCC melanoma staging and classification. J Clin Oncol, 2009. 27(36): p. 6199-206.

[39] American Joint Committee on Cancer, Melanoma of the Skin Staging, melanoma8.5x11.pdf, Editor 2009.

[40] Garbe, C., et al., Interdisziplinäre Leitlinien zur Diagnostik und Behandlung von Hauttumoren. 2005, Stuttgart, New York: Georg Thieme Verlag.

[41] Schlager, K. and M. Binder, Klinischer Vorhersagewert der Tumormarker S100ß, MIA und LDH bei Patienten mit malignem Melanom in Department of Dermatology2009, Medical University of Vienna: Vienna.

[42] C Scheibboeck, P Huber, S Weber, K Harmankaya, R Nemecek, J Weingast, M Binder, T Mehl, Ch Schuh S Dreiseitl: Prediction of metastatic events in patients with cutaneous melanoma. eTELEMED 2013, 12 40 40134.

[43] T. Takagi and M. Sugeno, Fuzzy Identification of Systems and its applications to modeling and Control, IEEE Transactions on Systems, Man, and Cybernetics, 15(1): 116-132, Jan-Feb 1985.

[44] Varonen H, Kortteisto T, Kaila M, for the EBMeDS Study Group. What may help or hinder the implementation of computerized decision support systems (CDSSs): a focus group study with physicians. Fam. Pract. 2008 Jun; 25(3):1627.

[45] R. Goud, N.F. de Keizer, G. ter Riet, J.C.Wyatt, A. Hasman, I.M. Hellemans, et al., Effect of guideline based computerized decision support on decision making of multidisciplinary teams: cluster randomized trial in cardiac rehabilitation, BMJ 338 (2009) b1440.

[46] Shojania KG, Grimshaw JM: Evidence-based quality improvement: the state of the science. Health Aff. (Millwood) 2005, 24(1):138-150.

[47] Tierney WM: Improving clinical decisions and outcomes with information: a review. Int J Med Inf 2001, 62(1):1-9.

[48] Perreault L, Metzger J. A pragmatic framework for understanding clinical decision support. Journal of Healthcare Information Management. 1999; 13(2):5-21.

[49] Trivedi MH, Kern JK, Marcee A, Grannemann B, Kleiber B, Bettinger T, Altshuler KZ, McClelland A. Development and implementation of computerized clinical guidelines: barriers and solutions. Methods Inf. Med. 2002; 41(5):435-42.

Using an Inference Engine for AI in the Office Tactics Video Game

Arturo I Concepcion[1], Edward Munoz[2], Matthew Hawkins[3], and Diane Balane[4]
[1]School of Computer Science & Engineering, California State University, San Bernardino, CA, United States
[2]iMedRis Data Corp., Redlands, CA, United States
[3]School of Computer Science & Engineering, California State University, San Bernardino, CA, United States
[4]Department of Art, California State University, San Bernardino, CA, United States

Abstract - *FSM (or its improvements) is the most common method employed when implementing AI on video games. Its major advantages are its simplicity and the ease of implementation but its greatest disadvantage is the predictability of the next state, which could lead to the player predicting the next step the game will take. Although there are some improvements done in FSM to alleviate this predictability, the inference engine allows a reasoning process and could come up with a strategy or move that the player might not be expecting. The inference engine consists of three parts: the knowledge base, the agenda, and the working memory. This paper developed an inference engine using the scripting language of UDK and applied to a video game, OfficeTactics. The resulting AI is very diverse and provides a lot of options that makes the game more exciting and enjoyable to play.*

Keywords: expert system, inference engine, knowledge-base, AI agents, and video game.

1 Introduction

Office Tactics is a video game designed by Danny Vargas when he was an undergraduate student in the School of Computer Science & Engineering, California State University, San Bernardino, in 2011. The game is a turn-based strategy and involves office workers who were laid off from their jobs causing them to retaliate in rebellion against management.

The paper describes the use of an inference engine to implement the AI of the enemy and friendly NPCs in the video game. Finite-state machines are the most common structures that are used for AI because of its simplicity and ease of implementation that produces good results. In spite of its advantages, FSM has many disadvantages: subject to unbounded growth of states, hard to maintain, and predictability of actions by the NPCs.

There were extensions made to FSM to overcome the disadvantages mentioned above. One is extending the states to offer Enter/Exit blocks and allowing event notifications to produce random or probabilistic transitions. Another is to allow FSM to have a stack-based history to track past states [6]. This was also extended to allow a state to transition to an entirely new FSM, making the FSM hierarchical. Even with these extensions, FSMs cannot perform pathfinding, reasoning or learning and so we propose the use of inference engines to overcome these deficiencies [7].

2 Office Tactics Video Game

2.1 Story

Office Tactics is a game that circles around the conflict of choosing between human and machine labor. The story starts off with three of our main characters, the Friendlies: the corporate employees who serve as our protagonists as well as your playable characters in the game. The three Friendlies are called into a conference meeting, where it is announced by the Boss that they are all fired (Figure 1).

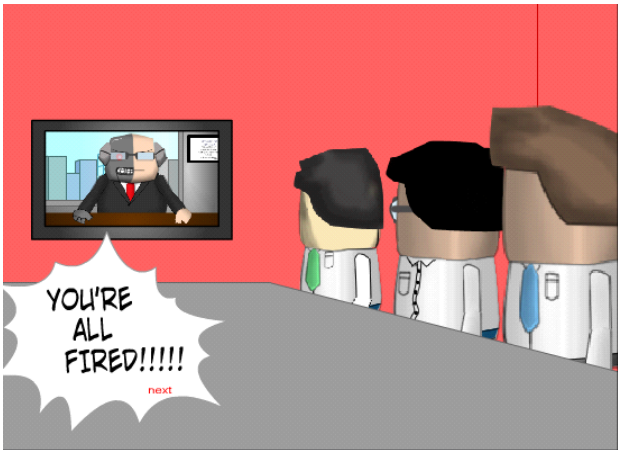


Figure 1. The corporate boss announces to the Friendlies that they are fired.

In a state of shock, the Friendlies are speechless as the Boss welcomes into the room their new replacements. To more of the Friendlies' shocks, they find that their replacements are, in fact, exact copies of themselves but in robotic form. The Boss

gladly explains to the Friendlies that the Robots are beneficial to both him and the company since they are more reliable, productive, obedient, and extremely cheap (Figure 2).

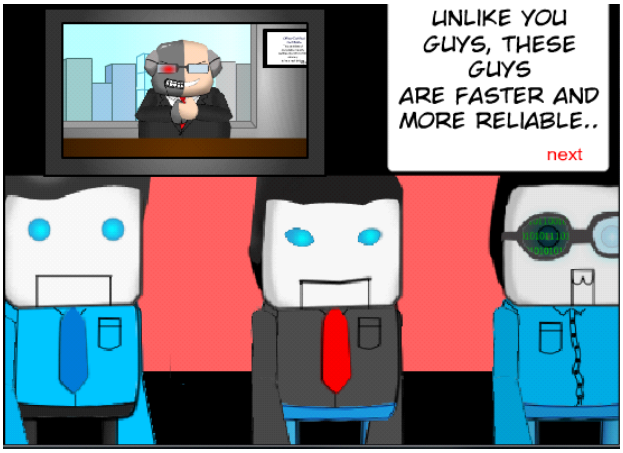


Figure 2. The corporate boss explains the advantages of having robotic employees versus human

Infuriated, one of the Friendlies, Joe Bob, stands up against this proposal and is challenged by the boss to stop him, thus forcing Joe Bob and his partners to fight for their right to voice against this decision as well as for the sake of keeping their job (Figure 3).

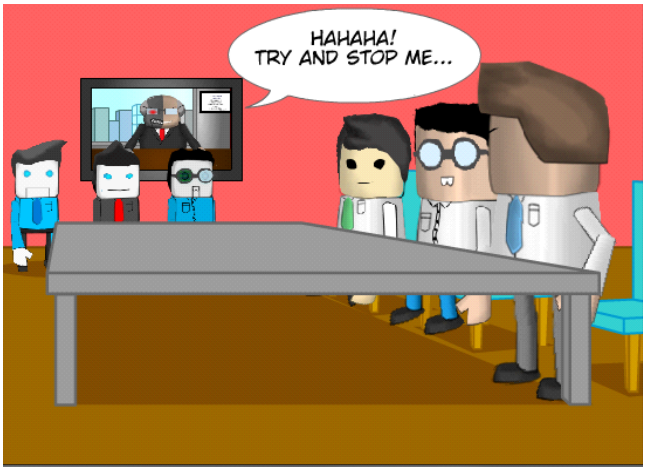


Figure 3. The Friendlies stand up to the corporate boss, opposing his plans

The game takes you through three levels, and in each level you play as the Friendlies, battling the Robots with each Friendly's different abilities. Some of the characters include:

- **Joe Bob (Brown hair, blue tie)** A regular employee that is also the most dedicated employee to the company. His abilities include Bash, Rush, Self Motivation, and Genuflect.
- **Slick Back (Black Hair, green tie)** A young man who is only interning for the company. Due to his young age and minor attachment to the company, he can come across as naïve or even irresponsible at times. His abilities include Bear Trap, Poison Dart, FlashBang, and Brain Drain.
- **John Doe (Black hair, glasses)** Although often picked on by other employees, John Doe is the IT Nerd as well as the brain of the company. The company would be nothing without his technical knowledge and skills and the employees know it. His abilities include Focused Shot, Rush, Self Motivation, and Genuflect.

Right after the final boss battle, the game switches to a cut scene of the infuriated, beaten Boss kicking the Friendlies out of his office. As the Friendlies stumble out of the office, the John Doe's glasses are knocked off onto the floor. His vision extremely blurred, John Doe drops down to his knees to look for the glasses, only to stumble upon the handle of a hidden file cabinet. The Friendlies gather together to force open the cabinet where they find what looks like a steering wheel to a ship. After the Boss leaves his office for a bathroom break, the Friendlies sneak into the office to figure out where the steering wheel belongs. Slick Back finds its rightful place by accidently locking the steering wheel into the Boss's desk. The Friendlies gather around to find out what it does by turning the wheel, which causes the building break off the ground and levitate into the distance, ending the game.

2.2 Game Design

Office Tactics is a turn-based strategy game for Windows PCs, similar to other games such as *Ogre Battle* and *Final Fantasy Tactics*. It is designed as a single player game centered around unit management and strategy. The game is built using the public version of Unreal Engine 3. *Office Tactics* challenges the player's ability to use a limited number of units to beat the ten planned levels of the game. Each of the player's units will level with each map completed and each enemy they defeat, but are gone for good when taken out by an enemy, making it so that the player has to be careful with how they deploy their units and how they approach each situation. There are six different unit types, with the AI having access to special Boss types as well. The player has a choice between either a melee or ranged physical type, a direct damage or control 'magic' type, and a healer or a buff/debuff type. The player starts with one of each, along with an extra ranged and melee character, though they cannot use all of them at once. Each level has a set limit on how many characters a player can have, determined by the cost of the units and the cap on spending for each level; typically around 2500 with each unit being 500.

Each level has a different mix of enemy units based on both the classes that the player can use, and Boss enemies that are either a combination of 2 of two class types, or a unique NPC that has a whole different move set from a normal class. The maps themselves are designed around a cartoony image of an office in chaos, with cover and obstacles based on cubicle walls and other things that may be found in an office, or whatever odd ball thing that might seem interesting or comical from a visual standpoint.

The characters are based on different character tropes and archetypes that would be found in something like *Office Space* or *The Office*, while the main antagonist can best be described as Gordon Gekko-1000. Enemies are designed to look like mechanical versions of the base classes, but as if built by a modern tech company; sleek, smooth, and cheaply made.

All the in-game audio and music is completely original, created specifically for the game. The music is meant to be light and comedic, with 5 unique tracks that play over each map, with a specific song for certain unique events like the tutorial and the final boss battle. The audio includes the effects that play with each unit's actions, and different office ambiance cues as the player looks around the map.

2.3 Game Mechanics

Mechanically the game is a turn-based system where, starting with the human player, each side takes turns controlling their units. On that player's turn, each unit may move and perform up to one action in any order. The game is played on a grid board, with the size based on the size of the map; the average map is a 64x64 sized grid, with each square being a 1 unit by 1 unit square in Unreal. At the start of a game, the player selects the units they wish to use from a list that lets them see their stats and abilities, along with the costs of each unit and the limit they have on that map. Units are deployed one at a time, until the player has spent the amount of points they want up to the cap.

Every unit has a set of seven stats that determine their effectiveness in combat. These stats are their Hit Points (HP), Ability Points (AP), Physical Power/Defense, Ability Power/Defense, and Movement speed. Each of these except for Movement is determined by a unit's class and level, with a unit's level determined by how much Experience it has gained. Ability Points (AP) are spent in units to use their special attacks, which are different depending on the unit using them, and any effect they have is based on the unit's Ability Power; if it is an offensive ability, it deals damage based on a comparison of the attacker's Ability Power to the defender's Ability Defense. Every unit has a basic attack that is either melee or ranged, depending on the unit, and deals damage similar to an Ability, except using the Physical stats. Movement speed is what determines how many squares a unit can move each turn. Movement can be blocked by other units and obstacles, making placement and movement order important.

Every attack and ability has a set of stats as well that, when combined with the each unit's base stats, determines how powerful they are. The stats are Range, Power, Area of Effect, which has either a cone, line, or burst shape; and AP cost for Abilities. Most units have a basic attack range of one square, and can attack in any direction around them in either a horizontal or vertical line, but not diagonally. Attacks and Abilities that have a range beyond one are targeted in the same manner as movement, with the action's Area of Effect shown on the grid. Bursts are centered on the player's cursor and can be centered anywhere in the action's range; meaning that units outside the actual range can be hit if they are still within the Area of Effect's radius. Line and cone actions are centered on the unit using it, and fire out in the selected direction similar to a Range one attack, but either a line of squares in that direction if a Line, or in a cone of squares in that direction if a Cone. As an example a basic ranged Attack, and certain abilities, is effectively an attack with an Area of Effect Burst of zero.

When a unit's HP reaches zero it is considered defeated, and if an NPC, the player unit that defeated it gains a set amount of Experience determined by the level of the unit compared to the type and level of the NPC defeated. When a unit gains enough Experience to level, they gain it immediately and have their stats recalculated, along with regaining health and ability points up to their new maximum at that level.

Between levels each unit's HP and AP is replenished, however in game each of the caster types has a way to replenish AP, and the healer class obviously can restore the HP of units on the player's side; either with a weak Area of Effect heal, or a stronger single target heal.

Finally, the current goal of each level is to clear it of all enemy units, allowing the player to advance to the next level of the game. There are plans to include different mission types, but those are time dependent.

3 Inference Engine

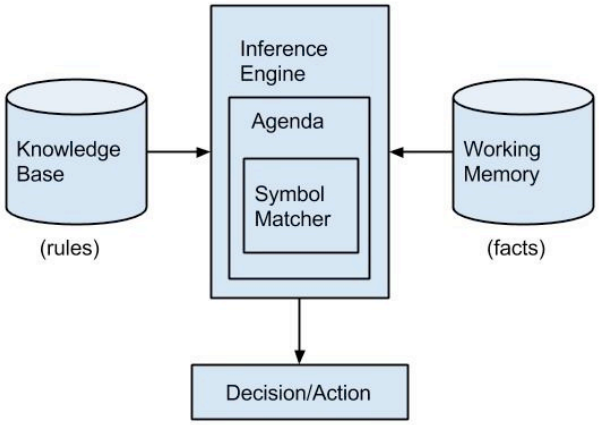


Figure 4. Architecture of an expert system showing the inference engine

An inference engine, see Figure 4, is designed to employ methods of plausible reasoning. Indeed when designing an inference engine there are many methods that can be implemented to perform reasoning, such methods include forward chaining, backward chaining, fuzzy logic, and bayesian logic, to name a few.

Among the many different methods of reasoning that could have been used for *Office Tactics,* forward chaining stood out as being the most practical and easiest method to implement into the game. Forward chaining allows for the inference engine to make use of the if-then structure of the rule base in order to locate justified rules that allow the engine to conclude the consequent (Then Clause), as a result new information is added to the games working memory [5].

For example, suppose the goal is determine the best possible move that the AI should make based on the following rules:

- **If** Target distance is between 6 and 10 **Then** Move to target
- **If** Target is 1 space away **Then** check Player's available action points
- **If** Player's available action points are less than 10 **Then** perform basic attack

Assuming the following facts:

- Target is 1 space away
- Player's available action points is 6

With forward chaining, the inference engine can ascertain that the best possible move is to perform a basic attack in three steps:

- Target is 1 space away

Based on that logic, the inference engine is then asked to check the Player's available action points.

- Player's available action points is less than 10

Based on rule 3, the inference engine can derive

- Perform basic attack

Just as the name "forward chaining" sounds, it is based on the fact that the inference engine starts with data and reasons its way to the best possible move.

3.1 Knowledge Base

A knowledge base is comprised of two key components: facts, and rules. Facts are assertions that change rapidly through the course of a program, and generally represent short-term information corresponding to rules that have been proven true. Rules form the representation of knowledge that originates from an expert on a specific domain. Unlike the short-term duration of facts, rules serve as long-term information about how to generate new facts. In most expert systems, rules are expressed in natural language typically following an IF conditional THEN consequent structure, for example:

Rule 1
IF the target is 1 space away, **AND** available action points is less than 10 **THEN** the AI should perform a basic attack.
Rule 2
IF the target is 1 space away, **AND** available action points is greater than or equal to 10 **AND** the target is below 50% health points. **THEN** the AI should perform a stronger ability.

These rules cannot directly be embedded in program code due to the nature of natural language; instead they can be represented by decision trees, semantic nets, or predicate calculus.

3.2 Agenda

An Agenda is a prioritized list of rules prepared by the inference engine [8]. Rules put onto the agenda are satisfied by the facts from working memory. When the inference engine locates facts that satisfy the conditional portion of a rule it adds the rule to the agenda. In order for a rule to be put on the agenda the entire conditional portion of the rule has to be proven true, even when there are multiple patterns that need to be satisfied, for example:

Rule 2
IF the target is 1 space away, **AND** available action points is greater than or equal to 10 **AND** the target is below 50% health points. **THEN** the AI should perform a stronger ability.

In this example, Rule 2 has two conditions: the target has to be 1 space away, and available action points need to be greater than or equal to 10. If both of these conditions are met then this rule gets added to the agenda. When the inference engine is ready to fire rules from the agenda there are several methods that can be applied, such methods include FIFO (First In First Out), precedence, and number of antecedents.

- **First In First Out** works by adding rules to the agenda in the order in which they were proven true in the knowledge base, and the inference engine fires those rules in the same order in which they appear in the agenda.
- **Precedence** is a system in which rules are given precedence values, and rules with higher precedence are fired first.
- **Number of Antecedents** is a system that fires rules based on the number of antecedents. According to this kind of logic, rules that have more antecedents have more requirements, which are likely to be more accurate, and more likely to provide the shortest path to the goal.

3.3 Working Memory

Working memory consist of a collection of facts that are stored to be used later on by the rules. The inference engine uses working memory to retrieve known facts in an attempt to satisfy the conditional portion of a rule. Facts that are applied to rules generate new facts that are added to the working memory forming a continuous cycle.

4 Implementation Using UDK Scripting Language

All the components of the inference engine used by the Office Tactics video game were written in the Unreal Development Kit's native scripting language UnrealScript. UnrealScript is used for authoring game code, and game events. Similar to high-level programming languages such as Java, and C++, UnrealScript is object-oriented. The language was designed to be simple, yet powerful enough for high-level game programming.

4.1 Working Memory Implementation

The goal of working memory is to provide new facts that have been generated by previous calls to the inference engine. The engine constantly updates simple facts, such as an enemy units current health, throughout game play. For example, a call to working memory is made in the beginning of each of the AI unit's turns. This call is also done before any reference to knowledge base to ensure that the facts being retrieved are those that have been updated by the previous turn.

```
var CorPIE_WM  WorkingMem;
WorkingMem = new ()  class 'CorpIE_WM' ;
```

Figure 5. The above snippet of code illustrates a call to working memory in UnrealScript.

Most of the information needed by the working memory is data regarding the player units that are targeted by AI units. Information such as ability points and health points are of primary concern. Before executing any rules in the knowledge base calls to working memory are made to retrieve facts regarding the AI's target. Some examples of functions used by working memory to retrieve data include:

```
/ / TODO:  Get pawn's hit points accessor function
function int getCurrentHealthPoints (CorpPawn pawn)
{
  return pawn.CurrentHealthPoints;
}
```

Figure 6. The above snippet of code is an accessor function that returns a pawn's current health points

```
/ /  TODO:  Get pawn's ability points functions
function int getCurrentAbilityPoints (CorpPawn
pawn)
{
  return pawn.CurrentAbilityPoints;
}
```

Figure 7. The above snippet of code is an accessor function that returns a pawn's current ability points

4.2 Knowledge Base Implementation

The rules in a knowledge base are expressed in natural language, but unfortunately computers don't have the capabilities to understand natural language due its ambiguity. However natural language is an exceptional start when designing the expert rules that will be included into the knowledge base. Once a list of expert knowledge has been accumulated it is only a matter of translating those rules from natural language into high-level computer language such as UnrealScript. For example, when determining the actions of a support unit, several facts are needed to ascertain the best move that should be made. Consider the following rules as an example:

Rule 3
IF the target is 1 space away **AND** available action points is greater than or equal 20, **AND** the target has 50% or less health, **AND** only allies are in the area. **THEN** perform a Group Heal.
Rule 4
IF the target is 1 space away **AND** available action points is greater than or equal to 10, **AND** the target has 50% or less health. **THEN** perform a heal on target.

Rules 3, and 4 can be translated into UnrealScript, the following is an example of the kind of an if-then structure taken from the knowledge base.

```
If(TargetPawnDistance == 1)
{
  if(UnitActionPoints  >== 20 &&
                TargetPawnObjective.CurrentHealthPoints
                <=  (0.50
                *float(TargetPawnObjective.MaxHealthPoi
                nts)) &&
                GroupHealArea.Length > 1 &&
                AllyOnlyArea(1))
                {
                  / /GroupHeal
                AgendaQueue.AddItem(Ability);
                }
                else if(UnitActionPoints  >= 10 &&
                  TargetPawnObjective.CurrentHealthPoin
                ts <= (0.50
                *float(TargetPawnObjective.MaxHealthPoi
                nts)))
                {
                  / /Heal
                  AgendaQueue.AddItem(Ability0);
                }
}
```

Figure 8. The above snippet of code represents rule 3 and rule 4 in UnrealScript.

Even though the original design for the rules are in natural language the statements can easily be expressed through the use of comparative operators, variables, and objects in UnrealScript. The 'then' portion of each rule is equally as imperative to the knowledge base as the conditional portion, because it is within this block of code that new rules enter the life cycle of the game. When a rule has been justified, a keyword, or symbol gets pushed onto a data structure for later use. For this particular kind of expert system a **symbol** represents various problems concepts, actions, and is used for applying strategies to reach a certain conclusion. In UnrealScript a symbol is represented as a constant, and stands for some concept related to the Office Tactics video game. Examples of the symbols used in the Office Tactics video game include:

- **Ability [0-5]**: This symbol is used to represent the type of ability a unit should use in a certain situation, in which case the AI goes into an Ability Standby State.
- **BasicAttack**: This symbol is used notify the system that a basic attack should be used on a target, in which case the AI goes into an Attack Standby State.
- **MoveToTarget**: This symbol is used to notify the system that the AI should move toward a target, in which case the AI goes into a Move Standby State.
- **Wander** This symbol is used to flag the system that there are no moves to be made, in which case the AI is allowed to make to traverse the map randomly.

These symbols were given names that represent the actions that they embody. These rules once proven to be true will cause symbols to be added onto a data structure that serves as the agenda of the inference engine.

4.3 Agenda Implementation

The agenda implementation for Office Tactics works a little differently than other expert systems. Generally an agenda would fire rules base on their precedence level. Rules with higher precedence leave the agenda quicker than those that have lower precedence. However for Office Tactics the agenda works off a FIFO principle. The agenda for this implementation is a queue that allows the inference engine to fire rules in the same order in which they appear in the agenda. A queue based implementation allowed for easier manageability, and development. The set of symbols representing different rules can be added onto the agenda in many different combinations, allowing for the AI to perform actions that are less predictable, and more likely to lead to greater results than a static finite state machine.

```
Function Agenda ()
{
  local int Action;
  NumberOfActions = AgendaQueue.Length;

  Foreach AgendaQueue(Action)
  {

    if(Action < 6)
    {
      ThisPawn.SetAbility(Action);
      If(NumberOfActions == 1)
      {
        GotoState('SpecialAbilityStandby');
      }
                  }
    else if(Action == 6)
    {
      if(NumberOfActions == 1)
      {
        GotoState('AttackStandby');
      }
    }
    else if(Action == 7)
    {
        GotoState('Standby');
    }
```

```
    else if(Action == 8)
    {
        WanderState();
    }
  }

  AgendaQueue.Remove(0, AgendaQueue.Length );
}
```

Figure 9. The above snippet of code represents the structure of the agenda used in the Office Tactics video game.

Based on the order in which the rules are added onto the agenda, and the unique constant value that each symbol is represented by, the agenda directs the system to the appropriate state until all the rules have been fired.

5 Conclusion

Historically expert systems have been less than enthusiastically received by the software engineering community [2], and has led many designers to favor finite state machine in their designs. However, FSM have their limitations such the lack of knowledge manipulation, predictability, and rigid design. These pitfalls in FSM are generally resolved through the use of Inference logic provided by an expert system. Some notable benefits of expert system integration include [1,3,4]:

- **Overcoming toy domains:** When integrated into software the expert system is able to be targeted at specific components, therefore, a small scale inference engine can make a meaningful contribution to large scale projects.
- **Ability to manipulate knowledge:** Experts systems allow of the manipulation of knowledge, this differs from conventional programming that generally only manipulates data.
- **Use of symbols:** A symbol is simply a string of characters that can represent various problem concepts that apply to strategies, and heuristics to reach a conclusion.
- **Quality of analysis:** Due to the fact that an expert system's knowledge base is comprised of information from multiple specialists in a specific domain it has been proven to perform better than their human counterparts.
- **Control:** Experts systems allow for improved control over various aspects of operations. Through the use of symbols to represent facts and ideas, users of the system are able to identify and monitor the progress of each inference result, which is unlike traditional systems that require the backtracking of chained events.

Inference engine logic is most effective in applications that require vast amount of expert knowledge and in cases where the best choice out of many permutations of outcomes is required. For these reasons inference logic is most often integrated into video game artificial intelligence to handle the high demand of possible outcomes that a complex game can produce. Such an example includes the application in Chinese chess, where the aim of the inference engine is to find the best move in the game's large decision tree. The endgame knowledge base for Chinese chess aids in determining infrequent winning nodes or inevitable draw nodes, which are then removed from consideration. As a result, the chances of determining a winning node go up as well as the engines chances of finding the best way to win. When dealing with large complex systems, it is apparent that FSMs have their limitation, as the possible permutation of outcomes become too large they become increasingly more difficult to manage. For reasons such as this, it is imperative to take advantage of the control, and expertise that an inference engine can provide for complex game systems.

References :

[1] Forsyth, R. (1984). Expert Systems Principles and Case Studies. New York, NY: Chapman and Hall Computing.

[2] Gillies, A. C. (1991). The Integration of Expert Systems Into Mainstream Software. New York, NY: Chapman and Hall Computing.

[3] Goldenthal, N. (1987). Expert Systems and Artificial Intelligence. Cleveland, OH: Weber Systems, Inc.

[4] Laswell, Lawrence. K. (1989). Collision Theory vs. Reality in Expert Systems. Wellesley, MA: QED Information Science, Inc.

[5] Robin. "Rule Based Expert Systems." *Artificial Intelligence: Articles On Artificial Intelligence,* November 1st, 2010. September 29, 2013. <http://intelligence.worldofcomputing.net/expert-systems-articles/rule-based-expert-systems.html#.Uku--IashcY>.

[6] Tozour, Paul, "Stack-Based Finite-State Machines," AI Game Programming Wisdom 2, Charles River Media, 2003.

[7] "Introduction to Game Development," 2nd Ed., Edited by Steve Rabin, Charles River Media, 2010.

[8] "Expert Systems/The Agenda." *Wikibooks*, January 3rd, 2008. September 29, 2013. <http://en.wikibooks.org/wiki/Expert_Systems/The_Agenda>.

Topic-Model based Query Intent Prediction for Search by Multiple Examples

Mingzhu Zhu, Chao Xu, and Yi-Fang Brook Wu
Department of Information Systems, New Jersey Institute of Technology, Newark, New Jersey, USA

Abstract – *It is often difficult for users to express their information needs as keywords. Search-By-Multiple-Examples (SBME) is a new search paradigm that provides an easier way for users to express their information needs as multiple relevant examples (called query examples) rather than as a simple string of keywords. One key issue of SBME is to identify user's true information needs from the query examples, which may contain multiple topics. However, none of the previous studies on SBME considers the issue of topic diversity of the query examples. In this research, we explore the solutions to the topic diversity issue in SBME through adopting topic modeling techniques to predict the likelihood that the query examples belonging to a topic. The learned topic distributions from the query examples are used to build query vectors for document ranking. Using Mean Average Precision (MAP) and precision at k (p@k), experiments conducted on two benchmark datasets show that the proposed method outperforms the two baselines significantly, especially when a large number of query examples are available.*

Keywords: Information need, Search by Multiple Examples, Topic Modeling, Information Retrieval

1 Introduction

A key to the success of using modern keyword-based IR systems to satisfy an information need lies in how users express it using keywords as queries. However, it is often difficult for many users to express their information needs using a simple string of keywords, especially when the information needs are complex. There are also many cases where the keyword-based search is unsatisfactory, as most of the knowledge in the modern era is a combination of concepts, topics, methods, and ideas. For instance, with the growing popularity of interdisciplinary studies, researchers often need to search articles related to multiple domains (e.g., searching articles on using machine learning techniques for entity extraction in bioinformatics), it would be difficult for them to express the information needs using keywords, especially when they are not familiar with the domain of the articles of interest. An alternative search paradigm called Search by Multiple Examples (SBME) has been proposed recently to overcome the shortcomings of the keyword-based search. It allows users to express their information needs using multiple examples rather than as a simple string of keywords [1][2][3]. In the article search scenario, having some articles of interest at hand, the researcher can use them to represent their information needs.

The previous studies on SBME can be classified into three categories, according to the methodologies of adopting the query examples for documents ranking through 1) building concept networks using citation based graph analysis techniques [1]; 2) using Positive Unlabeled learning (PU learning) models by considering the query examples as positive data and the documents in the database as unlabeled data [2][3]; and 3) selecting important terms from the query examples to build query vectors for conducting traditional keyword-based search.

The main limitation of the graph analysis based method is that some important and relevant documents for an information need may never be cited as they are new articles or their importance has not been identified by the community. The PU learning based methods can be used in a broader area, but they are not efficient for online search when all the documents in the online database are considered as unlabeled data, as the high volume of the unlabeled data can downgrade the efficiency of a PU learning algorithm dramatically. Although the term selection methods (e.g., centroid method) are efficient as the terms can be used directly to conduct standard keyword-based search using modern IR systems, these keyword selection methods may be biased as they consider the query examples belonging to a single topic, which may actually belong to multiple topics. For instance, a set of documents related to "Information Retrieval" may contain some other topics such as "business" or "biology" as the background information. It is biased to build query vectors from the query examples without considering their topical diversity.

Based on the hypotheses that the true information needs consist of some aspects of the query examples, and each document can be related to more than one topic, we employ a topic model to predict the probabilities that the query examples belong to a certain topic. Then query vectors are built using the latent topic distribution information for document ranking.

Using Mean Average Precision (MAP) and precision at k (p@k) measures, experimental results on two benchmark datasets indicate that the proposed method outperforms the baselines significantly, especially when a large number of query examples are available.

In the following sections, we will review the related work in section 2, describe the topic modeling based search intent prediction methods in section 3, illustrate the experimental design and results in section 4, and present conclusions and discussions in Section 5.

2 Related Work

In this section, we will first present Search by Multiple Examples and Topic Modeling, which form the background of this research. Then we review the studies on information intent modeling.

2.1 Search by Multiple Examples

Different from the traditional keyword-based search, SBME allows users to express their information needs using a set of relevant examples rather than a set of simple string of keywords. Although it is a valuable research topic, there are only a few studies in the literature [1][2][3]. In [1], the entire list of references of a group of papers is adopted to find related publications. The query is defined as a small set of papers that are relevant to the research task at hand. More relevant articles are selected by "optimizing an object function based on a fine-grained notion of influence between documents." The authors define a directed, acyclic graph for each concept in the document collection to capture how ideas travel between documents. In the graph, nodes represent articles that contain the concept and the edges represent citations and common authorship. The degree of influence is captured by calculating a weight for each edge between two nodes. The weight of the edge represents the probability of direct influence from the head to the tail of the edge with respect to the concept that is contained by the articles (nodes) in the graph. Given these edge weights, they define a probabilistic, concept-specific notion of influence between any two articles in the document collection. One concern of this method is that not all documents are cited because some articles may be newly published or their importance has not been recognized by the community.

Another direction for SBME is based on PU learning. Employing the same idea of using multiple examples to express users' information needs, Zhang and Lee [2] consider the problem of SBME from the perspective of PU learning in two scenarios: online queries and offline queries. For the online queries, they propose to learn a one-class SVM model from the query set; for offline queries, they propose to learn an SVM model from the query set as well as the entire corpus. Zhu et al. [4] also adopt PU learning and propose an under-sampling based method to solve the class imbalance problem: the size of positive examples is much smaller that the size of the unlabeled data.

The main issue of these studies is that it is inefficient to consider the entire data collection as unlabeled data, as the high volume of documents in an online database may downgrade the efficiency of a PU learning algorithm dramatically.. Moreover, the one-class SVM performs poorly in PU learning [5], especially when only a small number of positive examples are available, as it does not adopt useful information in the unlabeled examples. Thus the one-class SVM is actually not a good approach for online search.

Duh and Kirchhoff [3] present a transductive learning framework exploring how to improve ranking performance using partially labeled data. They adopt KernelPCA to generate better features from the unlabeled data and use the features via Boosting for learning different ranking functions adapted to the individual test queries. As the method in [3] needs both positive and negative training data, it is not applicable in the setting where users only have positive examples.

In this study, we aim to generate high quality queries from the query examples by employing topic modeling techniques to represent users' true information needs. Then query vectors are built using the topic distribution information to rank documents in the online database directly.

2.2 Topic Modeling

In machine learning and nature language processing, topic models [6][7][8] are statistical models that uncover the hidden abstract "topics" in document collections. The basic idea behind topic models is that documents are mixtures of topics, where a topic is a probability distribution over words. Using these models, we can develop new ways to organize, search, and browse the large size of document collections. Applications of statistical topic models in text analysis have received much attention in recent years, especially in information retrieval and text mining fields [9][10][11][12]. For instance, in [9][11], the authors adopt topic models to extract scientific research topics; in [12], an LDA-based document model is proposed for ad-hoc retrieval in the language modeling framework.

An early topic model is the probabilistic Latent Semantic Analysis (pLSA) proposed by Hofmann [7]. The assumption behind pLSA is that the interdependence between words in a document can be explained by the latent topics which the document belongs to. The word occurrences in a document are conditionally independent on an assigned topic. The most popular topic model is the Latent Dirichlet Allocation (LDA) [6], which is extended from the pLSA. The basic assumption behind LDA is that documents are associated with latent topics, and the corpus is modeled as a Dirichlet distribution of the topics, where each topic is characterized by a distribution over words. Based on this assumption, each document is represented as a probability distribution over some topics, and each topic is represented as a probability distribution over a number of words.

Using LDA, the topic distributions of each document can be inferred from the data collection. The results can be used to get the most likely topics that the query examples belong to. Then the set of words that are associated with these topics can be obtained. They will be chosen for representing users' true information needs to retrieve a set of potentially relevant documents from the database for the PU learning based ranking process.

2.3 Modeling Information Need

The quality of a query result is mainly determined by two factors: 1) the quality of user's queries that represent the information needs; and 2) the effectiveness of the ranking module of an IR system. Therefore, for a given IR system, the key to improve the quality of the search result for an information need is to improve the quality of the query. However, there is still a gap between the user information need and the query that is often represented as text, which brings a big challenge for modern IR systems.

One approach for improving the query quality is query formulation. For instance, handcrafted controlled vocabularies are used for improving the quality of the original query. However, the effectiveness of such methods is largely depending on the quality of the controlled vocabularies, which may need extensive human input and oversight for creation and maintenance.

Another popular method for improving query quality is relevance feedback [20], which aims to improve the quality of the original query adopting the information from the labeled documents in order to reduce the distance between the input text query and user's true information need. For instance, the famous relevance feedback method, Rocchio algorithm, computes the query vector by taking into account both the documents in the positive set and the negative set using the following formula:

$\vec{q} = \frac{1}{|P|}\sum_{i=1}^{|P|} X_i - \frac{1}{|N|}\sum_{j=1}^{|N|} X_j$, where $X_i \in P$ *and* $X_j \in N$.

where P denotes the relevant document set, N denotes the irrelevant document set.

Relevance feedback has been shown effective in practice, but users are often reluctant to provide explicit feedback [13]. Pseudo feedback is an alternative method that assumes a certain number of top-ranked documents from the returned results of the original query to be relevant documents to update the original query [14][15]. However, some of the top ranked documents may be actually irrelevant in practice. The precision of the ranking system may be downgraded because of the polysemy effect: the added terms may have different meanings from user's intended meaning.

Query log based studies have been proposed to model the information needs associated with a query by analyzing the large sets of query log data. Wen et al. [16] compute query similarity using query distance and click through data. Fonseca et al. [17] adopt association rule methods for mining query logs and query sessions to discover the correlation between queries. Zhao et al. [18] use query session to compute query similarities by calculating their popularity over time. Different from these methods, which rely on large sets of log data, our method predicts query intents by utilizing the topic distributions from the query examples employing topic modeling techniques.

3 Research Methods

In this section, we present a topic model based method to predict user' true information needs from the query examples for SBME. We begin with an overview of the research framework (see Figure 1) and then describe each step in detail.

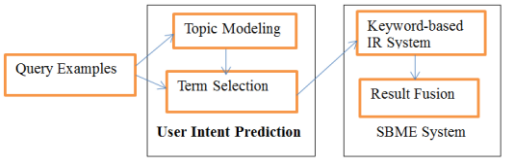


Figure 1. The Research Framework.

3.1 Overview

When user's search intent is represented using multiple examples, which are most likely belonging to multiple topics, it is biased to use the traditional methods such as centroid to build a query vector simply using the centroid of the vectors of the query examples. We adopt a topic model to conduct topic analysis on the query examples to obtain topic distributions, which are used to predict the most likely topics that the query examples may belong to. Then for each topic, we rank the terms based on the probability that they are belonging to the topic. These terms for each topic along with the corresponding probability values are used to build a query vector to conduct standard keyword-based queries using a traditional keyword-based IR system. Several queries are needed when the query examples are predicted as belonging to multiple topics. In this case, the result fusion module is used to rank all the documents that are returned from the keyword-based searches using the built queries.

3.2 Topic Model based Query Vector Construction

Based on the hypothesis that user's query examples belong to multiple topics, and the topics with higher probabilities are more likely to represent user's true information need, we adopt the most widely used topic model LDA to predict the topic distributions of the query examples. Using LDA, two probability distributions can be obtained: p(t|d), the probability that document d belongs to topic t; and p(w|t), the probability of a term w under topic t. Table 1 shows an example of the four topic distributions for a set of query examples with 10 documents. We can see

that only the 5th document is assigned to topic 2 (T_2) with probability 0.127, and none of the other documents is assigned to this topic. This suggests that the terms of topic 2 may not represent the true information need. On the other hand, four documents are assigned to topic 1 (T_1) with probability higher than 0.5 (i.e., the scores in bold), and three documents are assigned to topic 4 (T_4) with a probability higher than 0.7 (i.e., the scores underlined). This suggests that it is more likely that the terms of T_1 and T_4 should be used to represent the true information need.

In the following part, we formalize the idea of using topic modeling for information need prediction.

Table 1. An Example of Document Topic Distributions

Doc#	1	2	3	4	5	6	7	8	9	10
T_1	**0.79**	**0.981**	0.353	**0.515**	0.468	0	0	**0.566**	0	0.282
T_2	0	0	0	0	0.127	0	0	0	0	0
T_3	0.129	0.019	0.451	0.117	0.165	0.897	0	0.158	0	0
T_4	0.081	0	0.196	0.369	0.241	0.103	1	0.276	1	0.718

Let $D=\{D_1,D_2,\ldots D_n\}$ denotes the set of query examples, let K denotes the number of topics to train an LDA model. The likelihood that the query document D belonging to topic T_i is calculated as follows:

$$L(D,T_i)=\sum_{j=1}^{j=n} p(T_i|D_j)$$

Given a topic T_i, the probability that a term w should be selected is denoted as $p(T_i|w)$, which is calculated using the following formula:

$$p(T_i|w)=\frac{p(w|T_i)*p(T_i)}{p(w)}$$

where p(w) denotes the global distribution of term w, and $P(T_i)$ is calculated using the following formula:

$$p(T_i)=\frac{1}{n}\mathrm{L}(\mathrm{D},\mathrm{T}_i)$$

Let $S_i=\{w_{i1}, w_{i2},\ldots, w_{im}\}$ denote the terms that are associated with topic Ti, where wij denotes the jth term under topic Ti. The corresponding query vector with respect to Ti can be represented as $\vec{S_i}=<p(T_i|w_{ij}),\ldots p(T_i|w_{im})>$, where $p(T_i|w_{ij})$ is the weight of term wij.

3.3 Result Fusion

For any given document d in the database, the similarity score between d and S_i can be computed using the cosine method:

$$sim(\vec{s_i},\vec{d})=\frac{\vec{s_i}\cdot\vec{d}}{\|\vec{s_i}\|\|\vec{d}\|}=\frac{\sum_{j=1}^{j=m} p(T_i|w_{ij})*w_{dj}}{\sqrt{\sum_{j=1}^{j=m} p(T_i|w_{ij})^2 * \sum_{j=1}^{j=m} w_{dj}{}^2}}$$

where $\vec{d}$ represents the vector of document d, and w_{dj} denotes the weight of the feature w_{ij} in the vector $\vec{d}$. The ranking score of document d can be determined using the maximum value of the scores between d and the query vector of each topic (i.e., $\vec{s_i}, i\epsilon[1,k]$):

$$Rank(d)=\max\left(sim(\vec{s_i},\vec{d})\right), where\ i\epsilon[1,K] and\ i\epsilon N$$

4 Experiments and Results

In this section, we report the results of the experiments conducted on two benchmark datasets to evaluate the performance of the proposed method (called topic method) and the two baselines in a PU learning based framework. The first baseline is the tfidf based term selection method, which adopts the top terms that have the highest tfidf scores to build the query vector, and the second baseline is the centroid method that builds the query vector using the centroid of the query examples. We used Stanford core NLP[1] for stop words removal, stemming and lemmatization, and developed the IR system for this research using Lucene[2].

4.1 Data Collections

Two benchmark datasets are used for the empirical evaluation. The first one is the Reuters-21578 dataset, which is a commonly used benchmark news article collection in text classification. There are 135 potential topic categories. The same as previous studies [4], only the most frequent 10 documents are used as the source of user' query examples, while all other documents are used to form the unlabeled dataset.

The second dataset is the WebKB collection [19], which contains web pages gathered from university computer

[1] http://nlp.stanford.edu/software/corenlp.shtml
[2] http://lucene.apach.org

science departments. The pages are grouped into seven categories. In this study, we only use the four classes: course, faculty, project, and student, which contain most frequent instances.

4.2 Evaluation Methods

Two standard IR evaluation measures, Mean Average Precision (MAP) [20] and p@k, are adopted for evaluation. MAP represents the mean of the average precision (AP) scores. Let L be a ranked list of documents, and R be the relevant documents being retrieved. The AP score is calculated using the following formula:

$$\mathrm{AP(L)} = \frac{1}{|R|}\sum_{k=1}^{n}\big(p(k)\times rel(k)\big)$$

where $|R|$ is the number of retrieved relevant documents; k is the rank in the sequence of the retrieved documents; n is the number of retrieved documents; p(k) is the precision at cut-off k in the list; rel(k) is an indicator function equaling 1 if the item at rank k is a relevant document, and zero otherwise The MAP score is the mean arithmetical value of the AP scores.

In IR, users are often interested in the precision of the top returned documents [21], which is denoted as p@k. Compared with MAP, p@k is a more user-oriented measure, as users hope to find relevant documents by only scanning the top few (e.g. 30) documents in the returned search results.

4.3 Experimental Design

Each run of an experiment was conducted using the procedure proposed in [4]. For each topic of interest in a data collection, the documents belong to it were considered as positive examples to form the positive pool, and other documents in the same data collection formed the negative pool. Altogether, we obtained 10 positive and 10 corresponding negative pools from the Reuters data collection, 4 positive and 4 corresponding negative pools from the WebKB data collection, respectively. From a positive pool, a subset of |P| examples was randomly sampled to simulate user' query examples. The documents in the negative pool and the remaining documents in the positive pool form the searching space. For each rank, Average Precision, and p@k (k=10, 20, 30) were calculated. The MAP and average p@k scores for each run of the experiments were calculated using the mean of Average Precisions, and the mean of p@k for all the selected topics. For each data collection, we carried out 10 runs for an experiment, and the final MAP and average p@k scores were calculated by averaging over the MAP and average p@k scores from the 10 runs.

4.4 Experimental Results

To train an LDA model, the size of the latent topics (denoted as K) must be predefined. We randomly sample a subset of documents from the Reuters and the WebKB dataset respectively to conduct experiments trying different numbers of K to see how the system performance changes when the topic number varies. On average, for the sampled Reuters dataset, the maximum system performance achieved when K=20; for the sampled WebKB dataset, the maximum system performance achieved when K=10. Therefore, we chose K=20 for the Reuters dataset and K=10 for the WebKB dataset for the performance comparison.

We compare the performance of the proposed method (denoted as Topic) with two baselines: tfidf based method (denoted as Tfidf) and the centroid method (denoted as Centroid). The results are shown in Table 2 and Table 3 for the two benchmark datasets respectively. The best results for each experiment are presented in bold. It can be observed that with the increase of |P|, the performance of all the methods tend to increase. This observation is consistent with that in the previous studies [4]. In all cases, the topic based method performs better than the two baselines. When |P| is small (e.g., |P|=2), the topic based method outperforms the baselines slightly in terms of p@10, but the performance improvement is significant in terms of MAP. However, when |P| is large, (e.g. |P|=20), the topic based method outperforms the baselines significantly in terms of both MAP and p@10. For instance, for the Reuters dataset, when |P|=20, the p@10 value for the topic based method is about 10% higher than the centroid method, and 6% higher than the Tfidf method. From the WebKB dataset, when |P|=20, the p@10 value for the topic based method is 17% higher than the centroid method and 6% higher than the Tfidf based method. This observation indicates that the prosed method can deal with the topic diversity problem by adopting the topic distribution information in the query examples, thus the system performance can be improved.

Table 2. Experimental results on Reuters dataset, where the bold numbers indicate the best performance

|P|	Algorithm	MAP	p@10	p@20	p@30
2	Topic	**0.604**	**0.812**	**0.805**	**0.791**
	Centroid	0.551	0.788	0.759	0.745
	Tfidf	0.442	0.789	0.732	0.703
5	Topic	**0.702**	**0.925**	**0.913**	**0.893**

	Centroid	0.684	0.883	0.868	0.825
	Tfidf	0.452	0.857	0.783	0.742
10	Topic	**0.825**	**0.935**	**0.928**	**0.873**
	Centroid	0.773	0.887	0.894	0.826
	Tfidf	0.520	0.915	0.838	0.815
20	Topic	**0.842**	**0.943**	**0.925**	**0.914**
	Centroid	0.781	0.847	0.886	0.838
	Tfidf	0.56	0.886	0.856	0.845
50	Topic	**0.837**	**0.875**	**0.87**	**0.892**
	Centroid	0.813	0.825	0.831	0.84
	Tfidf	0.573	0.903	0.865	0.821

Table 3. Experimental results on WebKB dataset, where the bold numbers indicate the best performance.

|P|	Algorithms	MAP	p@10	p@20	p@30
2	Topic	**0.482**	**0.697**	**0.648**	**0.651**
	Centroid	0.457	0.652	0.613	0.621
	Tfidf	0.435	0.577	0.607	0.625
5	Topic	**0.522**	**0.697**	**0.688**	**0.672**
	Centroid	0.411	0.632	0.605	0.613
	Tfidf	0.423	0.615	0.622	0.604
10	Topic	**0.599**	**0.765**	**0.750**	**0.739**
	Centroid	0.555	0.713	0.707	0.686
	Tfidf	0.459	0.732	0.665	0.675
20	Topic	**0.655**	**0.808**	**0.793**	**0.836**
	Centroid	0.612	0.633	0.624	0.66
	Tfidf	0.509	0.73	0.71	0.68
50	Topic	**0.679**	**0.755**	**0.776**	**0.770**
	Centroid	0.61	0.713	0.705	0.716
	Tfidf	0.516	0.74	0.727	0.713

5 Analysis and Conclusions

Previous studies on SBME assume query examples contain only one topic; in reality, they would most likely contain multiple topics. The ignorance of the topic diversity issue of the query examples in the previous studies on SBME motivates us to explore how to predict user's true information needs from the query examples. Based on the hypothesis that the most likely topics that the query examples belonging to are more likely representing user's true information needs, we adopt topic modeling for information need prediction. The experiments on two benchmark datasets endorse the hypothesis.

One concern of the proposed method is that the number of topic K should be predefined. We sampled a subset of documents from the Reuters and WebKB dataset respectively to investigate how the system performance changes when K varies. The results indicated that: 1) the change of K indeed affects the ranking performance; the system performance downgrade when K is too small or too large, and 2) on average, setting K=20 on the sampled Reuters dataset and K=10 on the sampled WebKB dataset leads to the highest MAP of the proposed method.

The system performance also depends on the size of query examples. The experimental results show that the topic based method performs better than the baselines with more relevant examples used as the query. This is because the more

the query examples, the more likely that they are belonging to different topics. The prosed method can deal with the topic diversity problem by adopting the topic distribution information in the query examples, thus improving the system performance.

Previous studies demonstrate the effectiveness of SBME, but they are not scalable for online search. It is inefficient for the PU learning based method using all the documents in an online database as unlabeled data, which are often huge. The semi-supervised learning methods can downgrade dramatically when a huge number of documents are used for training. Our research of building high quality keyword queries from query examples makes it possible to conduct SBME online by leveraging the modern IR techniques.

6 References

[1] K. El-Arini and C Guestrin. Beyond keyword search: discovering relevant scientific literature. *Proceedings of the 17th ACM SIGKDD international conference on knowledge discovery and data mining, 439-447, 2011*.

[2] D. Zhang and W.S. Lee. Query-By-Multiple-Examples using Support Vector Machines. *Journal of Digital Information Management*. 7(4): 202-210, 2009.

[3] K. Duh and K. Kirchhoff. Learning to rank with partially-labeled data. *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, 251-258, 2008.

[4] M. Zhu, C. Xu, and Y. B. Wu. IFME: information filtering by multiple examples with under-sampling in a digital library environment. *Proceedings of the 13th ACM/IEEE-CS joint conference on Digital libraries*. 2013.

[5] Xiaoli Li, and Bing Liu. Learning to classify texts using positive and unlabeled data. *Proceeding of IJCAI 2003*, vol 3.587-592. 2003.

[6] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 3, 993-1022, 2003.

[7] T. Hofmann. Unsupervised Learning by Probabilistic Latent Semantic Analysis. *Machine Learning Journal*, 42(1), 177-196, 2001.

[8] T. L. Griffiths and M Steyvers. A probabilistic approach to semantic representation. *Proceedings of the 24th Annual Conference of the Cognitive Science Society*, 2002.

[9] T. L. Griffiths and M. Steyvers, Finding scientific topics, *Proceedings of the National Academy of Sciences of the United States of America*, 2004.

[10] M. Steyvers, P. Smyth, M. Rosen-Zvi, and T. Griffiths, Probabilistic author-topic models for information discovery, *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2004.

[11] D. Blei and J. Lafferty, Correlated topic models, *Advances in neural information processing systems*, vol. 18, p. 147, 2006.

[12] X. Wei and W. B. Croft, LDA-based document models for ad-hoc retrieval, *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, 2006.

[13] Joachims, T. (2002). Optimizing search engines using clickthrough data. *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining.*

[14] Tao, T., and Zhai, C. (2006). Regularized estimation of mixture models for robust pseudo-relevance feedback. *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*.

[15] Shen, X., and Zhai, C. (2005). Active feedback in ad hoc information retrieval. *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*.

[16] Wen, J.-R., Nie, J.-Y., and Zhang, H.-J.. Clustering user queries of a search engine. *Proceedings of the 10th international Conference on World Wide Web*, 2001.

[17] Fonseca, B. M., Golgher, P. B., Moura, E. S. d., & Ziviani, N. (2003). Using association rules to discover search engines related queries. *Proceedings of the First Latin American Web Congress (LA-WEB'03).*

[18] Zhao, Q., Hoi, S. C. H., Liu, T.-Y., Bhowmick, S. S., Lyu, M. R., and Ma, W.-Y. (2006). Time-dependent semantic similarity measure of queries using historical clickthrough data. *Proceedings of the 15th international conference on World Wide Web.*

[19] M. Craven, D. DiPasquo, D. Freitag, A. McCallum, T. Mitchell, K. Nigam, and S. Slattery. Learning to extract symbolic knowledge from the World Wide Web. *Proceedings of AAAI-98*, 1998.

[20] E. Agichtein, E. Brill, S. Dumais. Improving web search ranking by incorporating user behavior information. *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 19-26, 2006.

[21] C. D. Manning, C.D. P. Raghavan., H. Sch¨utze. Introduction to Information Retrieval. *Cambridge University Press*, 2008.

Semantic and spatial similarity in geographic information retrieval

N. Machado[1], **Y. Marrero**[1], **C. Balmaseda**[1] **and, A. Montoyo**[2]
[1]Computer Science Department, Agrarian University of Havana, San José de Las Lajas, Mayabeque, Cuba
[2]Director of Technical School, University of Alicante, Alicante, Spain

Abstract - *People use the similarity to store and retrieve information, also for learning and concept formation. Similarity plays a fundamental role in many applications such as decision-making systems, data mining and pattern recognition. The same applies to the spatial similarity in the processes of recovery and integration of spatial information. In this paper a methodology based on the semantic processing of geospatial information is proposed. It consists of five stages: conceptualization, synthesis, application processing, retrieval and management. It uses ontology of New Genetic Soil Classification of Cuba and applying the measure of semantic similarity DISC combined with an implementation of model TDD (Topology-Direction-Distance) to restore and support the user in the selection of spatial scenes. As a case study is considered the town of San José de Las Lajas located in the province of Mayabeque in western of Cuba*.

Keywords: geographic information retrieval; geo-ontology; similarity measure; spatial relations; semantic relations

1 Introduction

In recent years the similarity measures have been used for information retrieval and therefore in the domain of spatial information. Initially applied in search engines and they have gradually become more visible integrated in user interfaces.

The similarity is a key element for analogical inference [1]. People use the similarity to store and retrieve information comparing new situations with similar experiences occurred in the past also for learning and concept formation [2]. While equality comparison between two objects can be calculated by computers quickly and accurately similarity is a complex problem to calculate. Even still difficult to determine similarity plays a fundamental role in many applications such as decision-making systems, data mining and pattern recognition. The same applies to the spatial similarity in the processes of recovery and integration of spatial information [3].

Unlike the similarity between documents, which works with the comparison of words, the spatial similarity depends on different elements such as relations and spatial distribution, geometric and thematic attributes and semantic relations [4]. In the semantic similarity between geospatial concepts is important to consider spatial relations in the calculation process.

Most semantic representations of spatial concepts are based on features or descriptors of the information they have an ambiguous character in the explanation of what these features represent. Many of the approach used to determine the semantic similarity apply to general domains, do not focus on the particular properties of the concepts in the spatial domain [5] and use a shared ontology that semantically related concepts. Models based on the semantic relations generally include two types: synonymy (equivalence) and hyponymy (is-a). In the spatial domain is important to consider the relation of meronymy (is part of) it provides a better representation between spatial concepts.

In this paper a methodology based on the semantic processing of geospatial information is proposed. It consists of five stages: conceptualization, synthesis, application processing, retrieval and management. It uses ontology of New Genetic Soil Classification of Cuba and applying the measure of semantic similarity DISC combined with an implementation model TDD (Topology-Direction-Distance) to restore and support the user in the selection of spatial scenes. As a case study is considered the town of San Jose de Las Lajas located in the province of Mayabeque in western Cuba.

This document is organized into five main sections. A summary of the most relevant work related similarity measures applied to geospatial data is presented in section 2. The proposed methodology is presented in Section 3. Then in Section 4 the main results of a first implementation and testing of the system are discussed. Finally are presented in Section 5 the conclusions and future research.

2 Related work

Currently much research is aimed at finding ways to formally codify the context and geographical relations. Comparison of concepts describing the meaning of data in distributed sources of information is an approach that aims to become an efficient solution to process geographic data from the semantic point of view [6].

The study of information retrieval systems in the field of Geosciences, concentrated their efforts on studying other ways of representing knowledge, change the way to store and organize data and search control mechanisms that extract information accuracy, to match exactly or similarly thrown responses to queries made by users.

The semantic similarity is essential for semantic processing of geospatial data. Sets the degree of semantic interoperability between different data and establish the basis for the recovery and integration of semantic information are the goals of several approaches [7].

In GIS similarity is particularly important because of the difficulty in obtaining satisfactory representations of geographical phenomena and there are variety of formalizations of spatial properties such as shape, location and spatial relations [8].

In [4] propose a similarity measure that integrates four models, the geometric model, feature model, model transformation and structured alignment model to calculate the similarities and differences between spatial scenes and layer level. Applies the priority order topology, direction, distance and processing costs are reduced. Both features are implemented through the application of weights.

An approach to integrate spatial relations and semantic similarity measures between geospatial concepts considering different spatial relations are key parts of the semantic description of the geo-data is presented in [9]. A spatial relations represented by a natural language according to the computational model are selected Sharif et al. The work is focused on measuring semantic distances at the conceptual level. An example of integration of geographic information system level are ruled by the GIS ontologies (SIGGO) acting as a system integrator regardless of the model [8].

The concept of ontology has attracted increasing attention in the community of information science because of its ability to achieve a representation of shared knowledge. Ontologies are importance in the creation and use of standards for data exchange as well as in solving problems of heterogeneity and interoperability of geographic data. Ontologies can be used as an alternative to represent the data and explicitly knowledge about them.

To the knowledge of semantic processing geospatial data stored in ontologies is essential to calculate the semantic similarity, which is essential in processing data queries from users and is the basis for the recovery and integration of semantic information [2].

3 Application ontology in spatial data extraction

The methodology is based on a semantic processing that locates a particular scene with determine characteristics. In addition images are retrieving. These images show to user a visualization of the characteristics and properties of the objects.

Apply semantic similarity measures and employs an ontology that includes geographical information relating to Soil Resources. The area selected for study was the municipality of San José de Las Lajas located in Mayabeque, one province in the western region of Cuba. Soils are classified according to their structure and composition defines in the New Genetic Soil Classification of Cuba. The method consists of five stages: Concept, Synthesis, Application Processing, Recovery and Management..

3.1 Ontology of geographic information

Relations in GI can be divided into spatial properties and non-spatial properties; the following is the discussion of these two kinds of relation in the ontology.

Spatial properties of semantic relations in this paper can be divided into the following three.

1) Topological relation: reflects the logical relation between geospatial entities. The topological relations for geographic information retrieval have very important significance. In this paper the topological relations including Contain, Touch, Overlap, Cover, Disjoint, Equal, CoverBy and Internal 8 kinds of relations, in which Cover and CoverBy, Contain and Internal are the inverse relation.

2) Direction relation: is one geographic entity relative the other geographic entities of orientation relation. The 8 well-known direction relations: North, Northwest, West, Southwest, South, Southeast, East, Northeast, may also have left, right, up, down and other direction relations. In our approach, take the central geographic entity of a map as the first one starting position to determine the other geographic entities relative the central geographic entity of the direction relations; then take the first one starting position as the center and around to find a geographic entity as a reference, accordance with this method to determine the spatial location relations between the geographic entities. Take A as the first one starting reference entity, build other geographic entities relative A entity the spatial orientation relations; Then in the

geographical entities set(F,G,H) choosing one of them as the next reference entity, in our approach, we according to the clockwise direction choice F as the second reference entity, then build other geographical entities(excluding the A entity which has been built spatial orientation relations) relative F geographic entity the spatial orientation relations. To build all the spatial orientation relations between geographical entities the same method is used.

In the spatial orientation relations, A and B have the following inverse relations:

- If B is located in the North to A, then A is located in the South to B;
- If B is located in the Northwest to A, then A is located in the Southeast to B;
- If B is located in the West to A, then A is located in the East to B;
- If B is located in the Southwest to A, then A is located in the Northeast to B;
- If B is located in the South to A, then A is located in the North to B;
- If B is located in the Southeast to A, then A is located in the Northwest to B;
- If B is located in the East to A, then A is located in the West to B;
- If B is located in the Northeast to A, then A is located in the Southwest to B.

The above orientation relations will be used in spatial reasoning, using OWL to build their mutual relations; System can infer the spatial relations between them by Jena.

3) Distance: usually people use far, close, farther and closer, etc. to describe the distance between the two geographic entities, but this kind of description is not accurate and easily leads to ambiguity. In our approach, the distance relations we use the object centroid of standard Euclidean distance.

Non-spatial properties the semantic relation, mainly due to people in expression the same class of geographic entities using difference concepts, as for "Mayabeque", it is known as the "river", also is known as "beach"; and a "Province" in Cuba. The semantic heterogeneity of non-spatial properties is generally divided into the following two: heterogeneous synonyms, i.e. we use a different vocabulary to express the same meaning, such as "region", also known as "area"; isomorphism homograph, i.e. we use the same vocabulary to express different meaning.

For the above-mentioned problems, there have a large number of the spatial properties and non-spatial properties. Firstly, the domain conceptualization describing working through a hierarchy of objects with their relationships and characteristics is performed. It was use the OWL built a description of the geographic entity and semantic relations.

The use of ontologies as tools to represent knowledge in a specific domain can assist management system to get places where the syntactic comparison of strings is not enough. These structures provide a way for representing knowledge. They are more than taxonomy; in them the meanings of the terms are not ambiguous, making them semantically independent of the user and the context. Identify classes of objects, their relationships and hierarchies of concepts within a specific domain.

Figure I. Fragment of the ontology

```
http://www.semanticweb.org/ontologies/2013/0/OntologySoil.owl
#Alta_Actividad_Arcillosa_Amarillento -->
   <owl:Class
rdf:about="#Alta_Actividad_Arcillosa_Amarillento">
      <owl:equivalentClass>
         <owl:Restriction>
            <owl:onProperty rdf:resource="#subtipo"/>
            <owl:someValuesFrom
rdf:resource="#Nodular_Ferruginoso"/>
         </owl:Restriction>
                        </owl:equivalentClass>
      <owl:equivalentClass>
         <owl:Restriction>
            <owl:onProperty rdf:resource="#subtipo"/>
            <owl:someValuesFrom rdf:resource="#Tipico"/>
         </owl:Restriction>
      </owl:equivalentClass>
      <owl:equivalentClass>
         <owl:Restriction>
            <owl:onProperty rdf:resource="#TipeOf"/>
            <owl:someValuesFrom rdf:resource="#Alitico"/>
         </owl:Restriction>
      </owl:equivalentClass>
      <owl:equivalentClass>
         <owl:Restriction>
            <owl:onProperty rdf:resource="#genero"/>
            <owl:someValuesFrom
rdf:resource="#depositos_binarios"/>
         </owl:Restriction>
      </owl:equivalentClass>
      <rdfs:subClassOf rdf:resource="#Alitico"/>
   </owl:Class>
```

In this work the ontology describes the characteristics of the soils of Mayabeque in the western region of Cuba following the New Version of Genetic Soil Classification (see Figure I).

Following ontology instance data stored in PostGIS to establish a relationship of images showing the soils according to the classification maps and the concepts of the ontology. The integration of spatial data in ontologies must be done by a previous process of annotation data.

The simplest way to integrate spatial data ontology is completely storing the data in the ontology, ie, the bytes representing spatial data to be embedded in the ontology structure in any section provided therefor. But not enough to fully embed the data in the ontology it is necessary to make a correspondence between the geographic objects in the data and concepts associated with them. After creating the instances of application ontology, the OWL document will be built.

3.2 Query processing

Within the processing module of the application user's query is enriched becoming syntactic query into a semantic representation equivalent, from the relationships between the concepts of an ontology. The first step that suffers the search string is to simplify and standardize the words in your query, removing irrelevant words. For this step the morphological analyzer Freeling is used.

The second step aims to recognize objects and spatial operators that connect them. Finally, activated terms in the ontology or similar search terms concepts. To identify similar concepts in the ontology algorithm DIS-C developed by the CIC-IPN in Mexico is used.

3.3 Recovery module

Then with all the concepts obtained (equal and similar) begins the search. When a match is found the associated relations and their respective classes or instances are extracted. Four types of matches can occur: complete, partial, atomic or no match; dependent similarity threshold although it is very unlikely that there is no coincidence. Having identified the same or similar objects are placed based ontology space scenes to which they belong. Comparison of spatial scenes is performed taking into account the spatial similarity according TDD model [4]. Images semantically related to concepts are retrieved of the knowledge base and maps detailing its spatial location. To view the enlarged image, select it and the system also gives a description of the image.

When an image is added in the knowledge base, the characteristics are extracted and related to the concepts of the ontology. Once a user uses an image, these features are stored as user preferences and from these the user can request a new search, finally showing images related to preferences and others that are related to these, but they are not within of user preferences. Finally, the system displays images with ordered lists of the map where they are located soils that meet user requirements. In addition to showing similar concepts that may be of interest.

The user of the system, on the other hand, have a profile with your data and preferences stored, obtained as a result of the image search soils, which will be stored according to the most frequent user searches. If you have already done several searches and have preferences on your profile, then you can do a custom search where to select semantically related images with this will show other related preference not in the profile.

4 Results

A platform for experimentation which is named as SIGeopedia is developed. The main objective is the implementation and testing of the developed methods.

This platform provides the basic functionality in existing conventional GIS accessible through the toolbar, (see Figure II), these are:

- Loading and displaying data layers.
- Interaction with the map (Pan, Zoom, Reset).
- Check the data.

Figure II. Experimental platform

The platform also has a map viewer where the loaded data through spatial component is displayed (see Figure III).

Figure III. Maps viewer

The management system allows searching for information on the different types of soils. When you select the search, for example, types of soil "*red ferralitic*", a map of the region with the location of this soil type is displayed (see Figure III). In addition, all images related to the soil type are showing. If the user clicks on any of these images that are displayed with

a wider view, showing the description of it and the concepts of the ontology with semantic relations.

Geographic relations are used to qualitative and quantitative description the basic spatial relations of concept sets, including Topology, Direction and Distance, in which Topology including Contain, Touch, Overlap, Cover, Disjoint, Equal, CoverBy and Internal 8 kinds of spatial topological relations; Direction including isNorthOf, isEastOf, isSouthOf, isWestOf, isNorthEastOf, isSouth-NorthOf, isNorthWestOf and isSouthWestOf 8 kinds of orientation relationship, these properties all have transitivity, some properties have the inverse relations, e.g. isWestOf and isEastOf have the inverse relationship; Distance represents the spatial distance. Some relations can be seen in Table I.

Table I. Experimental platform

	Type	Tag
Relation	**Topology**	**isCotain, isTouch**
	Direction	**isNorthOf, isNorthEastOf**
	Distance	**metresOf, kilometresOf**

Furthermore a saving in time is achieved in the performance of spatial analysis. This is because of a series of steps that take some time and in which the commission of errors can be deleted, such as:

- The search data layers with which you want to work as these are conceptualized in the ontology generated.
- Creating a query in SQL language as the user navigates freely ontology that also could be implemented based search methods in natural language.

Examples of queries the system may not give answers automatically using conventional methods would be:

What rivers are in the town of San Jose?

What are the regions surrounding Mampostón lake?

This type of query can`t be obtained through SQL language as both the database and the ShapeFile file because no table named with the terms used in the query exist. Therefore, through the use of ontology the system can easily understand what you want to retrieve. In addition recoveries were performed without the need to build a query in SQL language making it possible to expand its use to non-expert users.

Another aspect is that the data recovery tasks using the structure and proposed methods turn out to be simpler and faster than conventional methods for non-expert users in GIS applications, as it enables these to browse the ontology and perform different types of analysis from recoveries without the need to know the analytical tools of GIS. The results show that information retrieval using the proposed method increases the quality of analyzes because it reduces possible errors in the operations to be performed, for example, to create the query.

These results show that the use of ontology as an intermediate layer between the recovery system and the spatial database allows the handling of objects through its conceptual representation (abstraction). The system has tools for semantic processing of geospatial data. This allows different types of analyzes performed on these data types can be made by both specialists and non-experts in the use of GIS.

5 Conclusions

In this paper it is proposed a methodology to optimize the management of geospatial semantic similarity measures applied in the recovery process. The application of semantic similarity measure makes to avoid returning empty result, but be careful with the results that can be completely removed from what the user is looking.

Using semantic analysis is possible to approach a form of intelligent processing, which is not due solely to a syntactic coincidence but without forgetting the advantages of the classic geospatial approach.

This work helps to reduce the lack of systems that integrate semantic aspects together with spatial aspects, hence the integration model presented in this paper, semantic analysis exposes as an excellent complement to traditional mutual geospatial analysis techniques.

In future work the proposed large volumes of data and new similarity measures are applied to identify the geographic objects architecture experience. You need to find elements to better linkages between formal semantics and information needed for graphical display of geographic information.

6 References

[1] A. B. Markman, "Constraints on Analogical Inference.," Cognitive Science., vol. 21(4), pp. 373–418, 1997.

[2] A. Schwering, "Approaches to Semantic Similarity Measurement for Geo-Spatial Data - A Survey," Transactions in GIS, vol. 12(1), pp. 5–29, 2008.

[3] M. A. Rodríguez and M. J. Egenhofer, "Determining semantic similarity among entity classes from different ontologies.," IEEE Transactions on Knowledge and Data Engineering, vol. 15(2), pp. 442–456, 2003.

[4] B. Li and F. Fonseca, "TDD - A Comprehensive Model for Qualitative Spatial Similarity Assessment.," Spatial Cognition and Computation, vol. 6(1), pp. 31–62, 2006.

[5] A. Schwering, "Semantic Similarity Measurement including Spatial Relations for Semantic Information Retrieval of Geo-Spatial Data," Münster, Deutschland, 2006.

[6] R. Larín, "Nuevo tipo de ontología para la representación semántica de objetos geoespaciales," Centro de Aplicaciones de Tecnologías de Avanzada, La Habana, 2013.

[7] K. Janowicz, M. Raubal, A. Schwering, and W. Kuhn, "Semantic Similarity Measurement and Geospatial Applications," Workshop on Semantic Similarity Measurement and Geospatial Applications, 2008.

[8] F. Fonseca, "Ontology-Driven Geographic Information Systems," The University of Maine, The Graduate School, Maine, USA, 2001.

[9] A. Schwering and M. Raubal, "Spatial Relations for Semantic Similarity Measurement," Ordnance Survey, 2005.

A Hybrid Search Method to Find Minimal Length Paths for Navigation Route Planning

Yong-Hu Lee [1,2] **and Sang-Woon Kim** [2]
[1] Samsung SDS Co., Ltd.,
Suwon, 443-822 South Korea
yongwho@naver.com
[2] Dept. of Computer Eng., Myongji University,
Yongin, 449-728 South Korea
kimsw@mju.ac.kr

Abstract

This paper reports an experimental result on combining multiple search methods in order to find the minimal length path for navigation route planning. In particular, in order to incorporate the strengths of A algorithm and Dijkstra's algorithm, a framework of unifying the two algorithms is investigated. In the unified search algorithm, the $OPEN$ lists of the A* algorithm and the Dijkstra's algorithm are (ir)regularly switched, so that the computational cost is reduced and the route that could not be searched by A* algorithm can be discovered. The experimental results, obtained using artificial and real-city map data, demonstrate that the proposed hybrid algorithm can reduce computational cost significantly and shows similar results with the suggested route from the origin to the destination in Dijkstra's algorithm.*

Keywords: Navigation route planning, Dijkstra's algorithm, A* algorithm, Learning real-time A* algorithm

1. Introduction

An automotive navigation system is a road guidance system that explores the best route from the starting point to the destination, presenting a guide using the optimal or near-optimal route search. For general path search, methods, such as Dijkstra's algorithm [1] and A* search algorithm [2], are used. For navigation, two-way search mechanisms, including bidirectional Dijkstra's algorithm and bidirectional A* algorithms, are used in order to reduce search time [3]. Stochastic time-dependent planning [4], through which a search area is divided into smaller areas, is also used. Additionally, there has been an ongoing research on the shortest route search mechanism, including LRTA* (Learning Real-Time A*) [5], that is suitable in an operating environment with various vehicles.

In [3] and [5], it is reported that, in finite domains where the goal state is reachable from any state and all action costs are non-zero, LRTA* finds a goal on every trial. Additionally, if the initial heuristic function is admissible then LRTA* will converge to an optimal solution after a finite number of trials. Compared to A* algorithm, LRTA* with a lookahead of one has a considerably shorter first-move lag and finds sub-optimal solutions much faster. However, converging to an optimal solution can be expensive in terms of the number of trials and the total execution cost. In order to address this problem, three extensions have been proposed in the literature: deeper lookahead LRTA* [7], [6], optimality weighted LRTA* [8], and backtracking LRTA* [9]. Recently, in [5], the three extensions have been combined into a single algorithm called Learning Real-Time Search (LRTS) and compared along several dimensions.

With regard to A* algorithm, there are two well-known issues: first, as the navigation path increases, the computational cost increases exponentially; second, it is more difficult to find an accurate route using the A* algorithm compared to using the Dijkstra's algorithm. Compared to Dijkstra's algorithm, A* algorithm, which uses heuristic values (h), may be able to reduce the amount of computation, but it is not capable of finding certain useful paths in the case where there is an error in calculating the h values. In this study, a unified route planning of A* and Dijkstra's algorithms is proposed

based on the fact that the computational cost increases as the number of nodes stored in the $OPEN$ list increases in A* algorithm. By separating the $OPEN$ lists of the two algorithms before processing, the increase in $OPEN$ list size can be prevented. The farther the distance between the origin and the destination, the more dramatic the increase in $OPEN$ list size. However, such an increase can be prevented by using this present technique. The simulation demonstrates that the proposed hybrid search method reduces the computational cost of the conventional A* algorithm, and is also able to find the route that A* algorithm could not detect. In addition to that, it provides the result of the optimal route in less time.

The significant contribution of this paper is the demonstration that the efficiency of navigation route planning can be improved using hybrid techniques when learning a near-optimal path and, at the same time, minimizing the learning time and memory required. In particular, a unified route planning of A* and Dijkstra's algorithms is considered based on the fact that the computational cost increases as the number of nodes between the origin and the destination increases. By a means of switching the two algorithms before processing continuously, the increase in the computational cost can be prevented.

The remainder of the paper is organized as follows. In Section 2, a brief introduction to traditional search algorithms, such as Dijkstra's algorithm, A* algorithm, and LRTA* algorithm, is provided. Then, in Section 3, a hybrid method unifying the two search algorithms for selecting minimal length paths for navigation route planning is presented. In Section 4, the experimental setup and results obtained using artificial and real-city map data are presented. Finally, in Section 5, the concluding remarks and limitations that deserve further study are presented.

2. Related Work

In this section, the Dijkstra's and A* algorithms, which are closely related to the present empirical study, are briefly reviewed. The details of the algorithms can be found in the related literature [1], [2].

2.1 Dijkstra's Algorithm

The problem of selecting an optimal or near-optimal path for navigation route planning is a problem of selecting the shortest path between two nodes in graph-search applications. A systematic method to achieve this is to search the whole solution space to select the optimal or near-optimal path using Dijkstra's algorithm, which can be explained as follows. Here, the problem is to find the path of minimum total length between two given nodes s and t, which denote the start node and the terminal node, respectively. Also, a set, S, contains the nodes for which the path of minimum length from s is known.

1) Find the nearest neighbor node (n) connected to the origin node (s). Thus, only s and n exist in S at the beginning.
2) Expand node n, generating all of its successors (n_i). For each successor n_i, if $n_i \notin S$, add n_i to S. If n_i is the destination node (t), terminate the search.
3) Go to Step 2 after finding the nearest neighbor node (n) connected to any node in S, and repeat the step till all available nodes are added to S.

2.2 A* Algorithm

Since the above algorithm is time consuming, another approach can be considered, namely, A* algorithm, in which the optimal or near optimal (minimum total length) paths are selected by searching the solution space in a specified direction that is determined by using a heuristic function $h(n)$. Here, the heuristic function is an estimate of the cost (i.e. the length) of an optimal solution path from the node n to a set of terminal nodes, including the destination node t. This scheme can reduce the searching time significantly. The procedure that utilizes the heuristic function can be formalized as follows, where the evaluation function f is such that its value, $f(n)$, at any node n estimates the sum of the cost of the minimal cost path from node n to a goal node. That is, $f(n)$ is an estimate of the cost of a minimal cost path constrained to go through node n and estimated as follows [10], [11].

$$f(n) = g(n) + h(n), \tag{1}$$

where $g(n)$ is the cost of the path in the search tree from s to n given by summing the arc costs encountered while tracing back through the pointers from n to s. Meanwhile, for the estimate $h(n)$, we depend on heuristic information from the problem domain. On the basis of what we have briefly discussed, A* algorithm based search scheme can be summarized as follows.

1) Put the start node s on a list named $OPEN$ and compute $f(s)$. Thus, only s exists in $OPEN$ at the beginning.
2) Choose a node with the smallest f from $OPEN$ and name the node n. Remove n from $OPEN$ and put it on a list called $CLOSED$. If n is the

destination node, terminate the search (with the solution path obtained by tracing back through the pointers). If $OPEN$ is empty, exit with failure (which means that there is no possible route); otherwise continue.

3) Expand the node n, generating all of its successors (n_i's). If there are no successors, go to Step 2. For each successor n_i, compute $f(n_i)$. Associate with n_i's that were not already on either $OPEN$ or $CLOSED$ the f values just computed. Put these nodes on $OPEN$ and direct pointers from them back to n.
4) Associate with those successors that were already on $OPEN$ or $CLOSED$ the smaller of the f values just computed and their previous f values. Put on $OPEN$ those successors on $CLOSED$ whose f values were thus lowered, and redirect to n the pointers from all nodes whose f values were lowered.
5) Go to Step 2 and repeat the steps.

2.3 LRTA* Algorithm

LRTA* [3], [6] is one of the most well known learning real-time heuristic search algorithms. The LRTA* algorithm works to find the the optimal or near-optimal route by interleaving planning and execution in an on-line decision-making manner. In particular, in the current state s, the LRTA* with a lookahead of depth k considers the successors (i.e., neighbors in depth k) of s. For each neighbor state, two values are measured: the execution cost of getting there from the current state and the estimated execution cost of getting to the closest goal state from the neighbor state. LRTA* then travels to the state with the lowest value. That is, $h'(s)$ is an estimate of the cost of a minimal cost path constrained to go through node s and estimated as follows.

$$h'(s) = min\{c(s,n) + h(n)\}, \quad (2)$$

where $c(s,n)$ is the cost of the path in the search tree from s to n given by summing the arc costs encountered, while $h(n)$ is the estimated distance from node n to goal nodes, depending on heuristic information from the problem domain. Additionally, it updates the heuristic value of the current state if the minimum f-value is greater than the heuristic value of the current state. On the basis of what we have briefly discussed, the LRTA* algorithm based search scheme can be summarized as follows.

1) Perform lookahead at depth k from the current state s, updating its heuristic estimate accordingly.
2) Select the state s' with minimum value of $c + h$ among the successors of s as the new current state.

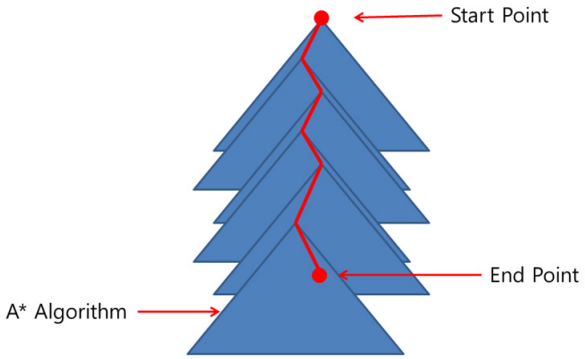


Fig. 1. Plot of the searching area for route planning in the LRTA* algorithm.

3) After taking an action that passes from s to s', go to Step 1 and repeat the steps.

Fig. 1 presents the method of path-finding in the LRTA* algorithm. In the figure, the triangular areas in blue color indicate the paths examined in order to search for the optimal or near-optimal route within depth k, whereas the line in red color indicates the route that is finally selected.

3. Proposed Hybrid Algorithm

As mentioned previously, in this paper A* algorithm and Dijkstra's algorithm are unified for more efficient route planning. First, in order to combine the two algorithms, parameter levels are defined as follows. The node nearest to the departure point is Level 1, whereas the node nearest to Level 1 is Level 2. Similarly, the node nearest to Level l is Level $l + 1$. In order to resolve the problem regarding the A* algorithm, a hybrid search, which uses an algorithm different from the one used for Level l, is used for Level $l + 1$. For instance, if Dijkstra's algorithm is used to search from Level 1 to Level 9, the search algorithm is changed to A* algorithm when Level 10 is reached. The proposed search algorithm to find a near-optimal route using a hybrid technique can be described with the following steps. The inputs are: an origin node (s), a destination node (t), and the number of the level for switching (T). On the basis of what we have briefly discussed, a hybrid algorithm for the navigation route planning is summarized as follows.

1) Initialization:
   a) Create the $OPEN$ and $CLOSED$ lists, DO and DC for the Dijkstra's algorithm and AO and AC for the A* algorithm, where the

related parameters (i.e. f, g, and h) are also defined.

b) The origin point s is set as node n. After calculating the parameters relative to the node n, add n to DO. Thus, only n exists in $OPEN$ list at the beginning, with its level being $l \leftarrow 0$.

2) Perform the following steps as were done in the Dijkstra's algorithm:

a) Expand node n, generating all of its successors (n_i's). For each successor n_i, if $n_i \notin DO$, add n to DO after computing $f(n_i)$ and $l \leftarrow l + 1$.

b) Choose the node with the smallest f from DO and name it n. Add n to DC. If n is the destination node t, terminate the search. When DO is empty, it is considered that there is no possible route.

c) For the node n, if $l < T$, then go to Step 2; otherwise (i.e. l is equal to T) go to Step 3.

3) Perform the following steps as were done in the A* algorithm:

a) For all the nodes of DO, compute the f values, which are $(g + h)$'s, and move them into AO.

b) Choose the node with the smallest f from AO and name it n. Add n to AC.

c) After clearing up DO and AO, go to Step 2.

Here, we can see that the performance of the hybrid algorithm relies heavily on how well the threshold-level (T), which is determined as an experimental constant, is selected. Thus, in order to improve the performance, we need to ensure that the threshold is well chosen.

4. Computer Simulation

The run-time characteristics of the proposed hybrid algorithm are reported in the following subsections. First, in order to evaluate the performance of the hybrid search algorithm, a computer simulation was performed using an artificial map data consisting of $1600(40 \times 40)$ search nodes (points). A test was done in three scenarios for the route from the upper left origin (S) to the lower right destination (E) in the search area, at Level-1, Level-4, and Level-10, respectively. Fig. 2 (a), (b), and (c) present the results obtained with the proposed hybrid algorithm at the three levels, respectively. In the figures, the black line indicates the paths examined in order to search for the optimal or near-optimal route, whereas the pink line indicates the route that is finally selected.

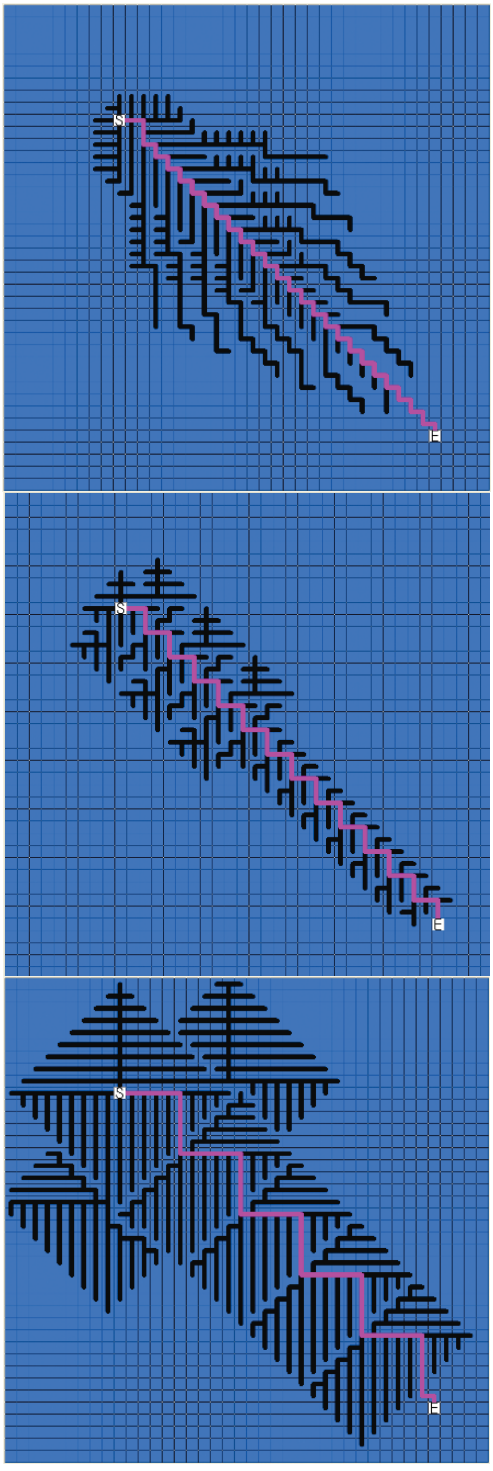


Fig. 2. Plots of the searching areas of the proposed hybrid method for route planning: (a) top, (b) middle, and (c) bottom; (a) - (c) are obtained with Level-1, Level-4, and Level-10, respectively.

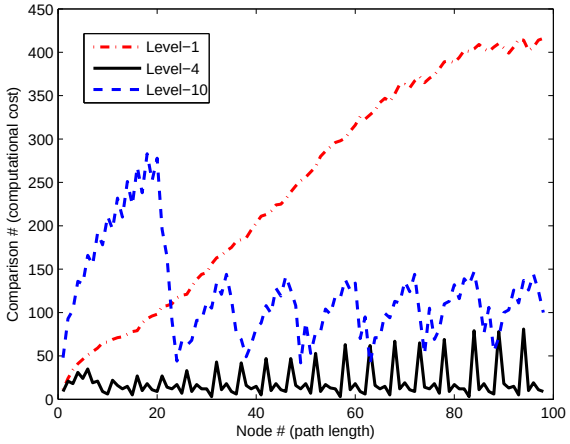


Fig. 3. Comparison of the number of comparisons (i.e. computational cost) to search for the minimal value of the $OPEN$ list in the three scenarios (Level-1, Level-4, and Level-10 as shown in Fig. 2 (a), (b), and (c), respectively).

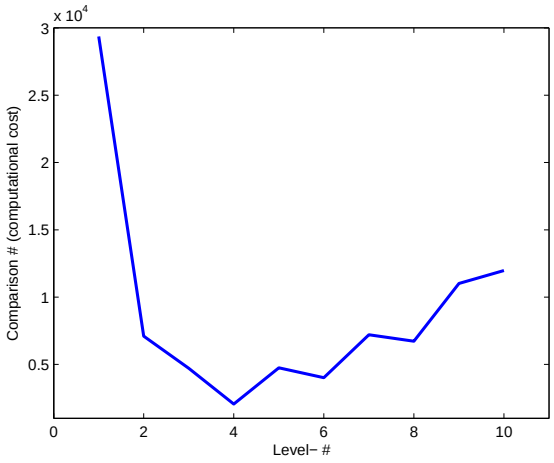


Fig. 4. Plot of comparing the number of comparisons (i.e. computational cost) needed for finding the minimal length paths when increasing the level number (Level-#) for switching from 1 to 10.

The observations obtained from the three figures are the followings. First, Fig. 2 (a) shows the same result as A* algorithm searched at Level-1. Thus, the route planning was performed only using A* algorithm, without using Dijkstra's algorithm. Second, Fig. 2 (b) shows the result of Level-4 search, using Dijkstra's algorithm for the first three nodes, and then changing the algorithm for route search. Based on the nodes set by Dijkstra's algorithm, A* algorithm was performed in succession. Finally, Fig. 2 (c) shows the result of Level-10 search, using Dijkstra's algorithm from the origin up to the 9 levels, then switching to A* algorithm for route planning. These figures show that the farther the distance between the origin and the destination, the greater the increase in computational cost. It is also shown that the hybrid algorithm examines a route in a wider range, compared to A* algorithm.

Fig. 3 presents the number of comparisons (i.e. computational cost) to search for the minimal value of the $OPEN$ list in the three scenarios of route planning. In the plot, the x-axis denotes the length of the path (the number of nodes) between the origin S and the destination E, while the y-axis indicates the number of comparisons (computational cost) to search for the minimal f value on the $OPEN$ list.

From Fig. 3, as can be seen in Fig. 2 (a), (b), and (c), it can be observed that, for the hybrid search method of Level-1, which corresponds to the A* algorithm, the computational cost increases dramatically as the length of the paths increases. Meanwhile, for the hybrid search method of Level-4, the computational cost gradually increases. In other words, a farther distance between the origin and the destination represents a greater decrease in computational cost at the Level-4 hybrid search method. This observation demonstrates that the proposed hybrid mechanism works well.

With regard to this observation, the following is an interesting issue to investigate: *is the computational cost to a minimum at any level number?* In order to investigate this issue, we repeated the above experiment with increasing the level number from 1 to 10, i.e. Level-1 to Level-10. Fig. 4 presents a comparison of the computational costs needed for increasing the level number (Level-#).

From the figure, it can be noted that, for the hybrid algorithm, contrary to the case of using A* algorithm, the computational cost is *remarkably* decreased when the level number of 4 value is used. As the level number increases from 4 to 10 values, the computational cost increased *steadily*, not abruptly. This observation demonstrates again that the proposed hybrid mechanism works well.

Secondly, in order to further investigate the advantage of the hybrid search method, and especially to determine which types of significant map data are more suitable for the proposed scheme, we repeated the experiment with a few real-city map data. For example, Fig. 5 presents four minimal length paths between two points (S and E) in Seoul, which are selected using Dijkstra's algorithm [1],

Table 1. Numbers of nodes and links of the real map data of four the largest cities in Korea, such as Seoul, Busan, Daejeon, and Daegu).

Nodes & links	Map data of four cities			
	Seoul	Busan	Daejeon	Daegu
Node #	8,565	2,565	1,889	1,438
Link #	22,182	7,671	5,618	4,471

Table 2. Comparison of the computational costs and the minimal length paths within the parentheses for the real map data of the four largest cities. Here, for ease of comparison, each value (computational cost) is divided by 1000. Also, for each column, the lowest cost is highlighted with a * marker.

Search methods	Minimal length paths			
	Seoul	Busan	Daejeon	Daegu
Dijkstra	532.6 (285.1)	54.3 (349.2)	77.3 (281.6)	45.5 (465.5)
A*	*29.6 (296.0)	14.7 (358.2)	47.0 (294.4)	6.8 (472.7)
LRTA* (k = 16)	38.5 (308.8)	21.4 (503.0)	18.7 (337.0)	14.2 (477.3)
LRTA* (k = 32)	101.3 (306.9)	57.8 (501.3)	28.2 (293.4)	39.0 (475.6)
LRTA* (k = 64)	267.5 (306.9)	149.3 (524.2)	31.2 (293.4)	85.4 (474.1)
Hybrid method	41.1 (291.3)	*14.2 (356.4)	*11.2 (293.2)	*5.1 (472.3)

A* algorithm [2], LRTA* (k = 16, 32, and 64) [5], and the proposed hybrid method, respectively. The details of the other city map data are omitted here in the interest of compactness, but the numbers of nodes and links of the map data are summarized in Table 1.

Also, Table 2 presents a numerical comparison of the computational costs (and the minimal length paths between S and E) obtained using the four algorithms in the four real-city map data. Here, Level-15 was used for the hybrid method.

From Table 2 as well as Fig. 5, it can be observed that, for the four search methods of Dijkstra's, A*, LRTA* (k = 16, 32, and 64), and the hybrid method, the minimal length paths are almost the same. In contrast, the computational costs are different. More specifically, for example, consider the results of the last column (Daegu). All of the minimal length paths in $(\cdot)$ are around the value of 470×10^3. However, for the computational cost, the proposed hybrid method wins with the value of 5.1 (i.e., 5073 divided by 1000), which is highlighted with a * marker. Meanwhile, in the table, the computational cost of the hybrid method for Seoul map data is $41,111$, which is not highlighted. This observation demonstrates that when considering the minimal length paths for navigation route planning, the computational cost can be reduced by using a hybrid technique, even though it is not always the best.

5. Conclusions

In this paper, a unified navigation route planning of A* algorithm and Dijkstra's algorithm was examined in order to reduce the computational cost despite having a large map data. A* algorithm was not only used as a basic planning method, but also used for route planning in order to incorporate the strengths of Dijkstra's algorithm. The $OPEN$ lists of A* algorithm and the Dijkstra's algorithm are irregularly switched, so that the computational cost is reduced and the route that could not be searched by A* algorithm can be discovered. In order to evaluate the performance of the proposed hybrid method, artificial and real-city map data have been experimented and compared to the computational costs and the route search results of the traditional search algorithms. The simulation results demonstrate that the proposed hybrid method can reduce computational cost significantly and shows similar results with the suggested route from the origin to the destination in Dijkstra's algorithm. In this paper, the searching range of the Dijkstra's algorithm was limited to a maximum of Level-10 (for artificial map) or Level-15 (for real-city map). However, when the level number was changed, the final route and the computational cost were changed. Thus, further study is recommended to find the optimal level while taking into consideration the actual road conditions.

Acknowledgment

The work of the second author was generously supported by the National Research Foundation of Korea funded by the Korean Government (Ministry of Science, ICT and Future Planning; NRF-2012R1A1A2041661). The author is grateful to this support.

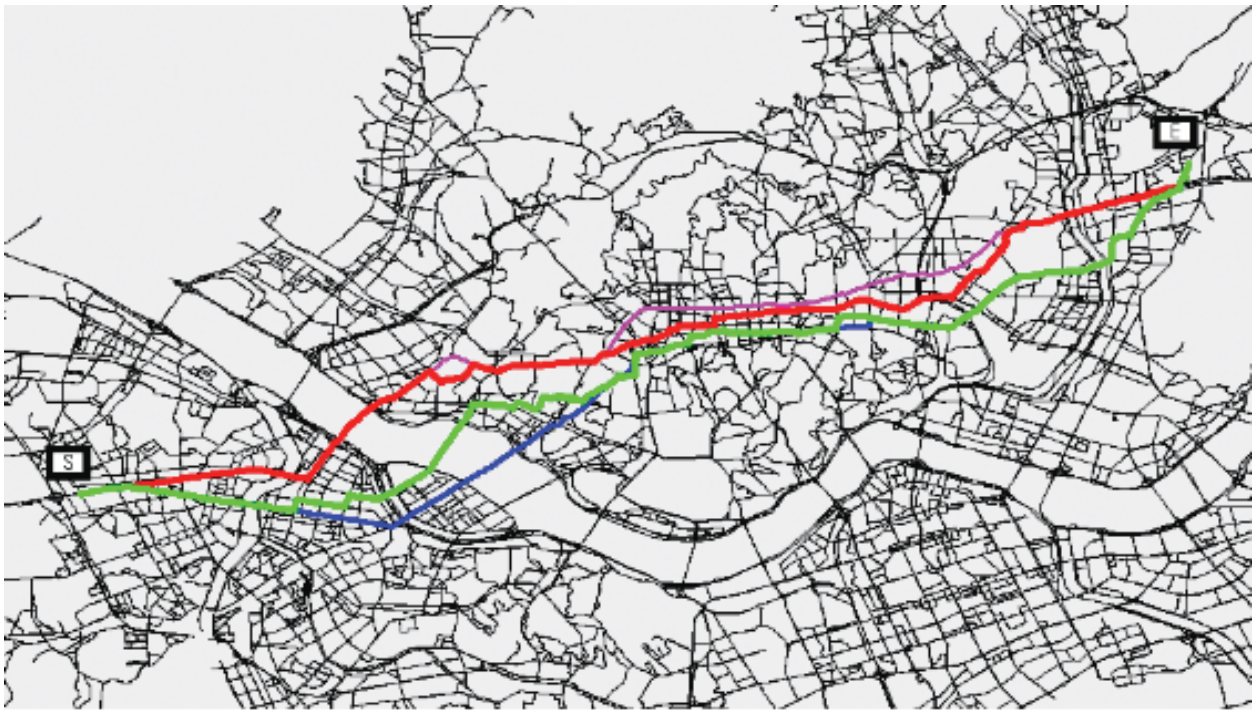


Fig. 5. Plot of a real-city (Seoul) map data, where four paths between two points (S and E) are selected using Dijkstra's algorithm (pink line), A* algorithm (blue line), LRTA* (green line), and the proposed method (red line), respectively.

References

[1] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*, vol. 1, pp. 269–271, 1959.

[2] P. Hart and N. Nilsson and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE Trans. Systems Science and Cybernetics*, vol. 4, no. 2, pp. 100–107, 1968.

[3] R. E. Korf, "Real-time heuristic search," *Artificial Intelligence*, vol. 42, no. 2, pp. 189–211, 1990.

[4] I. C. M. Flinsenberg, *Route Planning Algorithms for Car Navigation*. PhD Thesis, Technische Universiteit Eindhoven, The Netherlands, 2004.

[5] V. Bulitko and G. Lee, "Learning in real time search: A unifying framework," *Journal of Artificial Intelligence Research*, vol. 25, pp. 119–157, 2006.

[6] V. Bulitko, "Lookahead pathologies and meta-level control in real-time heuristic search," In *Proc. of the 15th Euromicro Conference on Real-Time Systems (ECRTS 2003)*, pages 13–16, Porto, Portugal, July 2 - 4, 2003.

[7] S. Koenig, "A comparison of fast search methods for real-time situated agents," In *Proc. of the 3rd International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS '04)*, pages 864–871, New York, NY, July 19 - 23, 2004.

[8] M. Shimbo and T. Ishida, "Controlling the learning process of real-time heuristic search," *Artificial Intelligence*, vol. 146, no. 1, pp. 1–41, 2003.

[9] L. -Y. Shue and S. -T. Li and R. Zamani, "An intelligent heuristic algorithm for project scheduling problems," In *Proc. of the 32nd Annual Meeting of the Decision Sciences Institute (2001 DSI)*, San Francisco, CA, November 17 - 20, 2001.

[10] N. J. Nilsson, *Principles of Artificial Intelligence*. Tioga Publishing Co., Palo Alto, CA, 1980.

[11] L. H. O. Rios and L. Chaimowicz, "A survey and classification of A* based best-first heuristic search algorithms," In *Proc. of the 20th Brazilian Symposium on Artificial Intelligence (SBIA 2010)*, volume 6404, pages 253–262, Sao Bernardo do Campo, Brazil, October 23-28, 2010.

SESSION

LEARNING METHODS + ONTOLOGY, ONTOLOGICAL ARGUMENTS, AND LOGIC + KNOWLEDGE REPRESENTATION AND KNOWLEDGE BASED SYSTEMS

Chair(s)

TBA

A Protocol for Intelligent Agent Systems Enabling Incidental Learning on the Web

H. Wang
School of Computing and Information Systems
Athabasca University
Alberta, Canada

Abstract - *Incidental learning is learning something without intent, which usually happens during the time that is not dedicated to learning that thing. Examples of such time include work, game play and other activities for leisure. In today's life of many people, a lot of time is spent on the surfing the Web. Should incidental learning be effectively implemented for individuals using the Web, it would make learning much more effective and efficient. This can be especially true for adult learners as they most likely have less time dedicated to learning, or don't have any dedicated time for learning at all. In this paper, we present a protocol that is intended to be used in the design and implementation of intelligent agent systems for enabling incidental learning on the Web. The protocol prescribes the key requirements for such an intelligent agent system to meet, and depicts the steps such an intelligent agent system needs to take when rendering an incidental learning session on the web.*

Keywords: *intelligent agent, incidental learning, web-based learning, e-learning, computer protocol*

1 Introduction

Incidental learning is learning without intent, which happens at the time that is not dedicated to learn what is learned through the incidental learning session [1]. By incidental learning on the Web, we mean that the learner has incidentally learned something useful, while browsing the web, without the intent to learn that thing. The learning of that thing is incidental to the learner. To make incidental learning more meaningful for the learner, however, what to be learned through it should not be completely random. The learning outcomes from incidental learning should better be part of the learner's general goals [2] [3], or should at least serve the learner's interest. In order for an intelligent agent system to do this, the following key requirements must be met:

1. The learner must be capable of learning – this is the very basic requirement for incidental learning, or any learning to occur [4] [5]. An agent system designed for enabling incidental learning on the web must do its best to know whether the learner has the capability to learn, and what the learner is capable of learning. A learner may be capable of learning X+Y but not ready to learn X*Y. It won't help the learner at all by keeping firing up incidental learning sessions that teach the learner something he or she is incapable of.

2. Learner must have the desire to learn -- without any desire to learn, no learning can be effective. The desire can be fostered gradually through certain enabling technologies. This may be the hardest part for an agent to do – to know whether the learner has the desire to learn at a specific time. The information may be gathered during the time the learner is browsing the web, or after two or three tries of firing up an incidental learning session – when the learner has simply ignored. In any case, it is important for the agent system to know if the learner has the desire to learn at a given time; otherwise it would definitely be an annoying thing to the learner to keep firing up one incidental learning session after another.

3. Knowing the learning goals of the leaner [6] [7] – what topics and subjects the learner is interested in, at what level the learner is in a particular area or topic.

4. Locating triggers to fire up incidental learning sessions – the intelligent agent system needs to be fired up at the right time while the learner is on the web. The triggers should be within the web document the learner is reading, together with the learner's motion such as mouse click.

5. Finding the proper learning content – the system must be able to get the right learning content to form a learning session for the learner. The content can be in a learning object repository, or mined from the web on the fly.

6. Presenting an effective incidental learning session – the session must be well controlled and properly sized in terms of both content and duration.

An intelligent agent system [8] that implements incidental learning on the web for web users must address these key requirements. Fortunately today, however, we have at least two conveniences in developing and implementing such a system.

The first one is that many individuals are spending a lot of time on the Web. This would make incidental learning on the web be able to occur more often. Imagine that such an "incident" could occur only once a year or even only once a lifetime, it might be enough to trigger a scientific invention for a well-prepared scientist, but wouldn't be enough for someone to learn substantially on a subject. As such, it makes more sense now for us to develop enabling technologies and systems for incidental learning on the web because more and more people will benefit from the technologies and systems more often.

The second convenience we are given is the unprecedented availability on the world-wide web of knowledge of almost any subject. This greatly ensures the feasibility of the development of incidental learning-enabling technologies and systems in terms of source of knowledge and information. Because the information and knowledge needed to feed the learners on the Web is widely available, an intelligent agent should be able to collect the particular knowledge and information especially for an individual based on his/her overall learning goals and feed the individual in a way without his or her notice. As such, the enabling technologies and systems developed for incidental learning, such as the protocol reported here, could be used to learn knowledge in various topics and subjects!.

1.1 Significance and Contributions

Incidental learning brings at least two benefits to learners. One is that it effectively utilizes the times or tiny fractions of time that are not designated for learning; the second is that people will be able to learn with a pleasure but without pressure. The learners would be much happier to learn with less or no stress. Imagining that you could learn something effectively for a purpose or purposes while you are browsing the Web for pleasure, and imagining that students could build their knowledge and skills to satisfy the requirements of a course or courses while they are working or playing with the Web, isn't it wonderful and very significant in education?

The benefits can be more obvious for adult students taking online courses or other distance learning courses. As we know, most of these students are part-time as they often have other business to do other than completing courses. They cannot afford to take leave to study in traditional universities in the classroom, and they don't usually have a big trunk of their time to spend on a course study. However, they do spend time on the web for various reasons and purposes. If an intelligent web agent system [9] is available for enabling incidental learning on the web, it will help these learners greatly, because these students will be able to utilize fractions of their time to learn while they are working, playing or doing other things. The protocol presented in this paper is intended to be useful guide for developing intelligent agent systems that enables incidental learning on the web.

2 Steps of incidental learning on the web

Incidental learning may happen to someone with or without consciousness. In the former case, the learning session happens incidentally while the learner is doing something else, but during the session, the learner consciously knows she or he is in the learning session, and knows what she or he is supposed to learn. With consciousness, the learner could be more motivated during the session, and the learning could be more effective.

In the latter case, not only does the learning session, if it can be called a learning session, occur incidentally, the learner doesn't even know he or she is in a learning session, and may not even know what she or he is learning either, in an extreme case.

For some learners, learning with a clear purpose or consciously should be more effective, while for learners who are really tired of learning certain things in certain area, incidental learning unconsciously should be more effective too.

While the learners may learn something in an incidental learning session on the web consciously or unconsciously, the intelligent agent system that enables incidental learning on the web must know what a particular learner is supposed to learn in a specific learning session, what content should be delivered to the learner, how the content should be presented to the learner, when the incidental learning session should be fired up, and when it should closed. In general, the process of incidental learning on the web can be depicted as follows:

1. Monitor user's activities on the web within a browser;

2. Identify the point of interest within the context of what the learner is doing, or browsing in the case of web-based incidental learning;

3. Develop the point of interest further into a specific learning objective or objectives that are part of the learner's learning goals at the time;

4. Get the proper learning content to support the learning objective or objectives, The proper learning content can be dynamically generated from the web or retrieved from learning object repositories [10] [11] [12];

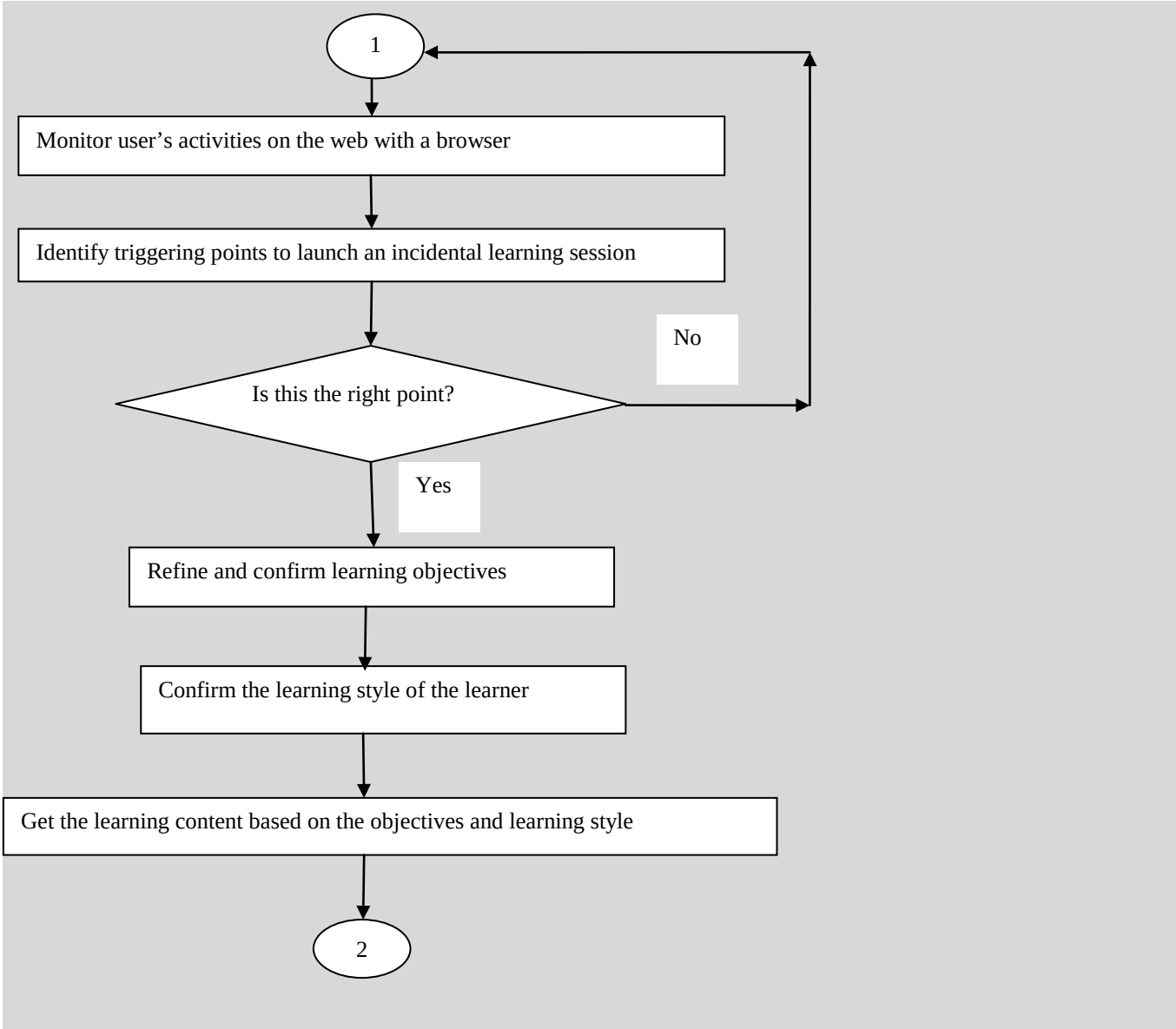
5. Form a small learning session with the proper learning content to suite the learner's personality and learning style;

6. Present the learning session to the learner and/or bring the learner into the learning session;

7. Monitor the incidental learning session.

8. Closing the session once the learner is done, or has left the session, and book keeping the learning session.

These eight steps are the core of the protocol. Any intelligent agent system that enables incidental learning on the web should follow these eight steps. The protocol can then be depicted in a flowchart as shown in figure 1.



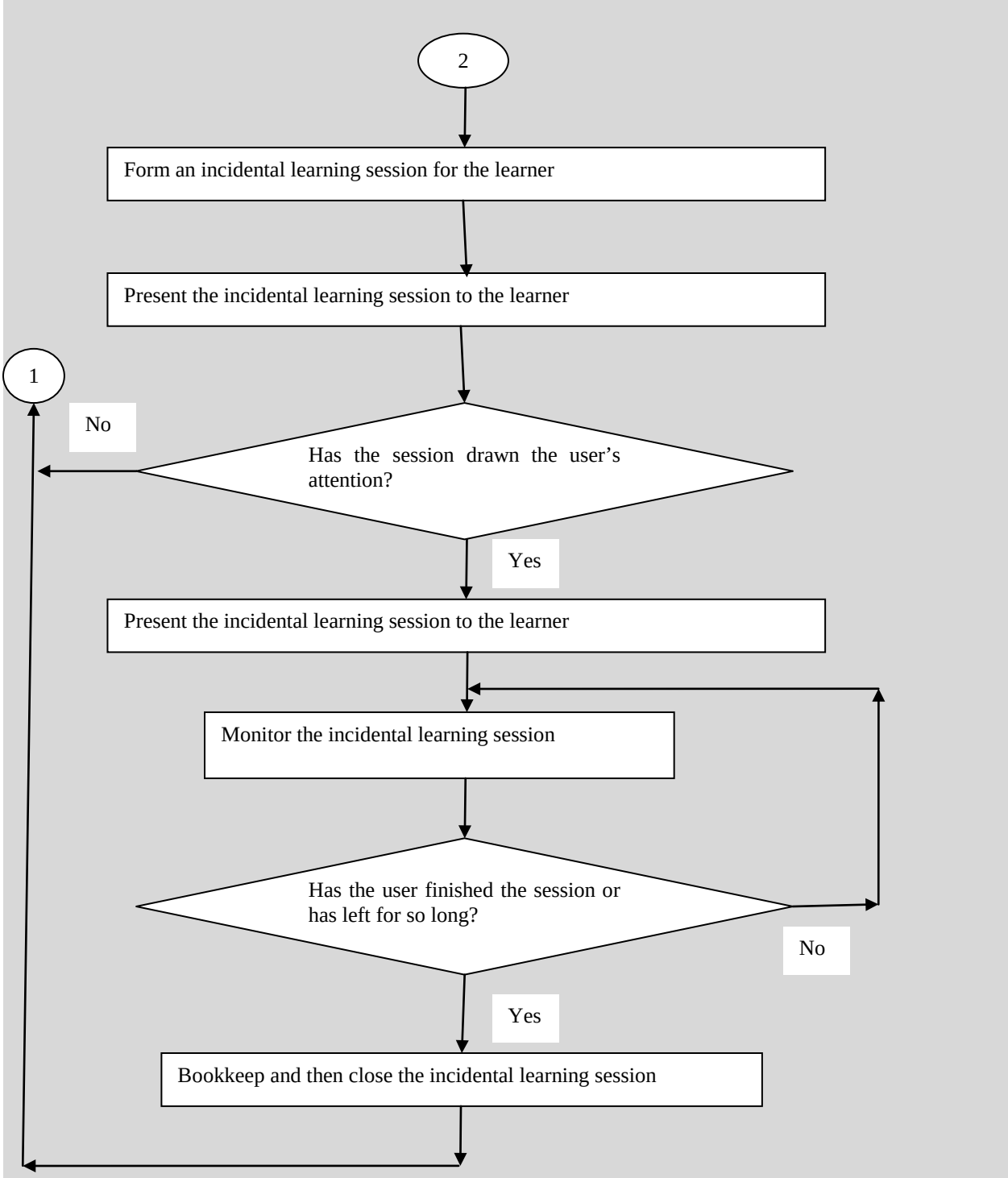


Figure 1 -- a protocole for intelligent agent enabling incidental learning on the web

Some more strategic details can be added to the protocol. One of these details is that, if the user keeps ignoring the incidental learning session fired-up by the intelligent agent, the agent should stop doing it after a certain number of tries, or at least should present a vitally different incidental learning session next time, by refining the learning objectives and rethinking the user's learning style.

3. Types of Incidental Learning enabled by the protocol

The unique feature of incidental learning is that learning occurred incidentally in the view of the learner. However, although the occurrence of learning may seem to be random (that's why incidental learning is also called random learning), the knowledge and skills acquired through it don't need to be random, especially when the incidental learning session is enabled by an intelligent agent system. That is to say, the learning outcome can be either expected or unexpected by the learner, depending on how the intelligent agent chose the learning objectives when forming a learning session. We then have the following two types of incidental learning supported by the protocol:

1. incidental learning from which something expected has been learned, we use letter E to refer to such learning, use letter E with a subscript to refer a learning goal that may consists of small gradients of learning goals or knowledge, denoted by letter e with or with a subscript. Figure 2 depicts the scenario where all learning sessions deliver expected learning outcomes to the learner. If the i, j, k and l are the same, it means all incidental learning sessions are for the same consistent learning outcomes expected by the learner;

2. incidental learning from which something unexpected has been learned, we use letter U to refer to such learning, use letter U with a subscript to refer a learning goal that may consists of small gradients of learning goals or knowledge, denoted by letter u with or with a subscript. Figure 3 depicts the scenario where all learning sessions deliver unexpected learning outcomes to the learner. If the i, j, k and l are the same, it means all incidental learning sessions are for the same consistent learning outcomes but unexpected by the learner;

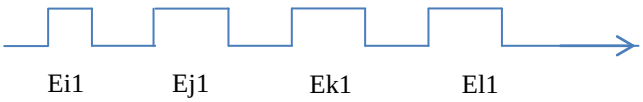


Figure 2 - incidental learning with expected learning outcomes

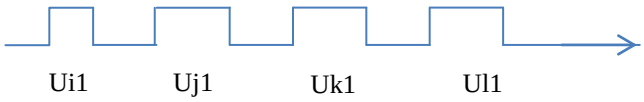


Figure 3 - incidental learning with unexpected learning outcomes

There can be cases where incidental learning sessions are mixed with expected and unexpected learning outcomes. To the question of why incidental learning sessions with unexpected learning outcomes are even needed, the answer is that, some learners may not even know what he or she wants to learn. In these cases, it is necessary for the intelligent agent to set the learning path for the learner.

4. Discussions

We presented in this paper a protocol for the implementation of intelligent agent systems that enable incidental learning on the web. The protocol has prescribed the key requirements that must be met when implementing such an intelligent system, and the steps such an intelligent agent system must take when rendering an incidental learning session to a web user.

When implementing such an intelligent system to enable incidental learning on the web, one needs to overcome several technical challenges. The first challenge is to understand the learner in terms of his or her interests, short and long term learning goals as well as the learning style, and the second challenge is to getting the right learning content and to form an interesting and effective learning session, and the last but not least challenge is to monitor the learner's activities and behaviors both in and outside the learning session, in order to make the right decision on when to render and when to close an incidental learning session. Further research and investigation into these challenges are needed, and findings from investigation into these challenges will be reported in forthcoming papers..

5. References

[1] Baskett, H. K. M., Workplace Factors Which Enhance Self-directed Learning, a report of a project on self-directed learning in the workplace, a paper presented to the Seventh International Symposium on Self-Directed Learning, West Palm Beach, Florida, January 21-23, 1993. Available online at http://www.eric.ed.gov/PDFS/ED359354.pdf

[2] Baylor, A.L. Perceived disorientation and incidental learning in a Web-based environment: Internal and external factors J of Educational Multimedia and Hypermedia Vol. 10/3 227-251:2001

[3] Elley, W.B. (Dez 2005), [http//cela.albany.edu/reports/inpraise/main.html In praise of incidental learning: Lessons from some empirical findings on language acquisition: 1997]

[4] Kerka, Sandra 2000, Trends and Issues 2000 Alert No.18, Incidental learning

[5] Marsick, V.J. & KJ Watkins (2001). Informal and incidental learning, new directions for adult and continuing education 89 25-34.

[6] Mayer, R.E., Heiser, J. & Lonn,S. (2001). Cognitive constraints on multimedia learning: when presenting more material results in less understanding. Journal of educational psychology, 93 pp.187-198

[7] Schank, Roger, C. & Ch. Cleary (1995). Engines for Education, Lawrence Erlbaum Ass. Publishers Hillsdale NJ, pp. 95-105 (Incidental learning chapter).

[8] Chunyu Kit; Ng, J.Y.H., "An Intelligent Web Agent to Mine Bilingual Parallel Pages via Automatic Discovery of URL Pairing Patterns," Web Intelligence and Intelligent Agent Technology Workshops, 2007 IEEE/WIC/ACM International Conferences on , vol., no., pp.526,529, 5-12 Nov. 2007

[9] Lee, R.S.T.; Liu, J.N.-K.; Yeung, K.S.Y.; Sin, A.H.L.; Shum, D.T.F., "Agent-Based Web Content Engagement Time (WCET) Analyzer on e-Publication System,"Intelligent Systems Design and Applications, 2009. ISDA '09. Ninth International Conference on , vol., no., pp.67,72, Nov. 30 2009-Dec. 2 2009

[10] Boddu, S.B.; Anne, V.P.K.; Kurra, R.R.; Mishra, D.K., "Knowledge Discovery and Retrieval on World Wide Web Using Web Structure Mining,"Mathematical/Analytical Modelling and Computer Simulation (AMS), 2010 Fourth Asia International Conference on , vol., no., pp.532,537, 26-28 May 2010

[11] Cooley, R.; Mobasher, B.; Srivastava, J., "Web mining: information and pattern discovery on the World Wide Web," Tools with Artificial Intelligence, 1997. Proceedings., Ninth IEEE International Conference on , vol., no., pp.558,567, 3-8 Nov 1997

[12] Toth, P., "E-learning and web mining," Intelligent Systems and Informatics (SISY), 2012 IEEE 10th Jubilee International Symposium on , vol., no., pp.439,444, 20-22 Sept. 2012

The Cognitive, Interactive Training Environment (CITE)

Dr. James A. Crowder
Raytheon Intelligence, Information, and Services
16800 E. Centretech Parkway, Aurora, Colorado 80011

Shelli Friess MA, NCC
Relevant Counseling
P.O. Box 4193, Englewood, CO 80155

Abstract *- Utilizing software to partially or fully automate tasks is now commonplace. However, the capabilities of the software performing these tasks typically do not improve over time (as humans would perform the same tasks). Here we discuss the use of software that learns and improves through the use a Human Operator acting as a Mentor for the software, until the software is capable of performing the desired operations autonomously and with improvements. We will discuss the rule of Human Interaction Learning (HIL) as the operator changes from manager to mentor to monitor while the software evolves from learner to performer. Presented will be a discussion of the Levels of Automation of Design and Action Selection (Parauraman, Sheridan, and Wickens), and the cognitive software architectures required to allow the system to learn and evolve. We will walk through examples, illustrating how the software would change over time (memory, artificial emotions, etc.), such that it will perform the actions learned. Included will be a discussion of the utility of Human Mentored Software (HMS) and its role in future systems design and implementation, including discussion of a human-computer collaboration system, called the Cognitive, Interactive Training Environment (CITE). The CITE system learns and improves through the use of a Human Operator acting as a Mentor for the software, until the software is capable of performing the desired operations autonomously and with improvements, as well as the software active as a mentor for the human. CITE provides for Human Interaction Learning (HIL); as the human operator's role changes from manager to mentor to monitor while the system evolves from learner to performer. Here, CITE will be explored as a means to allow human-AI communication to allow humans (particularly soldiers) to learn Cognitive Resilience as a result of CITE interactions.*

Keywords: Cognitive Self-Regulation, Human Interaction Learning, Human-Computer Collaboration, Cognitive Resilience.

1. Introduction

Developing "Warrior Resiliency" has been a focus of armies since the dawn of time. There has been much research over the last decades to understand and provide systems and methodologies to develop and enhance cognitive resiliency in soldiers. The ability to adapt to adversity and overcome barriers in all walks of life is critical to a soldier's overall mental health and strength. And while physical resilience is very important, lack of psychological resilience can cripple a soldier just as easily as physical impairments. In addition to peak physical performance, each soldier must be well-balanced psychologically and socially in order to sustain the intense rigors of military life, especially when it must be balanced with home, family, and community life.

What is described here is an automated, interactive, cognitive system, called the Cognitive, Interactive Training Environment (CITE, pronounced "KITE") that will provide the necessary cognitive psychological training to provide the warfighter with the ability to maintain mission readiness and psychological self-regulation before, during, and after stressful situations, whether in combat or at home. The CITE system utilizes advances in Artificial Cognitive Systems developed by Raytheon combined with Linguistic Ontological Technologies developed by Purdue University to create an interactive environment capable of providing training to and adapting to individual soldier's needs and requirements.

Included will be a discussion of the utility of Human Mentored Software (HMS) and its role in future systems design and implementation, including discussion of the human-computer collaboration system, CITE system. CITE system learns and improves through the use of a Human Operator acting as a Mentor for the software, until the software is capable of

performing the desired operations autonomously and with improvements, as well as the software active as a mentor for the human. CITE provides for Human Interaction Learning (HIL); as the human operator's role changes from manager to mentor to monitor while the system evolves from learner to performer. Here, CITE will be explored as a means to allow human-AI communication to allow humans (particularly soldiers) to learn Cognitive Resilience as a result of CITE interactions.

CITE will provide automated, interactive training and capture and report metacognitive indicators and metrics that allow complete assessment of psychological resilience. CITE will gather and assess metacognitive indicators like:

- Problem solving skills
- Social skills
- Relationship Skills
- Self-awareness
- Emotional self-regulation
- Cognitive self-regulation

CITE will provide the soldier with training to assist in development of the self-monitoring and self-assessment skills needed for psychological self-regulation. Figure 1 illustrates the block diagram of the CITE system.

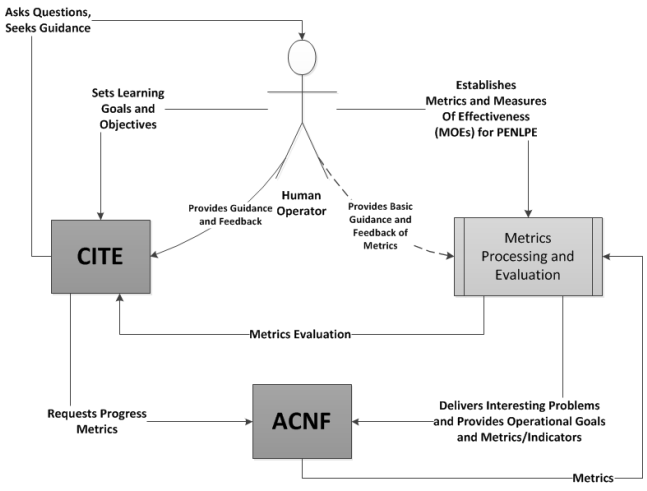


Figure 1 – Block Diagram of the CITE System

2. Human Interaction Learning and Warrior Resilience

The Office of Naval Research (ONR) website describes Warrior Resilience[1] as:

> "...improving the cognitive agility, flexibility and capacity of expeditionary warfighters by making them mentally tough, resilient to stress and well-adapted to chaotic, irregular environments"

In order to affect cognitive resiliency among warfighters, that must deal with a host of complex environments, both on and off the battlefield, an interactive training system must have a comprehensive understanding of the psychological, physical, and social elements and their interrelationships with warfighter performance. What we describe here is a comprehensive, Cognitive Interactive Training Environment (CITE, pronounces "kite") that will provide extensive, interactive, cognitive training and evaluation to provide warfighters with psychological self-evaluation, and self-awareness, and self-regulation skills to improve their cognitive performance throughout deployment as well as off the battlefield. We describe advanced instructional methods, based on Dr. Peter Levines [Levine 1997], autonomic nervous system states, that will provide cognitive behaviour training, metrics, and biomarkers that include environmental, contextual, and social components to affect real cognitive resiliency. The CITE system will provide the cognitive training tools to develop cognitive self-regulation and mitigation strategies to reduce cognitive dissonance among warfighters.

3. Cognitive Resiliency and Memory Development

Cognitive Resiliency develops in the brain through training that results in the learned ability to respond or self-regulate to severe

[1] http://www.onr.navy.mil/en/Media-Center/Fact-Sheets/Mental-Resilience-Cognitive-Agility.aspx

psychological changes which may result from many forms of trauma and/or change; physical, emotional, environmental, or social. These learned abilities, then, get stored as memories within the human brain. Memories, in general, are divided according to the functions they serve [Newell 2003]. To qualify as a "memory" a cognitive input must cause both enduring changes within the nervous system (affect the autonomous nervous system states) and must also affect emotional and motivational responses and goals [LeDoux 1996]. A memory must induce some change that affects the nervous system and drives some physical change, in addition to modifying the human conceptual ontology, brought about by the memory being in the class of things that are affected by input, and therefore, affect other forms of behaviour. There are no memories that are neutral from a behavioural standpoint (Crowder and Friess 2010a].

The underlying issues and challenges facing Artificially Intelligent systems today are not new. Information processing and dissemination within these types of systems have generally been expensive to create, operate and maintain. Other artificially intelligent system challenges involve information flow throughout the system. If flow is not designed carefully and purposefully, the flood of information via messages within these systems and between their software and hardware components can cause delays in information transfer, delaying or stunting of the learning process which can result in incorrect or catastrophic decisions.

4. Procedural Memory Development and Resiliency

One of the main divisions of human memory is "Procedural Memory." Procedural memory is a form of implicit memory that includes classical conditioning and the acquisition of skills. Procedural memory creation contains central pattern generators that form as a result of teaching or practice and are formed independently of conscious or declarative memory. In his work on Procedural Memory and contextual Representation, Kahana showed that retrieval of implicit procedural memories is a cue-dependent process that contains both semantic and temporal components (Kahana, Howard, and Plyn 2008)]. Creation of Procedural Memories is tied to not only repetition of tasks, but also to the richness of the semantic association structure (Landauer and Dumais 1997). In order to provide cognitive resilience, the CITE system provides interactive training that allows warfighters to create procedural memories, or "scripts" that have emotional, social, and psychological triggers and provide the skills required at the time for cognitive self-evaluation, self-awareness, and self-regulation to present or reduce psychological disorders, or problems, caused by trauma, either physical, psychological, or environmental [Crowder and Friess 2011a].

The CITE interactive training will provide a cognitive system that will interact and learn from the warfighter, developing strategies and training scenarios specific to that warfighter, thus allowing the warfighter to develop procedural memory strategies, i.e., implicit procedural memories, that will "kick in" under specific emotional memory queues, based on physical, emotional, psychological, and/or environmental events that the warfighter encounters. CITE will develop a model, or picture, or the warfighter's prefrontal cortex, based on the cognitive interactions with the system. This prefrontal cortex, or mediator, model (see Figure 2) allows CITE to understand the warfighters particular cognitive processes and what drives changes between emotional and cognitive states [Crowder and Friess 2011b] for that warfighter.

Based on these derived cognitive models, cognitive interactions between CITE and the warfigher will affect procedural memory creation, which will allow self-assessment, self-awareness, and self-regulation, driving cognitive

self-soothing procedures to be initiated, greatly reducing mental stress and thus the possibilities of mental disorders [Crowder and Friess 2010b].

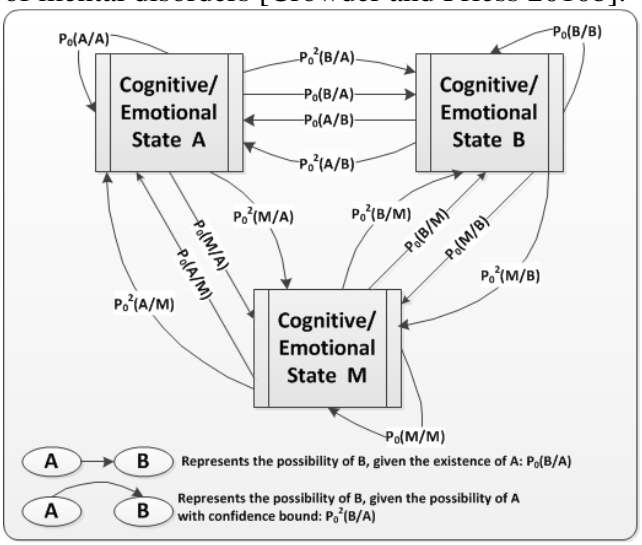


Figure 2 – Derived Warfighter Prefrontal Cortex Model, based on CITE interactions

5. The CITE Artificial Cognitive Neural Framework

The CITE warfighter cognitive models are derived through human-machine interactions and stored within the CITE Artificial Cognitive Neural Framework[2] (see Figure 3 below). In order to understand the world we live in, humans synthesize models that enable us to reason about what we perceive. Situations warfighters find themselves in, as well as the information they receive comes from a variety of sources, rendering it fuzzy. These diverse sources often do not have consistent contextual bases and this introduces ambiguity into the correlation and inferences the warfighter applies to the combined information. We have the ability to perceive the world we see and form our own concepts to describe and make decisions. To do this, we use language fuzzily and we communicate fuzzily, adapting and evolving our communication and processing to best fit the needs of our personal and conceptual views, along with our goals and vision for where we need to grow and evolve to [Zadeh 2004]. In order to understand and provide individualized resiliency training for each warfighter, CITE must be able to organize information from the warfighter semantically into meaningful fuzzy concepts and models that provide a conceptual ontology [Raskin, Taylor, and Hempelmann 2010, and Taylor and Raskin 2010] of the individual warfighters cognitive abilities.

The purpose of these warfighter cognitive models is to provide cognitive knowledge products that reflect the state of being for the individual warfighter that includes metrics to measure cognitive resiliency for all aspects of life, to include battlefield, family and community.

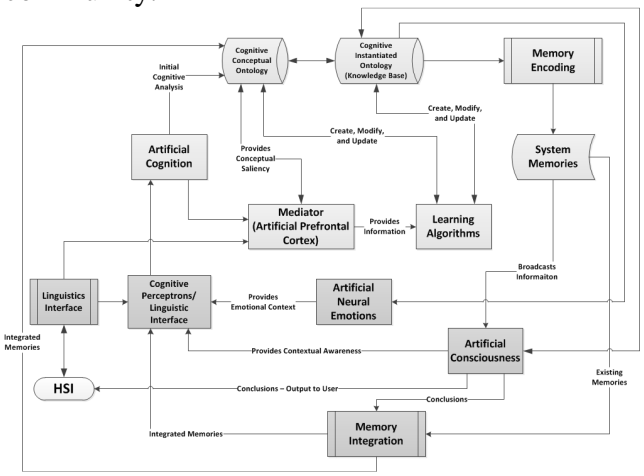


Figure 3 – The CITE Artificial Cognitive Neural Framework

6. Conclusions and Discussion

The purpose of these warfighter cognitive models is to provide cognitive knowledge products that reflect the state of being for the individual warfighter that includes metrics to measure cognitive resiliency for all aspects of life, to include battlefield, family and community. Much more research is needed, including extensive work in human/AI interaction to determine the interfaces necessary to implement the cognitive collaboration.

[2] Patent Pending

7. References

1. Crowder, J., and Friess, S. 2010a. Artificial Emotions and Emotional Memory. *Proceedings of the 11th Annual International Conference on Artificial Intelligence*, Las Vegas, Nevada.
2. Crowder, J. A., and Friess, S. 2010b. Artificial Neural Diagnostics and Prognostics: Self-Soothing in Cognitive Systems. *Proceedings of the 11th Annual International Conference on Artificial Intelligence*, Las Vegas, Nevada.
3. Crowder, J. and Friess, S. 2011a. Metacognition and Metamemory Concepts for AI Systems. *Proceedings of the 12th Annual International Conference on Artificial Intelligence*, Las Vegas, Nevada.
4. Crowder, J. and Friess, S. 2011b. The Artificial Prefrontal Cortex: Artificial Consciousness. *Proceedings of the 12th Annual International Conference on Artificial Intelligence*, Las Vegas, Nevada.
5. Crowder J., and Friess, S. 2012. Artificial Psychology: The Psychology of AI. *Proceedings of the International Multi-Conference on Information and Cybernetics*, Pasadena, CA.
6. Kahana, M. J., Howard, M. W., and Polyn, S. M. 2008. Associative retrieval processes in episodic memory. In H. L. Roediger III (Ed.), *Cognitive psychology of memory. Vol. 2 of Learning and memory: A comprehensive reference, 4 vols. (J. Byrne, Editor).* Oxford: Elsevier.
7. LaBar KS and Cabeza. 2006. Cognitive Neuroscience of Emotional Memory. *Nat Rev Neurosci 7: 54-64*.
8. LeDoux JE. 1996. The Emotional Brain. *Simon and Schuster*, New York, NY.
9. Marsella, S., and Gratch J., "*A Step Towards Irrationality: Using Emotion to Change Belief*." 1st International Joint Conference on Autonomous Agents and Multi-Agent Systems, Bologna, Italy (July 2002).
10. Levine, P. 1997. Walking the Tiger: Healing Trauma. *North Atlantic Books*, Berkeley, CA.
11. Newell, A. 2003. Unified Theories of Cognition. *Harvard University Press*, Cambridge, MA.
12. Raskin, V., Taylor, J. M., & Hempelmann, C. F. 2010. Ontological semantic technology for detecting insider threat and social engineering. *New Security Paradigms Workshop*, Concord, MA.
13. Taylor, J. M., & Raskin, V. 2010. Fuzzy ontology for natural language. *29th International Conference of the North American Fuzzy Information Processing Society,* Toronto, Ontario, Canada.
14. Zadeh, L. 2004. A note of Web Intelligence, World Knowledge, and Fuzzy Logic. *Data and Knowledge Engineering, vol. 50, pp. 291-304*.

A Population-Based Incremental Learning Tool for Finite Impulse Response (FIR) Filters Design

Katherine Ortega, Freddy Bolanos
Electrical and Automatic Engineering Department
National University of Colombia
Medellin, Colombia
{ikortegag,fbolanosm}@unal.edu.co

***Abstract*—This paper describes a software tool for synthesizing Finite Impulse Response (FIR) filters, by means of an optimization engine called Population-Based Incremental Learning (PBIL). The tool is devoted to obtain the combination of fixed-point filter coefficients which better approximates the desired response in the frequency domain. Such an approach is quite different from several filter design methodologies, in which quantification of the coefficients takes place at the end of the design process. Some results are derived from comparing the mean absolute error between the desired frequency response and the response of the approximations obtained with the PBIL optimization engine, and the approximation obtained with classical methods.**

I. Introduction

The design process of digital filters involves two choices. In the first place, Infinite Impulse Response (IIR) filters imply the use of less memory blocks, but may be unstable and exhibit issues when dealing with low-precision implementations. On the other hand, Finite Impulse Response (FIR) filters usually require more memory blocks for an equivalent performance, when compared with IIR filters. The main advantage of FIR filters relies in that they are always stable, and may be implemented by using low-precision coefficients [1].

Since the time domain response of a stable IIR filter approximates to zero if time grows toward infinity, some design methodologies are aimed to obtain first an IIR filter and then truncate its response in time domain, in order to obtain a FIR approximation [2]. The more coefficients (taps) are present in the FIR approximation, the more accurate will be the filter behavior with respect to the desired response. As an undesirable consequence of the truncating process, the FIR response exhibits a sort of oscillation in the frequency domain, which is referred as Gibbs oscillations. In order to minimize the effect of such oscillations, several windowing methods have been proposed in literature [3]. If the filter implementation by using floating-point coefficients is prohibitive, the last stage of a traditional design flow requires the quantification of such coefficients, in order to adjust them to fixed-point platforms.

The quality of the FIR approximation in the design process described above, is affected by three main factors. In the first place the finite nature of the FIR response in time domain introduces error with respect to the desired filter shape. The way in which such an error is minimized is to increase the filter size (i.e., the amount of filter coefficients or taps). Secondly, windowing allows reducing Gibbs oscillations, but also affects the shape of the FIR filter, yielding differences with respect to the desired response in frequency domain. Several windows have been proposed in literature. Each of them has different behaviors and features, with respect to distortion performed to the filter in frequency domain. Finally, the quantification process at the end of the design process, also affects the quality of the approximation. The way for improve such a quality is to reduce the distance between contiguos levels in the quantification process, which is accomplished by increasing the bit resolution of the coefficients.

This paper describes a PBIL approach, which is used as an optimization engine to adjust the filter coefficients progressively. Such coefficients represent the response of the FIR filter in time domain. Each coefficient has a limited precision, which corresponds to its resolution in bits. This means that the quantification issue is taken into account along the design process and not only at the end, as it happens with traditional design methodologies.

The PBIL optimization algorithm uses a population of potential solutions in order to approach to a given optimum, as it happens with evolutionary algorithms [4], [5]. The main difference of PBIL algorithms with respect to other population-based approaches, relies in the form in which such population is represented. Instead of having a single representation for each individual of the population, PBIL uses a probability array, in order to represent all the individuals at once. The stochastic information provided by the array leads the optimization process, and allows to perform a design space exploration, which means that the algorithm surveys the solutions space, looking for those solutions which optimize the problem at hand. The PBIL optimization engine has been successfully used in several design problems, and its performance may be better than with other population-based techniques, due to its compact population representation [6], [7], [8].

The rest of this document is organized as follows. Section II describes briefly the PBIL algorithm, and some important related subjects such as the learning rate, the probability array and the learning rate parameter. Section III describes the software tool developed for FIR filter design. Some implementation results, as well as a simple comparison with respect to traditional design methodologies is given in this section too. Finally, conclusion and future work is presented in section IV.

II. The PBIL algorithm

The core of the PBIL optimization engine is an array which stores the probabilities of occurrence of each feature of a given solution. At early stages of the optimization process, the population is very disperse, meaning that all potential solutions are candidates to be an optimum. This is reflected in the probability array by the fact that all potential features in the array are equally probable. As the algorithm converges toward a given optimum, some entries of the matrix have more weight than others, which means that the array is pointing to a set of solutions and some other sets of solutions are being discarded in the searching process. At the end of the optimization process, entries of the probability array are very close either to 0 or 1. This combination of values is pointing toward the optimal solution found by the algorithm. Algorithm 1 describes the basic PBIL approach.

Algorithm 1: Basic PBIL algorithm.

Input: A probability array, called P.
Output: An optimized solution to the problem at hand.

```
begin
    P = Initialize_Array(P);
    repeat
        Pop = Create_Population(P);
        Fitness = Evaluate_Population(Pop);
        Best = Choose_Best(Pop, Fitness);
        P = Update_Array(P, Best, LR);
    until (Stopping_Criterion = false);
    return Best;
end
```

The routine $Initialize\_Array$ in Algorithm 1 is aimed to give initial values to all probabilities in array P. Probability values are set in such a way that all solution features be equally probable. This favors the population diversity at early stages of the convergence process, and avoids the algorithm to get stuck at a local minimum. At the beginning of each algorithm's iteration, a set of solutions (population) is created from the array P, by means of the $Create\_Population$ routine. Individuals created in such routine exhibits those features which have more probability of appearance, according with the P array.

The $Evaluate\_Population$ routine gives a fitness value to each individual of the population just created. Fitness is a measure of the optimality of each solution. By using as a selection criterium the fitness values, the best solution of the population is chosen, by means of the $Choose\_Best$ routine. The obtained best individual is used to update the P array, in order to increase those probabilities associated with the features present in such an individual. Consequently, those features which appear more frequently in the best solutions found, will increase their corresponding probabilities, and will be candidates to be part of the optimal solution. The algorithm may stop the iterations when the probabilities in the P array are concentrated in some entries, or when all probabilities in the array are close to their extreme values, i.e., 0 and 1.

Some PBIL implementations use the array's Entropy to determine whether it is time to stop the iterative process [7]. Such an Entropy value may be calculated in the same way as for information theory. Let us suppose that the PBIL probability array takes the form of a linear vector with N entries. In such conditions, entropy of the array may be caculated as described in Equation (1).

$$H = -\sum_{k=0}^{N-1} P_k \times Log\left(P_k\right) \tag{1}$$

Where H is the entropy of the array P, and P_k is the k_{th} entry of the same array. When the represented population contains a high diversity of individuals, the value of H approaches to its maximum. Alternatively, when all the probability entries of the array get closer to their extreme values (0 or 1), value of H approaches to zero.

As shown in Algorithm 1, the $Update\_Array$ routine uses three input arguments: The P array, the best found solution, and a third parameter, called LR. LR is often referred as Learning Rate and controls the speed of convergence of the algorithm. A High value of LR leads to faster convergence processes, at expenses of a reduction of the quality of the found optimum. The latter is because when a good solution is found, the associated probabilities are increased in a very abrupt way, forcing the population to converge quickly to such a solution. This gives no chance to explore another good solutions (which may be better that the current one) and is referred in literature as exploitation of the solutions space. Alternatively, if the value of LR is low, the algorithm has the chance to evaluate more carefully the solutions space and, hence, produce better solutions at the end of the optimization process. Such an strategy is known as exploration of the solutions space, and implies that the algorithm will last more time finding an optimal solution. Adaptive PBIL solutions use the Entropy values to adjust the LR parameter, in order to set convergence speeds according with the convergence process status.

III. Software Tool for FIR filters design

As already mentioned, the developed software tool uses a PBIL optimization engine, in order to adjust the fixed-point coefficients of a FIR filter to approximate an IIR response. Figure 1 shows the PBIL array representation for the filter design problem.

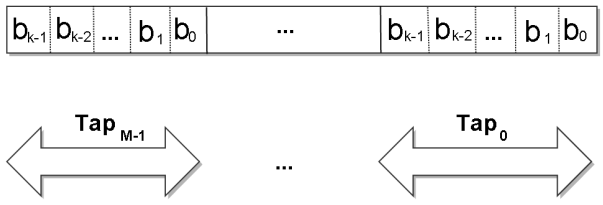


Fig. 1. PBIL array representation.

As depicted in Figure 1, the PBIL array takes the form of a $k \times M$ entries vector, where M is the number of taps (coefficients) which constitute the FIR filter, and k is the bit resolution used for representing each tap. A single entry

of such vector represents the probability of the k bit in the corresponding tap representation to be set to 1. As already mentioned, at the end of the PBIL algorithm execution each entry on the array is very close either to 0 or to 1, so the optimal solution may be easily derived from this array.

The software tool was developed in the Matlab GUIDE and has two main parts. Firstly, the desired frequency response of the filter must be specified. Although any shape may serve as a target filter, the developed tool only deals with four classical forms, namely, ideal filter, Butterworth, and Chebyshev type I and II. In the second stage of the design process with the proposed tool, the number of taps and the bit resolution of each tap, must be specified. Then, the PBIL optimization process may begin.

Let's illustrate the approximation process with a simple example. As explained before in Figure 1, the FIR filter representation is an array of $M \times k$ bits, meaning that the designed filter will be composed by M taps, each with k bits in its binary representation. For the example we are about to show, M and k are both equal to 8 (i.e., a FIR filter with M taps, each with $8-$bit precision). The later means that the PBIL array representation will take the form of a $64-$probabilities vector.

In such vector representation, which is depicted in Figure 2, the entry b_{ij} is associated with the probability of the bit j of tap i to be equal to one in the optimized solution. Routine $Initialize\_Array$ of Algorithm 1, firstly sets to 0.5 all the values of the probability vector, depicted in Figure 2. This is done because all binary combinations start being considered as equally potential solutions to the FIR filter design problem. The value of 0.5 for these probabilities allows to explore the solutions space at first stages of the optimization process.

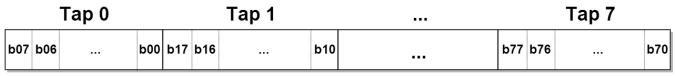


Fig. 2. The PBIL array representation for a $M = 8, k = 8$ FIR filter.

At each iteration of the algorithm, a new population is created by using on the probabilities vector depicted in Figure 2. If a given entry in the vector (i.e., a feature) exhibits a higher probability value, the associated bit will appear as 1 more frequently in the solutions population. In other words, each entry represents the probability of appearance of each feature in the final optimized solution.

Once the population is created, it is necessary to evaluate its fitness with respect to the desired FIR filter. Such a task is performed by the $Evaluate\_Population$ routine. The approximation error, which serves as a fitness value to the PBIL algorithm, corresponds the mean absolute difference between the desired frequency response, and the response of the best solution found so far. Equation (2) describes such calculations.

$$E = \langle |H_{Desired} - H_{Best}| \rangle \tag{2}$$

In Equation (2), $\langle x \rangle$ represents the mean value of x, $H_{Desired}$ corresponds to the desired IIR response which will be approximated, and H_{Best} refers to the response of the best solution found so far. It must be highlighted that the absolute difference of Equation (2) takes into account the phase response of the filter, so complex values are considered in the mean error calculation. The error (E) of Equation (2) is calculated along a set of 2048 frequency samples, uniformly distributed in the range: $0 \leq f \leq 0.5$, being the value 0.5 equivalent to half of the sample rate.

By using the mean absolute error of Equation (2) as criterium, it is possible to find the best solution of the current population. Such best solution will have the form of a $64-$bit vector, as explained before. This allows to update the probabilities of the PBIL vector: Some probability values will be slightly increased (i.e., those entries in which the best solution is equal to 1), and the remaining ones will be slightly decreased, because the associated entries in the best solution are equal to 0. The way in which probabilities of the PBIL vector are updated, takes the form of a modified Hebbian rule [7], [8].

As explained so far, the probabilities in the PBIL vector will gradually accumulate the information of the best solutions found so far in the iterative process. After an exploration process in which several good solutions have been used to update the PBIL array, entries in such array will converge to their extreme values (i.e., 0 or 1). If all entries in the PBIL vector are very close either to 0 or 1, the algorithm is near of reaching an optimal solution, and the iterative process may be stopped. A measure of how close is the algorithm of finding a solution, is given by the Entropy of the PBIL array. Normalized Entropy provides a measure of diversity of the PBIL array, and may be calculated as shown in Equation (3).

$$H_N = -\frac{2}{M} \times \sum_{i=0}^{M-1} \sum_{j=0}^{k-1} b_{ij} \times Log_2 (b_{ij}) \tag{3}$$

At the start of the PBIL optimization process, values of H_N are very close to 1, meaning that the associated vector represents a high population diversity. As the algorithm converges iteratively, the Entropy value in Equation (3) decreases and approaches to 0. The later means that probability entries in the vector are very close to either 0 or 1. In fact, the proximity of the vector's Entropy to 0 may be used as stopping criterium by the PBIL algorithm.

At the end of the iterative process, the optimal solution may be easily derived from the entries in the vector probability, by simply rounding its values. A binary vector of $M \times k$ entries, representing the M taps of the FIR filter, each with a precision of k bits, will be obtained. For the example previously described in Figure 2, the 64 probabilities of the PBIL vector are rounded to 64 binary values (bits), which represents a FIR filter with $M = 8$ and $k = 8$.

Figure 3 depicts the software application for FIR filter design, based on the PBIL optimization algorithm. Such application was developed by using the Matlab Guide environment. Several tuning tasks must be performed for the sake of establishing the desired shape of the filter, which was labeled as $H\_Desired$ in Equation (2):

- Defining the shape of the desired filter, meaning ideal, Butterworth, Chebyshev I or Chebyshev II.

- Defining of the kind of the response, i.e., lowpass, highpass, bandpass, or band reject.
- Setting up the cutoff frequency of the filter, filter order (except for ideal filters), and the ripple allowable in the response of bandpass or band reject for Chebyshev filters.

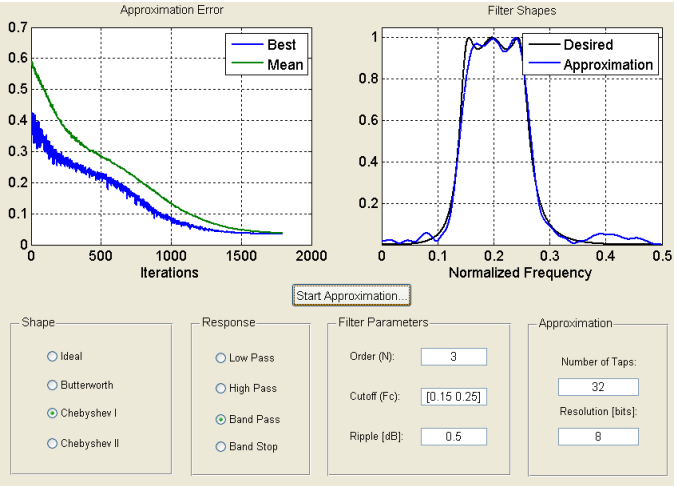


Fig. 3. Aspect of the FIR filter design tool

Figure 3 also shows how to set the values of M (number of Taps of the approximation) and k (bit resolution of each tap). Such figure depicts the final results of an optimization process for a FIR filter with $M = 32$ taps, each with a resolution of $k = 8$ bits.

Two graphs top the application in Figure 3: The first-left graph depicts the evolution of error as defined in Equation (2), with respect to the number of iterations of the PBIL algorithm. Two errors are represented in such graph: the mean error of the whole population of solutions, and the error of the best solution found so far in the optimization process. On the other hand, the second-right graph shows the frequency response of the best filter found so far, along with the desired-filter frequency response.

In order to compare the application depicted in Figure 3, a traditional design flow for FIR filters were developed as a reference instance [3], [1]. Such a design flow involves the following steps:

1) If the desired filter holds the ideal shape, the $H\_Desired$ response may be derived directly: the response is equal to 1 in the bandpass, and 0 elsewhere. However, if other shape is being considered (Butterworth, Chebyshev), an infinite impulse response (i.e., IIR) version of the system is obtained by generating the suitable polynomials for each kind of filter.
2) Starting from the IIR implementation of the filter, a total of M taps are generated, by storing the first M samples of the system's response, when a unitary impulse is applied as input.
3) A Hamming window is used in this first FIR version of the filter, for the sake of avoiding Gibbs oscillations in the obtained response [2].
4) The M coefficients (taps) of the filter are quantized to a set of 2^k allowable values, in order to represent them as $k-$bit words for comparison purposes.

Table I summarizes some of the obtained results with the PBIL optimization tool, with respect to the traditional design flow just described. Several filter shapes (Butterworth, Chebyshev) and responses (lowpass, highpass, bandpass, and band reject) were tested. For each test, a target FIR filter with $M = 32$ and $k = 8$ was optimized by using the PBIL tool, and was compared with respect to the filter obtained with the traditional design flow just described. It must be highlighted that a FIR filter with $M = 32$ taps implemented with fixed-point coefficients, with a precision of $k = 8$ bits, is considered poor with respect to current FIR filter implementations. If either M or k are increased, the optimization process counts with more freedom degrees for adjusting the filter approximation, which may lead to lower errors. The selection of such values for M and k, derives from the fact that this application tool is devised for the implementation of the FIR filters in low performance platforms.

TABLE I. ERROR RESULTS FOR THE PBIL TOOL

Filter Shape	Order	PBIL Error	Reference Error
Butterworth, Highpass	2	0.1515	0.1378
Chebyshev I, Bandpass	3	0.0339	0.1248
Chebyshev II, Lowpass	4	0.0462	0.2452
Butterworth, Stopband	5	0.0632	0.4042

Table I compares the PBIL optimization tool with the traditional design flow, in terms of error with respect to the desired response, as defined in Equation (2). Third column of Table I shows the error obtained with the PBIL tool for several filter designs, whilst fourth column shows the corresponding reference error. When trying to approximate a Butterworth filter, which has a smoother shape, errors derived from filters obtained both from the PBIL tool, and the traditional design flow, exhibits higher values. As a consequence of the finite nature of its response in time domain, FIR filters are more prone to oscillate in frequency domain. If the desired response is smooth, i.e., with no oscillations, the error, as defined in Equation 2, may be strongly increased. Nevertheless, for the sake of performing a fair test, the same constraints were imposed to the FIR filters designed both with the PBIL tool and the reference design flow, so a row-to-row comparison is suitable in Table I.

Results depicted in Table I show that errors obtained from the PBIL tool are similar to those which were derived from the traditional design flow, for small-order filters (i.e., small values of M). However, as the order of the desired filter increases, quality of solutions obtained with the PBIL tool may improve even an order of magnitude with respect to the traditionally-designed filters.

The mean convergence time for the tests performed to the PBIL tool, is in the order of several minutes, when was running on Matlab 2011a for windows, over a PC with 8 Gbytes of RAM memory and a Core i7 processor. The LR parameter was set to 0.01 for all the simulations depicted in Table I. As already mentioned, increasing the value of LR may improve the convergence time, at expense of the decreased quality of the obtained solutions.

IV. Conclusion

A tool for FIR filter synthesis has been described and tested. Simulation results are promising and shows a good performance of the obtained filters in terms of mean absolute error, measured as the mean difference between the desired frequency response, and the frequency response obtained from the optimized filter. Preliminary results suggest that PBIL tool may improve the quality of the obtained FIR filters with respect to traditionally-designed systems.

One of the advantages of the proposed PBIL-based tool with respect to other design methodologies, relies in the fact that its filter coefficients (taps) are represented by using fixed-point numbers along the whole optimization process. Alternatively, several filter design methodologies perform quantization of the coefficients at the end of the design process, which is why the error may be increased.

As future work, it is intended to implement an adaptive version of the optimization engine for the PBIL tool, as a way for improving the convergence speed. Such adaptive version of the algorithm may take advantage of the Entropy values calculated over the probability array, which are currently being used as stop criterion in the algorithm's iterations. Such Entropy values may serve as an indicator of the convergence status of the algorithm and provide a way to change the LR parameter accordingly.

Also as a future work, it is proposed the implementation of a weighting criterium , which may serve as a way to reduce the absolute error in some zones of the desired response. This improvement is intended to provide the designer with a strategy for increasing the quality of the approximation in some critical bands of the frequency response.

Acknowledgment

The authors would like to thank to the National University of Colombia, for its support in the realization of the present work.

References

[1] J. G. Proakis and D. K. Manolakis, *Digital Signal Processing (4th Edition)*, 4th ed. Prentice Hall, 2006.

[2] A. Ambardar, *Analog and digital signal processing*, ser. PWS Foundations in engineering series. PWS Pub., 1995, no. v. 1. [Online]. Available: http://books.google.com.mx/books?id=JnVGAAAAYAAJ

[3] A. V. Oppenheim and R. W. Schafer, *Digital Signal Processing*. Prentice–Hall, 1999.

[4] E. Hughes, "Optimisation using population based incremental learning (pbil)," in *Optimisation in Control: Methods and Applications (Ref. No. 1998/521), IEE Colloquium on*, 1998, pp. 2/1–2/3.

[5] M. Schmidt, K. Kristensen, and T. Randers Jensen, "Adding genetics to the standard pbil algorithm," in *Evolutionary Computation, 1999. CEC 99. Proceedings of the 1999 Congress on*, vol. 2, 1999, pp. –1534 Vol. 2.

[6] L. jun Fan, B. Li, Z. quan Zhuang, and Z.-Q. Fu, "An approach for dynamic hardware /software partitioning based on dpbil," in *Natural Computation, 2007. ICNC 2007. Third International Conference on*, vol. 5, 2007, pp. 581–585.

[7] F. Bolanos, J. E. Aedo, F. Rivera, and N. Bagherzadeh, "Mapping and scheduling in heterogeneous noc through population-based incremental learning," *Journal of Universal Computer Science (JUCS)*, vol. 18, no. 7, pp. 901 – 916, apr 2012.

[8] F. Bolanos, F. Rivera, J. Aedo, and N. Bagherzadeh, "From {UML} specifications to mapping and scheduling of tasks into a noc, with reliability considerations," *Journal of Systems Architecture*, vol. 59, no. 7, pp. 429 – 440, 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1383762113000696

Error Analysis for Vector-Valued Regularized Least-Squares Algorithm

Yong Liu and Shizhong Liao
School of Computer Science and Technology
Tianjin University, Tianjin 300072, P. R. China
{yongliu,szliao}@tju.edu.cn

Abstract— *Vector-valued regularized least-squares algorithm (RLS) on vector-valued reproducing kernel Hilbert space (RKHS) has recently received increasing interest in various machine learning problems such as multi-task learning and multi-view learning, but error analysis of the vector-valued RLS is still widely unknown. In this paper, we derive an error bound of the vector-valued RLS, which consists of two parts: sample error bound and approximation error bound. We first present the sample error bound through the concentration inequalities of function-valued random variables. Under a suitable assumption of the approximation error, we propose the total error bound with the derived sample error bound. Furthermore, together with a Ysybakov function, we also present an error bound of the multi-class classification problem in terms of the error bound derived for the vector-valued RLS.*

Keywords: Consistency, vector-valued regularized least-squares algorithm, multi-class classification, multi-view learning

1. Introduction

The regularized least-squares algorithm (RLS) on a reproducing kernel Hilbert space (RKHS) of real-valued functions (i.e., when the output space is equal to $\mathbb{R}$) has been extensively studied in the literature [1]–[6]. In [1], a covering number technique is used to obtain the error bounds expressed in terms of suitable complexity measures of the regression function. In [2], the covering techniques are replaced by estimates of integral operators through concentration inequalities of vector-valued random variables. In [3], entropy methods are used to establish the upper bounds. In [4], [6], the eigenvalues of the integral operator are used as a complexity measure for error analysis.

Following the development of multi-task learning and multi-view learning methods, the vector-valued RLS on a vector-valued RKHS (i.e., when the output space is equal to $\mathbb{R}^d$) has recently attracted considerable attention in the machine learning community. A study of vector-valued learning with kernel methods is started in [7] where the vector-valued RKHS is adopted and the representer theorem for Tikhonov regularization is generalized to the vector-valued setting. In [8], [9], they derive conditions which ensure that the operator-valued kernel is universal (which means that on every compact subset of the input space, every continuous function with values in output space can be uniformly approximated by sections of the kernel). Instead of studying operator-valued kernels and their corresponding RKHS from the perspective of extending Aronszajn's pioneering work [10] to the vector or function valued, Kadri et al. [11] target at advancing the understanding of feature spaces associated with operator-valued kernels. In [6], they study the asymptotic performances of the vector-valued RLS for a suitable class of priors and a assumption that the regression function belong to the RKHS.

Although the vector-valued RLS has recently attracted considerable attention, its error analysis is still widely unknown. In this paper, based on the fact that scalar positive defined kernels can be extended to cope with vector-functions using operator-valued positive kernels, we extend the results of error bounds of the scalar RLS to the vector-valued RLS. We first present finite sample error bounds for the vector-valued RLS both in vector-valued RKHS norm and square integrable norm through the concentration inequalities of function-valued random variables. Then, with the derived sample error bounds, we propose total error bounds under a suitable assumption of approximation error. Furthermore, we consider to use the vector-valued RLS for multi-class classification. Together with a Ysybakov function, we apply the error bounds derived for the vector-valued RLS regression to the multi-class classification problem for error analysis.

The rest of the paper is organized as follows. In Section 2 we consider the vector-valued learning and present the setup of the problem, as well as the basic notions behind the theory of vector-valued RKHS. In Section 3 we present the error bounds for vector-valued RLS regression and discuss their consequences. In Section 4 we generalize the above results to multi-class classification problem. We end in Section 5 with conclusion.

2. Preliminaries and Notations

The problem of supervise learning amounts to inferring an unknown functional relation given a finite training set of examples $\boldsymbol{z} = \{(x_i, y_i)\}_{i=1}^n$. More precisely, the training examples are assumed to be identically and independently distributed according to a fixed, but unknown probability

measure $\rho(x,y)$ on $\mathcal{Z}=\mathcal{X}\times\mathcal{Y}$, where usually $\mathcal{Y}\subseteq\mathbb{R}$. Here we are interested in vector-valued learning where $\mathcal{Y}\subseteq\mathbb{R}^d$. A learning algorithm is a map from a training set $\boldsymbol{z}$ to an estimator $f_{\boldsymbol{z}}:\mathcal{X}\to\mathcal{Y}$.

A good estimator should generalize to future examples, this translates into the requirement of having small expected risk

$$\mathcal{E}(f)=\int_{\mathcal{X}\times\mathcal{Y}}\|y-f(x)\|_d^2 d\rho(x,y),$$

where $\|\cdot\|_d$ denotes the euclidean norm in $\mathbb{R}^d$. The minimizer of the expected risk over the space of all the measurable $\mathcal{Y}$-valued functions on $\mathcal{X}$ is the $regression\ function$

$$f_\rho(x)=\int_{\mathcal{Y}} y d\rho(y|x),\quad x\in\mathcal{X},$$

where $\rho(y|x)$ is the conditional distribution at x induced by ρ. Thus the quality of an estimator $f_{\boldsymbol{z}}$ can be assessed by $\|f_{\boldsymbol{z}}-f_\rho\|_\rho$, where

$$\|f_{\boldsymbol{z}}-f_\rho\|_\rho=\left\{\int_{\mathcal{X}}\|f_{\boldsymbol{z}}(x)-f_\rho(x)\|_d^2 d\rho_{\mathcal{X}}(x)\right\}^{1/2},$$

$\rho_{\mathcal{X}}$ is the marginal distribution of ρ on $\mathcal{X}$.

2.1 Vector-Valued RKHS

In the following we will introduce the vector-valued RKHS. You may refer to [7] for further details and references.

Let $\mathcal{Y}^{\mathcal{X}}$ denote the vector space of all functions $f:\mathcal{X}\to\mathcal{Y}$, $\mathcal{L}(\mathcal{Y})$ the Banach space of bounded linear operators on $\mathcal{Y}$. Note that for $\mathcal{Y}\subseteq\mathbb{R}^d$, the space $\mathcal{L}(\mathcal{Y})$ is the space of $d\times d$ matrices. A function

$$K:\mathcal{X}\times\mathcal{X}\to\mathcal{L}(\mathcal{Y})$$

is said to be an $operator\ valued\ positive\ definite\ kernel$ if for each pair $(x,y)\in\mathcal{X}\times\mathcal{X}$, $K(x,y)\in\mathcal{L}(\mathcal{Y})$ is a self-adjoint operator and

$$\sum_{i,j=1}^n\langle y_i,K(x_i,x_j)y_j\rangle_d\geq 0$$

for every finite set of examples $\{(x_i,y_i)\}_{i=1}^n\subset\mathcal{X}\times\mathcal{Y}$.

For each $x\in\mathcal{X}$ and $y\in\mathcal{Y}$, we form a function $K_xy=K(\cdot,x)y\in\mathcal{Y}^{\mathcal{X}}$ defined by

$$(K_xy)(t)=K(t,x)y\ \text{ for all } t\in\mathcal{X}.$$

Similarly to the scalar case, it can be shown that for any given operator valued kernel K, a unique RKHS $\mathcal{H}_K$ can be defined by considering the completion of the space

$$\operatorname{span}\left\{\sum_{i=1}^n K_{x_i}y_i\Big|\ x_i\in\mathcal{X},y_i\in\mathcal{Y}\right\}$$

with respect to the norm $\|\cdot\|_K$ induced by the inner product

$$\langle f,g\rangle_K=\sum_{i,j=1}^n\langle K(x_j,x_i)\beta_i,w_j\rangle_d,$$

for any

$$f,g\in\operatorname{span}\left\{\sum_{i=1}^n K_{x_i}y_i\Big|\ x_i\in\mathcal{X},y_i\in\mathcal{Y}\right\}$$

with $f=\sum_{i=1}^n K(\cdot,x_i)\beta_i$ and $g=\sum_{i=1}^n K(\cdot,x_i)w_i$. By definition, the kernel K has the following reproducing property, for all $y\in\mathcal{Y}$ and $x\in\mathcal{X}$,

$$\langle f(x),y\rangle_d=\langle f,K_xy\rangle_K\ \text{ for all }\ f\in\mathcal{H}_K. \tag{1}$$

Denote $\kappa=\sqrt{\sup_{x\in\mathcal{X}}\|K(x,x)\|}$. Then (1) implies that

$$\|f\|_\infty:=\sup_{x\in\mathcal{X}}\|f(x)\|_d\leq\kappa\|f\|_K \tag{2}$$

for all $f\in\mathcal{H}_K$.

In this paper, we assume that $\kappa<\infty$ and for some $D\geq 0$, $\|y\|_d\leq D$ almost surely, thus $\|f_\rho\|_\rho\leq D$.

2.2 Vector-Valued RLS Algorithm

In this subsection, we will introduce the vector-valued RLS on the vector-valued RKHS. In this framework the hypothesis space $\mathcal{H}_K$ is a given vector-valued RKHS induced by the operator valued positive definite kernel K, and for any $\lambda>0$, the vector-valued RLS estimator $f_{\boldsymbol{z},\lambda}$ is defined as the solution of the minimizing problem

$$f_{\boldsymbol{z},\lambda}=\operatorname*{arg\,min}_{f\in\mathcal{H}_K}\left\{\frac{1}{n}\sum_{i=1}^n\|f(x_i)-y_i\|_d^2+\lambda\|f\|_K^2\right\}. \tag{3}$$

We know from [7], [12] that the solution $f_{\boldsymbol{z},\lambda}$ uniquely exists, and is given by

$$f_{\boldsymbol{z},\lambda}=\left(\frac{1}{n}S_{\boldsymbol{x}}^*S_{\boldsymbol{x}}+\lambda I\right)^{-1}\frac{1}{n}S_{\boldsymbol{x}}^*\boldsymbol{y}, \tag{4}$$

where the operator $S_{\boldsymbol{x}}^*:\mathcal{Y}^n\to\mathcal{H}_K$ is given by

$$S_{\boldsymbol{x}}^*\boldsymbol{y}=S_{\boldsymbol{x}}^*(y_1,\ldots,y_n)=\sum_{i=1}^n K_{x_i}y_i,$$

and the operator $S_{\boldsymbol{x}}^*S_{\boldsymbol{x}}:\mathcal{H}_K\to\mathcal{H}_K$ is given by

$$S_{\boldsymbol{x}}^*S_{\boldsymbol{x}}f=\sum_{i=1}^n K_{x_i}f(x_i).$$

Our goal is to understand how $f_{\boldsymbol{z},\lambda}$ approximates f_ρ and how the decay of the regularization parameter $\lambda=\lambda(n)$ leads to convergence rates. For the scalar RLS, the rates for this approximation in $L^2_{\rho_X}$ ($\|f_{\boldsymbol{z},\lambda}-f_\rho\|_\rho$) have been considered in [1], [13]–[16], and the approximation in the space $\mathcal{H}_K$ ($\|f_{\boldsymbol{z},\lambda}-f_\rho\|_K$) has been shown in [2], [14]. In this paper, we extend the results of the scalar RLS to the vector-valued RLS. Furthermore, we generalize our results of the vector-valued RLS to multi-class classification problem for error analysis.

3. Error Bounds for Vector-Valued RLS Regression

A data-free limit of (3) is

$$f_\lambda = \arg\min_{f\in\mathcal{H}_K} \left\{ \|f - f_\rho\|_\rho^2 + \lambda \|f\|_K^2 \right\}. \quad (5)$$

By [1], we know that the solution of (5) is

$$f_\lambda = (L_K + \lambda I)^{-1} L_K f_\rho, \quad (6)$$

where I is identity operator and $L_K : L^2 \to \mathcal{H}_K$ is an integral operator defined by

$$(L_K f)(t) = \int_{\mathcal{X}} K(t,x) f(x) d\rho_{\mathcal{X}}(x).$$

We will deal with the error $\|f_{\boldsymbol{z},\lambda} - f_\rho\|_K$ by dividing it into two parts $\|f_{\boldsymbol{z},\lambda} - f_\lambda\|_K$ and $\|f_\lambda - f_\rho\|_K$. The first term, $\|f_{\boldsymbol{z},\lambda} - f_\lambda\|_K$, is called the sample error which is made by approximating f_λ through a finite training set $\boldsymbol{z}$. The second term, $\|f_\lambda - f_\rho\|_K$, depends on the choice of $\mathcal{H}_K$ but is independent of sampling, which is called the approximation error.

Theorem 1: Let $\boldsymbol{z} = \{(x_i, y_i)\}_{i=1}^n$ be randomly drawn according to ρ, for all $0 < \delta < 1$, with confidence $1 - \delta$,

$$\|f_{\boldsymbol{z},\lambda} - f_\lambda\|_K \le \frac{6\kappa D \log(2/\delta)}{\sqrt{n}\lambda}. \quad (7)$$

Proof: See in Appendix.A. ■

For the scalar RLS, the error bounds $\|f_{\boldsymbol{z},\lambda} - f_\lambda\|_K$ have been given in [2], [14]. In [14], they show that with confidence $1 - \delta$,

$$\|f_{z,\lambda} - f_\lambda\|_K \le \frac{c_1 \log(4/\delta)}{\sqrt{n}\lambda} \left(30 + \frac{c_2 a}{3\sqrt{n\lambda}} \right),$$

in [2], with confidence $1 - \delta$,

$$\|f_{z,\lambda} - f_\lambda\|_K \le \frac{c_3 \log(2/\delta)}{\sqrt{n}\lambda},$$

where c_1, c_2 and c_3 are some constants.

To the best of our knowledge, the error bound $\|f_{\boldsymbol{z},\lambda} - f_\lambda\|_K$ for the vector-valued RLS on the vector-valued RKHS had never been given before. Our result fills this gap. By theorem 1, we find that the convergence rate of vector-valued RLS is $O(\frac{1}{\sqrt{n}\lambda})$ as the same as that of the scalar RLS in [2], [14].

Using Theorem 1, we will prove our total error estimate in the $\|\cdot\|_K$ norm.

Theorem 2: Let $\boldsymbol{z} = \{(x_i, y_i)\}_{i=1}^n$ be randomly drawn according to ρ, and assume the approximation error $\|f_\lambda - f_\rho\|_K$ satisfies

$$\|f_\lambda - f_\rho\|_K \le c\lambda^\beta,$$

where $c > 0$ and $\beta > 0$. Then, for any $0 < \delta < 1$, with confidence $1 - \delta$,

$$\|f_{\boldsymbol{z},\lambda} - f_\rho\|_K \le 2\log(2/\delta) \left\{ \frac{3\kappa D}{\sqrt{n}\lambda} + c\lambda^\beta \right\}. \quad (8)$$

Setting $\lambda = (2\kappa D)^{\frac{1}{\beta+1}} \left(\frac{1}{n}\right)^{\frac{1}{2(\beta+1)}}$, we have

$$\|f_{\boldsymbol{z},\lambda} - f_\rho\|_K \le 4c \log(2/\delta) (2kD)^{\frac{\beta}{\beta+1}} \left(\frac{1}{n}\right)^{\frac{\beta}{2(\beta+1)}}. \quad (9)$$

Proof: Note that

$$\|f_{\boldsymbol{z},\lambda} - f_\rho\|_K \le \|f_{\boldsymbol{z},\lambda} - f_\lambda\|_K + \|f_\lambda - f_\rho\|_K.$$

By Theorem 1 and the assumption $\|f_\lambda - f_\rho\|_K \le c\lambda^\beta$, with confidence $1 - \delta$,

$$\|f_{\boldsymbol{z},\lambda} - f_\rho\|_K \le 2\log(2/\delta) \frac{3\kappa D}{\sqrt{n}\lambda} + c\lambda^\beta.$$

Since $0 < \delta < 1$, we have $2\log(2/\delta) > 1$. Therefore,

$$\|f_{\boldsymbol{z},\lambda} - f_\rho\|_K \le 2\log(2/\delta) \left\{ \frac{3\kappa D}{\sqrt{n}\lambda} + c\lambda^\beta \right\}.$$

Minimize the $\frac{3\kappa D}{\sqrt{n}\lambda} + c\lambda^\beta$ over $\lambda > 0$, and we obtain

$$\lambda = (2\kappa D)^{\frac{1}{\beta+1}} \left(\frac{1}{n}\right)^{\frac{1}{2(\beta+1)}}.$$

With this choice of λ, we can obtain Theorem 2. ■

Remark 1: If f_ρ is in the range of L_K^r and $\frac{1}{2} < r \le 1$, the approximation error $\|f_\lambda - f_\rho\|_K \le \lambda^{r-\frac{1}{2}} \|L_K^{-1} f_\rho\|_\rho$, which implies that the assumption $\|f_\lambda - f_\rho\|_K \le c\lambda^\beta$ in Theorem 2 is reasonable.

For the scalar RLS, if f_ρ is in the range of L_K, [14] show that

$$\|f_{\boldsymbol{z},\lambda} - f_\rho\|_K \le c_4 \left(\frac{(\log(4/\delta))^2}{n} \right)^{\frac{1}{6}}.$$

In [2], they improve the above result and obtain that

$$\|f_{\boldsymbol{z},\lambda} - f_\rho\|_K \le c_5 \log(2/\delta) \left(\frac{1}{n}\right)^{\frac{1}{6}}.$$

For the vector-valued RLS, if we also assume that f_ρ is in the range of L_K as the same as that of scalar RLS in [2], [14], then

$$\|f_\lambda - f_\rho\|_K \le \lambda^{\frac{1}{2}} \|L_K^{-1} f_\rho\|_\rho.$$

Therefore, the β in theorem 2 is equal $\frac{1}{2}$. In this case, by Theorem 2, we have

$$\|f_{\boldsymbol{z},\lambda} - f_\rho\|_K \le c_6 \log(2/\delta) \left(\frac{1}{n}\right)^{\frac{1}{6}},$$

the convergence rate of the vector-valued RLS is equal to that of the scalar RLS. When we consider the extreme case, that is, $\beta \to \infty$, the convergence rate is $O(\frac{1}{\sqrt{n}})$.

Using Theorem 2 and $\|f_{\boldsymbol{z},\lambda} - f_\rho\|_\rho \le \kappa \|f_{\boldsymbol{z},\lambda} - f_\rho\|_K$, it is easy to obtain the following corollary.

Corollary 1: Let $\boldsymbol{z} = \{(x_i, y_i)\}_{i=1}^n$ be randomly drawn according to ρ, and assume $\|f_\lambda - f_\rho\|_K \le c\lambda^\beta$, where $c > 0$ and $\beta > 0$. Then, for any $0 < \delta < 1$, with confidence $1 - \delta$,

$$\|f_{\boldsymbol{z},\lambda} - f_\rho\|_\rho \le 2\kappa \log(2/\delta) \left\{ \frac{3\kappa D}{\sqrt{n}\lambda} + c\lambda^\beta \right\}. \quad (10)$$

Setting $\lambda = (2\kappa D)^{\frac{1}{\beta+1}} \left(\frac{1}{n}\right)^{\frac{1}{2(\beta+1)}}$, we have

$$\|f_{\boldsymbol{z},\lambda} - f_\rho\|_\rho \leq 4\kappa c \log(2/\delta)(2kD)^{\frac{\beta}{\beta+1}} \left(\frac{1}{n}\right)^{\frac{\beta}{2(\beta+1)}}. \quad (11)$$

In [6], under the assumptions that $\rho \in \mathcal{P}(b,c)$ (see Definition 1 in [6]), $f_\rho \in \mathcal{H}_K$ and the eigenvalues t_n of the integral operator L_K satisfy

$$\alpha \leq n^b t_n \leq \beta,$$

they obtain that

$$\lim_{\tau\to\infty} \limsup_{n\to\infty} \sup_{\rho\in P(b,c)} \mathbb{P}_{\boldsymbol{z}\sim\rho^n} \left[\|f_{\boldsymbol{z},\lambda} - f_\rho\|_\rho > \tau \left(\frac{1}{n}\right)^{\frac{bc}{bc+1}}\right] = 0,$$

where $b < \infty$ and $1 \leq c \leq 2$. The above assumptions may be too strong and therefore may not be satisfied in general cases. In this paper, we only assume that

$$\|f_\lambda - f_\rho\|_K \leq c\lambda^\beta,$$

and if $\beta \geq \frac{2bc}{1-bc}$ and $bc < 1$, by theorem 2, our result yields faster convergence rate. In addition, our proof is much simpler than theirs.

4. Application to Multi-Class Classification

In multi-class classification the examples belong to one of d ($d > 2$) classes. Let $\rho(k|x)$ be the conditional probability for each class, $k = 1, \ldots, d$. A classifier is a function $c : \mathcal{X} \to \{1, 2, \ldots, d\}$, assigning each input point to one of the d classes.

The classification performance can be measured via the misclassification probability

$$R(c) = \mathbb{P}[c(x) \neq y].$$

It is easy to check that the minimizer of the misclassification probability is given by the Bayes rule, defined as

$$b(x) = \arg\max_{k\in\{1,\ldots,d\}} \rho(k|x).$$

In order to use the vector-valued RLS for multi-class classification, we define a coding, that is, a one-to-one map

$$M : \{1, 2, \ldots, d\} \to \mathcal{Y}$$

where $\mathcal{Y} = \{l_1, \ldots, l_d\} \subset \mathbb{R}^d$. In this paper, we define the coding as $l_1 = (1, -1, -1, \ldots, -1)$, $l_2 = (-1, 1, -1, \ldots, -1)$,..., $l_d = (-1, -1, -1, \ldots, 1)$.

We use superscripts to index vector components, so that the squared loss can be written as

$$\|l - f(x)\|_d^2 = \sum_{j=1}^d (l^j - f^j(x))^2.$$

Since the coding is one-to-one, the probability for each coding vector l_k is given by $\rho(k|x)$. The expected risk

$$\mathcal{E}(f) = \int_{\mathcal{X}\times\mathcal{Y}} \|y - f(x)\|_d^2 d\rho(y|x) d\rho_{\mathcal{X}}(x) = \int_{\mathcal{X}} \sum_{k=1}^d \|l_k - f(x)\|_d^2 \rho(k|x) d\rho_{\mathcal{X}}(x),$$

is minimized by the regression function f_ρ, which is expressed as

$$f_\rho(x) = (f_\rho^1(x), \ldots, f_\rho^d(x)) = \int_{\mathcal{Y}} y d\rho(y|x) = \sum_{k=1}^d l_k \rho(k|x).$$

We can write the i-th component of the regression function as

$$f_\rho^i(x) = \sum_{k=1}^d l_k^i \rho(k|x) = \sum_{k=1, k\neq i}^d -\rho(k|x) + \rho(i|x) = \sum_{k=1}^d -\rho(k|x) + \rho(i|x) + \rho(i|x) = 2\rho(i|x) - 1,$$

since $\sum_{k=1}^d \rho(k|x) = 1$. By the definition of the Bayes rule, we have

$$b(x) = \arg\max_{j\in\{1,\ldots,d\}} f_\rho^j(x). \quad (12)$$

The above calculation is simple, but shows us the useful facts: First, the vector-valued RLS algorithm approximating the regression function can be used to learn the Bayes rule for a multi-class problem. Second, once we obtained an estimator for the regression function, Equation (12) shows that the natural way to define a classification rule is to take the argmax of the components of the estimator.

Based on the above idea, the vector-valued RLS estimator $f_{\boldsymbol{z},\lambda}$ for multi-class is defined as the solution of the minimization problem

$$f_{\boldsymbol{z},\lambda} = \arg\min_{f\in\mathcal{H}_K} \left\{\frac{1}{n}\sum_{i=1}^n \|f(x_i) - \bar{l}_i\|_d^2 + \lambda\|f\|_{\mathcal{H}_K}^2\right\}. \quad (13)$$

The classifier is given by

$$c(x) = \arg\max_{i\in\{1,\ldots,d\}} f_{\boldsymbol{z},\lambda}^i(x).$$

Instead of estimating the error

$$\mathbb{E}_\rho(I(c(x) \neq b(x))),$$

where $I(c(x) \neq b(x)) = 1$ if $c(x) \neq b(x)$, $I(c(x) \neq b(x)) = 0$ otherwise, in this paper, for applying the previously derived results derived for vector-valued RLS to multi-class classification problems for error analysis, we consider to estimate

$$\|\mathrm{sgn}(f_{\boldsymbol{z},\lambda}) - \mathrm{sgn}(f_\rho)\|_\rho,$$

where $\mathrm{sgn}(f) = (\mathrm{sgn}(f^1), \ldots, \mathrm{sgn}(f^d))$, $\mathrm{sgn}(f^i(x)) = 1$ if $f^i(x) \geq 0$ and $\mathrm{sgn}(f^i(x)) = -1$ otherwise.

Remark 2: If $\mathrm{sgn}(f_{\boldsymbol{z},\lambda})$ approximates $\mathrm{sgn}(f_\rho)$, it implies that $c(x)$ approximates $b(x)$.

4.1 An Error Bound for Multi-Class Classification

In order to estimate the error bound $\|\mathrm{sgn}(f_{\boldsymbol{z},\lambda}) - \mathrm{sgn}(f_\rho)\|_\rho$, we first denote the $misclassification\ set$ of the classifier $\mathrm{sgn}(f_{\boldsymbol{z},\lambda})$ as

$$\mathcal{X}_{f_{\boldsymbol{z},\lambda}} = \Big\{ x \in \mathcal{X} \big| \exists i \in \{1, 2, \ldots, d\}, \\ \mathrm{sgn}(f^i_{\boldsymbol{z},\lambda})(x) \neq \mathrm{sgn}(f^i_\rho)(x), \Big\}.$$

Note that

$$\|\mathrm{sgn}(f_{\boldsymbol{z},\lambda}) - \mathrm{sgn}(f_\rho)\|^2_\rho = \\ \int_{\mathcal{X} \setminus \mathcal{X}_{f_{\boldsymbol{z},\lambda}}} \|\mathrm{sgn}(f_{\boldsymbol{z},\lambda})(x) - \mathrm{sgn}(f_\rho)(x)\|^2_d d\rho_{\mathcal{X}}(x) + \\ \int_{\mathcal{X}_{f_{\boldsymbol{z},\lambda}}} \|\mathrm{sgn}(f_{\boldsymbol{z},\lambda})(x) - \mathrm{sgn}(f_\rho)(x)\|^2_d d\rho_{\mathcal{X}}(x) = \\ 0 + \int_{\mathcal{X}_{f_{\boldsymbol{z},\lambda}}} \|\mathrm{sgn}(f_{\boldsymbol{z},\lambda})(x) - \mathrm{sgn}(f_\rho)(x)\|^2_d d\rho_{\mathcal{X}}(x).$$

Note that

$$\int_{\mathcal{X}_{f_{\boldsymbol{z},\lambda}}} \|\mathrm{sgn}(f_{\boldsymbol{z},\lambda})(x) - \mathrm{sgn}(f_\rho)(x)\|^2_d d\rho_{\mathcal{X}}(x) \leq \\ 4d \cdot \rho_{\mathcal{X}}(\mathcal{X}_{f_{\boldsymbol{z},\lambda}}),$$

so we have

$$\|\mathrm{sgn}(f_{\boldsymbol{z},\lambda}) - \mathrm{sgn}(f_\rho)\|^2_\rho \leq 4d \cdot \rho_{\mathcal{X}}(\mathcal{X}_{f_{\boldsymbol{z},\lambda}}), \qquad (14)$$

where $\rho_{\mathcal{X}}$ is the marginal distribution of ρ on $\mathcal{X}$.

In the following, we show that $\mathrm{sgn}(f_{\boldsymbol{z},\lambda})$ approximates $\mathrm{sgn}(f_\rho)$ well if $f_{\boldsymbol{z},\lambda}$ is a good approximation of f_ρ. To this end, we introduce a function motivated by the Tsybakov condition [17] with noise exponent q $(0 < q \leq \infty)$: for some constant $c_q > 0$, $\exists i \in \{1, 2, \ldots, d\}$,

$$\rho_{\mathcal{X}} \left(\left\{ x \in \mathcal{X} \big| 0 < |f^i_\rho(x)| \leq c_q t \right\} \right) \leq t^q. \qquad (15)$$

Definition 1: The $Tsybakov\ function$ associated with the probability distribution ρ on $\mathcal{X} \times \mathcal{Y}$ is defined to be the function $S = S_\rho : [0, 1] \to [0, 1]$ given by, $\exists i \in \{1, 2, \ldots, d\}$

$$S(C) = \rho_{\mathcal{X}} \left(\left\{ x \in \mathcal{X} \big| f^i_\rho(x) \in [-C, C], \right\} \right), \qquad (16)$$

Let $0 < q < \infty$, we define the q-coefficient as follows (if it is finite)

$$a_q = a_{q,\rho} = \sup_{0 < C < 1} \frac{S(C)}{C^q}. \qquad (17)$$

By the above definitions, it is easy to verify that for $0 < q < \infty$, the Tsybakov condition (15) holds if and only if $a_q < \infty$ and $S(0) = 0$. We say that ρ has (hard) margin $\tau > 0$ if $S(L) \equiv 0$ when $L \in [0, \tau)$.

Proposition 1: Let $\boldsymbol{z} = \{(x_i, y_i)\}^n_{i=1}$ be randomly drawn according to ρ having q-coefficient $a_q < \infty$ for some $0 < q < \infty$, then

$$\|\mathrm{sgn}(f_{\boldsymbol{z},\lambda}) - \mathrm{sgn}(f_\rho)\|^2_\rho \leq 4d \cdot a_q \kappa^q \|f_{\boldsymbol{z},\lambda} - f_\rho\|^q_K.$$

Proof: By the definition of misclassification set $\mathcal{X}_{f_{\boldsymbol{z},\lambda}}$, $\exists i \in \{1, 2, \ldots, d\}$,

$$\mathcal{X}_{f_{\boldsymbol{z},\lambda}} = \left\{ x \in \mathcal{X} \big| \mathrm{sgn}(f^i_{\boldsymbol{z},\lambda})(x) \neq \mathrm{sgn}(f^i_\rho)(x) \right\},$$

we know that for $x \in \mathcal{X}_{f_{\boldsymbol{z},\lambda}}$, $\exists i \in \{1, 2, \ldots, d\}$ such that

$$\mathrm{sgn}(f^i_{\boldsymbol{z},\lambda})(x) \neq \mathrm{sgn}(f^i_\rho)(x).$$

Therefore, $\exists i \in \{1, 2, \ldots, d\}$,

$$|f^i_\rho(x)| \leq |f^i_{\boldsymbol{z},\lambda}(x) - f^i_\rho(x)| \leq \\ \|f_{\boldsymbol{z},\lambda}(x) - f_\rho(x)\|_d \leq \\ \|f_{\boldsymbol{z},\lambda} - f_\rho\|_\infty.$$

This means that the set $\mathcal{X}_{f_{\boldsymbol{z},\lambda}}$ is a subset of (or equal to)

$$\left\{ x \in \mathcal{X} \big| f^i_\rho \leq \|f_{\boldsymbol{z},\lambda} - f_\rho\|_\infty, \exists i \in \{1, 2, \ldots, d\} \right\}.$$

By the definition of Tsybakov function, we have

$$\rho(\mathcal{X}_{f_{\boldsymbol{z},\lambda}}) \leq S(\|f_{\boldsymbol{z},\lambda} - f_\rho\|_\infty).$$

By (14) and $\|f\|_\infty \leq \kappa \|f\|_K$, we have

$$\|\mathrm{sgn}(f_{\boldsymbol{z},\lambda}) - \mathrm{sgn}(f_\rho)\|_\rho \leq \\ 4d \cdot S(\|f_{\boldsymbol{z},\lambda} - f_\rho\|_\infty) \leq \\ 4d \cdot S(\kappa \|f_{\boldsymbol{z},\lambda} - f_\rho\|_K).$$

According to the definition of q-coefficient, it is easy to verify that

$$S(\kappa \|f_{\boldsymbol{z},\lambda} - f_\rho\|_K) \leq a_q (\kappa \|f_{\boldsymbol{z},\lambda} - f_\rho\|_K)^q.$$

This verifies the desired bound for $\|\mathrm{sgn}(f_{\boldsymbol{z},\lambda}) - \mathrm{sgn}(f_\rho)\|_\rho$. ■

This proposition shows that $\mathrm{sgn}(f_{\boldsymbol{z},\lambda})$ approximates $\mathrm{sgn}(f_\rho)$ well if $f_{\boldsymbol{z},\lambda}$ is a good approximation of f_ρ in $\|\cdot\|_K$. When ρ has hard margin $\tau > 0$, $S(C) = 0$ for $C < \tau$, it is sufficient to consider the case $\|f_{\boldsymbol{z},\lambda} - f_\rho\|_K \geq \frac{\tau}{\kappa}$ in Proposition 1.

Combining Theorem 2 and Proposition 1 yields the following result.

Corollary 2: Let $\boldsymbol{z} = \{(x_i, y_i)\}^n_{i=1}$ be randomly drawn according to ρ having $a_q < \infty$ for some $0 < q < \infty$ and $\|f_{\boldsymbol{z},\lambda} - f_\rho\| \leq c\lambda^\beta$. Setting

$$\lambda = (2\kappa D)^{\frac{1}{\beta+1}} \left(\frac{1}{n} \right)^{\frac{1}{2(\beta+1)}}.$$

Then with confidence $1 - \delta$,

$$\|\mathrm{sgn}(f) - \mathrm{sgn}(f_\rho)\|^2_\rho \leq 4d \cdot a_q \cdot (Q \log(2/\delta))^q \left(\frac{1}{n} \right)^{\frac{q\beta}{2(\beta+1)}},$$

where $Q = 4c\kappa^q (2kD)^{\frac{\beta}{\beta+1}}$.

Remark 3: The theoretical analysises of the multiclass empirical risk minimization methods in multiclass classification have been given in [18]–[20]. In this paper, we use the vector-valued RLS for multiclass classification, and present the specific convergence rate of the error bound (most of the above work only studied the consistency of multiclass classification, but didn't give the specific convergence rate of error bound).

5. Conclusion

The error analysis of the scalar RLS algorithm has been extensively studied in the literature, but little work has focused on the error analysis of the vector-valued RLS. In this paper, we propose the error bounds of the vector-valued RLS for general operator valued kernels. Furthermore, we consider to use the vector-valued RLS for multi-class classification, and derive an error bound for the multi-class classification problem.

Our analysis extensively uses the special properties of the square loss function, henceforth it would be interesting to extend our approach to other loss functions. We think that our results may be improved by taking into account more information about the structure of the hypothesis space.

Acknowledgments

The work is supported in part by the National Natural Science Foundation of China under grant No. 61170019, the Natural Science Foundation of Tianjin under grant No. 11JCYBJC00700.

Appendix.A

Lemma 1 (De Mol et al. [21] Proposition 6): Let H be a Hilbert space and let ξ be a random variable on $(\mathcal{Z},\rho)$ with values in H. Assume $\|\xi\| \leq C$ almost surely. Denote $\sigma^2(\xi) = \mathbb{E}(\|\xi\|^2)$. Let $\{z_i\}_{i=1}^n$ be independent random drawers of ρ. For any $0 < \delta < 1$, with confidence $1-\delta$,

$$\left\| \frac{1}{n}\sum_{i=1}^{m}[\xi_i - \mathbb{E}(\xi_i)] \right\| \leq \frac{2C\log(2/\delta)}{n} + \sqrt{\frac{2\sigma^2(\xi)\log(2/\delta)}{n}}.$$

Proof: [Proof of Theorem 1] By (4) and (6), we have

$$f_{\boldsymbol{z},\lambda} - f_\lambda = \left(\frac{1}{n}S_{\boldsymbol{x}}^* S_{\boldsymbol{x}} + \lambda I\right)^{-1} \times \left\{\frac{1}{n}S_{\boldsymbol{x}}^*\boldsymbol{y} - \frac{1}{n}S_{\boldsymbol{x}}^* S_{\boldsymbol{x}} f_\lambda - \lambda f_\lambda\right\}.$$

Note that

$$\frac{1}{n}S_{\boldsymbol{x}}^*\boldsymbol{y} - \frac{1}{n}S_{\boldsymbol{x}}^* S_{\boldsymbol{x}} f_\lambda = \frac{1}{n}\sum_{i=1}^{n} K_{x_i}(y_i - f_\lambda(x_i)),$$

and by the definition (6) of f_λ, we have

$$\lambda f_\lambda = L_K(f_\rho - f_\lambda).$$

It follows that, for all $\boldsymbol{z} = \{(x_i,y_i)\}_{i=1}^n$, and $\lambda > 0$,

$$f_{\boldsymbol{z},\lambda} - f_\lambda = \left(\frac{1}{n}S_{\boldsymbol{x}}^* S_{\boldsymbol{x}} + \lambda I\right)^{-1} \Lambda.$$

where

$$\Lambda = \frac{1}{n}\sum_{i=1}^{n} K_{x_i}(y_i - f_\lambda(x_i)) - L_K(f_\rho - f_\lambda).$$

Since $S_{\boldsymbol{x}}^* S_{\boldsymbol{x}}$ is positive semidefinite operator, it is easy to see that

$$\|f_{\boldsymbol{z},\lambda} - f_\lambda\|_K \leq \left\| \left(\frac{1}{n}S_{\boldsymbol{x}}^* S_{\boldsymbol{x}} + \lambda I\right)^{-1} \right\| \|\Lambda\|_K \leq \frac{1}{\lambda}\|\Lambda\|_K.$$

Denote random variable $\xi = K_x(y - f_\lambda(x))$ on $(\mathcal{Z},\rho)$ with values in $\mathcal{H}_K$. According to the reproducing property, we have

$$\|\xi\|_K = \|y - f_\lambda(x)\|_d \sqrt{\|K(x,x)\|} \leq \kappa(D + \|f_\lambda\|_\infty),$$

and

$$\sigma^2(\xi) \leq \kappa^2 \int_{\mathcal{Z}} \|f_\lambda(x) - y\|_d^2 d\rho.$$

Note that the definition of the regression function yields

$$\int_{\mathcal{Z}} \|f(x) - y\|_d^2 d\rho - \int_{\mathcal{Z}} \|f_\rho - y\|_d^2 d\rho = \|f - f_\rho\|_\rho^2. \quad (18)$$

Recall the definition (5) of f_λ. Setting $f = 0$ yields

$$\|f_\lambda - f_\rho\|_\rho^2 + \lambda\|f_\lambda\|_K^2 \leq \|f_\rho\|_\rho^2.$$

Hence

$$\|f_\lambda\|_K \leq \|f_\rho\|_\rho/\sqrt{\lambda}$$

and

$$\|f_\lambda - f_\rho\|_\rho^2 \leq \|f_\rho\|_\rho^2 \leq D^2.$$

Recall the Eq.(2), we have

$$\begin{aligned}\|\xi\|_K \leq& \kappa(D + \|f_\lambda\|_\infty) \leq \\ & \kappa(D + \kappa\|f_\lambda\|_K) \leq \\ & \kappa(D + \kappa\|f_\rho\|_\rho/\sqrt{\lambda}) \leq \\ & \kappa D(1 + \kappa/\sqrt{\lambda}).\end{aligned}$$

By (18), we have

$$\begin{aligned}&\int_{\mathcal{Z}} \|f_\rho(x) - y\|_d^2 d\rho = \\ &\int_{\mathcal{Z}} \|f(x) - y\|_d^2 d\rho - \|f - f_\rho\|_\rho^2 \leq \\ &\int_{\mathcal{Z}} \|f(x) - y\|_d^2.\end{aligned}$$

Setting $f = 0$, then

$$\int_{\mathcal{Z}} \|f_\rho(x) - y\|_d^2 d\rho \leq \int_{\mathcal{Z}} \|0 - y\|_d^2 d\rho \leq D^2,$$

thus

$$\int_{\mathcal{Z}} \|f_\lambda(x) - y\|_d^2 d\rho = \int_{\mathcal{Z}} \|f_\rho - y\|_d^2 d\rho + \|f_\lambda - f_\rho\|_\rho^2 \leq 2D^2.$$

Note that

$$\mathbb{E}(\xi) = \int_{\mathcal{X}} K_x \int_d (y - f_\lambda(x)) d\rho(y|x) d\rho_{\mathcal{X}}(x) = L_K(f_\rho - f_\lambda).$$

This means that

$$\Lambda = \frac{1}{n}\sum_{i=1}^n K_{x_i}(y_i - f_\lambda(x_i)) - L_K(f_\rho - f_\lambda) = \frac{1}{n}\sum_{i=1}^n [\xi(z_i) - \mathbb{E}(\xi)].$$

Using the lemma 1, with confidence $1-\delta$, we have

$$\|\Lambda\| \leq \frac{2\kappa(D + \|f_\lambda\|_\infty)\log(2/\delta)}{n} + \kappa\sqrt{\frac{2\int_{\mathcal{Z}} \|f_\lambda(x) - y\|_d^2 d\rho \log(2/\delta)}{n}} \leq \frac{2\kappa D(1 + \kappa/\sqrt{\lambda})\log(2/\delta)}{n} + 2\kappa D\sqrt{\frac{\log(2/\delta)}{n}}.$$

If $\kappa/\sqrt{n\lambda} \leq 1/(3\log(2/\delta))$, the above estimate can be bounded further as

$$\|\Lambda\| \leq \frac{2\kappa D\log(2/\delta)}{n} + \frac{2\kappa D\log(2/\delta)}{\sqrt{n}}\frac{\kappa}{\sqrt{n\lambda}} + \frac{2\kappa D\log(2/\delta)}{\sqrt{n}}\frac{1}{\sqrt{\log(2/\delta)}} \leq \frac{6\kappa D\log(2/\delta)}{\sqrt{n}}.$$

This yields the bound when $\kappa/\sqrt{n\lambda} \leq 1/(3\log(2/\delta))$.

When $\kappa/\sqrt{n\lambda} > 1/(3\log(2/\delta))$, we have

$$\frac{6\kappa D\log(2/\delta)}{\sqrt{n}\lambda} \geq 2D/\sqrt{\lambda}.$$

In this case, we use

$$\|f_\lambda\|_K \leq \|f_\rho\|_\rho/\sqrt{\lambda} \leq D/\sqrt{\lambda},$$

and the trivial bound

$$\|f_{\boldsymbol{z},\lambda}\| \leq D/\sqrt{\lambda}$$

seen from (4) by setting $f = 0$. Then there holds

$$\|f_{\boldsymbol{z},\lambda} - f_\lambda\|_K \leq 2D/\sqrt{\lambda}$$

with probability 1. So the desired inequality also holds in the second case. This proves Theorem 1. ■

References

[1] F. Cucker and S. Smale, "Best choices for regularization parameters in learning theory: on the bias-variance problem," *Foundations of Computational Mathematics*, vol. 2, no. 4, pp. 413–428, 2002.

[2] S. Smale and D.-X. Zhou, "Learning theory estimates via integral operators and their approximations," *Constructive Approximation*, vol. 26, no. 2, pp. 153–172, 2007.

[3] V. Temlyakov, "Approximation in learning theory," *Constructive Approximation*, vol. 27, no. 1, pp. 33–74, 2008.

[4] I. Steinwart, D. Hush, and C. Scovel, "Optimal rates for regularized least squares regression," in *Proceedings of the 22nd Conference on Learning Theory (COLT 2009)*, 2009, pp. 79–93.

[5] Y. Liu, S. Jiang, and S. Liao, "Eigenvalues perturbation of integral operator for kernel selection," in *Proceedings of the 22nd ACM International Conference on Information and Knowledge Management (CIKM 2013)*, 2013, pp. 2189–2198.

[6] A. Caponnetto and E. D. Vito, "Optimal rates for the regularized least-squares algorithm," *Foundations of Computational Mathematics*, vol. 7, no. 3, pp. 331–368, 2006.

[7] C. A. Micchelli and M. Pontil, "On learning vector-valued functions," *Neural Computation*, vol. 17, no. 1, pp. 177–204, 2005.

[8] A. Caponnetto, C. A. Micchelli, M. Pontil, and Y. Ying, "Universal multi-task kernels," *Journal of Machine Learning Research*, vol. 9, pp. 1615–1646, 2008.

[9] C. Carmele, E. D. Vito, and A. Toigo, "Vector valued reproducing kernel Hilbert spaces of integrable functions and Mercer theorem," *Analysis and Applications*, vol. 4, no. 4, pp. 377–408, 2006.

[10] N. Aronszajn, "Theory of reproducing kernels," *Transactions of the American Mathematical Society*, vol. 68, pp. 337–404, 1950.

[11] H. Kadri, A. Rabaoui, P. Preux, E. Duflos, and A. Rakotomamonjy, "Functional regularized least squares classification with operator-valued kernels," in *Proceeding of the 28th International Conference on Machine Learning (ICML 2011)*, 2011, pp. 993–1000.

[12] H. Q. Minh and V. Sindhwani, "Vector-valued manifold regularization," in *Proceeding of the 28th International Conference on Machine Learning (ICML 2011)*, 2011, pp. 57–64.

[13] E. D. Vito, A. Caponnetto, and L. Rosasco, "Model selection for regularized least-squares algorithm in learning theory," *Foundations of Computational Mathematics*, vol. 5, no. 1, pp. 59–85, 2005.

[14] S. Smale and D.-X. Zhou, "Shannon sampling II: Connections to learning theory," *Applied and Computational Harmonic Analysis*, vol. 19, no. 3, pp. 285–302, 2005.

[15] Q. Wu, Y. Ying, and D.-X. Zhou, "Learning rates of least-square regularized regression," *Foundations of Computational Mathematics*, vol. 6, no. 2, pp. 171–192, 2006.

[16] Z. Tong, "Leave-one-out bounds for kernel methods," *Neural Computation*, vol. 15, no. 6, pp. 1397–1437, 2003.

[17] A. B. Tsybakov, "Optimal aggregation of classifiers in statistical learning," *The Annals of Statistics*, vol. 32, pp. 135–166, 2004.

[18] D.-R. Chen and T. Sun, "Consistency of multiclass empirical risk minimization methods based on convex loss," *Journal of Machine Learning Research*, vol. 7, pp. 2435–2447, 2006.

[19] A. Tewari and P. L. Bartlett, "On the consistency of multiclass classification methods," *Journal of Machine Learning Research*, vol. 8, pp. 1007–1025, 2007.

[20] Y. Guermeur, "VC theory of large margin multi-category classifiers," *Journal of Machine Learning Research*, vol. 8, pp. 2551–2594, 2007.

[21] C. D. Mola, E. D. Vito, and L. Rosasco, "Elastic-net regularization in learning theory," *Journal of Complexity*, vol. 25, pp. 201–230, 2009.

A Computationally Generated Ontological Argument Based on Spinoza's *The Ethics*: Part 1

Jack K. Horner
PO Box 266, Los Alamos NM 87544
jhorner@cybermesa.com

ICAI 2014

Abstract

The comments accompanying Proposition (Prop.) 11 ("God ... necessarily exists") in Part I of Spinoza's The Ethics contain sketches of at least three distinct ontological arguments. The first of these is suspiciously short. But worse is true: even the proposition "God exists" (GE), an implication of Prop. 11, cannot be derived from the definitions and axioms of Part I (the "DAPI") of The Ethics; thus, Prop. 11 cannot be derived from the DAPI, either. To show this, I describe an automated first-order model-generator for the DAPI of The Ethics and use it to prove that Prop. 11 is independent of those definitions and axioms. In companion papers, I augment the DAPI with some auxiliary assumptions that I believe Spinoza would accept and that sustain an automated derivation of (GE). The analysis demonstrates how an automated deduction system can augment traditional methods of textual exegesis.

Keywords: automated deduction, textual exegesis, Spinoza

1.0 Introduction

There are two objectives of this paper. The first is to explore some aspects of the validity, that is, whether there is a derivation that follows by inference rules alone from the premises of an argument of interest, of one of Spinoza's arguments for Prop. 11 ([1]). The second is to demonstrate how an automated deduction system can augment traditional methods of textual exegesis. I take no position on whether any proposition in [1] is true.

Ontological arguments are arguments for the claim that

(GE) God exists

based on premises which purport to derive from some source other than observation of the world ([11]), that is, from reason alone. They have a long history in the philosophical literature, extending to at least Anselm ([12]).

The comments accompanying Prop. 11 ("God, or substance, consisting of infinite attributes, of which each expresses eternal and infinite essentiality, necessarily exists") in Part I of Spinoza's *The Ethics* ([1]) contain sketches of at least three distinct ontological arguments. The first argument sketch of Proposition 11 in [1] (p. 51) is remarkably short. If we deny Proposition 11, Spinoza asserts, then we are claiming that God does not exist. But to deny God's existence, he continues, is to deny that God's

essence involves God's existence. That, however, is absurd, he argues, by virtue of Prop. 7 ("Existence belongs to the nature of substances"; [1], p. 48).

Can the promise of this sketch be redeemed within the resources of the definitions and axioms of Part I (the "DAPI") of [1]? Whatever Spinoza may have had in mind, it seems highly likely that Df. VI of Part I of [1], "By God, I mean ... a substance consisting in infinite attributes, of which each expresses eternal and infinite essentiality" ([1], p. 45), would play some role in a more explicit derivation. From Def. VI, at least, we could derive that "God is substance", and from that, together Prop. 7, we could derive (GE). To deny that God exists, therefore, would be, by modus tollens, to deny that God is substance, which would contradict Df. VI, an "absurdity" in the sense Spinoza appears to use that term in the first argument sketch for Prop 11.

If something like this is Spinoza's intent, there is a further problem. Spinoza's definition of God (Df. VI, ""By God I mean a being absolutely infinite -- that is, a substance ...") contains an identification of "a being absolutely infinite" with "a substance ...". This identification, however, does not follow from the meanings of "a being absolutely infinite" and "substance", and thus must be argued. Spinoza does not do so.

We might hope, by careful textual analysis, to flesh out that sketch along the lines Spinoza suggests. But no search for a derivation of Prop. 11 from the DAPI proper, I will argue, can succeed, because (GE), which is an implication of Prop. 11, cannot be derived from the DAPI. If (GE) cannot be derived from the DAPI, then Prop. 11 cannot be derived from the DAPI, either.

2.0 Method

The method used in this paper to investigate the first argument sketch of Prop. 11 of [1] relies heavily on an automated deduction system (ADS, [2]). It is in some ways similar to the method of [3], which is likely to be the first published use of [2] in "conventional" philosophical literature. [2] and its predecessors have been widely used in mathematical logic applications (see, for example, [8], [9]) for at least two decades.

mace4 ([2]) is a software framework that searches for models ([5]) of a set of propositions expressed in a first-order language ([7]). To prove the independence of (GE) from the DAPI, it suffices to show that there is a model that satisfies the DAPI and also satisfies the negation of (GE) ([5]).

The DAPI were rendered in the *mace4* framework and the result used to show the independence of the Prop. 11 from the DAPI. In this rendering, I attempted to adhere rigorously to the text in [1], presuming nothing about the meanings of terms beyond what is explicitly stated in [1].

The script shown in Section 3.0 was executed on a Dell Inspiron 545 with an Intel Core2 Quad CPU Q8200 (clocked @ 2.33 GHz) and 8.00 GB RAM, running under the *Windows Vista Home Premium /Cygwin* operating environment.

3.0 Results

Figure 1 shows the *mace4* script used to show the independence of (GE) from the DAPI.

```
assign(iterate_up_to, 10).
set(verbose).

formulas(theory).
     % DEFINITIONS


     %  Definition of self-caused.
     SelfCaused(x) <-> ( EssenceInvExistence(x) &
                         NatureConcOnlyByExistence(x) )
             # label("Definition I: self-caused").



     %  Definite of "finite after its kind".
     FiniteAfterItsKind(x) <->  ( CanBeLimitedBy(x,y) & SameKind(x,y)  )
                           # label("Definition II: finite after its kind").

    %  Definition of substance.
    Substance(x)  <-> InItself(x) &
                      CanBeConceivedThruItself(x)  # label("Definition III: substance").

    %  Definition of attribute.
    Attribute(x) <-> IntPercAsConstEssSub(x)  # label("Definition IV: attribute").

    %  Definition of mode.
    Mode(x) <->  ( ( Modification(x,y) & Substance(y) ) |
                   ( ExistsIn(x,y) & ConceivedThru(x,y)) )    # label("Definition V: mode").

    % Definition of God.
    God(x) <-> ( Being(x) & AbsolutelyInfinite(x) )  # label("Definition VI: God").

    % Definition of absolutely infinite.
    AbsolutelyInfinite(x) <-> ( Substance(x) & ConstInInfAttributes(x) &
                               ( AttributeOf(y,x) -> ( ExpressesEternalEssentiality(y) &
                                                       ExpressesInfiniteEssentiality(y) ) ) )
                          # label("Definition VI: absolutely infinite").

    % Definition of free.
    Free(x) <-> ( ExistsOnlyByNecessityOfOwnNature(x)  &
                  ( ActionOf(y,x) -> DeterminedBy(y,x) ) )    # label("Definition VII:  free").

    %  Definition of necessary.
    Necessary(x) <-> ( (  ExternalTo(y,x) & DeterminedByFixedMethod(x,y) &
                        DeterminedByDefiniteMethod(x,y) ) &
                        ( IsMethod(Action(y) | IsMethodExistence(y) ) ) )
                                  # label("Definition VII: necessary").

    %  Definition of eternity.
    Eternity(x) <->  ExistConcFollowFromDefEternal(x)  # label("Definition VIII: eternity").



    %  AXIOMS

    %  Axiom I.  Everything which exists, exists either in itself
    %            or in something else.
    Exists(x) <-> ( ExistsIn(x,x) | (ExistsIn(x,y) &  (x != y) ) )
                          # label("Axiom I").
```

```
    %  Axiom II.  That which cannot be conceived through itself must
    %             be conceived through something else.
    -( ConceivedThru(x,x) ) -> (ConceivedThru(x,y) & (x != y) )
                          # label("Axiom II").


    %  Axiom III.  From a given definite cause an effect necessarily
    %              follows; and, on the other hand, if no definite
    %              cause be granted, it is impossible that an effect
    %              can follow.
    DefiniteCause(x)  -> ( EffectNecessarilyFollowsFrom(y,x) &
                         ( -DefiniteCause(x) -> -EffectNecessarilyFollowsFrom(y,x) ) )
              # label("Axiom III").

    %  Axiom IV.  The knowledge of an effect depends on and involves
    %              the knowledge of a cause.
    KnowledgeOf(x,y)  <-> IsACause(x,y)
               # label("Axiom IV: The knowledge of an effect
                 depends on and involves the knowledge of a cause").



    %  Axiom V.  Things which have othing in common cannot be understood,
    %             the one by the means of the other
    %             the one by means of the other; the conception of one
    %             does not involve the conception of the other.
    HaveNothingInCommon(x,y) -> ( (-CanBeUnderstoodInTermsOf(x,y) ) &
                                  (-CanBeUnderstoodInTermsOf(y,x) ) &
                                  ( -ConceptionInvolve(x,y) )  &
                                 (-ConceptionInvolves(y,x) ) )
               # label("Axiom V: Things which have nothing in common cannot
                     be understood, the one by means of the other.").



    %  Axiom VI.  A true idea must correspond with its idea or object.
    TrueIdea(x) -> ( CorrespondWith(x,y) & (  IdeateOf(y,x) | ObjectOf(y,x) ) )
               # label("AxiomVI").


    %  Axiom VII.  If a thing can be conceived as non-existing, its
    %              essence does not involve its existence.
    CanBeConceivedAsNonExisting(x) -> -EssenceInvExistence(x)
               # label("Axiom VII").



%    Negate "God exists".
     -( God(x) -> Exists(x) )
              # label("Negate *God exists*").


end_of_list.
```

Figure 1. *mace4* ([2]) script for generating a model that shows the independence of (GE) from the DAPI. For a detailed description of the *mace4* syntax and semantics, see [2].

```
============================== MODEL ================================

interpretation( 2, [number=1, seconds=0], [

        function(Action(_), [ 0, 0 ]),
        function(IsMethodExistence(_), [ 0, 0 ]),
        function(|(_,_), [
                         0, 0,
                         0, 0 ]),
        relation(AbsolutelyInfinite(_), [ 1, 1 ]),
        relation(Attribute(_), [ 0, 0 ]),
        relation(Being(_), [ 1, 1 ]),
        relation(CanBeConceivedAsNonExisting(_), [ 0, 0 ]),
        relation(CanBeConceivedThruItself(_), [ 1, 1 ]),
        relation(ConstInInfAttributes(_), [ 1, 1 ]),
        relation(DefiniteCause(_), [ 0, 0 ]),
        relation(EssenceInvExistence(_), [ 0, 0 ]),
        relation(Eternity(_), [ 0, 0 ]),
        relation(ExistConcFollowFromDefEternal(_), [ 0, 0 ]),
        relation(Exists(_), [ 0, 0 ]),
        relation(ExistsOnlyByNecessityOfOwnNature(_), [ 0, 0 ]),
        relation(ExpressesEternalEssentiality(_), [ 0, 0 ]),
        relation(ExpressesInfiniteEssentiality(_), [ 0, 0 ]),
        relation(FiniteAfterItsKind(_), [ 0, 0 ]),
        relation(Free(_), [ 0, 0 ]),
        relation(God(_), [ 1, 1 ]),
        relation(InItself(_), [ 1, 1 ]),
        relation(IntPercAsConstEssSub(_), [ 0, 0 ]),
        relation(IsMethod(_), [ 0, 0 ]),
        relation(Mode(_), [ 0, 0 ]),
        relation(NatureConcOnlyByExistence(_), [ 0, 0 ]),
        relation(Necessary(_), [ 0, 0 ]),
        relation(SelfCaused(_), [ 0, 0 ]),
        relation(Substance(_), [ 1, 1 ]),
        relation(TrueIdea(_), [ 0, 0 ]),
        relation(ActionOf(_,_), [
                         0, 0,
                         0, 0 ]),
        relation(AttributeOf(_,_), [
                         0, 0,
                         0, 0 ]),
        relation(CanBeLimitedBy(_,_), [
                         0, 0,
                         0, 0 ]),
        relation(CanBeUnderstoodInTermsOf(_,_), [
                         0, 0,
                         0, 0 ]),
        relation(ConceivedThru(_,_), [
                         1, 0,
                         0, 1 ]),
        relation(ConceptionInvolve(_,_), [
                         0, 0,
                         0, 0 ]),
        relation(ConceptionInvolves(_,_), [
                         0, 0,
                         0, 0 ]),
        relation(CorrespondWith(_,_), [
                         0, 0,
                         0, 0 ]),
        relation(DeterminedBy(_,_), [
                         0, 0,
                         0, 0 ]),
        relation(DeterminedByDefiniteMethod(_,_), [
                         0, 0,
                         0, 0 ]),
        relation(DeterminedByFixedMethod(_,_), [
                         0, 0,
                         0, 0 ]),
        relation(EffectNecessarilyFollowsFrom(_,_), [
                         0, 0,
                         0, 0 ]),
```

```
        relation(ExistsIn(_,_), [
                          0, 0,
                          0, 0 ]),
        relation(ExternalTo(_,_), [
                          0, 0,
                          0, 0 ]),
        relation(HaveNothingInCommon(_,_), [
                          0, 0,
                          0, 0 ]),
        relation(IdeateOf(_,_), [
                          0, 0,
                          0, 0 ]),
        relation(IsACause(_,_), [
                          0, 0,
                          0, 0 ]),
        relation(KnowledgeOf(_,_), [
                          0, 0,
                          0, 0 ]),
        relation(Modification(_,_), [
                          0, 0,
                          0, 0 ]),
        relation(ObjectOf(_,_), [
                          0, 0,
                          0, 0 ]),
        relation(SameKind(_,_), [
                          0, 0,
                          0, 0 ])
]).

============================== end of model ==========================
```

Figure 2. A "domain-size 2" *mace4* model that shows the independence of (GE) from the DAPI. For a detailed description of the *mace4* syntax and semantics, see [2].

4.0 Discussion and conclusions

Sections 2.0 and 3.0 motivate at least two observations:

1. Figure 2 shows that (GE) cannot be derived from the DAPI.

2. As [3] notes, an ADS can be used augment the traditional methods of exegesis of any text. An ADS is not a panacea, of course. It cannot "transform sense-free random noise into divine omniscience" ([10]). Indeed, there is no guarantee that an ADS will produce results of interest in a time we care to wait. That said, an ADS is often astonishingly good at providing insights that may not otherwise be obvious.

3. Companion papers show that the DAPI can be augmented with some plausible assumptions that allow the derivation of (GE).

5.0 Acknowledgements

This work benefited from discussions with Ed Zalta, Paul Oppenheimer, Paul Spade, Tom Oberdan, and Joe Van Zandt. For any infelicities that remain, I am solely responsible.

6.0 References

[1] Spinoza. *The Ethics* (published posthumously, 1677). In Benedict de Spinoza. *On the Improvement of the Understanding, The Ethics, Correspondence.* Unabridged trans. by RHM Elwes (1883). Dover reprint, 1955.

[2] McCune WW. *prover9 and mace4*. February 2009. http://www.cs.unm.edu/~mccune/prover9/.

[3] Oppenheimer P and E Zalta. A computationally-discovered simplification of the ontological argument. *Australasian Journal of Philosophy* 89/2 (2011), 333-349.

[4] Horner JK. *spinoza_ethics*, an automated deduction system for Part I of Spinoza's *The Ethics*. 2013. Available from the author on request.

[5] Chang CC and HJ Keisler. *Model Theory*. North-Holland. 1990.

[6] Kant. *The Critique of Pure Reason.* Trans. by N Kemp Smith. St. Martin's. 1929.

[7] Church A. *Introduction to Mathematical Logic.* Part I. Princeton. 1956.

[8] *Journal of Automated Reasoning.* Springer.

[9] Horner JK. An automated deduction of implication-type-restricted Foulis-Holland theorems from orthomodular quantum logic. In eight parts. *Proceedings of the 2013 International Conference on Artificial Intelligence*. Las Vegas NV, July 2013. CSREA Press. 2013. pp. 374-429.

[10] Clarke AC. *Rendezvous with Rama.* Spectra. 1990.

[11] Oppy G. Ontological Arguments. *Stanford Encyclopedia of Philosophy.* 2011. http://plato.stanford.edu/entries/ontological-arguments/.

[12] Anselm. *Prosologion.* Circa 1050. Trans. by SN Deane. In *St. Anselm: Basic Writings.* Second Edition. Open Court. 1962.

[13] Horn A. On sentences which are true of direct unions of algebras. *Journal of Symbolic Logic* 16 (1951), 14–21.

[14] Hennessy JL and DA Patterson. *Computer Architecture: A Quantitative Approach.* Fourth Edition. Morgan Kaufmann. 2007

A Computationally Generated Ontological Argument Based on Spinoza's *The Ethics*: Part 2

Jack K. Horner
PO Box 266, Los Alamos NM 87544
jhorner@cybermesa.com

ICAI 2014

Abstract

The comments accompanying Proposition (Prop.) 11 ("God ... necessarily exists") in Part I of Spinoza's The Ethics contain sketches of at least three distinct ontological arguments. The first of these is suspiciously short. But worse is true: even the proposition "God exists" (GE), an implication of Prop. 11, cannot be derived from the definitions and axioms of Part I (the "DAPI") of The Ethics; thus, Prop. 11 cannot be derived from the DAPI, either. In a companion paper, I describe an automated first-order model-generator for the DAPI of The Ethics and use it to prove that Prop. 11 is independent of those definitions and axioms. In this paper, I augment the DAPI with some auxiliary assumptions that I believe Spinoza would accept and which sustain an automated derivation of (GE). The analysis demonstrates how an automated deduction system can augment traditional methods of textual exegesis.

Keywords: automated deduction, textual exegesis, Spinoza

1.0 Introduction

There are two objectives of this paper. The first is to explore some aspects of the validity, that is, whether there is a derivation that follows by inference rules alone from the premises of an argument of interest, of one of Spinoza's arguments for Prop. 11 ([1]). The second is to demonstrate how an automated deduction system can augment traditional methods of textual exegesis. I take no position on whether any proposition in [1] is true.

Ontological arguments are arguments for the claim that

(GE) God exists

based on premises which purport to derive from some source other than observation of the world ([11]), that is, from reason alone. They have a long history in the philosophical literature, extending to at least Anselm ([12]).

The comments accompanying Prop. 11 ("God, or substance, consisting of infinite attributes, of which each expresses eternal and infinite essentiality, necessarily exists") in Part I of Spinoza's *The Ethics* ([1]) contain sketches of at least three distinct ontological arguments. The first argument sketch of Proposition 11 in [1] (p. 51) is remarkably short. If we deny Proposition 11, Spinoza asserts, then we are claiming that God does not exist. But to deny God's existence, he continues, is to deny that God's

essence involves God's existence. That, however, is absurd, ([1], p. 48), he argues, by virtue of Prop. 7 ("Existence belongs to the nature of substances").

I argue in a companion paper that no derivation of Prop. 11 from the DAPI can succeed, because (GE), which is an implication of Prop. 11, cannot be derived from the DAPI. Here, I augment the DAPI with some plausible propositions I believe Spinoza would have accepted, and use that combination to derive (GE).

2.0 Method

The method used in this paper to investigate the first argument sketch of Prop. 11 of [1] relies heavily on an automated deduction system (ADS, [2]). It is in some ways similar to the method of [3], which is likely to be the first published use of [2] in "conventional" philosophical literature. [2] and its predecessors have been widely used in mathematical logic applications (see, for example, [8], [9]) for at least two decades.

prover9 ([2]) is a software first-order automated deduction framework that searches for derivations of given propositions, given a set of propositions.

The script shown in Section 3.0 was executed on a Dell Inspiron 545 with an Intel Core2 Quad CPU Q8200 (clocked @ 2.33 GHz) and 8.00 GB RAM, running under the *Windows Vista Home Premium /Cygwin* operating environment.

3.0 Results

Figure 1 shows the *prover9* script used to derive (GE) from the DAPI conjoined with some plausible auxiliary assumptions

```
formulas(assumptions).
     % DEFINITIONS


     %  Definition of self-caused.
     SelfCaused(x) <-> ( EssenceInvExistence(x) &
                         NatureConcOnlyByExistence(x) )
             # label("Definition I: self-caused").



     %  Definite of "finite after its kind".
     FiniteAfterItsKind(x) <->  ( CanBeLimitedBy(x,y) & SameKind(x,y)  )
                           # label("Definition II: finite after its kind").

    %  Definition of substance.
    Substance(x)  <-> InItself(x) &
                      CanBeConceivedThruItself(x)  # label("Definition III: substance").

    %  Definition of attribute.
    Attribute(x) <-> IntPercAsConstEssSub(x)  # label("Definition IV: attribute").

    %  Definition of mode.
    Mode(x) <->  ( ( Modification(x,y) & Substance(y) ) |
                   ( ExistsIn(x,y) & ConceivedThru(x,y)) )    # label("Definition V: mode").
```

```
% Definition of God.
God(x) <-> ( Being(x) & AbsolutelyInfinite(x) )  # label("Definition VI: God").

% Definition of absolutely infinite.
AbsolutelyInfinite(x) <-> ( Substance(x) & ConstInInfAttributes(x) &
                          ( AttributeOf(y,x) -> ( ExpressesEternalEssentiality(y) &
                                                  ExpressesInfiniteEssentiality(y) ) ) )
                     # label("Definition VI: absolutely infinite").

% Definition of free.
Free(x) <-> ( ExistsOnlyByNecessityOfOwnNature(x)  &
              ( ActionOf(y,x) -> DeterminedBy(y,x) ) )    # label("Definition VII:  free").

%  Definition of necessary.
Necessary(x) <-> ( (  ExternalTo(y,x) & DeterminedByFixedMethod(x,y) &
                    DeterminedByDefiniteMethod(x,y) ) &
                    ( IsMethod(Action(y) | IsMethodExistence(y) ) ) )
                              # label("Definition VII: necessary").

%  Definition of eternity.
Eternity(x) <->  ExistConcFollowFromDefEternal(x)  # label("Definition VIII: eternity").



%  AXIOMS

%  Axiom I.  Everything which exists, exists either in itself
%            or in something else.
Exists(x) <-> ( ExistsIn(x,x) | (ExistsIn(x,y) &  (x != y) ) )
                     # label("Axiom I").



%  Axiom II.  That which cannot be conceived through itself must
%             be conceived through something else.
-( ConceivedThru(x,x) ) -> (ConceivedThru(x,y) & (x != y) )
                     # label("Axiom II").


%  Axiom III.  From a given definite cause an effect necessarily
%              follows; and, on the other hand, if no definite
%              cause be granted, it is impossible that an effect
%              can follow.
DefiniteCause(x)  -> ( EffectNecessarilyFollowsFrom(y,x) &
                     ( -DefiniteCause(x) -> -EffectNecessarilyFollowsFrom(y,x) ) )
          # label("Axiom III").

%  Axiom IV.  The knowledge of an effect depends on and involves
%              the knowledge of a cause.
KnowledgeOf(x,y)  <-> IsACause(x,y)
           # label("Axiom IV: The knowledge of an effect
             depends on and involves the knowledge of a cause").


%  Axiom V.  Things which have othing in common cannot be understood,
%             the one by the means of the other
%             the one by means of the other; the conception of one
%             does not involve the conception of the other.
HaveNothingInCommon(x,y) -> ( (-CanBeUnderstoodInTermsOf(x,y) ) &
                              (-CanBeUnderstoodInTermsOf(y,x) ) &
                              ( -ConceptionInvolve(x,y) )  &
                             (-ConceptionInvolves(y,x) ) )
           # label("Axiom V: Things which have nothing in common cannot
                 be understood, the one by means of the other.").



%  Axiom VI.  A true idea must correspond with its idea or object.
TrueIdea(x) -> ( CorrespondWith(x,y) & (  IdeateOf(y,x) | ObjectOf(y,x) ) )
           # label("AxiomVI").
```

```
    %  Axiom VII.  If a thing can be conceived as non-existing, its
    %              essence does not involve its existence.
    CanBeConceivedAsNonExisting(x) -> -EssenceInvExistence(x)
              # label("Axiom VII").


    % AUXILIARY ASSUMPTIONS (JKH added).

    Substance(x) -> Being(x)
           # label("Auxiliary assumption 1: if x is a substance, x is a being").

    (InItself(x) & CanBeConceivedThruItself(x)) -> (Modfication(y,x) -> PriorTo(x,y))
           # label("Auxiliary assumption 2: if x is in itself and can be conceived
               through itself, then if y is a modification of x, x is prior to y").


    KnowledgeOf(x,y) -> CanBeUnderstoodInTermsOf(y,x)
           # label("Auxiliary assumption 3: if x is knowledge of y, then y
                    can be understood in terms of x").



    InItself(x) -> SelfCaused(x)
           # label("Auxiliary assumption 4: if x is in itself, x is self-caused").


    HasInfiniteAttributes(x) ->  EssenceInvExistence(x)
      # label("Auxiliary assumption 5: if x has infinite attributes,
                x has the property that x's essence involves x's existence").

    AbsolutelyInfinite(x) -> HasInfiniteAttributes(x)
      # label("Auxiliary assumption 6: If x is absolutely infinite,
                  then x has infinite attributes").



    Being(x) -> HasEssence(x)
           # label("Auxiliary assumption 7: If x has being, then x has essence").


    (EssenceInvExistence(x) & HasEssence(x)) -> Exists(x)
           # label("Auxiliary assumption 8: if the essence of x involves
               the existence of x and x has essence, then x exists").

end_of_list.



formulas(goals).

%    Propositions XI.  God exists.
     God(x) -> Exists(x)
              # label("Proposition XI. God exists").

end_of_list.
```

Figure 1. *prover9* ([2]) script for generating a derivation of (GE) from the DAPI conjoined with some plausible auxiliary assumptions. For a detailed description of the *prove9* syntax and semantics, see [2].

Figure 2 shows the derivation obtained by executing the script shown in Figure 1.

```
============================== Prover9 ===============================
```

```
Prover9 (32) version 2009-02A, February 2009.
Process 1080 was started by Owner on Owner-PC,
Tue Dec  3 09:49:20 2013
The command was "../bin/prover9".
============================== end of head ===========================

...

============================== PROOF =================================

% Proof 1 at 0.06 (+ 0.01) seconds.
% Length of proof is 27.
% Level of proof is 6.
. . .

1 SelfCaused(x) <-> EssenceInvExistence(x) & NatureConcOnlyByExistence(x) # label("Definition I:
self-caused") # label(non_clause).  [assumption].
3 Substance(x) <-> InItself(x) & CanBeConceivedThruItself(x) # label("Definition III: substance")
# label(non_clause).  [assumption].
6 God(x) <-> Being(x) & AbsolutelyInfinite(x) # label("Definition VI: God") # label(non_clause).
[assumption].
7 AbsolutelyInfinite(x) <-> Substance(x) & ConstInInfAttributes(x) & (AttributeOf(y,x) ->
ExpressesEternalEssentiality(y) & ExpressesInfiniteEssentiality(y)) # label("Definition VI:
absolutely infinite") # label(non_clause).  [assumption].
21 InItself(x) -> SelfCaused(x) # label("Auxiliary assumption 4: if x is in itself, x is self-
caused") # label(non_clause).  [assumption].
24 Being(x) -> HasEssence(x) # label("Auxiliary assumption 7: If x has being, then x has
essence") # label(non_clause).  [assumption].
25 EssenceInvExistence(x) & HasEssence(x) -> Exists(x) # label("Auxiliary assumption 8: if the
essence of x involves the existence of x and x has essence, then x exists") # label(non_clause).
[assumption].
26 God(x) -> Exists(x) # label("Proposition XI. God exists") # label(non_clause) # label(goal).
[goal].
28 -SelfCaused(x) | EssenceInvExistence(x) # label("Definition I: self-caused").  [clausify(1)].
30 -InItself(x) | SelfCaused(x) # label("Auxiliary assumption 4: if x is in itself, x is self-
caused").  [clausify(21)].
35 -Substance(x) | InItself(x) # label("Definition III: substance").  [clausify(3)].
40 -AbsolutelyInfinite(x) | Substance(x) # label("Definition VI: absolutely infinite").
[clausify(7)].
64 -God(x) | Being(x) # label("Definition VI: God").  [clausify(6)].
65 -God(x) | AbsolutelyInfinite(x) # label("Definition VI: God").  [clausify(6)].
66 God(c1) # label("Proposition XI. God exists").  [deny(26)].
72 -AbsolutelyInfinite(x) | InItself(x).  [resolve(40,b,35,a)].
80 AbsolutelyInfinite(c1).  [resolve(66,a,65,a)].
97 -InItself(x) | EssenceInvExistence(x).  [resolve(30,b,28,a)].
113 InItself(c1).  [resolve(80,a,72,a)].
127 -Being(x) | HasEssence(x) # label("Auxiliary assumption 7: If x has being, then x has
essence").  [clausify(24)].
129 Being(c1).  [resolve(66,a,64,a)].
136 -EssenceInvExistence(x) | -HasEssence(x) | Exists(x) # label("Auxiliary assumption 8: if the
essence of x involves the existence of x and x has essence, then x exists").  [clausify(25)].
138 EssenceInvExistence(c1).  [resolve(113,a,97,a)].
153 -Exists(c1) # label("Proposition XI. God exists").  [deny(26)].
188 HasEssence(c1).  [resolve(129,a,127,a)].
191 -HasEssence(c1) | Exists(c1).  [resolve(138,a,136,a)].
192 $F.  [copy(191),unit_del(a,188),unit_del(b,153)].

============================== end of proof ==========================
```

Figure 2. *prover9* derivation of (GE) from the script shown in Figure 3. For further detail on the syntax and semantics of *prover9*, see [2].

The derivation shown in Figure 2 uses only three of the auxiliary assumptions of the script shown in Figure 1 (the other auxiliary assumptions are used in the derivation of

propositions not essential to the purposes of this paper):

- Auxiliary assumption 4: If x is in itself, x is self-caused
- Auxiliary assumption 7: If x is a being, then x has essence
- Auxiliary assumption 8: If the essence of x involves the existence of x and x has essence, then x exists.

It is difficult to imagine any serious objection to any of these assumptions within the context of Spinoza's philosophy (in that context, they are arguably definitional), and I believe Spinoza would have accepted them.

4.0 Discussion and conclusions

Sections 2.0 and 3.0 motivate at least two observations:

1. A companion paper shows that (GE) cannot be derived from the DAPI. Figure 2 shows that (GE) can be derived from the DAPI conjoined with a few plausible assumptions.

2. As [3] notes, an ADS can be a useful tool for augmentation of traditional methods of exegis. An ADS is not a panacea, of course. Indeed, there is no guarantee that an ADS will produce results of interest in a time we care to wait. That said, an ADS is often astonishingly good at providing insights that may not otherwise be obvious.

5.0 Acknowledgements

This work benefited from discussions with Ed Zalta, Paul Oppenheimer, Paul Spade, Tom Oberdan, and Joe Van Zandt. For any infelicities that remain, I am solely responsible.

6.0 References

[1] Spinoza. *The Ethics* (published posthumously, 1677). In Benedict de Spinoza. *On the Improvement of the Understanding, The Ethics, Correspondence.* Unabridged trans. by RHM Elwes (1883). Dover reprint, 1955.

[2] McCune WW. *prover9 and mace4*. February 2009. http://www.cs.unm.edu/~mccune/prover9/.

[3] Oppenheimer P and E Zalta. A computationally-discovered simplification of the ontological argument. *Australasian Journal of Philosophy* 89/2 (2011), 333-349.

[4] Horner JK. *spinoza_ethics*, an automated deduction system for Part I of Spinoza's *The Ethics*. 2013. Available from the author on request.

[5] Chang CC and HJ Keisler. *Model Theory.* North-Holland. 1990.

[6] Kant. *The Critique of Pure Reason.* Trans. by N Kemp Smith. St. Martin's. 1929.

[7] Church A. *Introduction to Mathematical Logic*. Part I. Princeton. 1956.

[8] *Journal of Automated Reasoning.* Springer.

[9] Horner JK. An automated deduction of implication-type-restricted Foulis-Holland theorems from orthomodular quantum logic. In eight parts. *Proceedings of the 2013 International Conference on Artificial Intelligence*. Las Vegas NV, July 2013. CSREA Press. 2013. pp. 374-429.

[10] Clarke AC. *Rendezvous with Rama.* Spectra. 1990.

[11] Oppy G. Ontological Arguments. *Stanford Encyclopedia of Philosophy.* 2011. http://plato.stanford.edu/entries/ontological-arguments/.

[12] Anselm. *Prosologion.* Circa 1050. Trans. by SN Deane. In *St. Anselm: Basic Writings.* Second Edition. Open Court. 1962.

[13] Horn A. On sentences which are true of direct unions of algebras. *Journal of Symbolic Logic* 16 (1951), 14–21.

[14] Hennessy JL and DA Patterson. *Computer Architecture: A Quantitative Approach.* Fourth Edition. Morgan Kaufmann. 2007

A Computationally Generated Ontological Argument Based on Spinoza's *The Ethics*: Part 3

Jack K. Horner
PO Box 266, Los Alamos NM 87544
jhorner@cybermesa.com

ICAI 2014

Abstract

The proposition "If x is substance, x exists" (SE), an implication of Proposition (Prop.) 7 of Part I of Spinoza's The Ethics *, is assumed by one of the ontological arguments in that text. However, (SE) cannot be derived from the definitions and axioms of Part I (the "DAPI") of* The Ethics*; thus, Prop. 7 cannot be derived from the DAPI, either. To show this, I describe an automated first-order model-generator for the DAPI and use it to prove that Prop. 7 is independent of the DAPI. In companion papers, I augment the DAPI with some auxiliary assumptions that I believe Spinoza would accept and that sustain an automated derivation of (SE). The analysis demonstrates how an automated deduction system can augment traditional methods of textual exegesis.*

Keywords: automated deduction, textual exegesis, Spinoza

1.0 Introduction

There are two objectives of this paper. The first is to explore some aspects of the validity, that is, whether there is a derivation that follows by inference rules alone from the premises of an argument of interest, of one of Spinoza's arguments for Prop. 7 ([1]). The second is to demonstrate how an automated deduction system can augment traditional methods of textual exegesis. I take no position on whether any proposition in [1] is true.

The proposition

(SE) If x is substance, x exists

is an implication of Prop. 7 ("Existence is involved in the nature of substances", [1], p. 48) of Part I of Spinoza's *The Ethics*. Prop. 7 is of interest in its own right, and is also assumed by one of the ontological arguments ([11], [12]) for Prop. 11 ("God … exists") of [1].

The comments accompanying Prop. 7 in Part I of *The Ethics* ([1]) are intended as an argument for Prop. 7. Substance, Spinoza asserts there, cannot be produced by anything external. It must, therefore, be its own cause. And that implies, Spinoza concludes, that the essence of substance involves its existence – that is to say, existence belongs to the nature of substance.

Taken at face, this sketch is too brief to be convincing. Can its promise be redeemed within the resources of the definitions and axioms of Part I (the "DAPI") of [1]? We might hope, by careful textual analysis, to flesh out the sketch along the lines Spinoza suggests. But no search for a derivation of Prop. 7 within the DAPI proper, I will argue, can succeed, because (SE) cannot be derived from the DAPI. If (SE) cannot be derived from the DAPI, then Prop. 7 cannot be derived from the DAPI, either.

2.0 Method

The method used in this paper to investigate the argument sketch of Prop. 7 of [1] relies heavily on an automated deduction system (ADS, [2]). It is in some ways similar to the method of [3], which is likely to be the first published use of [2] in "conventional" philosophy. [2] and its predecessors have been widely used in mathematical logic applications (see, for example, [8], [9]) for at least two decades.

mace4 ([2]) is a software framework that searches for models ([5]) of a set of propositions expressed in a first-order language ([7]). To prove the independence of (SE) from the DAPI, it suffices to show that there is a model that satisfies the DAPI and also satisfies the negation of (SE) ([5]).

The DAPI were first rendered in the *mace4* framework and the result used to show the independence of the (SE) from the DAPI. In this rendering, I attempted to adhere rigorously to the text in [1], presuming nothing about the meanings of terms beyond what is explicitly stated in [1].

The script shown in Section 3.0 was executed on a Dell Inspiron 545 with an Intel Core2 Quad CPU Q8200 (clocked @ 2.33 GHz) and 8.00 GB RAM, running under the *Windows Vista Home Premium /Cygwin* operating environment.

3.0 Results

Figure 1 shows the *mace4* script used to show the independence of (SE) from the DAPI.

```
assign(iterate_up_to, 10).
set(verbose).

formulas(theory).

     % DEFINITIONS


     %  Definition of self-caused.
     SelfCaused(x) <-> ( EssenceInvExistence(x) &
                         NatureConcOnlyByExistence(x) )
             # label("Definition I: self-caused").



     %  Definite of "finite after its kind".
     FiniteAfterItsKind(x) <->  ( CanBeLimitedBy(x,y) & SameKind(x,y)  )
                           # label("Definition II: finite after its kind").

    %  Definition of substance.
    Substance(x)  <-> InItself(x) &
                      CanBeConceivedThruItself(x)  # label("Definition III: substance").
```

```
%  Definition of attribute.
Attribute(x) <-> IntPercAsConstEssSub(x)  # label("Definition IV: attribute").

%  Definition of mode.
Mode(x) <->  ( ( Modification(x,y) & Substance(y) ) |
               ( ExistsIn(x,y) & ConceivedThru(x,y)) )    # label("Definition V: mode").

% Definition of God.
God(x) <-> ( Being(x) & AbsolutelyInfinite(x) )  # label("Definition VI: God").

% Definition of absolutely infinite.
AbsolutelyInfinite(x) <-> ( Substance(x) & ConstInInfAttributes(x) &
                           ( AttributeOf(y,x) -> ( ExpressesEternalEssentiality(y) &
                                                   ExpressesInfiniteEssentiality(y) ) ) )
                      # label("Definition VI: absolutely infinite").

% Definition of free.
Free(x) <-> ( ExistsOnlyByNecessityOfOwnNature(x)  &
              ( ActionOf(y,x) -> DeterminedBy(y,x) ) )    # label("Definition VII:  free").

%  Definition of necessary.
Necessary(x) <-> ( (  ExternalTo(y,x) & DeterminedByFixedMethod(x,y) &
                    DeterminedByDefiniteMethod(x,y) ) &
                    ( IsMethod(Action(y) | IsMethodExistence(y) ) ) )
                              # label("Definition VII: necessary").

%  Definition of eternity.
Eternity(x) <->  ExistConcFollowFromDefEternal(x)  # label("Definition VIII: eternity").



%  AXIOMS

%  Axiom I.  Everything which exists, exists either in itself
%            or in something else.
Exists(x) <-> ( ExistsIn(x,x) | (ExistsIn(x,y) &  (x != y) ) )
                      # label("Axiom I").



%  Axiom II.  That which cannot be conceived through itself must
%             be conceived through something else.
-( ConceivedThru(x,x) ) -> (ConceivedThru(x,y) & (x != y) )
                      # label("Axiom II").


%  Axiom III.  From a given definite cause an effect necessarily
%              follows; and, on the other hand, if no definite
%              cause be granted, it is impossible that an effect
%              can follow.
DefiniteCause(x)  -> ( EffectNecessarilyFollowsFrom(y,x) &
                     ( -DefiniteCause(x) -> -EffectNecessarilyFollowsFrom(y,x) ) )
          # label("Axiom III").

%  Axiom IV.  The knowledge of an effect depends on and involves
%              the knowledge of a cause.
KnowledgeOf(x,y)  <-> IsACause(x,y)
           # label("Axiom IV: The knowledge of an effect depends on and
                involves the knowledge of a cause").



%  Axiom V.  Things which have othing in common cannot be understood,
%             the one by the means of the other
%             the one by means of the other; the conception of one
%             does not involve the conception of the other.
HaveNothingInCommon(x,y) -> ( (-CanBeUnderstoodInTermsOf(x,y) ) &
                              (-CanBeUnderstoodInTermsOf(y,x) ) &
                              ( -ConceptionInvolve(x,y) )  &
```

```
                              (-ConceptionInvolves(y,x) ) )
                # label("Axiom V: Things which have nothing in common cannot be understood,
                       the one by means of the other.").


    %  Axiom VI.  A true idea must correspond with its idea or object.
    TrueIdea(x) -> ( CorrespondWith(x,y) & (  IdeateOf(y,x) | ObjectOf(y,x) ) )
                # label("AxiomVI").


    %  Axiom VII.  If a thing can be conceived as non-existing, its
    %              essence does not involve its existence.
    CanBeConceivedAsNonExisting(x) -> -EssenceInvExistence(x)
                # label("Axiom VII").




%    Proposition VII.  Negate  "substance -> existence".
     -( Substance(x) -> Exists(x) )
              # label("Proposition XI. Negate substance implices exists").

end_of_list.
```

Figure 1. A *mace4* ([2]) script for generating a model that shows the independence of the (SE) from the DAPI. For a detailed description of the *mace4* syntax and semantics, see [2].

Figure 2 depicts a model, produced by the script shown in Figure 1, that shows the independence of (SE) from the DAPI.

```
============================== Mace4 =================================
Mace4 (32) version 2009-02A, February 2009.
Process 2160 was started by User on User-PC,
Tue Dec 10 05:20:54 2013
The command was "../bin/mace4".
============================== end of head ===========================


...

============================== MODEL =================================

interpretation( 2, [number=1, seconds=0], [

        function(Action(_), [ 0, 0 ]),
        function(IsMethodExistence(_), [ 0, 0 ]),

        function(|(_,_), [
                         0, 0,
                         0, 0 ]),
        relation(AbsolutelyInfinite(_), [ 0, 0 ]),
        relation(Attribute(_), [ 0, 0 ]),
        relation(Being(_), [ 0, 0 ]),
        relation(CanBeConceivedAsNonExisting(_), [ 0, 0 ]),
        relation(CanBeConceivedThruItself(_), [ 1, 1 ]),
        relation(ConstInInfAttributes(_), [ 0, 0 ]),
        relation(DefiniteCause(_), [ 0, 0 ]),
        relation(EssenceInvExistence(_), [ 0, 0 ]),
        relation(Eternity(_), [ 0, 0 ]),
        relation(ExistConcFollowFromDefEternal(_), [ 0, 0 ]),
```

```
relation(Exists(_), [ 0, 0 ]),
relation(ExistsOnlyByNecessityOfOwnNature(_), [ 0, 0 ]),
relation(ExpressesEternalEssentiality(_), [ 0, 0 ]),
relation(ExpressesInfiniteEssentiality(_), [ 0, 0 ]),
relation(FiniteAfterItsKind(_), [ 0, 0 ]),
relation(Free(_), [ 0, 0 ]),
relation(God(_), [ 0, 0 ]),
relation(InItself(_), [ 1, 1 ]),
relation(IntPercAsConstEssSub(_), [ 0, 0 ]),
relation(IsMethod(_), [ 0, 0 ]),
relation(Mode(_), [ 0, 0 ]),
relation(NatureConcOnlyByExistence(_), [ 0, 0 ]),
relation(Necessary(_), [ 0, 0 ]),
relation(SelfCaused(_), [ 0, 0 ]),
relation(Substance(_), [ 1, 1 ]),
relation(TrueIdea(_), [ 0, 0 ]),
relation(ActionOf(_,_), [
                  0, 0,
                  0, 0 ]),
relation(AttributeOf(_,_), [
                  0, 0,
                  0, 0 ]),
relation(CanBeLimitedBy(_,_), [
                  0, 0,
                  0, 0 ]),
relation(CanBeUnderstoodInTermsOf(_,_), [
                  0, 0,
                  0, 0 ]),
relation(ConceivedThru(_,_), [
                  1, 0,
                  0, 1 ]),
relation(ConceptionInvolve(_,_), [
                  0, 0,
                  0, 0 ]),
relation(ConceptionInvolves(_,_), [
                  0, 0,
                  0, 0 ]),
relation(CorrespondWith(_,_), [
                  0, 0,
                  0, 0 ]),
relation(DeterminedBy(_,_), [
                  0, 0,
                  0, 0 ]),
relation(DeterminedByDefiniteMethod(_,_), [
                  0, 0,
                  0, 0 ]),
relation(DeterminedByFixedMethod(_,_), [
                  0, 0,
                  0, 0 ]),
relation(EffectNecessarilyFollowsFrom(_,_), [
                  0, 0,
                  0, 0 ]),
relation(ExistsIn(_,_), [
                  0, 0,
                  0, 0 ]),
relation(ExternalTo(_,_), [
                  0, 0,
                  0, 0 ]),

relation(HaveNothingInCommon(_,_), [
                  0, 0,
                  0, 0 ]),
relation(IdeateOf(_,_), [
                  0, 0,
                  0, 0 ]),
relation(IsACause(_,_), [
                  0, 0,
                  0, 0 ]),
relation(KnowledgeOf(_,_), [
                  0, 0,
                  0, 0 ]),
```

```
        relation(Modification(_,_), [
                          0, 0,
                          0, 0 ]),
        relation(ObjectOf(_,_), [
                          0, 0,
                          0, 0 ]),
        relation(SameKind(_,_), [
                          0, 0,
                          0, 0 ])
]).

============================== end of model ==========================
```

Figure 2. A "domain-size 2" *mace4* model that shows the independence of (SE) from the DAPI. For a detailed description of the *mace4* syntax and semantics, see [2].

4.0 Discussion and conclusions

Sections 2.0 and 3.0 motivate at least two observations:

1. Figures 1 and 2 show that (SE) cannot be derived from the DAPI.

2. As [3] notes, an ADS can be a useful augmentation of the traditional methods of textual exegesis. An ADS is not a panacea, of course. Indeed, there is no guarantee that an ADS will produce results of interest in a time we care to wait. That said, an ADS is often astonishingly good at providing insights that may not otherwise be obvious.

3. A companion paper shows that the DAPI can be augmented with some plausible assumptions that allow the derivation of (SE).

5.0 Acknowledgements

This work benefited from discussions with Ed Zalta, Paul Oppenheimer, Paul Spade, Tom Oberdan, and Joe Van Zandt. For any infelicities that remain, I am solely responsible.

7.0 References

[1] Spinoza. *The Ethics* (published posthumously, 1677). In Benedict de Spinoza. *On the Improvement of the Understanding, The Ethics, Correspondence.* Unabridged trans. by RHM Elwes (1883). Dover reprint, 1955.

[2] McCune WW. *prover9 and mace4*. February 2009. http://www.cs.unm.edu/~mccune/prover9/.

[3] Oppenheimer P and E Zalta. A computationally-discovered simplification of the ontological argument. *Australasian Journal of Philosophy* 89/2 (2011), 333-349.

[4] Horner JK. *spinoza_ethics*, an automated deduction system for Part I of Spinoza's *The Ethics*. 2013. Available from the author on request.

[5] Chang CC and HJ Keisler. *Model Theory.* North-Holland. 1990.

[6] Kant. *The Critique of Pure Reason.* Trans. by N Kemp Smith. St. Martin's. 1929.

[7] Church A. *Introduction to Mathematical Logic.* Part I. Princeton. 1956.

[8] *Journal of Automated Reasoning.* Springer.

[9] Horner JK. An automated deduction of implication-type-restricted Foulis-Holland theorems from orthomodular quantum logic. In eight parts. *Proceedings of the 2013 International Conference on Artificial Intelligence*. Las Vegas NV, July 2013. CSREA Press. 2013. pp. 374-429.

[11] Oppy G. Ontological Arguments. *Stanford Encyclopedia of Philosophy.* 2011. http://plato.stanford.edu/entries/ontological-arguments/.

[12] Anselm. *Prosologion.* Circa 1050. Trans. by SN Deane. In *St. Anselm: Basic Writings.* Second Edition. Open Court. 1962.

[13] Horn A. On sentences which are true of direct unions of algebras. *Journal of Symbolic Logic* 16 (1951), 14–21.

[14] Hennessy JL and DA Patterson. *Computer Architecture: A Quantitative Approach.* Fourth Edition. Morgan Kaufmann. 2007

A Computationally Generated Ontological Argument Based on Spinoza's *The Ethics*: Part 4

Jack K. Horner
PO Box 266, Los Alamos NM 87544
jhorner@cybermesa.com

ICAI 2014

Abstract

The proposition "If x is substance, x exists" (SE), an implication of Proposition (Prop.) 7 of Part I of Spinoza's The Ethics *, is assumed by one of the ontological arguments in that text. In a companion paper, I show (SE) cannot be derived from the definitions and axioms of Part I (the "DAPI") of* The Ethics*; thus, Prop. 7 cannot be derived from the DAPI, either. In this paper, I augment the DAPI with some auxiliary assumptions that I believe Spinoza would accept and that sustain an automated derivation of (SE). The analysis demonstrates how an automated deduction system can augment traditional methods of textual exegesis.*

Keywords: automated deduction, textual exegesis, Spinoza

1.0 Introduction

There are two objectives of this paper. The first is to explore some aspects of the validity, that is, whether there is a derivation that follows by inference rules alone from the premises of an argument of interest, of one of Spinoza's arguments for Prop. 7 ([1]). The second is to demonstrate how an automated deduction system can augment traditional methods of textual exegesis. I take no position on whether any proposition in [1] is true.

The proposition

(SE) If x is substance, x exists

is an implication of Prop. 7 ("Existence is involved in the nature of substances", [1], p. 48) of Part I of Spinoza's *The Ethics*. Prop. 7 is of interest in its own right, and is also assumed by one of the ontological arguments ([11], [12]) for Prop. 11 ("God … exists") of [1].

In a companion paper, I show that no search for a derivation of Prop. 7 within the DAPI proper, can succeed, because (SE) cannot be derived from the DAPI. If (SE) cannot be derived from the DAPI, then Prop. 7 cannot be derived from the DAPI, either.

Here, I augment the DAPI with some plausible propositions I believe Spinoza would have accepted, and use that combination to derive (SE).

2.0 Method

The method used in this paper to investigate the argument sketch of Prop. 7 of [1] relies heavily on an automated deduction system (ADS, [2]). It is in some ways similar to the method of [3], which is likely to be the first published use of [2] in "conventional" philsophy. [2] and its predecessors have been widely used in mathematical logic applications (see, for example, [8], [9]) for at least two decades.

prover9 ([2]) is a software first-order automated deduction framework that searches for derivations of given propositions, given a set of propositions.

The script shown in Section 3.0 was executed on a Dell Inspiron 545 with an Intel Core2 Quad CPU Q8200 (clocked @ 2.33 GHz) and 8.00 GB RAM, running under the *Windows Vista Home Premium /Cygwin* operating environment.

3.0 Results

Figure 1 shows the *prover9* script used to show that (SE) can be derived from the DAPI conjoined with a few plausible assumptions.

```
formulas(assumptions).

     % DEFINITIONS


     %  Definition of self-caused.
     SelfCaused(x) <-> ( EssenceInvExistence(x) &
                         NatureConcOnlyByExistence(x) )
             # label("Definition I: self-caused").



     %  Definite of "finite after its kind".
     FiniteAfterItsKind(x) <->  ( CanBeLimitedBy(x,y) & SameKind(x,y)  )
                           # label("Definition II: finite after its kind").

    %  Definition of substance.
    Substance(x)  <-> InItself(x) &
                      CanBeConceivedThruItself(x)  # label("Definition III: substance").

    %  Definition of attribute.
    Attribute(x) <-> IntPercAsConstEssSub(x)  # label("Definition IV: attribute").

    %  Definition of mode.
    Mode(x) <->  ( ( Modification(x,y) & Substance(y) ) |
                   ( ExistsIn(x,y) & ConceivedThru(x,y)) )    # label("Definition V: mode").

    % Definition of God.
    God(x) <-> ( Being(x) & AbsolutelyInfinite(x) )  # label("Definition VI: God").

    % Definition of absolutely infinite.
    AbsolutelyInfinite(x) <-> ( Substance(x) & ConstInInfAttributes(x) &
                                ( AttributeOf(y,x) -> ( ExpressesEternalEssentiality(y) &
                                                        ExpressesInfiniteEssentiality(y) ) ) )
                          # label("Definition VI: absolutely infinite").

    % Definition of free.
    Free(x) <-> ( ExistsOnlyByNecessityOfOwnNature(x)  &
                  ( ActionOf(y,x) -> DeterminedBy(y,x) ) )    # label("Definition VII:  free").
```

```
    %  Definition of necessary.
    Necessary(x) <-> ( (  ExternalTo(y,x) & DeterminedByFixedMethod(x,y) &
                       DeterminedByDefiniteMethod(x,y) ) &
                       ( IsMethod(Action(y) | IsMethodExistence(y) ) ) )
                                 # label("Definition VII: necessary").

    %  Definition of eternity.
    Eternity(x) <->  ExistConcFollowFromDefEternal(x)  # label("Definition VIII: eternity").



    %  AXIOMS

    %  Axiom I.  Everything which exists, exists either in itself
    %            or in something else.
    Exists(x) <-> ( ExistsIn(x,x) | (ExistsIn(x,y) &  (x != y) ) )
                          # label("Axiom I").



    %  Axiom II.  That which cannot be conceived through itself must
    %             be conceived through something else.
    -( ConceivedThru(x,x) ) -> (ConceivedThru(x,y) & (x != y) )
                          # label("Axiom II").


    %  Axiom III.  From a given definite cause an effect necessarily
    %              follows; and, on the other hand, if no definite
    %              cause be granted, it is impossible that an effect
    %              can follow.
    DefiniteCause(x)  -> ( EffectNecessarilyFollowsFrom(y,x) &
                           ( -DefiniteCause(x) -> -EffectNecessarilyFollowsFrom(y,x) ) )
              # label("Axiom III").

    %  Axiom IV.  The knowledge of an effect depends on and involves
    %              the knowledge of a cause.
    KnowledgeOf(x,y)  <-> IsACause(x,y)
               # label("Axiom IV: The knowledge of an effect depends on and
                    involves the knowledge of a cause").



    %  Axiom V.  Things which have othing in common cannot be understood,
    %             the one by the means of the other
    %             the one by means of the other; the conception of one
    %             does not involve the conception of the other.
    HaveNothingInCommon(x,y) -> ( (-CanBeUnderstoodInTermsOf(x,y) ) &
                                  (-CanBeUnderstoodInTermsOf(y,x) ) &
                                  ( -ConceptionInvolve(x,y) )  &
                                 (-ConceptionInvolves(y,x) ) )
               # label("Axiom V: Things which have nothing in common cannot be understood,
                      the one by means of the other.").



    %  Axiom VI.  A true idea must correspond with its idea or object.
    TrueIdea(x) -> ( CorrespondWith(x,y) & (  IdeateOf(y,x) | ObjectOf(y,x) ) )
               # label("AxiomVI").


    %  Axiom VII.  If a thing can be conceived as non-existing, its
    %              essence does not involve its existence.
    CanBeConceivedAsNonExisting(x) -> -EssenceInvExistence(x)
               # label("Axiom VII").

% AUXILIARY ASSUMPTIONS (JKH added).

    Substance(x) -> Being(x)
           # label("Auxiliary assumption 1: if x is a substance, x is a being").
```

```
(InItself(x) & CanBeConceivedThruItself(x)) -> (Modfication(y,x) -> PriorTo(x,y))
       # label("Auxiliary assumption 2: if x is in itself and can be conceived
           through itself, then if y is a modification of x, x is prior to y").


KnowledgeOf(x,y) -> CanBeUnderstoodInTermsOf(y,x)
       # label("Auxiliary assumption 3: if x is knowledge of y, then y
                can be understood in terms of x").



InItself(x) -> SelfCaused(x)
       # label("Auxiliary assumption 4: if x is in itself, x is self-caused").


HasInfiniteAttributes(x) ->  EssenceInvExistence(x)
  # label("Auxiliary assumption 5: if x has infinite attributes,
            x has the property that x's essence involves x's existence").

AbsolutelyInfinite(x) -> HasInfiniteAttributes(x)
  # label("Auxiliary assumption 6: If x is absolutely infinite,
              then x has infinite attributes").



Being(x) -> HasEssence(x)
       # label("Auxiliary assumption 7: If x has being, then x has essence").


(EssenceInvExistence(x) & HasEssence(x)) -> Exists(x)
       # label("Auxiliary assumption 8: if the essence of x involves
           the existence of x and x has essence, then x exists").



%    Proposition VII.  Negate  "substance -> existence".
     Substance(x) -> Exists(x)
              # label("Proposition VII: Substance exists").


end_of_list.
```

Figure 1. A *prover* ([2]) script for generating a derivation of (SE) from the DAPI conjoined with 'AUXILIARY ASSUMPTIONS'. For a detailed description of the *prover9* syntax and semantics, see [2].

Figure 2 shows the derivation of (SE) from the script in Figure 1.

```
============================== Prover9 ===============================
Prover9 (32) version 2009-02A, February 2009.
Process 2624 was started by User on User-PC,
Tue Dec 10 05:24:01 2013
The command was "../bin/prover9".
============================== end of head ===========================

...

============================== PROOF =================================

% Proof 1 at 0.09 (+ 0.05) seconds.
% Length of proof is 22.
```

```
% Level of proof is 5.
% Maximum clause weight is 2.
% Given clauses 0.

1 SelfCaused(x) <-> EssenceInvExistence(x) & NatureConcOnlyByExistence(x) # label("Definition I:
self-caused") # label(non_clause).  [assumption].
3 Substance(x) <-> InItself(x) & CanBeConceivedThruItself(x) # label("Definition III: substance")
# label(non_clause).  [assumption].
18 Substance(x) -> Being(x) # label("Auxiliary assumption 1: if x is a substance, x is a being")
# label(non_clause).  [assumption].
21 InItself(x) -> SelfCaused(x) # label("Auxiliary assumption 4: if x is in itself, x is self-
caused") # label(non_clause).  [assumption].
24 Being(x) -> HasEssence(x) # label("Auxiliary assumption 7: If x has being, then x has
essence") # label(non_clause).  [assumption].
25 EssenceInvExistence(x) & HasEssence(x) -> Exists(x) # label("Auxiliary assumption 8: if the
essence of x involves the existence of x and x has essence, then x exists") # label(non_clause).
[assumption].
26 Substance(x) -> Exists(x) # label("Proposition VII. Substance exists") # label(non_clause) #
label(goal).  [goal].
28 -SelfCaused(x) | EssenceInvExistence(x) # label("Definition I: self-caused").  [clausify(1)].
30 -InItself(x) | SelfCaused(x) # label("Auxiliary assumption 4: if x is in itself, x is self-
caused").  [clausify(21)].
35 -Substance(x) | InItself(x) # label("Definition III: substance").  [clausify(3)].
43 -Substance(x) | Being(x) # label("Auxiliary assumption 1: if x is a substance, x is a being").
[clausify(18)].
44 Substance(c1) # label("Proposition VII. Substance exists").  [deny(26)].
82 -InItself(x) | EssenceInvExistence(x).  [resolve(30,b,28,a)].
87 InItself(c1).  [resolve(44,a,35,a)].
118 -Being(x) | HasEssence(x) # label("Auxiliary assumption 7: If x has being, then x has
essence").  [clausify(24)].
120 Being(c1).  [resolve(44,a,43,a)].
128 -EssenceInvExistence(x) | -HasEssence(x) | Exists(x) # label("Auxiliary assumption 8: if the
essence of x involves the existence of x and x has essence, then x exists").  [clausify(25)].
130 EssenceInvExistence(c1).  [resolve(87,a,82,a)].
148 -Exists(c1) # label("Proposition VII. Substance exists").  [deny(26)].
187 HasEssence(c1).  [resolve(120,a,118,a)].
190 -HasEssence(c1) | Exists(c1).  [resolve(130,a,128,a)].
191 $F.  [copy(190),unit_del(a,187),unit_del(b,148)].

============================== end of proof ==========================
```

Figure 2. A derivation of (SE) from the script shown in Figure 1. For a detailed description of the *prover9* syntax and semantics, see [2].

4.0 Discussion and conclusions

Sections 2.0 and 3.0 motivate several observations:

1. A companion paper shows that (SE) cannot be derived from the DAPI proper.

2. The derivation shown in Figure 2 uses only four of the auxiliary assumptions of the script shown in Figure 1 (the other auxiliary assumptions are used in the derivation of propositions not essential to the purposes of this paper):

- Auxiliary assumption 1: If x is a substance, x is a being
- Auxiliary assumption 4: If x is in itself, x is self-caused
- Auxiliary assumption 7: If x is a being, then x has essence
- Auxiliary assumption 8: If the essence of x involves the existence of x and x has essence, then x exists.

It is difficult to imagine any serious objection to any of these assumptions within the context of Spinoza's philosophy (in that

context, they are arguably definitional), and I believe Spinoza would have accepted them.

3. As [3] notes, an ADS can be a useful augmentation of the traditional methods of textual exegesis. An ADS is not a panacea, of course. Indeed, there is no guarantee that an ADS will produce results of interest in a time we care to wait. That said, an ADS is often astonishingly good at providing insights that may not otherwise be obvious.

5.0 Acknowledgements

This work benefited from discussions with Ed Zalta, Paul Oppenheimer, Paul Spade, Tom Oberdan, and Joe Van Zandt. For any infelicities that remain, I am solely responsible.

7.0 References

[1] Spinoza. *The Ethics* (published posthumously, 1677). In Benedict de Spinoza. *On the Improvement of the Understanding, The Ethics, Correspondence.* Unabridged trans. by RHM Elwes (1883). Dover reprint, 1955.

[2] McCune WW. *prover9 and mace4.* February 2009. http://www.cs.unm.edu/~mccune/prover9/.

[3] Oppenheimer P and E Zalta. A computationally-discovered simplification of the ontological argument. *Australasian Journal of Philosophy* 89/2 (2011), 333-349.

[4] Horner JK. *spinoza_ethics*, an automated deduction system for Part I of Spinoza's *The Ethics*. 2013. Available from the author on request.

[5] Chang CC and HJ Keisler. *Model Theory.* North-Holland. 1990.

[6] Kant. *The Critique of Pure Reason.* Trans. by N Kemp Smith. St. Martin's. 1929.

[7] Church A. *Introduction to Mathematical Logic.* Part I. Princeton. 1956.

[8] *Journal of Automated Reasoning.* Springer.

[9] Horner JK. An automated deduction of implication-type-restricted Foulis-Holland theorems from orthomodular quantum logic. In eight parts. *Proceedings of the 2013 International Conference on Artificial Intelligence.* Las Vegas NV, July 2013. CSREA Press. 2013. pp. 374-429.

[11] Oppy G. Ontological Arguments. *Stanford Encyclopedia of Philosophy.* 2011. http://plato.stanford.edu/entries/ontological-arguments/.

[12] Anselm. *Prosologion.* Circa 1050. Trans. by SN Deane. In *St. Anselm: Basic Writings.* Second Edition. Open Court. 1962.

[13] Horn A. On sentences which are true of direct unions of algebras. *Journal of Symbolic Logic* 16 (1951), 14–21.

[14] Hennessy JL and DA Patterson. *Computer Architecture: A Quantitative Approach.* Fourth Edition. Morgan Kaufmann. 2007

A Computationally Generated Ontological Argument Based on Spinoza's *The Ethics*: Part 5

Jack K. Horner
PO Box 266, Los Alamos NM 87544
jhorner@cybermesa.com

ICAI 2014

Abstract

The comments accompanying Proposition (Prop.) 11 ("God ... necessarily exists") in Part I of Spinoza's The Ethics contain sketches of at least three distinct ontological arguments. The first of these is suspiciously brief. I show in a companion papers that but worse is true: even the proposition "God exists" (GE), an implication of Prop. 11, cannot be derived from the definitions and axioms of Part I (the "DAPI") of The Ethics; thus, Prop. 11 cannot be derived from the DAPI, either. However, if we assume the propositions (a) "If x is substance, x exists" (a consequence of Prop. 7 of the DAPI), and (b) "God is substance", we can immediately derive (GE). In a companion paper, I show that (a) can be derived from the DAPI conjoined with some plausible assumptions. Here I show that (b) can be derived from the DAPI. From these two results, we can derive (GE). The analysis demonstrates how an automated deduction system can augment traditional methods of textual exegesis.

Keywords: automated deduction, textual exegesis, Spinoza

1.0 Introduction

There are two objectives of this paper. The first is to explore some aspects of the validity, that is, whether there is a derivation that follows by inference rules alone from the premises of an argument of interest, of one of Spinoza's arguments for Prop. 11 ([1]). The second is to demonstrate how an automated deduction system can augment traditional methods of textual exegesis. I take no position on whether any proposition in [1] is true.

Ontological arguments are arguments for the claim that

(GE) God exists

based on premises which purport to derive from some source other than observation of the world ([11]), that is, from reason alone. They have a long history in the philosophical literature, extending to at least Anselm ([12]).

The comments accompanying Prop. 11 ("God, or substance, consisting of infinite attributes, of which each expresses eternal and infinite essentiality, necessarily exists") in Part I of Spinoza's *The Ethics* ([1]) contain sketches of at least three distinct ontological arguments. The first argument sketch of Proposition 11 in [1] (p. 51) is remarkably short. If we deny Proposition 11, Spinoza asserts, then we are claiming that God does not exist. But to deny God's

existence, he continues, is to deny that God's essence involves God's existence. That, however, is absurd, ([1], p. 48), he argues, by virtue of Prop. 7 ("Existence belongs to the nature of substances").

On its surface, this sketch is too brief to be convincing. For the argument to be cogent it must at least somehow connect God's essence to God's existence. And if Spinoza's hints are to be taken at face, a fully articulated argument corresponding to this sketch will also have to connect these ideas to the nature of substances (via some consideration of Prop. 7).

If we assumed the following two propositions, we could derive (GE)

a) If x is substance, x has existence (an implication of Prop. 7 of the DAPI)
b) God is substance

In a companion paper, I show that although (a) cannot be derived from the DAPI, it can be derived from the DAPI conjoined with some plausible assumptions.

At first glance, it seems that Spinoza believes he has already established (b), by definition. That definition of God (Df. VI, ""By God I mean a being absolutely infinite -- that is, a substance ..."), however, contains an identification of "a being absolutely infinite" with "a substance ...". This identification, however, does not follow from the meanings of "a being absolutely infinite" and "substance", and thus must be argued. Spinoza does not do so.

2.0 Method

The method used in this paper to investigate the first argument sketch of Prop. 11 of [1] relies heavily on an automated deduction system (ADS, [2]). It is in some ways similar to the method of [3], which is likely to be the first published use of [2] in "conventional" metaphysics. [2] and its predecessors have been widely used in mathematical logic applications (see, for example, [8], [9]) for at least two decades.

prover9 ([2]) is a software first-order automated deduction framework that searches for derivations of given propositions, given a set of propositions.

The DAPI were first rendered in the *prover9* framework and the result used to derive (b). In this rendering, I attempt to adhere rigorously to the text in [1], presuming nothing about the meanings of terms beyond what is explicitly stated in [1].

The script shown in Section 3.0 was executed on a Dell Inspiron 545 with an Intel Core2 Quad CPU Q8200 (clocked @ 2.33 GHz) and 8.00 GB RAM, running under the *Windows Vista Home Premium /Cygwin* operating environment.

3.0 Results

Figure 1 is a *prover9* script that can be used to derive (b), "God is substance ".

```
formulas(assumptions).


     % DEFINITIONS


     %  Definition of self-caused.
     SelfCaused(x) <-> ( EssenceInvExistence(x) &
                         NatureConcOnlyByExistence(x) )
             # label("Definition I: self-caused").



     %  Definite of "finite after its kind".
     FiniteAfterItsKind(x) <->  ( CanBeLimitedBy(x,y) & SameKind(x,y)  )
                           # label("Definition II: finite after its kind").

    %  Definition of substance.
    Substance(x)  <-> InItself(x) &
                      CanBeConceivedThruItself(x)  # label("Definition III:
substance").

    %  Definition of attribute.
    Attribute(x) <-> IntPercAsConstEssSub(x)  # label("Definition IV:
attribute").

    %  Definition of mode.
    Mode(x) <->  ( ( Modification(x,y) & Substance(y) ) |
                   ( ExistsIn(x,y) & ConceivedThru(x,y)) )    #
label("Definition V: mode").

    % Definition of God.
    God(x) <-> ( Being(x) & AbsolutelyInfinite(x) )  # label("Definition VI:
God").

    % Definition of absolutely infinite.
    AbsolutelyInfinite(x) <-> ( Substance(x) & ConstInInfAttributes(x) &
                               ( AttributeOf(y,x) -> (
ExpressesEternalEssentiality(y) &

ExpressesInfiniteEssentiality(y) ) ) )
                           # label("Definition VI: absolutely infinite").

    % Definition of free.
    Free(x) <-> ( ExistsOnlyByNecessityOfOwnNature(x)  &
                  ( ActionOf(y,x) -> DeterminedBy(y,x) ) )    #
label("Definition VII:  free").

    %  Definition of necessary.
```

```
    Necessary(x) <-> ( (  ExternalTo(y,x) & DeterminedByFixedMethod(x,y) &
                        DeterminedByDefiniteMethod(x,y) ) &
                        ( IsMethod(Action(y) | IsMethodExistence(y) ) ) )
                                  # label("Definition VII: necessary").

    %  Definition of eternity.
    Eternity(x) <->  ExistConcFollowFromDefEternal(x)  # label("Definition
VIII: eternity").



    %  AXIOMS

    %  Axiom I.  Everything which exists, exists either in itself
    %            or in something else.
    Exists(x) <-> ( ExistsIn(x,x) | (ExistsIn(x,y) &  (x != y) ) )
                          # label("Axiom I").



    %  Axiom II.  That which cannot be conceived through itself must
    %             be conceived through something else.
    -( ConceivedThru(x,x) ) -> (ConceivedThru(x,y) & (x != y) )
                          # label("Axiom II").


    %  Axiom III.  From a given definite cause an effect necessarily
    %              follows; and, on the other hand, if no definite
    %              cause be granted, it is impossible that an effect
    %              can follow.
    DefiniteCause(x)  -> ( EffectNecessarilyFollowsFrom(y,x) &
                           ( -DefiniteCause(x) -> -
EffectNecessarilyFollowsFrom(y,x) ) )
               # label("Axiom III").

    %  Axiom IV.  The knowledge of an effect depends on and involves
    %              the knowledge of a cause.
    KnowledgeOf(x,y)  <-> IsACause(x,y)
              # label("Axiom IV: The knowledge of an effect
                depends on and involves the knowledge of a cause").



    %  Axiom V.  Things which have othing in common cannot be understood,
    %             the one by the means of the other
    %             the one by means of the other; the conception of one
    %             does not involve the conception of the other.
    HaveNothingInCommon(x,y) -> ( (-CanBeUnderstoodInTermsOf(x,y) ) &
                                  (-CanBeUnderstoodInTermsOf(y,x) ) &
                                  ( -ConceptionInvolve(x,y) )  &
                                 (-ConceptionInvolves(y,x) ) )
              # label("Axiom V: Things which have nothing in common cannot
                    be understood, the one by means of the other.").
```

```
    %  Axiom VI.  A true idea must correspond with its idea or object.
    TrueIdea(x) -> ( CorrespondWith(x,y) & (  IdeateOf(y,x) | ObjectOf(y,x) )
)
               # label("AxiomVI").


    %  Axiom VII.  If a thing can be conceived as non-existing, its
    %              essence does not involve its existence.
    CanBeConceivedAsNonExisting(x) -> -EssenceInvExistence(x)
               # label("Axiom VII").


end_of_list.


formulas(goals).

%    Proposition.  God is substance.
     God(x) -> Substance(x)
              # label("Proposition. God is substance").


end_of_list.
```

Figure 2. *prover9* script for deriving "God is substance" from the DAPI. For further detail on the syntax and semantics of *prover9*, see [2].

Figure 2 shows the derivation of (b) from the script shown in Figure 1.

```
============================== Prover9 ===============================
Prover9 (32) version 2009-02A, February 2009.
Process 4460 was started by User on User-PC,
Tue Dec 10 09:39:16 2013
The command was "../bin/prover9".
============================== end of head ===========================
. . .

============================== PROOF =================================

% Proof 1 at 0.06 (+ 0.05) seconds.
% Length of proof is 10.
% Level of proof is 4.
% Maximum clause weight is 2.
% Given clauses 0.

6 God(x) <-> Being(x) & AbsolutelyInfinite(x) # label("Definition VI: God") #
label(non_clause).  [assumption].
7 AbsolutelyInfinite(x) <-> Substance(x) & ConstInInfAttributes(x) &
(AttributeOf(y,x) -> ExpressesEternalEssentiality(y) &
ExpressesInfiniteEssentiality(y)) # label("Definition VI: absolutely
infinite") # label(non_clause).  [assumption].
```

```
26 God(x) -> Substance(x) # label("Proposition. God is substance") #
label(non_clause) # label(goal).  [goal].
51 -God(x) | AbsolutelyInfinite(x) # label("Definition VI: God").
[clausify(6)].
52 God(c1) # label("Proposition. God is substance").  [deny(26)].
54 -AbsolutelyInfinite(x) | Substance(x) # label("Definition VI: absolutely
infinite").  [clausify(7)].
60 AbsolutelyInfinite(c1).  [resolve(52,a,51,a)].
100 -Substance(c1) # label("Proposition. God is substance").  [deny(26)].
111 Substance(c1).  [resolve(60,a,54,a)].
112 $F.  [copy(111),unit_del(a,100)].

============================== end of proof ==========================
```

Figure 2. *prover9* derivation of(b), "God is substance". For further detail on the syntax and semantics of *prover9*, see [2].

4.0 Discussion and conclusions

Sections 2.0 and 3.0 motivate several observations:

1. The above arguments show (b) ("God is substance") can be derived from the DAPI.

2. (a) conjoined with (b) implies (GE).

3. As [3] notes, an ADS can be a useful tool for in textual exegesis. An ADS is not a panacea, of course. Indeed, there is no guarantee that an ADS will produce results of interest in a time we care to wait. That said, an ADS is often astonishingly good at providing insights that may not otherwise be obvious.

5.0 Acknowledgements

This work benefited from discussions with Ed Zalta, Paul Oppenheimer, Paul Spade, Tom Oberdan, and Joe Van Zandt. For any infelicities that remain, I am solely responsible.

7.0 References

[1] Spinoza. *The Ethics* (published posthumously, 1677). In Benedict de Spinoza. *On the Improvement of the Understanding, The Ethics, Correspondence*. Unabridged trans. by RHM Elwes (1883). Dover reprint, 1955.

[2] McCune WW. *prover9 and mace4*. February 2009. http://www.cs.unm.edu/~mccune/*prover9*/.

[3] Oppenheimer P and E Zalta. A computationally-discovered simplification of the ontological argument. *Australasian Journal of Philosophy* 89/2 (2011), 333-349.

[4] Horner JK. *spinoza_ethics*, an automated deduction system for Part I of Spinoza's *The Ethics*. 2013. Available from the author on request.

[5] Chang CC and HJ Keisler. *Model Theory*. North-Holland. 1990.

[6] Kant. *The Critique of Pure Reason.* Trans. by N Kemp Smith. St. Martin's. 1929.

[7] Church A. *Introduction to Mathematical Logic.* Part I. Princeton. 1956.

[8] *Journal of Automated Reasoning.* Springer.

[9] Horner JK. An automated deduction of implication-type-restricted Foulis-Holland theorems from orthomodular quantum logic. In eight parts. *Proceedings of the 2013 International Conference on Artificial Intelligence*. Las Vegas NV, July 2013. CSREA Press. 2013. pp. 374-429.

[11] Oppy G. Ontological Arguments. *Stanford Encyclopedia of Philosophy*. 2011. http://plato.stanford.edu/entries/ontological-arguments/.

[12] Anselm. *Prosologion.* Circa 1050. Trans. by SN Deane. In *St. Anselm: Basic Writings.* Second Edition. Open Court. 1962.

[13] Horn A. On sentences which are true of direct unions of algebras. *Journal of Symbolic Logic* 16 (1951), 14–21.

[14] Hennessy JL and DA Patterson. *Computer Architecture: A Quantitative Approach.* Fourth Edition. Morgan Kaufmann. 2007

A Knowledge Based System for Law Domain

Maria da Graça Pereira, Dr.
GEIASC
Caixa Postal 606,
88010-970 - Florianopolis, SC – Brazil
UNISUL

ICAI'2014

Abstract

The purpose of the present work is to develop an expert law system for producing a toolkit to dealing with Law Civil Procedural. The system will be able to encapsulate some of the expertise used by professors in Law Civil Procedural environment. The knowledge acquisition process involves both the elicitation of knowledge directly from experts. That is teachers and the analysis of reports from class with pupils. An application of the expert scheduling system is presented in order to verify its uses and to identify its principal difficulties and limitations. The model was implemented on an expert system shell. The evaluation of the system focused on the validity of the model. That is the degree at which the outcomes of the system resembled the outcomes of the human expertise being modeled in the knowledge base.

Keywords: Expert Systems, Knowledge Acquisition, Intelligent Software Law.

1 Introduction

The expert system developed in this study, in addition to the main objective, which is to teach the context of civil laws, also aims at offering support to the teachers involved in the learning this subject and to the graduate professors, who are the users of the system in class.

A further objective is to disseminate the technology used in Brazil in order to use learning systems to teach another subject [4], [6], [7]. This kind of systems is will used in Brazil in engineering and mathematics. The knowledge is give for a professor of the subject. This technology is completely mastered or widely known, although it is very much utilized in high schools and universities. However, its application

may also be advantageous for other kinds of learning.

During the development of the system and throughout the whole period of research with the students, guidelines were considered, establishing a goal in the solution of a problem, as presented by Polya [8].

The following points were studied and considered in developing the system:

*Understanding the problem: Why is the unknown? What are the data? What are the conditioners?

*Working out a plan: To find a connection between the known and the unknown, sometimes one may need to consider auxiliary problems if an immediate solution cannot be found, but eventually it will be necessary to arrive at a plan for solving the problem.

*Carrying out the Plan: At this point, the following questionings arise: Is it possible to verify clearly that the step is correct? Is it possible to show that it is correct?

*Retrospect: Examine the solution obtained, with the following questions: Is it possible to verify the result? Is it possible to verify the argument? Is it possible to get the result in a different way? Is it possible to perceive this at a glance? Is it possible to utilize the result or the method to solve some other problem?

The expert system that was developed considers all the points mentioned above when he presents the theory to the student, while he is questioning him/her and receiving his/her reply.

2 Artificial Intelligence in Learning Programming

Artificial intelligence is the part of computer sciences linked to the project of intelligent computer programs, i.e. systems that show characteristics that we usually associate with intelligence in human behavior (language comprehension, learning, reasoning, problem solving, etc.) [2], [12].

There are several different areas of artificial intelligence, such as: engineering of knowledge, pattern recognition, language processing and robotics.

Law systems have been research areas in artificial intelligence over the last twenty years. The first systems focused on the law and ordinary system. Subsequently more ambitious systems have incorporated the development of other areas in artificial intelligence. Conceptions through expert systems have especially influenced the development of many programs based on concepts of artificial intelligence. Some expert systems have been successful, when applied to law systems [3], [11].

A expert system is a computer system that models human knowledge of a particular domain and is capable of making intelligent decisions within that domain. This is usually done by applying a set of facts so as to imitate the thought processes of a human expert in order to reach conclusions or make decisions [9].

The formalisms utilized for the representation of knowledge are frames and rules

of production. Frames are utilized in the representation of objects and concepts through a set of attributes and operators. Each object is associated with a name and a series of attributes or slots with their respective values [13].

The rules of production specify actions that are expected to occur in certain situations. These rules, as computational elements, must adequately represent the knowledge involved within the reasoning that is being simulated in the system. The rule is executed only when any one of the data of the problem coincides with the conditions for the application of the rule found in its predecessor, inasmuch as the rules form a chain. This chaining process may either move forward or backward [1].

3 The Knowledge Based System

The system is composed of the four following modules:

(1) Initialization Module - In the first module, the procedures that initialize the information contained in the system get under way. At this point information is supplied regarding the number of elements to be used within the parameters of ideal efficiency for this process. This module receives information that enables the system to determinate the yield for the remaining modules. Here lists are utilized to store the necessary information for law domain in the first period, according to Civil Procedural environment [10].

(2) Second Module -This module covers the whole production the rules for frameworks, beginning with Abide by judgment: In this module, through procedures followed by technicians in Syncretism process: cognition phase and satisfied phase is modeled for the system, it is possible to obtain the programming for Judgment execution: art. 475-N, I, III, V and VII, Code of civil procedure, as well as, Obligation to do and obligation not to do: art. 461, Code of civil procedure. More obligation to delivery thing: art. 461, Code of civil procedure. And obligation to pay amount: art. 475-I, Code of civil procedure and finally various comparison is established between the programming.

(3) Third Module - After programming the second module, all the procedures that lead to the production of Free execution are considered. The Free execution is then to be joined to the framework, to obtain each Judgment execution: art. 475-N, II, IV and VI, Code of civil procedure:-Sentence:- Award,- Agreement to confirm by strange judgment. This module considers all the intermediary procedures until obtaining the elements that should be ready More:-Alimony (art. 733, Code of civil procedure),- Execution of obligation to pay against Public finances (art. 730, Code of civil procedure),- Execution against insolvent debtor (art. 748, Code of civil procedure). In this module all the procedures are also considered which comprise the process. In this case, it is possible to obtain programming of Free execution.

(4) Forth Module - In this last module all the procedures followed from the Free execution phase up to Foreclosure: art. 585, Code of civil procedure, as well as the subsequent release of the Obligation to do: art. 632, Code of civil procedure, obligation not to do: art. 642, Code of civil procedure, obligation to delivery thing: art. 621, right, and art. 629 not right, Code of civil procedure and obligation to pay amount: art. 646, Code of civil procedure. It comprises the final procedures and also presents the programming result.

The expert system we are developing will be submit to a series of tests and will be show is being suitable to the various situations studied. Specifically the system will be tested in cases that showed special characteristics, in terms of the result itself, that is, cases in which the result to be obtained by the system was known beforehand, through the information in control reports. In those cases where the system will be applied to known situations, with real data, it will be extracted from the Civil Procedural under study, the result obtained was the same as the expected one.

Through the results obtained in the application of the expert system, access can be gained to a series of complementary procedures, making it possible to attain near-optimization in sequencing. These actions are determined by the results obtained in the various processes of the system and, through their interaction with the laws, yield important data for highlighting the production process or correcting incidental faults.

The system constitutes a very important aid in decision-making. Comparative tests between reports on line of the plans created by the system, show clear advantages for the latter, mainly when the programming, inasmuch as they result in a better distribution of resources during the period and a better utilization of the resources in relation to production bottlenecks.

4 Conclusions

The expert system is an important resource in decision-making. The comparative tests that were made between the reports of laws already obtained and the plans created by the system, point to advantages for the latter , mainly regarding programming, favoring a better distribution of resources over a given period and a better utilization of these resources in terms of bottlenecks. The system permits expansion and other groups of different laws, with some degree of flexibility and without many modifications on the basis of the original knowledge. The techniques of elicitation, used in the acquisition of knowledge have helped overcome the greatest difficulty in this research, namely, the participation of expert in the process, due to their lack of time for interviews and for telling how they have solved their problems, so as to promote verbal protocols.

The development of the computational system was subdivided into modules and they, in

turn, were subdivided into stages that were rigorously programmed according to the basic idea of the production rules. This was done "a priori", that is, before developing the system, all the priorities, activities, resources and restrictions of production were determined and analyzed.

The evaluation of the system focused on the validity of the model, i. e. the degree at which the outcomes of the system resembled the outcomes of the human expertise being modeled in the knowledge base. A prescriptive method of validation was devised specifically for this study, involving both experts that had provided expertise for the system, and external experts.

5 References

[1]. Dym, Clive L. and Levitt, Raymond E. "Knowledge-Based Systems in Engineering". New York, McGraw-Hill, Inc. 1991.

[2]. Goldszein,M, and Carnota. R. "Inteligência Artificial Aplicada - Lógica Y Representación Del Conocimiento". Campinas, Editora da UNICAMP. 1986.

[3]. Hoeschl, H. C. "Sistema OLIMPO: Tecnologia da Informação Jurídica para o Conselho de Segurança da ONU". Tese de doutorado. UFSC, Florianópolis, SC. 2001.

[4]. Ijuris.."Ontoweb: Ferramenta Informacional do Governo Eletrônico", Instituto do Governo Eletrônico, Inteligência Jurídica e Sistemas. In: http://www.ijuris.org. Access: 20 may 2007.

[5]. Pereira, Maria da Graça. Vantagens da Utilização de Sistemas Especialistas na Programação da Produção", XIV ENEGEP 1994, 194- 196. 1994.

[6]. Pereira, M. G. "Demonstrative Prototype to Teach . Statistical Methodos". – IC-AI'04. 418-422. Las Vegas. USA. 2004

[7]. Pereira, M. G. "Artificial Intelligence – Techniques for Search Results in Programing Projects". IFORS 2005. Hawaii. USA. 2005.

[8]. Polya, G. "A Arte de Resolver Problemas". Editora Interciência. Rio de Janeiro. 1986.

[9]. Ragsdale, Cliff T.." Deriving Conclusions in Expert Systems when Knowledge is Incomplete". Computers Ops. Res.,20(1), 49- 58. 1993.

[10]. Santos, M. M. "Primeiras Linhas de Direito Processual Civil". Editora Saraiva. 2009.

[11]. Sriram, D.and Adey R. A. "Knowledge Based Expert Systems In Engineering: Planning end Design", Computacional Mechanics Publications, 36-47. 1987.

[12]. Waterman, Donald A. " A Guide to Expert Systems, Reading". Addison-Weskey. 1986

[13]. Watson, Ian D. et all. "A Knowledge Analysis Methodology Using an Intermediate Knowledge Representation Based on Conceptual Graphs".
Proceedings do the 9th Int. Workshop on Expert Systems and their Applications. Avignon, 1, 183-198. 1989.

Ontologies to coordinate multiple views: exploring document collections

Jorge Marques Prates, Rogério Eduardo Garcia, Danilo Medeiros Eler
Departamento de Matemática e Computação – Faculdade de Ciências e Tecnologia
UNESP-Universidade Estadual Paulista "Júlio de Mesquita Filho"
Presidente Prudente,SP - Brazil
jorgemprates@gmail.com, {rogerio,daniloeler}@fct.unesp.br

***Abstract*—Multiple views of data sets can provide assistance on discovering unforeseen associations among elements contained on these sets. Thus, users are able to explore data in distinct perspectives. Ontologies are formal representations which describe data relations explicitly. Representing the underlying data into ontology, the exploratory visualization can benefit from a semantic representation to create the mappings, and might be helpful to establish relations on multiple views. In this way, we propose the application of ontology to support the mapping in the coordination process, defining how the data elements are related. Also, we present two case studies applying ontologies on exploring text collections. After these presentation, the results are compared with traditional coordination techniques.**

***Keywords*-Ontology, semantic mapping, knowledge representation, coordinated and multiple views.**

I. Introduction

Multiple views of data sets allows users to explore data under varying perspectives [1], which may result in a better understanding and assimilation of the transmitted information. Exploring data by using complementary views helps to reveal uncover unforeseen relations in the underlying data [2], [3]. Furthermore, allows the visualization of data sets under distinct perspectives, since the primary concern is to reveal new relations [3], [4].

A important aspect of the exploration process is the employed coordination technique, which is responsible to establish relationships among elements from distinct views. For instance, by exploring documents collections, users can select a document in a view and automatically highlight documents covered by a topic (extracted from the selection in source view) in other views. To perform exploratory visualization, it is necessary to establish relations among the instances from different views. The mechanism responsible to determine these relations is called mapping.

However, the existing coordination techniques do not use semantics of data to establish the associations among elements, only relying on lexical representations to express the underlying data [5], [6]. This entails a limited exploration, due to a low discovery of the elements relationships.

A possible solution to perform a mapping based on semantic is by using ontologies. An ontology is a explicit specification of concepts, describing the meaning of a formal vocabulary and providing a specific conception of knowledge domains. Consists of logical axioms that contain the meaning of terms in a specific domain. These axioms represent hierarchies, concepts and their relationships, which may restrict the information, verify its correctness or to deduce new information [7], [8], [9].

Besides, ontologies contain classes that represent concepts and are usually organized into taxonomies (which organize knowledge by generalizing and specifying relations), allowing the application of inheritance mechanisms. New classes can be obtained by combination of classes, or from relationships restrictions. Moreover, classes present domain elements linked to a specific concept, which are represented by instances [10]. It is through relationships from this knowledge base, that is proposed a semantic mapping to bind instances present in distinct views.

In this paper we present the use of ontologies to express the information behind data elements from distinct views aiming to reveal previously hidden relationships among the elements, improving the mapping process. This technique, called *Semantic Coordination*, increases the user exploration power by discovering these new relationships.

In order to present our proposal and results of using ontologies on coordinating multiple views, the remaining part of this paper is organized as follows: Section II presents previous works, which use ontologies in coordinated multiple views. In Section III, is described the use of ontologies in the semantic coordination technique. Section IV shows the results, demonstrating the advantage of using this approach. Finally, the conclusions are presented in Section V.

II. Related works: ontologies in CMV

In this section, we present previous works that employ ontologies in CMV (Coordinated and Multiple Views). CMV allows users to explore collections by interacting with them and using complementary views [11]. The next approaches use ontologies to assist this process.

Seeling e Becks [12] propose system called *SWAPit*, in which ontologies are used to categorize the themes that define each document, based on similarity, categories and

related attributes. These system is designed to help users explore the associations between structured and unstructured data, including textual information, ontologies, document metadata, and sets of relational data.

Conceptually, the views are coupled by the associations among distinct data items. In addition, the system provides simultaneous exploration of a text repository categorized by similarity, the metadata descriptions of documents and the relational data associated to documents. According to the different types of information that need to be integrated into the multi-view environment, the following components are used: Similarity View, Relational View and Ontology View.

The ontology view provides schemes for the hierarchical classification of documents, thus each document can be associated to different classes in a schema. In addition, it may be used to help navigation in a set of documents, allowing the user to identify and present these documents by their classification.

In Figure 1 is showed a visual interface composed by four panels, which are color coded to indicate the origin of a selection. The panel of the document map is surrounded by a yellow rectangle, the area of tables is marked in red, whereas the area of the domain ontology is colored blue. Whenever the user selects a particular set of documents in a panel, the selection is simultaneously mirrored in the other two coordinates visions.

When the user makes a selection in ontology view, all documents that are associated with at least one of the names selected are highlighted in the document map by blue squares. Whereas in the table area table, tuples that are associated with at least one of the corresponding documents are highlighted by blue symbols.

A second application is proposed by Thai et al. [13]. In this application, text collections are explored through the use of ontologies in a system called IVEA, in which visualization components represent spheres of user interest. Thus, users are involved in the initial stage of the visualization process, defining their spheres of interest, the major entities and their relationships. The hierarchical structure of entities, defined in the ontology, allows users to interpret various aspects of a text collection in different levels of detail.

Each document in the collection can be represented by a record consisting of the number of variables (attributes), which values match the relevance values of these documents. This relevance value is the aggregate value of a document in relation to a class, in regard to their all direct instances and recursively, their all subclasses.

The visual interface of IVEA is shown in Figure 2. The collection overview, located in the upper right corner, shows a matrix based on multi-dimensional components. Also features a panel containing a document viewer, which allows users to read the document contents.

In addition, users can focus on specific entities and visualize the relevant data. This can be done by dragging a class

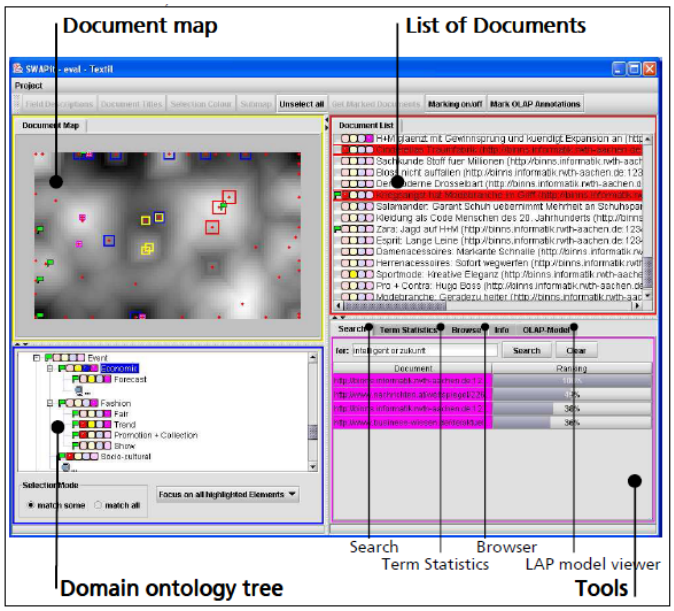


Figure 1. SWAPit interface, extracted from Seeling e Becks.

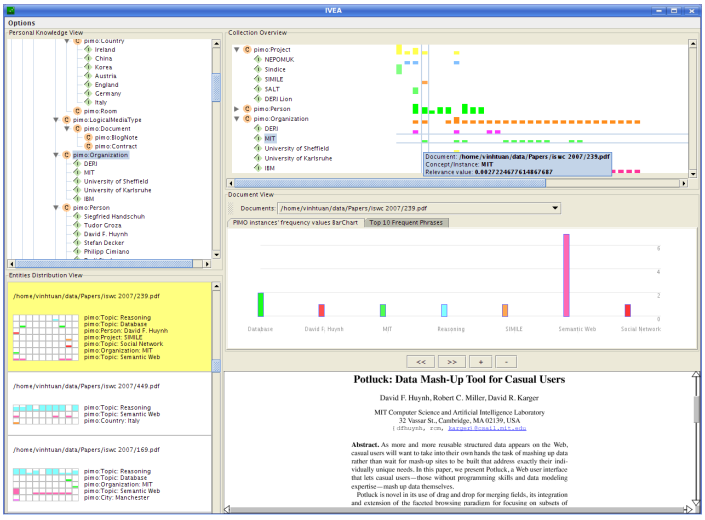


Figure 2. IVEA interface, extracted from Thai et al.

or an instance of *personal knowledge view* and dropping it to the overview screen. Once the hierarchical relationships are taken into account, it results in the inclusion of all the direct occurrences and recursively to all their subclasses.

Thus, the hierarchical relationship among ontology elements allows users to inspect a text collection in different granularity levels, expanding or minimizing the nodes which represent classes in the hierarchical structure.

III. Ontologies in Semantic Coordination

In this section, we propose a coordination technique using ontologies to provide a semantic mapping. For that, we

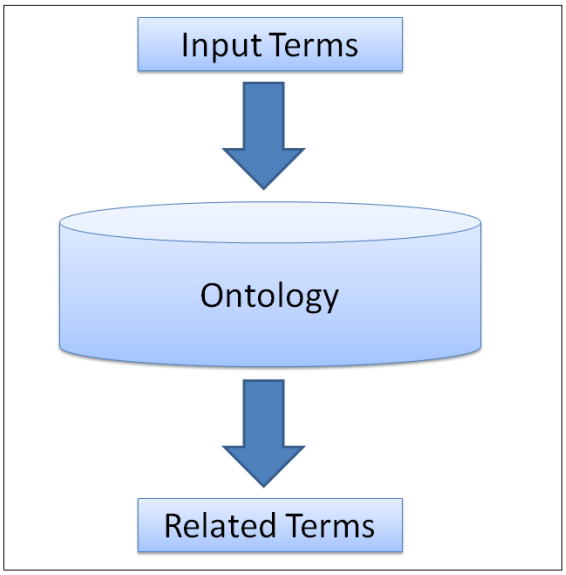


Figure 3. Matching process

use ontologies – a specification of a domain divided into concepts and their relations. These concepts are hierarchically organized, and axioms add meaning for each one, by declaring the entities that each concept is related to [14], [9].

Besides, they provide a vocabulary for representing knowledge (concepts and their relationships) from a specific domain through classes and their properties (functional, transitive, symmetric or inverse), instances and logical axioms [15]. Relying on an ontology, the proposed coordination technique can establish mappings supported by its hierarchy and relations. These mappings are determined directly by the relations declared in the ontology and in the properties that describes them. Besides, these relations provide the foundation to reasoning as well.

The first step of this technique consists on (after an user selection) the extraction of frequent terms from data items represented in the explored vision. These extracted terms play a key role in next step – the matching process, in which a number of correspondences must be found [16], [10]. In this process, illustrated in Figure 3, each term from input data set and terms from a ontology (written in *OWL* language) are compared, generating a new list of related terms.

The matching phase consists in the following steps: (i) matching between input terms and ontology concepts; (ii) matched class retrieval; (iii) retrieval of parent classes, subclasses and restrictions of retrieved class. After matching, the concepts related (obtained from ontology) are used to compose a new data set, which is used to identify related data items in the target views.

Finally, the third step consists in highlighting the documents which present the occurrence of terms obtained from matching process (terms from ontology) on target views. However, it is important to emphasize that it is necessary to use ontologies from same domain of knowledge than the explored data set, otherwise it is not possible to relate the elements.

IV. Results and Discussion

This section presents the results of two case studies using Semantic Coordination, in two different domains. Furthermore, the explored data is consisted of text document collections. Semantic Coordination was added as new coordination technique in a visualization tool called Projection Explorer (PEx) [17]. This tool already has some coordination techniques implemented, such as: *Identity Coordination, Topic Coordination, and Distance Coordination*. However, none of this techniques employ ontologies on the coordination process.

In a visual presentation on PEx, each document is represented by a circle projected into a plane. If two circles are close, the content of the documents represented by them is similar. Besides, it was adopted the rainbow scale color, in which blue indicates a null frequency of occurrence and red a maximum frequency.

For the case studies, tests and comparisons to a PEx coordination technique called *Topic Coordination* [18] were made. In this coordination technique, topics of interest are generated after the user selection in the source view. Then, the coordination process consists on searching topics from the selected documents, identifying documents that contains the selected topics. Finally, documents that are covered by the exact topics are highlighted in other views.

The first domain is composed by 300 documents, which describes x-rays and magnetic resonance images. These documents contain information about the images, the diagnostic and the pathology of a patient. To work with this domain we used an ontology about medical issues called *vHOG* [1] – an anatomical ontology for the lineage of vertebrates, based on an ontology called *CARO* (*Common Anatomy Reference Ontology*).

CARO is an ontology developed to facilitate the interoperability among existing ontologies for different species anatomies, besides it provides a model for building a new ontology for anatomy. This ontology includes 1184 terms, generally illustrated in Figure 4. These terms were used to compare expression patterns among species. Moreover, the vHOG is composed by structures which contain homologous relationships among the models of vertebrate species.

The first example shows a exploration for the term "muscle", using Topic Coordination. In Figure 5, after a selection on the left view, are highlighted in both views the documents covered by the term "muscle". For mapping

[1]The ontology was extracted from $http://www.obofoundry.org$

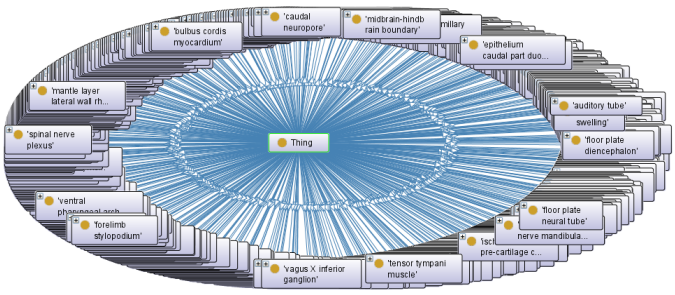


Figure 4. General view of vHOG ontology

these documents, it was only taken into account the lexical structure of this term. Thus, are only mapped documents which have the exact occurrence of the word "muscle". Using this approach documents covered by related terms to term"muscle" are not mapped and highlighted.

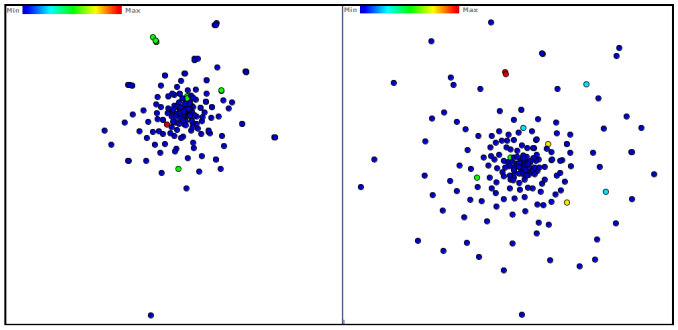


Figure 5. Projection of a document collection covered by the term "muscle"

In our approach we apply ontologies to map and coordinate elements in multiple views. Using Semantic Coordination, users are not only able to explore documents covered by terms related to the term "muscle". Mapping elements trough ontologies, it is possible to discover new related terms, expanding the coordination process. These new relations are revealed due to relations from ontology. In figure 6 are illustrated some of related terms to term "muscle".

Besides, users can filter which of related terms, are terms of his interest and then perform the mapping. In this example the term "diaphragm" was chosen for mapping in the target view. The relation between these two terms is illustrated in Figure 7, which shows a view from *Protégé* [2]. According to definition, "diaphragm" is the dome-shaped sheet of muscle that separates the chest from the abdomen. Thus, the relation between these terms is valid.

Using Semantic Coordination, are highlighted different elements than using the previous approach, as illustrated in

[2]A open source ontology editor – *protege.stanford.edu*

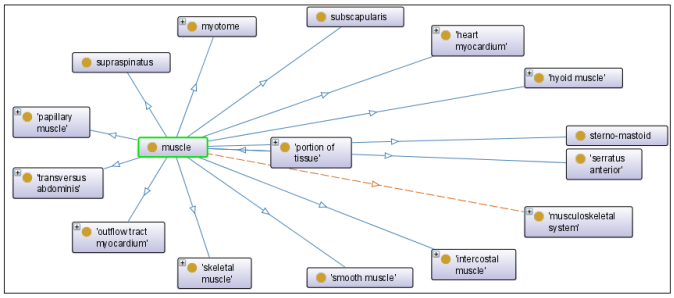


Figure 6. Related terms to term "muscle"

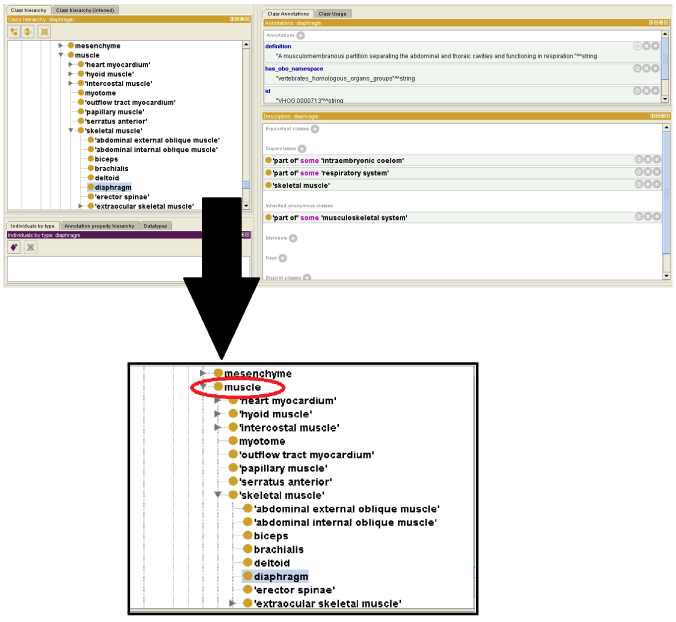


Figure 7. Relation between 'diaphragm" and "muscle", from Protégé view

Figure 8. Thus, it can be noticed that previously hidden connections are discovered through the use of ontology, which provide the reveal of these new relations.

The second domain is contained in Software Testing domain. It was composed by 232 articles of documents originated from five systematic reviews: Integration Testing of Aspect-Oriented Programs; Verification, Validation and Testing of Web-Services Composition; Mutation Testing of Aspect-Oriented Programs; Software Testing on Agile Methods; and Quality of Component-Based Software Development Processes [19].

This domain was mapped with the support of an ontology from the same domain, in this case, a software testing ontology called *Ontotest* [20]. This ontology was built in order to explore the different aspects related to testing activities, defining a common vocabulary for testing activities. The development of *OntoTest* is divided into two

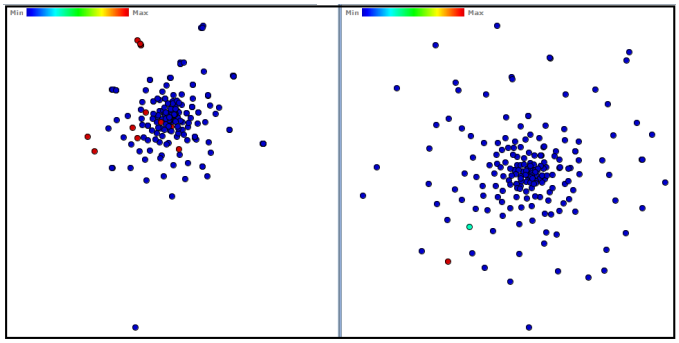


Figure 8. Projection of a document collection covered by the term "diaphragm"

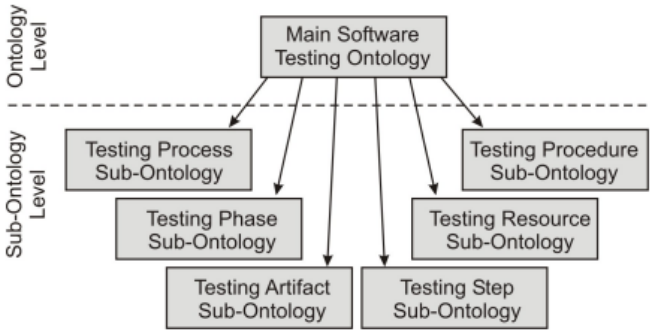


Figure 9. Ontotest structure, extracted from Barbosa et al.

layers, as illustrated in Figure 9. At the *Ontology Level*, the *Main Software Testing Ontology* deals with key concepts and relationships associated to testing. Whereas at the Sub-Ontology level, are handled specific concepts of the main ontology, as the process phases, artifacts, steps, procedures and resources for testing.

The second example shows the exploration of the term "training", using Topic Coordination. In Figure 10, after a selection on the left view, are highlighted in both views the documents covered by this term. Over again, are only mapped documents which have the exact occurrence of the term "training". However, documents covered by related terms are not highlighted.

Through ontology, it is verified that the term "training" is related to term "Organizational Activity". So, are mapped in Figure 11 documents which present the frequency of the term "Organizational Activity", i.e., are highlighted visual elements covered by this term. Over again, are detected distinct elements that using Topic Coordination.

The two case studies presented in this section shows the benefits of using ontologies on exploring documents collections, specifically in the coordinating process. This approach (Semantic Coordination) can be used as a resource

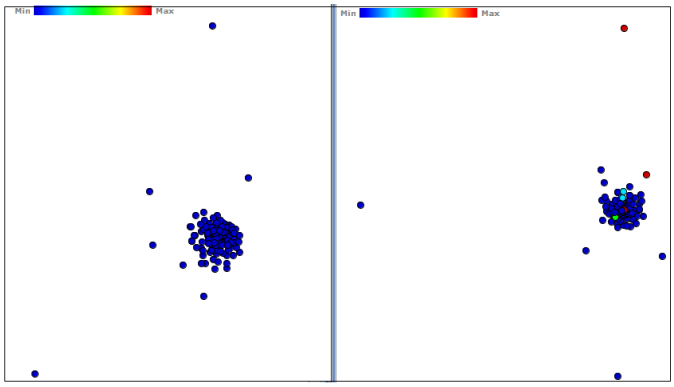


Figure 10. Exploration of a document collection covered by the term "training"

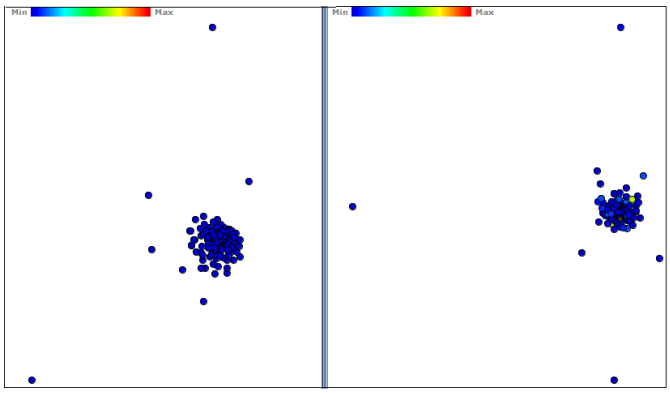


Figure 11. Exploration of a document collection covered by the term "Organizational Activity"

on filtering mapped documents, or on discovering new associations among elements. In both cases, increasing the possibilities of exploration from users.

V. Conclusion

Usually, traditional techniques limits the exploration user power. To deal with this limitation we propose a model relied on the concept that the mapping process, among multiple views, can be supported by underlying data [21]. In these approaches, mappings among elements from distinct views are established by data from each element represented in a view. For this purpose, we use ontologies in our approach, which provides a semantic mapping. This mapping expands the number of relationships among the elements, increasing the exploration power.

Opposed to previous presented works, we explore the structure of relationships that an ontology provides in the coordination process. Besides, this approach reveals a larger number of relations among elements than techniques that do not use ontologies on mapping process (i.e. Topic Coordination [18]).

In this way, our approach increases exploration of documents collections, finding similar elements (and not only identical ones) among distinct views on the coordination process. Furthermore, the same ontology can be used for various data sets, whereas they are contained in a same domain. However, Semantic Coordination is dependent on the quality of the ontology employed, because the terms must be well correlated. Thus the relations in ontology are crucial for the success of the mapping process.

Finally, this technique can be employed using distinct ontologies and datasets. Besides, it can be used on exploration of distinct domains of knowledge, the only restriction is that the ontology and the data set must belong to the same domain.

ACKNOWLEDGMENT

This work was partially supported by FAPESP (State of São Paulo Research Foundation), and CAPES (Coordination for the improvement of Higher Education Personnel).

REFERENCES

[1] C. North and B. Shneiderman, "Snap-together visualization: can users construct and operate coordinated visualizations?" *International Journal of Human-Computer Studies*, vol. 53, no. 5, pp. 715–739, Nov. 2000.

[2] D. A. Keim, "Information visualization and visual data mining," *IEEE Transactions on Visualization and Computer Graphics*, vol. 8, no. 1, pp. 1–8, 2002.

[3] H. Chen, "Towards design patterns for dynamic analytical data visualization," in *Proceedings Of SPIE Visualization and Data Analysis*, 2004, pp. 75–86.

[4] M. C. F. de Oliveira and H. Levkowitz, "From visual data exploration to visual data mining: a survey," *IEEE Transactions on Visualization and Computer Graphics*, vol. 9, pp. 378–394, 2003.

[5] C. E. Weaver, "Building highly-coordinated visualizations in improvise," in *Proceedings of the IEEE Symposium on Information Visualization*. austn, TX: IEEE, 2004, pp. 159–166.

[6] T. Pattison and M. Phillips, "View coordination architecture for information visualisation," in *Proceedings of the 2001 Asia-Pacific symposium on Information visualisation - Volume 9*, ser. APVis '01, vol. 1. Darlinghurst, Australia, Australia: Australian Computer Society, Inc., 2001, pp. 165–169.

[7] T. R. Gruber, "Toward principles for the design of ontologies used for knowledge sharing," *Int. J. Hum.-Comput. Stud.*, vol. 43, no. 5-6, pp. 907–928, Dec. 1995. [Online]. Available: http://dx.doi.org/10.1006/ijhc.1995.1081

[8] N. Guarino, *Formal Ontology in Information Systems: Proceedings of the 1st International Conference June 6-8, 1998, Trento, Italy*, 1st ed. Amsterdam, The Netherlands, The Netherlands: IOS Press, 1998.

[9] N. Guarino, D. Oberle, and S. Staab, *Handbook on Ontologies*. Springer Verlag, 2009, ch. What is an ontology?, pp. 201–221.

[10] J. Euzenat and P. Shvaiko, *Ontology matching*. Heidelberg (DE): Springer-Verlag, 2007.

[11] A. Heijs, "Requirements for coordinated multiple view visualization systems for industrial applications," in *Proceedings of the Fifth International Conference on Coordinated and Multiple Views in Exploratory Visualization*. Washington, DC, USA: IEEE Computer Society, 2007, pp. 76–79.

[12] C. Seeling and A. Becks, "Analysing associations of textual and relational data with a multiple views system," in *Proceedings of the Second International Conference on Coordinated & Multiple Views in Exploratory Visualization*. Washington, DC, USA: IEEE Computer Society, 2004, pp. 61–70.

[13] V. Thai, S. Handschuh, and S. Decker, "Tight coupling of personal interests with multi-dimensional visualization for exploration and analysis of text collections," in *12th International Conference Information Visualisation*. Washington, DC, USA: IEEE Computer Society, 2008, pp. 221–226.

[14] N. F. Noy and D. L. McGuinness, "Ontology development 101: A guide to creating your first ontology," Stanford University, Tech. Rep., 2001.

[15] C. Calero, F. Ruiz, and M. Piattini, *Ontologies for Software Engineering and Software Technology*. Berlin, Heidelberg: Springer-Verlag, 2006.

[16] Z. Aleksovski, M. Klein, W. ten Kate, and F. van Harmelen, "Matching unstructured vocabularies using a background ontology," in *Proceedings of the 15th international conference on Managing Knowledge in a World of Networks*. Berlin, Heidelberg: Springer-Verlag, 2006, pp. 182–197.

[17] F. V. Paulovich, M. C. F. Oliveira, and R. Minghim, "The projection explorer: A flexible tool for projection-based multidimensional visualization," in *XX Brazilian Symposium on Computer Graphics and Image Processing*. Washington, DC, USA: IEEE Computer Society, 2007, pp. 27–36.

[18] D. M. Eler, F. V. Paulovich, M. C. F. de Oliveira, and R. Minghim, "Coordinated and multiple views for visualizing text collections," in *12th International Conference Information Visualisation*. Washington, DC, USA: IEEE Computer Society, 2008, pp. 246–251.

[19] K. Felizardo, R. M. Martins, J. C. Maldonado, A. A. Lopes, and R. Minghim, "Content based visual mining of document collections using ontologies," in *II Workshop on Web and Text Intelligence (WTI)*, São Carlos, SP, Brazil, 2009, p. 1–8.

[20] E. F. Barbosa, E. Y. Nakagawa, and J. C. Maldonado, "Towards the establishment of an ontology of software testing," in *18th International Conference on Software Engineering and Knowledge Engineering (SEKE 2006)*, San Francisco, CA, July 2006, pp. 522–525, short Paper.

[21] M. Dörk, M. S. T. Carpendale, and C. Williamson, "Visualizing explicit and implicit relations of complex information spaces," *Information Visualization*, vol. 11, no. 1, pp. 5–21, 2012.

SESSION

MEDICAL APPLICATIONS, ASSISTIVE LIVING + HEALTHCARE

Chair(s)

TBA

Magnetic Resonance Imaging Registration With Artificial Neural Networks

Pramod Gadde, Xiao-Hua Yu
Department of Electrical Engineering, California Polytechnic State University
San Luis Obispo, CA 93407, USA

Abstract - *Image registration has widely been used in medical imaging, remote sensing, and computer vision. It relates multiple images of the same subject that were taken either at different time, or from different sensors, or from different points of views with a single coordinate system.*

In this study, the registration of MRI (magnetic resonance imaging) images is investigated with three different models based on artificial neural networks (ANN). Scale invariant feature transform (SIFT), discrete cosine transform (DCT), and discrete wavelet transform (DWT) is employed in each model, respectively. Two new models are investigated, e.g., SIFT-NN and DWT-NN. The performance of the above three models are studied and compared via computer simulations.

Keywords: Image registration, artificial neural networks, scale invariant feature transform, discrete cosine transform, discrete wavelet transform

1 Introduction

Image registration transforms different sets of data into the same coordinate system in order to align up and overlay multiple images ([1]). One image is often considered as the source image and the other one is often referred to as the target image to be registered. Image registration has been used widely in many fields such as medical imaging, remote sensing, and computer vision. It is especially important for medical research, where multiple images may be acquired at different time and each image may provide new information to doctors. However, these multiple images may not have the region of interest (e.g., a tumor) positioned in exactly the same spot or orientation; and thus need to be transformed to the same coordinate plane in order to overlap them. Image registration allows doctors to monitor the effects of treatments on patients in a certain region of interest over time.

In image registration research, the target image (to be registered) is considered as the source image with an affine transformation. Examples of affine transformations include translation, scaling, reflection, rotation, shearing etc., as well as the compositions of them in any combination and sequence. To represent affine transformations with matrices, we use homogeneous coordinates. That is, a two dimensional vector (x, y) is represented by a triplet $(x, y, 1)$. By using this augmented vector, it is possible to represent affine transforms in matrix form. For example, image translation can be represented as:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & \delta x \\ 0 & 1 & \delta y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (1)$$

where δx and δy represent the displacement on x and y, respectively. Similarly, rotation can be represented as:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (2)$$

where θ is the angle of rotation (clockwise).

Scaling (enlarging or shrinking) can be represented as:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (3)$$

where s_x and s_y are the scaling factor for x and y, respectively. When the absolute value of scaling factor is greater than 1, the transformation stretches the figures in the corresponding direction; otherwise, it shrinks them in that direction. Negative values of scaling factor flips (mirrors) the points in that direction. When $s_x \cdot s_y = 1$, the transform is called a "squeeze mapping" which preserves areas in the plane.

A complex transformation can be represented as a combination (multiplication) of the above matrices. For example performing a rotation after first scaling an image can be represented as:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \left(\begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \right) \quad (4)$$

In general, the four steps of registration are: feature detection; feature matching; transform model estimation; and image transform ([1]).

The first step is feature detection in which features are extracted from the image. Examples of features in the spatial domain are edges, corners, and intersections. Previous works in feature detection include extracting high contrast closed-boundary regions using segmentation methods ([1]), the virtual circle method ([5]), affinely invariant neighborhood algorithm based on Harris corner detector and edges ([2]), the approach based on representations of general line segments or elongated anatomic structures ([3]), the edge detection algorithms such as the Laplacian of Gaussian or Sobel operator, and the local curvature discontinuities detection method using the Gabor wavelets ([4]). The challenge is to ensure these features are robust to noise and complex transformations. For example, virtual circles can be extracted easily and are very effective for the registration of "clean" (noise-free) images; however, they are very sensitive to background noise and thus are not considered as strong features for the registration of noisy images.

In some medical applications, it is hard to extract strong features due to the lack of details in images. In order to perform registration on these types of images, an area-based method is used to extract and match features ([6]). This approach uses windows of a fixed size to match with images. Though this method works very well for simple transformations such as translation, it does not yield good results for more complicated situations. Besides, the areas of low intensity may be easily mismatched.

The approach based on Fourier transform has been applied for image registration ([1]). It is robust against narrow bandwidth noise and non-uniform time varying illumination disturbances. A 2-D (two-dimensional) Fourier transform is performed on both source and target images first; then the cross-power spectrum of both images are calculated and the location of the peak response is found. Note similar to the area based methods, this approach works well only for simple transforms such as translation.

There have been several studies on using neural networks for image registration in recent years ([7], [8], [9]). Most traditional approaches rely greatly on the specific features which are image dependent; thus additional image analysis is needed before registration. Besides, they can only estimate transformation parameters one at a time. For complicated image transform, an iterative process must be performed to obtain multiple parameters. In addition, for a new set of transform parameters, the whole procedure has to be repeated. The neural network based approach can estimate all the transformation parameters at once; and once it is fully trained, only forward computations are needed for a new set of parameters and thus its processing speed is very fast. With its nonlinear mapping ability, the neural network based approach is able to map complex input/output relationships and handle the cases with high distortion. The input to the neural network (feature vectors) include Fourier transform coefficients ([7]), DCT (discrete cosine transform) coefficients ([8]), centroid and outline points of image ([9]), low order Zernike moments ([10]). In [11], DCT coefficients and RBF (radial basis function networks) are employed.

In this study, the registration of MRI (magnetic resonance imaging) images is investigated with three different hybrid models using multi-layer feeedforward neural networks, including two novel models: SIFT-NN (scale invariant feature transform and neural network) and DWT-NN (discrete wavelet transform and neural network). The performance of the three proposed models are studied and compared via computer simulations.

The organization of this paper is as follows. Section 2 discusses the three different models with artificial neural networks. Computer simulation results are presented in section 3. Section 4 concludes the paper and gives directions for future works.

2 The proposed hybrid models for image registration

In this section, the three different hybrid models for image registration are discussed. First, the SIFT (scale invariant feature transform), DCT (discrete cosine transform), and DWT (discrete wavelet transform) are discussed in section 2.1, 2.2, and 2.3, respectively; then the neural network models for image registration are presented in section 2.4.

2.1 SIFT (Scale Invariant Feature Transform)

SIFT has been widely used in computer vision, image matching, and object recognition ([12], [13], [14]). The major advantage of using SIFT to extract features is that the features are invariant to rotation, scaling, and translation, and are robust to perspective changes. The SIFT algorithm has been used in image registration as well ([15]). The four main steps of SIFT algorithm are ([12], [13]):

1. Scale-space peak selection
2. Keypoint localization
3. Orientation assignment
4. Keypoint Descriptor.

To apply SIFT to an image, the image is first convolved with successive Gaussian kernels with different widths to produce blurred images separated by a constant factor; then DoG (Difference of Gaussian) images are obtained by subtracting adjacent image pairs in each octave. After each octave, the Gaussian image is down-sampled by a factor of 2; and the process repeated. The DoG operator is defined as:

$$DoG(x,y) = -\frac{1}{\sqrt{2\pi}}\left[\frac{1}{\sigma_1}e^{-\frac{(x^2+y^2)}{2\sigma_1^2}} - \frac{1}{\sigma_2}e^{-\frac{(x^2+y^2)}{2\sigma_2^2}}\right] \quad (5)$$

where σ_1 and σ_2 are the widths of Gaussian kernels; x and y represent the coordinate of a pixel in x and y direction. After all the DoG images are obtained, the keypoint candidates of the image are detected by checking each point against its 8 neighboring points (Fig. 1) as well as the 9 corresponding neighborhood points in the next and previous DoG images. The minimum and maximum of the points are selected.

(x-1, y-1)	(x-1, y)	(x-1, y+1)
(x, y-1)	(x, y)	(x, y+1)
(x+1, y-1)	(x+1, y)	(x+1, y+1)

Figure 1. Neighborhood of 8-connectivity rule

After all the keypoint candidates are detected, they are fitted to nearby data in order to get location, edge response, and peak magnitude. First, the location of the extreme can be found via Taylor series expansion. Define a vector $\boldsymbol{X}$:

$$\boldsymbol{X} = [x, y, \sigma]^T \quad (6)$$

$$D(\boldsymbol{X}) = D + \left[\frac{dD}{d\boldsymbol{X}}\right]^T (\Delta\boldsymbol{X}) + \frac{1}{2}(\Delta\boldsymbol{X})^T \frac{d^2D}{d\boldsymbol{X}^2}(\Delta\boldsymbol{X}) \quad (7)$$

where $\Delta\boldsymbol{X}$ is the offset from the keypoint candidate. The location of the extreme point can be found as:

$$(\Delta\hat{\boldsymbol{X}}) = -\left[\frac{d^2D}{d\boldsymbol{X}^2}\right]^{-1}\left(\frac{dD}{d\boldsymbol{X}}\right) \quad (8)$$

Thus, at the extreme point:

$$D(\Delta\hat{\boldsymbol{X}}) = D + \frac{1}{2}\left(\frac{dD}{d\boldsymbol{X}}\right)^T (\Delta\hat{\boldsymbol{X}}) \quad (9)$$

If $D(\Delta\hat{\boldsymbol{X}})$ is less than a certain threshold, that means the point has low contrast (i.e. sensitive to noise) and will be rejected (discarded).

Next, keypoint candidates that have high edge responses will be eliminated. Note the Difference of Gaussian function may have a strong response along edges even if the location along the edge is unstable ([16]). In general, a poorly defined keypoint candidate has a large principle curvature across the edge and a small curvature in the perpendicular direction. Consider the Hessian matrix:

$$H = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix} \quad (10)$$

where

$$D_{xx} = \frac{d^2D}{dx^2},\ D_{yy} = \frac{d^2D}{dy^2},\ D_{xy} = \frac{d^2D}{dxdy} \quad (11)$$

Let r be the ratio of the largest and smallest eigenvalues of H, then

$$\frac{[Tr(H)]^2}{Det(H)} < \frac{(r+1)^2}{r} \quad (12)$$

For each keypoint, the corresponding Gaussian smoothed image with the closest scale is selected; and each keypoint is assigned at least one orientation based on local image properties. The gradient magnitude, m, and orientations, θ, are calculated using pixel differences:

$$m = \sqrt{(L(x+1,y) - L(x-1,y))^2 + (L(x,y+1) - L(x,y-1))^2} \quad (13)$$

$$\theta = \tan^{-1}\left(\frac{L(x,y+1) - L(x,y-1)}{L(x+1,y) - L(x-1,y)}\right) \quad (14)$$

An orientation histogram with 36 bins covering 360 degrees of orientations is created at all points in a circular region around the keypoint. Each point in the region added to the histogram is, "weighted by its gradient magnitude and by a Gaussian-weighted circular window with a σ three times the scale of the keypoint" ([13]). The peak in the histogram is detected and its corresponding orientation, along with the orientations within 80% of the peak value, are assigned to the keypoint. This means it is possible to have multiple orientations assigned to one keypoint which improves the stability of matching significantly.

The first three steps of the SIFT algorithm ensure the invariance to location, scale, and rotation by assigning to each keypoint a location, scale, and orientations. The last step ensures the invariance for remaining parameters such as change in illumination. This is done by first sampling the gradient magnitudes and orientations around each keypoint. The Gaussian image selected is based on the scale of keypoint. Next, orientation histograms with 8 bins each are created over 4x4 neighborhood regions from the sampled magnitude and orientation values in a 16x16 neighborhood around each keypoint ([13]). A Gaussian weighting function is used to assign a weight to the magnitude of each sample point. The descriptor is a feature vector consisting of all the values in the histograms. Since there are 4x4 arrays of histograms with 8 bins each, there are 4x4x8 = 128 elements in each feature vector descriptor.

After the keypoints in the target and reference images are found, each keypoint in the reference image is matched against the keypoints in the target image by the nearest neighbor method. That is, the keypoints with the minimum Euclidean distance are matched. To avoid multiple matches

with similar Euclidean distance for the same point, the ratio of the Euclidean distance between the top two matches is considered. If this ratio is greater than 0.8, then the point is discarded and not matched ([13]).

2.2 The DCT (Discrete Cosine Transform)

Consider an image I with size M x N. The two dimensional DCT coefficients of I (i.e. u, v) can be calculated as:

$$X(u,v) = \alpha(u)\alpha(v)\sum_{m=0}^{M-1}\sum_{n=0}^{N-1} I(m,n)\cos\left(\frac{(2m+1)\pi u}{2N}\right)\cos\left(\frac{(2n+1)\pi v}{2M}\right) \quad (15)$$

$$\alpha(u) = \begin{cases} \sqrt{\frac{1}{N}} & \text{for } u = 0 \\ \sqrt{\frac{2}{M}} & \text{for } u = 1,2,\cdots,M-1 \end{cases} \quad (16)$$

$$\alpha(v) = \begin{cases} \sqrt{\frac{1}{N}} & \text{for } v = 0 \\ \sqrt{\frac{2}{N}} & \text{for } v = 1,2,\cdots,M-1 \end{cases} \quad (17)$$

where m and n represent the coordinate of a pixel in the image (m = 1, ..., M; n = 1, ..., N). DCT has strong "energy compaction" property ([16][17]) and has been widely used for image compression.

2.3 The Wavelet Transform

Wavelet decomposition has been used in many image processing applications, especially for image compression. In wavelet transform, a filter bank is often used to split an input signal into two bands. That is, wavelet transform can be considered as passing a signal (or image) through a set of low-pass and high-pass filters. For discrete-time DWT,

$$y(n) = \sum_{k=-\infty}^{\infty} x(k)h(2n-k) \quad (18)$$

where $x(n)$ is the signal; $h(n)$ is the filter impulse response, and $y(n)$ is the result of convolution. When $h(n)$ is low-pass filter, the output gives approximation coefficients; when $h(n)$ is high-pass filter, the output gives detail coefficients. The above procedure can be repeated for each sub-band after level 1 decomposition. Note the low-pass and high-pass filter must be a pair of quadrature mirror filters (i.e., the magnitude response of one filter must be symmetric with the magnitude response of the other filter with respect to $\pi/2$).

In image processing applications, two dimensional wavelet transform needs to be applied; i.e., 1-D transform is performed on x and y directions (rows and columns). For example, level 1 wavelet transform decomposes the image into four sub-bands (low-low, low-high, high-low, high-high).

2.4 The Proposed Neural Network Models

The proposed neural network model is shown in Fig. 2. First of all, the features of image are extracted using SIFT, DCT, or DWT. These feature vectors (i.e. transform coefficients) are the inputs of neural network. Multi-layer feedforward neural networks are employed; the outputs of neural network are registration parameters, such as the angle of rotation (in degree), scaling factors on x and y directions, and translation on x and y directions (in pixel).

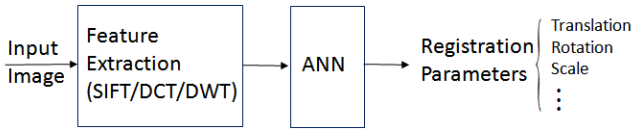


Fig. 2. The proposed neural network model

In DWT, a mother wavelet function $\psi(t)$ serves a basis function. It has been found that Morlet wavelet is closely related to human visual systems ([18]); thus in this research, we choose Morlet function (also called Gabor wavelet) as the mother wavelet function:

$$\psi(t) = ce^{j\omega_0 t}e^{\frac{-t^2}{2}} \quad (19)$$

where c is a constant; and ω_0 is called the central frequency. The waveform of Morlet function is shown in Fig. 3.

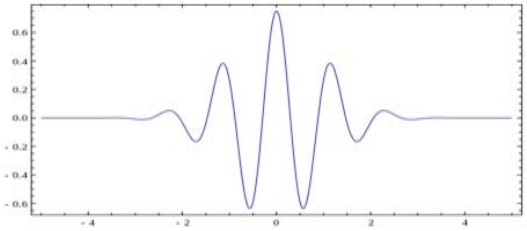


Fig. 3. Morlet wavelet

3 Simulation results

In this section, the performance of the above three models is studied. For SIFT-NN, 50 matched points are chosen as the neural network inputs. For DCT-NN and DWT-NN, 100 coefficients are used as the neural network inputs (i.e., a window size of 10 X 10). All 3 neural network models have 5 outputs which are image registration parameters, i.e., the angle of rotation (in degree), scaling factors on x and y directions, and translation on x and y directions (in pixel). The neural network employed has three hidden layers, with number of hidden neurons of 44, 25, and

10 in each layer. The activation function for hidden neurons is the logistic function; and neurons in the output layer are linear.

The original MRI image used in simulations is shown in Fig. 4. The neural networks are trained with 300 source-target image pairs using back propagation algorithm. Rotation angle ranges from -12˚ to 12˚, translation from -5 to 5 pixels, and scale factor from 0.9 to 1.1 in each direction. The training results are shown in Table 1, including the RMSE (root mean square error) and its range (maximum, minimum). For example, 0.5821 is the RMSE of rotation (in degree) for SIFT-NN, with its minimum value of 0.0747 and maximum value of 3.1876.

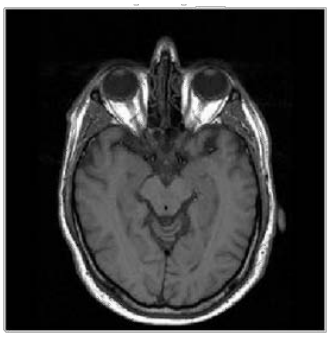


Fig. 4. MRI image for simulation

From Table 1, one can conclude that the novel SIFT-NN model yields the best training results in terms of RMSE for scaling and translation on both x and y directions; however, it yields the worst result for rotation. On the other hand, the DCT-NN performs the best on rotation, with moderate performance on scaling and translation. DWT-NN performs better than SIFT-NN on rotation, and is generally competitive with DCT-NN on scaling and translation.

Table 1. Training results

	SIFT-NN	DCT-NN	DWT-NN
Rotation	[0.0747, 3.1876] 0.5821	[0.0466, 1.8715] 0.4056	[0.0024, 3.6736] 0.4746
Scaling on *x*	[0.0002, 0.0932] 0.0023	[0.0004, 0.0984] 0.0032	[0.0001, 0.1040] 0.0033
Scaling on *y*	[0.0005, 0.0961] 0.0025	[0.0001, 0.1032] 0.0028	[0.00001, 0.0924] 0.0031
Translation *x*	[0.0038, 3.3261] 0.1398	[0.0017, 1.7980] 0.2183	[0.0067, 3.5945] 0.1779
Translation *y*	[0.0057, 3.5681] 0.1208	[0.0007, 1.7820] 0.1326	[0.0067, 4.8600] 0.1618

After the neural network is fully trained, the three hybrid models are tested with 25 source-target image pairs. The parameters of the target image are in the same range as the training sets. The testing results are shown in Table 2.

From Table 2, one can conclude that the DWT-NN yields the best testing results for scaling on both *x* and *y* directions while the DCT-NN model performs the best on rotation and translation. The SIFT-NN yields the moderate performance in most cases, except rotation.

Table 2. Testing results

	SIFT-NN	DCT-NN	DWT-NN
Rotation	[0.1135, 9.0103] 1.3415	[0.0233, 1.6972] 0.3148	[0.0678, 3.6755] 0.3873
Scaling on *x*	[0.0034, 0.0913] 0.0094	[0.0007, 0.0894] 0.01325	[0.0023, 0.063] 0.0057
Scaling on *y*	[0.0003, 0.0677] 0.0094	[0.0022, 0.0911] 0.0807	[0.0029, 0.0632] 0.0086
Translation *x*	[0.1247, 4.0535] 0.3026	[0.0238, 1.7761] 0.2343	[0.0215, 1.498] 0.43
Translation *y*	[0.0652, 4.2105] 0.2962	[0.0043, 1.1602] 0.2548	[0.0816, 1.7649] 0.4991

4 Conclusions

Image registration has widely been used in medical imaging and remote sensing. It relates multiple images of the same subject that were taken either at different time, or from different sensors, or from different points of views with a single image and coordinate system.

In this study, the registration of MRI images is investigated with three different hybrid models using artificial neural networks. Scale invariant feature transform (SIFT), discrete cosine transform (DCT), and discrete wavelet transform (DWT) are employed in each model, respectively. Two new neural network based models are proposed, i.e., SIFT-NN and DWT-NN. The performance of the three proposed models are studied and compared via computer simulations.

5 References

[1] B. Zitova, J. Flusser, "Image registration methods: a survey", Image and Vision Computing 21, 2003, pp. 977-1000.

[2] A. Noble, "Finding corners," Image and Vision Computing 6, 1988, pp. 121–128.

[3] N. Vujovic, D. Brzakovic, "Establishing the correspondence between control points in pairs of mammographic images," IEEE Transactions on Image Processing 6, 1997, pp 1388–1399.

[4] B. Manjunath, C. Shekhar, R. Chellapa, "A new approach to image feature detection with applications," Pattern Recognition 29, 1996, pp. 627–640.

[5] H. Alhichri, M. Kamel, "Virtual Circles: a new set of features for fast image registration," Pattern Recognition Letters 24, 2003, pp. 1181-1190.

[6] S. Damas, O. Cordon, J. Santamaria, "Medical Image Registration Using Evolutionary Computation: An Experimental Survey," IEEE Computational Intelligence Magazine, 2011, pp. 26-42.

[7] A. Abche, F. Yaacoub, A. Maalouf, E. Karam, "Image Registration based on Neural Network and Fourier Transform," Proceedings of the 28th IEE EMBS Annual International Conference, 2006, pp. 4803-4806.

[8] I. Elhanany, M. Sheinfeld, A. Beck, Y. Kadmon, N. Tal, D. Tirosh, "Robust Image Registration Based on Feedforward Neural Networks," IEE International Conference on Systems, Man, and Cybernetics, 2000, pp. 1507-1511.

[9] S. Wang, S. Lei, F. Chang, "Image registration Based on Neural Network," Proceedings of the 5th International Conference on Information Technology and Application in Biomedicine, 2008, pp. 74-77.

[10] J. Wu, J. Xie, "Zernike Moment-based Image Registration Scheme Utilizing Feedforward Neural Networks," Proceedings of the 5th World Congress on Intelligent Control and Automation, 2004, pp. 4046-4048.

[11] H. Sarnel, Y. Senol, D. Sagirlibas, "Accurate and Robust Image Registration Based on Radial Basis Neural Networks," 23rd International Symposium on Computer and Information Sciences, 2008, pp. 1-5.

[12] D. Lowe, "Object Recognition from Local Scale-Invariant Features," Proc. of the International Conference on Computer Vision, 1999, pp. 1150-1157.

[13] D. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," International Journal of Computer Vision, 2004, pp. 91-110.

[14] T. Hossain, S. Teng, G. Lu, M. Lackmann, "An Enhancement to SIFT-Based Techniques for Image Registration," 2010 Digital Image Computing: Techniques and Applications, pp. 166-171.

[15] I. El Rube, M. Sharks, A. Salem, "Image Registration Based on Multi-Scale SIFT for Remote Sensing Images," 3rd International Conference on Signal Processing and Communication Systems, 2009, pp. 1-5.

[16] N. Ahmed, T. Natarajan, K. Rao, "Discrete Cosine Transform," IEEE Transactions on Computers, January 1974, pp. 90-93.

[17] S. Khayam, "The Discrete Cosine Transform (DCT): Theory and Application," March 2003.

[18] N. Yamagishi, D. Callan, N. Goda, et. al., "Attentional modulation of oscillatory activity in human visual cortex," NeuroImage, Vol. 20, 2003, pp. 98 –113

Apply an Autonomous System for Identification of Some Living Activities in Bathroom

Fu-Hua Chou[1], Sue-Ping Yuan[2]
[1]Electronic Engineering, Chien Hsin University of Science and Technology, Taoyuan, Taiwan (R.O.C)
[2]CFO, Taiwan Implant Technology Company, Kaohsiung, Taiwan (R.O.C)

Abstract - *This paper presented an Intelligent Pervasive Care System for independent living aging peoples to provide the living care at home for all day long with three features. First of all is the designed structure that empowers it with the ability to infer the occurring events, actions, and purpose of the cared family. The second is the cooperating mechanism between multiple types of sensor agents. The third is an implementation of living activity detection and identification that uses non-vision and non-wear pressure sensor to sense and identify some activities in bathroom especially. There four activities, brushing teeth, rubbing the back, hula hooping, and washing face, testing for identification correction is taken in the final section, and the average correct rate based on those four build models is 93.75%.*

Keywords：Intelligent Homecare Systems, Living Activities Detection and Identification, Pressure floor.

1 Introduction

Under modern medical technology, the high-age population is growing up rapidly. The population of the whole world has going to aging society nowadays and future. According to the population evolution research report #14 from the Industrial Technology Research Institute (ITRI) of R.O.C., the rate of population that over 65 years old will be over 20 percentages at 2025 in Taiwan. In the same time, this proportion is just equal to the rate of population under 14 years old. Based on this fact, the elder compatriots may lack for care from their families in those moments. Therefore, developing autonomous home or health care mechanism (systems) for independent living elder people is a more important issue than past.

Based on the related research reports from Japan, Europe and USA, the average length of human life is promoted because of the fast growing in the fields of people nutrition, health care and the medical technology. Now, the average length of Male's life is 78 years old, and the length of female can be up to 82 years old. In other words, people will take about 14 years life after his retirement. According to the activity ability, we divide this retirement duration into three phases. In first phase, people can be cared her life by himself (about 7 to 9 years long). In the second phase, people must be nursing cause by body aging (about 3.5 to 4 years long). And before the end of life, people need the urgent medical care (about 1 to 1.5 years long).

In the aging of population circumstances, the aging-people care system is developed and forwards to personalized, autonomous, pervasive, remotely, and daylong [2,13,14]. Today, the ubiquitous computing was developed rapidly, and that technology application for the high-age people homecare systems is a more important issue [7,8]. Those systems can perform the complicated homecare task based on the competition to contribution between each heterogeneous and independent computing agent. Based on the above analysis, we realize three kinds of caring mode for those elder people. First mode is the live caring mode, digital care techniques about peoples living in home is the development issues [10,16,18]. Second mode is the health caring mode, in which developing medical equipments make sick peoples can be cared in home is the key issues [11,12,13]. Final mode is clinical caring mode, developing clinical equipments for assisting the therapeutics in the hospital is the key issues. Peoples in the first mode still own the ability to living alone. But they only need some suitable assistance or to remind for some important things in the home living. In addition, they also need immediate helps and to inform some particular peoples or official units when some accidents occurring in home [14]. Based on this need, this paper designs an Intelligent Pervasive Care System (also called IPCS) for independent living aging peoples to provide the living care at home for all day long [1,4,5].

Three features of IPCS were presented in this paper. First is the designed structure of the IPCS that empower the ability to infer the occurring events, actions, and purpose of the cared aging family based on the personalized behavior model. Second is the cooperating mechanism between multiple types of sensor agents. Those agents manage some sensing facilities that own non-vision and non-direct body touch abilities. Third is an implementation of living activity detection and identification that use non-vision and non-wear pressure sensor, and it can sense and identify some activities in bathroom especially. The personalized behavior model is built by the analysis of the regular activities exhibited in home living, and it is also presented in this paper. Based on this model, IPCS can identify the anomalistic activities through the competition mechanism upon the multi-sensor agents' subsystem [6][7].

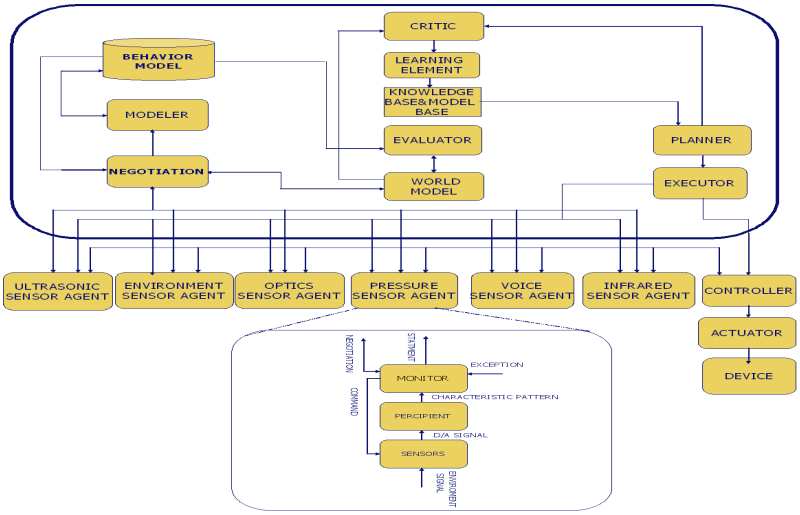


Figure 1 Intelligent Pervasive Care System

Then ICPS can immediately help and inform some particular peoples or official units through the Internet when the home accident is confirmed. Based on the designed architecture of IPCS's multi-sensor agents' subsystem, it empowered the ability to solve the uncertainty and unreliability problems even under the context without visual and directive touch sensing signal.

2 Intelligent Pervasive Care System

Now IPCS's architecture is shown in Figure 1. It consists five parts: multi-sensor agents, negotiation and modeler mechanism, evaluator and planner mechanism, executor, and controller [1]. This paper focuses on the multi-types sensing subsystem, and negotiation and modeler mechanism, and both will be presented in following sections. In general, the structural design of homecare distributed system can be divided into two kinds: partial distributed and fully distributed. The structure of partial distributed system has two operation units: administrator and service. The responsibility of administrator is to record and deal with the communication of information between agents. The operations of service are to progress management and execute the various negotiation processes. In detail, the service unit owns the power of negotiation between related agent, and the ability to perform some specified operations based on its knowledge about the working environment, such as, equipments control, specified object monitoring, and event type analysis. The ability to perform various negotiations based on integrated environment knowledge is empowered to all operation units in the structure of fully distributed [17].

2.1 Multi-Sensor Agents Subsystem

All sensing devices used by IPCS are commercialized, low cost, active but non-touch body, and non-visual detecting device. Such as body infrared sensor, pressure sensor, ultrasonic sensor, voice sensor, optics sensor, and other environment sensing device. The operations of all detecting agents are managed by the agent's community in IPCS. Each sensor agent0 can send some specific message via the community's negotiation mechanism. Some of message can commission other agent to verify the occurrence of specified event by its sensed evidence. Based on this negotiation mechanism, each sensing agent can verify the specified event by seeking specific symptom signal as occurring evidence. So the operation of agent's community for all sensing devices is divided into four. First, the sensor agent carries on the collection and analysis of each environmental signal. Second, convey and upgrade to the modeler when the specified symptom is confirmed. Third, accept the commission from other sensor agent to seek occurring evidence of specific symptom. Final, query related agent about some unsure symptoms and verify this return result. Figure 1 shows the structure of the single sensor agent in the community. It contains physical sensor, percipient, and monitor component.

The sensor entity is responsible for gathering the environmental signals, and transmitting them to the percipient component. It processes the pure signal to produce the characteristic value, and then transmits it to the monitor subsystem.

2.1.1 Monitor Subsystem

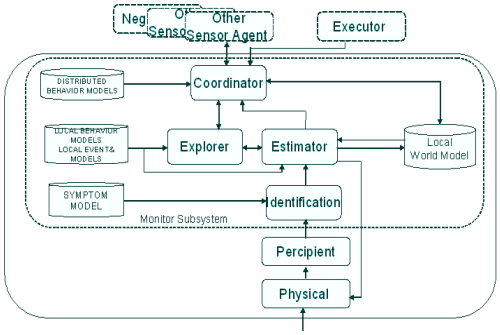


Figure 2 Structure of Monitor Subsystem and its interface

Monitor is the common member in all kinds of sensing agent. It identifies the entered sensing data, justifies the significance of identified symptoms, traces the change of event, and finds other appropriable sensor agents as partners for verifying some uncertain events. Its structure and interfacing is shown in Figure 2.

Monitor has designed to perform the following tasks. Firstly, it identifies the highest promising symptom by comparing the sensed characteristic, from its physical sensor, with the built symptom model. The second task is to analyze the similarity degree between two promising events, one is consisted by the serial identified symptoms based on sensed

signals, and another is known event built in the local behavior/event model. The third task is to evaluate the trustiness of the new identified event. The next task is to update the event with a high trustiness to the negotiation. Finally, monitor must accept another verification tasks about some specified events, or trace some new symptoms from the commission by the negotiation, executor, or other sensor agents.

2.1.2 Estimator

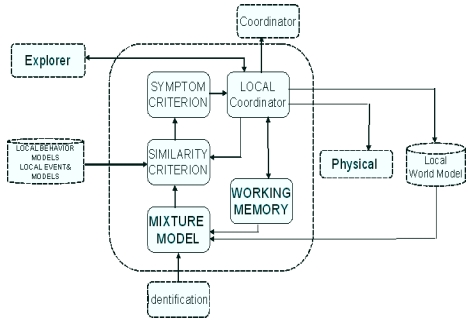


Figure 3 Structure of Estimator and its interface

Estimator is the key element of monitor subsystem. It is consisted by mixture model, similarity criterion, symptom criterion, and local coordinator. Its structure and interfacing are shown in Figure 3. The major task of estimator is to identify the highest promising symptom by comparing the sensed characteristics, from its physical sensor, with the built symptom model. And the second task is to evaluate the similarity degree of the promising event based on the known event that built in the local behavior/event model.

2.1.3 Coordinator

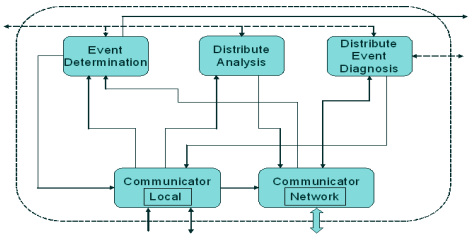


Figure 4 Structure of Coordinator and its interface

Coordinator is the communication window of a sensor agent. It is consisted by event determination, distribute analysis, and distribute event diagnosis. Its structure and interfacing is shown in Figure 4. The major task of coordinator is to record verifying symptom and justify verified symptoms in qualifying. Another important task assigned to the coordinator is to find out some partnered sensor agent from the sensing network for assisting in verifying some events, which are not be identified by this sensor agent.

2.2 Negotiation Subsystem

In order to correctly identify the status of cared person, ICPS uses many types of sensor agents, in each contains an array with distributed sensors to gather widely signals from its operating environment. In this progress, there are two situations often happened. Firstly, the information gathered from single sensor agent is not enough to analyze the state of cared person. In this moment, another sensor agent is required to be joined for identifying another type of symptom, and to offer its justification about commissioned event verification task. Secondly, when the verified symptoms from two different sensor agents are conflict, this situation will harm the performance of ICPS. Therefore, a negotiation subsystem is designed to solve this problem. Figure 5 shows its structure. There are six elements included into the negotiation, there are communicator, administrator, distribute event diagnosis, synchronicity, collision detection, and critical analysis.

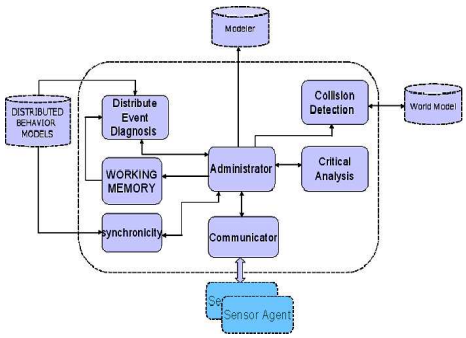


Figure 5 Structure of Negotiation and its interface

2.3 Modeler Subsystem and Behavior Model

The modeler subsystem infers the normal behavior of the cared person; this behavior model can analyze the state of people and predict next action. Otherwise, it can be used to evaluate the rate of accuracy about the behavior model. It will revise the behavior model while mistake occurring. The modeler utilizes fuzzy logic inference mechanism to solve the uncertainty in the environment.

To achieving the intelligent and independent purpose of looking after in the environment of intelligent surrounds, the behavior correctly distinguishing of cared person is an important factor. The architecture of behavior model consists five layers: purpose layer (P), behavior layer (B), action layer (A), event layer (E), and symptom layer (S). In addition, each layer has various rules to infer the possibility of each element. Finally, this model can emerge the purpose of behavior taken by the cared person on daily life. .

2.3.1 Purpose Layer

The purpose layer is the set of state element in top layer of behavior model. These elements' design are according to the vague intentions, they are generalized from the human life in a house. Such as rest, washing hands, and washing face, brushing teeth, etc.

2.3.2 Behavior Layer

This layer defines the single and clear behavior in daily life separately in these elements, such that each element maps to the single element of purpose layer. In other word, each element of purpose would map to multi element of behavior. For example, the person wake up in the morning, then wash

immediately. So the mental state will present the intention of "clean" on the purpose layer. Under this situation, the behavior layer will contain elements such as washing face, brushing teeth, etc.

2.3.3 Action Layer

The meaningful behavior would be ordered by series of time at many actions. This order is important for causality between each action. If the behavior model of cared person is created by traditional tree structure then it can't represent the relation of series. So, this paper adds two demonstrations: (1) rule-based to represent theses relation of element; (2) link of time series to represent these elements for relation of order in architecture of behavior model.

2.3.4 Event Layer

This layer causes the variation of model by the action of cared person. Each element would relate to multiple sensors from environment. These elements are different from the pattern of symptom. This event model contains the similarity of promising event and the weight associated with that event.

2.3.5 Symptom Layer

The symptom layer records the various status of sensor that expresses difference meaning for signal characteristic. Each element represents two meaning about specific place and specific characteristic pattern. It uses two crisp attributed values to produce the happening possibility of symptom by fuzzy inference engine. One is the similarity of symptom and another is the length difference of symptom.

3 Living Activity Detection and Identification

The specific family specific activity identification system means to train the activity modes of special users and add the features of the specific activities. Thus, in the private and specific environments including a bedroom, a bathroom and a toilet which are non-public spaces, the system shows well. The key to success of specific activity identification system results from identifying those rhythmic characteristics of users correctly. By taking use of the differences in every activity, it can recognize the correct activity and it is so called active identification. To face the need of senior family's activity identification in smart home care environments, this identification system with an activity characteristic model base to satisfy the need of different services still is ideal designs.

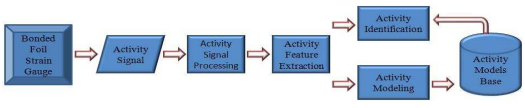


Figure 6 Living Activity Detection and Identification

Figure 6 presents the structure of the living activity detection and identification. The initial step is activity signal receiving by the pressure sensing floor with bonded foil strain gauges. The following step is activity signal processing, it purifies and enhances the activity signals from the environmental noise. The third step is the activity feature extraction, it extracts features about the activity frequencies and it's emerged sequence in time. There are two different processes following the activity feature extraction, one is the activity models building and another is the unknown activity identification. The activity modeling step is to construct a probability model for each kind of living activity based on the extracted features of this activity, and stores it into the activity models base for using by the activity identification. There is a pre-processing sub-step on the front of the activity modeling, the vector quantization, which is mainly to reduce the great numbers of calculation when in the follow up process of building models and identifying. So there use the vector quantization to simplify the activity signal characteristics. The final step in this process is the activity identification, it receives the extracted feature from an unknown activity, and compares those features with the activity characteristics of each activity model in the activity model, one after another until to find out the most matched activity model. The represented activity by the most matched activity model just is the identified outcome of this living activity identification.

3.1 Activity Signal Processing

The major purpose of the activity signal processing is to translate analogy signal produced from the activities of the senior families into a processed, purified and enhanced digital signal and that it can be utilized in building an activity model later. This kind of process mainly includes three parts of processing steps, includes sampling, end point detection and Hamming window, shown in figure 7. The first is the activity signals sampling, noise eliminating and purifying. The second is to detect the timing of meaningful activity signals emerged. The pressure sensing gauge is always on, but the cared senior family is not stand on this gauge all the time. The final is to enhance those meaningful activity signals and form them into the signal frames.

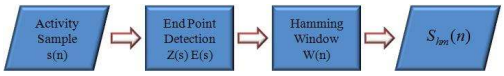


Figure 7 Activity Signal Processing

3.1.1 Sampling

Sampling process translates an activity analogy signal into digital data [3]. The sampling frequency used in this paper is 1000Hz, means there are 1000 activity signal samples per second. The variety of an activity signal is very wide during the time of sensed, and there imply many different frequency characteristics in whole section of the sampled activity data. So it needs to separate a whole section of activity data into several frames, then the variety of frequency characteristics in every frame of activity data are approximated linear. The continuously variety of sampled activity characteristic was broken at the beginning and ending point in each frame when a whole section of activity signal separated into several frames. Those two endpoints in each frame make a huge frequency response when they are translated to frequency domain. To avoid previous distortion it needs to overlap some part of the

sampled activity data between neighboring frames. The overlap range may be up to half of a frame.

3.1.2 End Point Detection

The end point detection is to find out the starting catch point of the human activity signal from the continuously sensing data [3], that not only includes activity signal also has biases signal and noises. There are two differences between the activity signal and biases (or noises). One is the activity signal owns oscillation amplitudes that is periodically changed crossing over a non-zero base line. And another is the energy that is coexisted with the activity signal. So there exist the zero-crossing rate, energy, inverse function and the variables of the spectrum methods can be used to detect the activity signal starting catch point. However, under the consideration about the time cost in detection computing, the zero-crossing rate coordinated with the energy contour method is used in this paper. The zero-crossing rate represents the times of the variety value of sampled data crossing through the zero point in each frame. Supposing that one signal passed through the zero point nine times, the zero-crossing rate of the signal is nine.

As analyzing activities of human, the zero-crossing rate of activity signals is low. If there is no activity signal, the zero-crossing rate is high instead. However, though human do not movement, there still exist noise. Because of the noise, the zero-crossing rate is high as well. There can't distinguish the conditions whether there is activity or not. Therefore, it is not enough if there only use the way of the zero-crossing rate. It also needs the energy contour. The energy contour expresses the intensity in every frame. Because the wave amplitudes of some parts of activities are bigger than others, so there can set a minimal activity signal amplitude as the critical value. It will be thought to have activity as long as the value of the energy in frames is over the critical value. Another way is that supposing there are no activities in the several frames before a frame with a section of activities, finding out the frames. And the values of energy of the frames are much bigger than the values of those non-activity frames. Thus it can be thought that there are activities in the section. The sum of the intensity of frames is usually represented by a logarithmic function. Taking the maximum intensity frame as the starting point and then compare it with the zero–crossing rate to make sure if there are activity actually.

3.1.3 Hamming Window

The discontinuity on both sides of frames will lead to a huge breakage of the continuity of the active spectrum in the frequency domain. To revise this breakage, making frames into windows to depress the signals on both sides of frames down and highlight the main signals in the middle of frames. With the overlapping of frames before, it can make the changes of frames on both sides have continuity effects. The sampled signal data processed by the Hamming window can be considered as the products of the activity signal processing pre-procedure. Following process for each product of data frame is the activity feature extraction. [3]

3.2 Activity Feature Extraction

The daily activity of family in home is sensed by the bonded foil strain gauge to become the family living activity signal, and then it is digitized, purified and enhanced by the activity signal processing pre-procedure. The main purpose in the activity feature extraction is to find out the frequency characteristics of living activity. That includes the characteristics of the distribution of frequency embedded in the activity signal, the rhythm of activity frequency variation, and the change of activity signal frequency in time. These extracted characteristics are the important criterions for distinguishing between different activities. Therefore, the activity feature extraction processing includes three main stages that are the frequency domain transfer, the activity rhythm analysis and the activity feature streams analysis. The relation of those three stages is shown in figure 5.

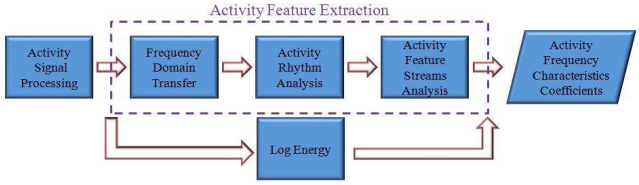


Figure 8 Processes of Activity Feature Extraction

3.2.1 Frequency Domain Transfer

In this stage, the main process is to find out the energy distribution of the activity signals in each frequency band. Different living activity shows different frequency and different energy distribution. Because of the energy distribution of activity signals in the time domain is not easily obtained, it translates the time domain into the frequency domain. In this paper, fast Fourier transform is used in the transform. [9]

3.2.2 Activity Rhythm Analysis

People take many different kind of living activity in home, and the difference between each activity is the rhythm of activity. Each kind of living activity owns a specific frequency distribution, which presents the way of frequency change by the response on different frequency bands. Activity rhythm analysis means to look for the degree of the response on different frequency of a sensed activity signals. In this paper, the process of the activity rhythm analysis is done by the Mel-scale frequency transform. The frequency distribution of the activity signals passes through a triangular digital band pass filter designed by the Mel-scale frequency to obtain the activity rhythm as the frequency characteristics of the kind of living activity.

3.2.3 Activity Feature Streams Analysis

For distinguishing between different kinds of living activity, the time sequence of activity frequency changing is another important characteristic besides the activity rhythm. After finding out the response in different frequency bands, there need to get the time varied characteristics of the change in frequency for these different activities, and take them as the living activity features. So the discrete cosine transform, DCT, is used to calculate the time varied characteristic value in the

change of the home activity signal frequencies.

It is insufficient for accurately to identify all kind of living activity, if that only use the Mel-scale frequency cepstral coefficients. Therefore, it must be added to the other parameters. In this paper, the logarithmic energy of signals is added except the 12-order coefficients of the Mel-scale frequency cepstral. Finally, there are 39-order coefficients to represent the feature of a frame of activity data that is composed by the previous 13-order characteristic coefficients and its first and second orders derivative coefficients. [

3.3 Activity Modeling and Identification

Each family member in daily life has his or her habits which are also the basic of mutual recognition. When the home activity pressure signal goes through the activity feature extraction and is obtained the activity frequency characteristics coefficients, the next step is activity identification processing. The activity identification processing can be divided into two parts, modeling and identifying. The modeling process is mainly to classify the characteristic coefficients of activity signals in accordance with actions and build activity models of the specific home activities. Identifying process is to compare the unknown activity frequency characteristics coefficients with all possible activity models and find out the most similar model. The activity condition of the most similar model is the identification result. Activity models in this paper are built by the Gaussian mixture mode. The Gaussian mixture model is a high-dimension probability density function. Through the high-dimension probability density function, it can be represented its variability by the way of probability statistics. The Gaussian mixture model is taken as the linear combination of the high-dimension probability density function. It can approximate any kinds of density function as long as there is a sufficient number of mixing vector components. Activity characteristics are usually with a smooth curve in probability density function. Thus, a limited number of the Gaussian density function is enough to form a density function of activity characteristics which is nearly smooth.

3.3.1 Gaussian Mixture Model

A complete Gaussian mixture density model (G) can be represented by the Gaussian density functions matrix (*P*), the means matrix (*V*), the covariances matrix (*Δ*), and the mixed weights matrix (*B*). And there can use it as a collection of these parameters.

$$G = \{P, V, \Delta, B\} \quad (1)$$

3.3.2 Identification of Activities

In the activity identifying process, there use a series of Gaussian mixture model, $G_1, G_2, ..., G_S$, to represent a series of activities, $S = \{1, 2, ..., S\}$. To a section of test activity signals, $X = \{\vec{x}_1, \vec{x}_2, ..., \vec{x}_t\}$, the purpose of activity identifying is to find out the model with the biggest probability in this series of activity models. It is shown as follow:

$$\hat{S} = \arg\max_{1 \le k \le S} p(X \mid G_k) \quad (2)$$

4 Experimental Settings and Results

The kernel element of the pressure signal acquisition floor used in following experiments is the bonded foil strain gauge (BFSG) pressure sensor [19]. The pressure signal acquisition floor was created by acrylic material with the size of 50cmX50cm. BFSG means that four foil gauges are bonded using industrial adhesives on a stainless beam which is flexible material to response the change in pressure. And the foil strain gauge sensing signal is joined together by a wheatstone bridge circuit because the strain gauge produced a corresponding change in resistance, and those change can be measured with a data acquisition.

In this paper, there focus on four living activities' processes of modeling and identification. They are: brushing teeth, rubbing the back, hula hooping, and washing face. There use 100 activity data to building activity model for each kind of activity, and another 100 activity data used to testing the correctness ratio of each identification model. The following figure 9 to 12 show the sensed pressure signal pattern of those four living activities. Vertical axis represents the sensed voltage value, and the horizontal axis represents the digitized sample points.

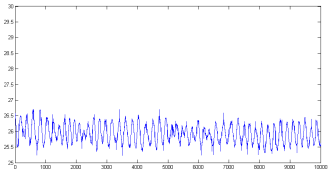


Figure 9 Signal pattern of brushing teeth

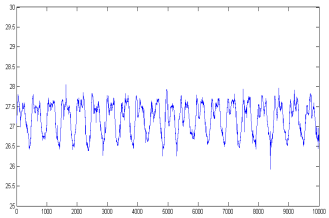


Figure 10 Signal pattern of rubbing the back

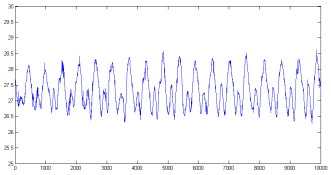


Figure 11 Signal pattern of hula hooping

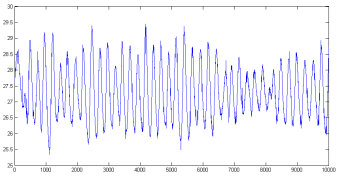


Figure 12 Signal pattern of washing face

Before the activity identification, some unknown living activity characteristics were extracted from its signal pattern.

Then the process compares the unknown activity characteristics one by one with four kinds of activity models for looking for the most similar activity model. In this paper, the identification error was composed by two errors that are the false reject (FR) and false acceptance (FA). The false reject means the ratio of correct activity signal misidentification to incorrect activity model in the system. The false acceptance is the ratio of incorrect activity signal identification to correct activity model in the system. Each activity model in the activity signal comparing process sets a gate value in order to ensure the quality of the identification. If the gate value is set too high that will lead FR to be high so does another situation that the gate value is set too low. These two situations will lead to the decrease of the correctness of the system identification. In this paper, the value of the gate value is set as the intersection point of the two curves of FR and FA. This point can make the equal error rate, EER, reach the minimum point. The definition of FA and FR are shown as follows:

$$FA = \frac{Amount\ of\ the\ wrong\ identifying\ in\ positive\ example}{Amount\ of\ the\ whole\ positive\ example} \quad (3)$$

$$FR = \frac{Amount\ of\ the\ wrong\ identifying\ in\ negative\ example}{Amount\ of\ the\ whole\ positive\ negative\ example} \quad (4)$$

And the EER is defined as follows：

$$EER = \frac{1}{2}(FA+FR) \quad (5)$$

Correctness rate (CR) is：

$$CR = \frac{Amount\ of\ the\ correctly\ identifying\ in\ positive\ example}{Amount\ of\ the\ whole\ positive\ example} \quad (6)$$

The identification correctness ratio of four living activities testing processes, brushing teeth, rubbing the back, hula hooping, and washing face, are showing in table 1.

Table 1 Identification outcomes comparison between activities

	CR	FR	FA	EER
brushing teeth	100%	0%	1.67%	0.84%
rubbing the back	95%	5%	1.67%	3.34%
hula hooping	90%	10%	0%	5%
washing face	90%	10%	1.67%	5.84%
average	93.75%	6.25%	1.25%	3.75%

5 Conclusion

This paper designs an Intelligent Pervasive Care System for independent living aging peoples to provide the living care at home for all day long. This system designed based on the structure of autonomous systems, and used multiple interlaced and non-visual sensors to recognize the behavior of concerned person in independent and concurrent. There four activities, brushing teeth, rubbing the back, hula hooping, and washing face, testing for identification correction is taken in the final section, and the average correct rate based on those four build models is 93.75%. Based on this system, the behavior tracking model can recognize the recent status of the cared person. So ICPS can immediately help and inform some particular peoples or official units through the Internet when the home accident is confirmed.

6 References

[1] Fu-Hua Chou, Chyong-Mei Chen and Kun-Feng Liu, "A Platform of Informationally Structured Space for the Autonomous Homecare System", Proceedings of the IEEE 2009 Workshop on Robotic Intelligence in Informationally Structured Space, Nashville Tennessee, USA, March 30 - April 2, 2009, p.114~120, NSC 97-2221-E-231-007 (EI)

[2] Fu-Hua Chou ,C.S. Ho, "HASLEARN: A Highly Autonomous System with Learning Behavior", Proceedings of IEEE International Conference on Systems, Man & Cybernetics (SMC-94), pp. 108-113, 10.1994

[3] Wai C. Chu, Speech coding algorithms: foundation and evolution of standardized coders, John Wiley & Sons, Inc., New Jersey, 2003

[4] Fu-Hua Chou, "Distributed Multistrategy Learning in Intelligent Highly Autonomous Systems", Journal of Ching-Yun Institute of Technology, Vol. 22, No.1, pp.65-70, 2002.6, NSC90-2212-E-231-007.

[5] Fu-Hua Chou, Chia-Min Wu & Che-Wun Chiou, "Independent Living Behavior Model of Elder Family for Autonomous Homecare Systems", Proceedings of 2006 International Conference on Business and Information(BAI2006), Pan Pacific Hotel, Singapore, 6180, 12-14.7.2006

[6] A. P. Dempster, N. M. Laird and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," Journal of the Royal Statistical Society, Series B, vol. 39, no. 1, pp. 1-38, 1977

[7] Hani Hagras,Victor Callaghan, Martin Colley, Graham Clarke, "Creating an Ambient-Intelligence Environment Using Embedded Agents," *IEEE, Intelligent Systems* , Volume 19, Issue 6, pp. 12–20, Nov-Dec,2004

[8] Toshio Hori, Yoshifumi Nishida, Hiroshi Aizawaand & Nobuyuki Yamasaki, "Networked Sensors for Monitoring Human Behavior," *IEEE Symposium on Computational Intelligence in Robotics and Automation*, pp. 900 – 905, July 2003.

[9] Cooley James, Lewis Peter, Welch Peter, "The Fast Fourier Transform and Its Applications,", IEEE Transactions on Education, vol.12, no.1, pp.27-34, March 1969

[10] Elizabeth D. Mynatt, Anne Sophie, Melenhorst, Arthur D. Fisk, and Wendy A. Rogers, "Aware Technologies for Aging in Place: Understanding User Needs and Attitudes," *IEEE Pervasive Computing*, Vol. 03, Issue 2, pp.36-41, April-June 2004.

[11] Mitsuhiro Ogawa, Shiroh Ochial, Ken Shoji, Minori Nishihara and Tatsuo Togawa, "An attempt of monitoring daily activities at home," *IEEE EMBS International Conference*, July, pp. 786 – 788, 2000

[12] Mitsuhiro Ogawa, Ryoji Suzuki, Sakuko Otake, Takeshi Izutsu, Tatsuo Togawa and Tsutomu Iwaya, "Long term remote behavioral monitoring of elderly by using sensors installed in ordinary housese," *IEEE EMBS International Conference*, May, pp. 233 – 235, 2002.

[13] Mark Perry, Alan Dowdall, Lorna Lines, and Kate Hone, "Multimodal and Ubiquitous Computing Systems: Supporting Independent-Living Older Users," *IEEE Transactions on Information Technology in Biomedicine*, Vol. 8, No. 3, pp.258-270, September 2004

[14] Matthai Philipose, Kenneth P. Fishkin, and Mike Perkowitz, Donald J. Patterson, Dieter Fox, and Henry Kautz Dirk, Hähnel, "Inferring Activities from interactions with Objects," *IEEE Pervasive Computing*, Vol. 03, Issue 4, pp.50-57, October-December 2004.

[15] L. R. Rabiner, A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition, in Proceeding of the IEEE, Volume, 77, No. 2, pp. 257-286, February. 1989.

[16] Thomas Schetter , Mark Campbell, "Derek Surka, Multiple agent-based autonomy for satellite constellations," *Artificial Intelligence*, p.p. 147–180, 2003.

[17] Chia-Min Wu, *Behavior Tracking Model of the Home Cared Persons*, M.S. thesis, Department of Electronic Engineering, Ching Yun University, Jung-Li, Taiwan, July 2006

[18] Akifumi Yamaguchi, Mitsuhiro Ogawa, Toshiyo Tamura & Tatsuo Togawa, "Monitoring behavior in the home using positioning sensors," *IEEE Medicine and Biology Society*, Vol. 20, Issue 4, pp. 1977 – 1979 , 1998.

[19] The Strain Gauge, "http://www.sensorland.com/HowPage002.html"

A comparative analysis for mobile and computer-based aphasia therapy

A. Kamuela Parker[1], **B. Heui-Seok Lim**[2]
[1]Department of Computer Science Education, Korea University, Anam-dong Seongbukgu, Seoul, South Korea
[2]Department of Computer Science Information and Engineering, Korea University, Anam-dong Seongbukgu, Seoul, South, Korea

Abstract *- A developing movement in the administration of aphasia therapy is the utilization of computer-based treatments and mobile-based treatments, with limited amount of tentative writings that prove the benefits of a computer and mobile approach. As researchers implement treatments from the computer and through mobile devices, it is essential to promote research that would progress treatments in a programmatic way that are significant to clinical trials. This paper examines the authenticity of two projects designed to aid in programmatic research on an aphasia treatment: computer-based software named MossTalk Words, and Constant Therapy an iPad based application. Results show the practicality and initial achievement of a structured and continuous therapy in providing evidence of the implementations of autonomous treatments.*

Keywords: *Aphasia, MossTalk Words, Constant Therapy*

1 Introduction

1.1 MossTalk Words

MossTalk Words was developed for those with aphasia whom possess a deficiency in word retrieval. Designed by researchers and clinicians, the software was proposed for a clinical environment as well as by patients in an autonomous environment. It offers practice in word comprehension and construction using cues and feedback. The two main treatment modules were designed after treatments that are used by clinicians that have been effective in experimental studies [1, 2](Figure1).

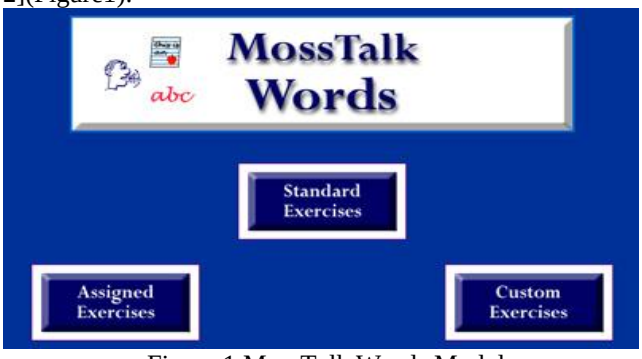


Figure 1 MossTalk Words Modules

1.2 Constant Therapy

Constant Therapy is a mobile rehabilitation application designed for brain rehabilitation that incorporates evidence based therapy in the recovery of aphasia patients. Therapy tasks were divided into language and cognitive tasks and were created based on a review of evidenced based treatment recommendations [3] (Figure 2). Under each main task exist sub-tasks, for example, Language consisted of, the following tasks; Naming Therapy: phoneme identification, category matching, feature matching rhyme judgment, syllable identification, and picture naming [3].

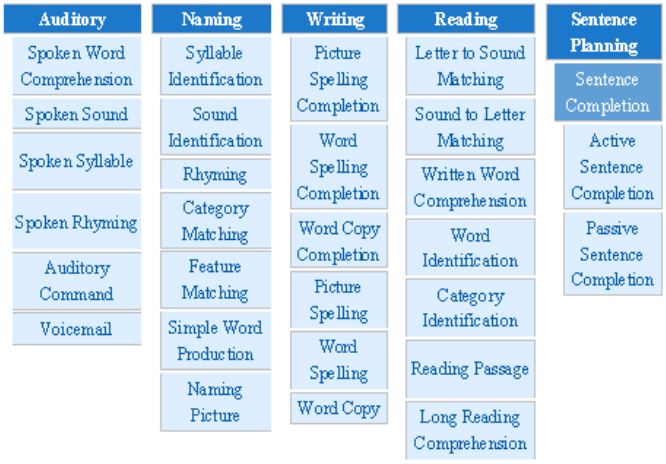


Figure 2 Constan Therapy task

2 Technical Analysis

2.1 MossTalk Words Training and Support

Researchers who perform research with patients with aphasia were asked to take part in a clinical research trial. For participants they were encouraged to; participate in a training program; complete evaluation forms; use the software to evaluate its effectiveness in a study of their design. Aphasia researchers and clinicians from sites across the United States met the standard and agreed to participate (Figure 3).
For those unable to attend the workshop, a videotape of the conference was given to them online. During the workshop organizers offered a summary of the software and its features;

trained them to use the treatment module; explanation of the reporting requirements; and assist with interaction among the colleagues. Preceding the workshop, participants received continued training and support through, e-mail and an electronic bulletin board.

Participating sites
Research sites
Boston University
Dalhousie University
Montclair State University
Old Dominion University
University of California - Davis
University of Colorado
University of Connecticut
Clinical sites
JFK Johnson Rehabilitation Center
Metro Health Center
New England Rehabilitation Center
Pennsylvania Veterans Administration Medical Center-Philadelphia
Rehabilitation Hospital of Indiana
Rush University

Figure 3 Reasearch and Clinician Sites

The organization of therapy tasks consisted of; multi-method matching, core vocabulary, and cued naming. The multi-method matching module consists of exercises that aid lexical comprehension and vocabulary expansion using printed or spoken words and images. The core vocabulary module focuses on patients with severe impairment in association. This module uses a progression of matching and naming exercises that restricts words with high implications. For the cued naming module, exercises in aiding in single word creation implementing an order from most common to uncommon in written and spoken cues. Vocabulary selection is based on categories, parts of speech, and linguistic characteristics.

2.2 Constant Therapy Training and Support

Constant Therapy is considered as a mobile application to aid individuals in rehabilitation in off-site treatment, the participants were selected through a standardized language test (Western Aphasia Battery) before the initial experiment took place. The experiment last for 10 weeks with respects to the patients advising clinician.

Therapy tasks were divided into several language and cognitive tasks. For language, the following tasks were designed; naming therapy, writing therapy, and reading therapy. For cognitive tasks the following task were designed; visuospatial processing, memory, mental flexibility, and problem solving. The language tasks were created from a database of concrete words spread from semantic categories. For reading comprehension, task extended across several topics with about 140 different items. The incentive for cognitive tasks covered anywhere from 50 items to 100 items per task.

3 Performance Analysis

The importance of a performance analysis is to obtain the effectiveness and efficiency of the system to aid in future implementation and design. In this section we review both the procedures in their performance analysis.

At the end of the first year, participants from the research sites had created a research proposal and in various stages of implementation such as data collection.

For the researchers who have developed projects, have then completed several studies, which lead to publications on relevance of the software, including its effectiveness for various language symptoms; its effectiveness when self-administered, and the impact of therapy [5, 6].

In conclusion to the first year of the project, a discussion took place to discuss the practicality and the implementation of the software. During that time an agreement was made that the data showed promising and encouraging outcomes with an interest in a continuous study and collection of data. In addition, a source of funding multiple research sites became a concern. In concluding the discussion, a preliminary evaluation was made to aid in the implementation and research for a multi-research study.

For Constant Therapy, in addition to an one hour per week clinician-interaction sessions, patients also logged into the iPad therapy software with an average duration of a 30 minutes. For each patient, a comparison graph was created showing accuracy and reaction time. With respects the patient's pre-evaluation test an average change was computed and the difference between the last two sessions and the first two sessions were outputted for evaluation. Due to the large sample size, a large amount of data has been collected leading to an unparalleled outcome that promotes the implementations and use of an iPad based therapy.

4 Discussion and Conclusion

4.1 Discussion MossTalk Words

Preliminary issues, such as, scheduling and intensity of treatment, must be attended to along with a developmental path toward clinical trials of rehabilitation interventions. A promising model for promoting development: identification of a treatment for critical effectiveness research, assist with organizations to provide a valued network, and offer methodological support to produce the optimal form of the intervention to be assessed.

4.2 Discussion Constant Therapy

This review of Constant Therapy emphasize that patients with brain damage are suitable to conduct therapy at home when provided with appropriate access to therapy and observed by a clinician. In addition, therapy tasks were personalized for each patient, based on standardized tests and performance and their baseline performance. Therefore,

patients showed much interest in practicing therapy tasks at home on a habitual basis. Patients also showed improvements in reaction time and accuracy on many therapy tasks that were dependent on the level of impairment, the level of difficulty and frequency of treatment.

4.3 Conclusion

In conclusion to the introduction of two autonomous therapy applications, the realization that a new form of therapy is proven to be effective in the rehabilitation of persons who suffer from aphasia. Within both cases, participants demonstrated improvements in accuracy and reaction time through computer and mobile treatment. The feasibility and success of a structured and continuous therapy with the use of a mobile device and/or personal computer provide accessibility and affordable means of receiving treatment and therapy.

5 References

[1] Howard, D., Patterson, K., Franklin, S., Orchard-Lisle, V., & Morton, J. (1985b).Treatment of word retrieval deficits in aphasia: A comparison of 2 therapy methods. Brain, 108 (Part4) 817-829

[2] Linebaugh, C., & Lehner, L. (1977). Cueing hierarchies and word retrieval: A treatment program. In R. H. Brookshire (Ed.), Clinical aphasiology: Conference proceedings (pp. 19-31). Minneapolis: BRK Publishers.

[3] Kiran, S & Sandberg, C. (2011). Treating communication problems in individuals with disordered language. R. Peach and L. Shapiro (Eds.). Cognition and Acquired Language Disorders: A Process-Oriented Approach. Elselvier. London. Pp 298-326

[4] Whyte J, Gordon W, Gonzalez-Rothi L: A phased developmental approach to rehabilitation research: The science of knowledge building. Archives of Physical Medicine and Rehabilitation. in press.

[5] Jokel, R.,Cupit,J.,Rochon, E., & Leonard, C. (2006). Computer-based intervention for anomiain progressive aphasia. Brain and Language, 99(1-2), 149-150.

[6] Ramsberger, G., & Marie, B. (2005). Self-administered moss talk words: A single subject design comparing treatment intensity replicated in three cases. 43rd Annual Meeting of the Academy of Aphasia, Amsterdam.

SESSION

ARTIFICIAL INTELLIGENCE, ALGORITHMS AND APPLICATIONS + COMPUTATIONAL SCIENCE AND INTELLIGENCE

Chair(s)

TBA

A Framework for Trust based shared Economy

Henry Hexmoor[1], Markum Reed[1], Purvag Patel[1], and Guy Fraker[2]
[1]Southern Illinois University
Carbondale, IL 62901, U.S.A.
[2]AutonomouStuff, LLC
Morton, IL

Abstract - *Network formation on trust and trustworthiness needs to be sustainable for any online trust based economy. It is important to have a matching game within our program in order to match up the most compatible individuals. Similar individuals can have higher levels of trust at the start of the relationship and thus have lasting partnerships. However, we must also create a network structure that allows us to single out more trustworthy individuals in the network; so that people have more access to these individuals in the network.*

Keywords: A Maximum of 6 Keywords

1 Introduction

Let us start by establishing some nomenclature. There are individuals who are strictly consumers, making up members of some set C where ci ϵ C will denote a prototypical member. Some individuals are providers of services and goods, as members of set P where $p_j \epsilon$ P and p_j denotes a prototypical member. Without loss of generality, P and C may contain corporate members as well as private citizens. There are individuals who are members of overlapping C and P denoted $C \cap P$. A subset of P denoted by X will denote elite members, X_k, who are established businesses[1]. A matchmaking, aggregation, and tracking platform is proposed to interface members of C to P. The aggregation API will solicit interested consumers and suggest pertinent providers. A central component of this system is to assign a *trust* scoring metric to members of C . By analogy to FICO[2] score, our scoring metric will consist of several components.

A consumer, C_i, is an existing member of group X, that individual will be assigned a kernel value, K, trust value of 1 for their basic trust value. A set of attributes A will be separately presented to each p_j and c_i where each attribute a is assigned a weight; important, neutral, unimportant. Thus, if someone is looking for a service, our system would inquire attributes of interest from c_i for preference elicitation and initial matching. This is at the core our aggregation component. Once the service is rendered, c_i would rate the service for feedback and suggested improvements for p_j. Lexicographic analysis[3] of inquiry and feedback will be used to fine tune attribute selection as well as postmortem rating of services. For example, if someone wants to loan an automobile from a provider, we can check the credibility of c_i by parsing various data from individual i's Facebook, Twitter, LinkedIn, and Google+. All major shared economy sites include reviews. This can be used as one of major attributes for computing trust. Individuals report values for their attributes with a value vector denoted as V. This is separate from their attribute weight vector W. The second component of the score is $T = \sum WV$. Thus, each p_j composes a trust score T for each c_i. Each trust score is used by each provider in determining quality of their potential consumer. Each consumer will be offered a trust score based on their personal weights, ω. This value differs from the trust score generated. The trust score is used as a feedback to

1 Such as Airbnb.com and Justshareit.com

2 The FICO mortgage score is between 300 and 850. Higher scores indicate lower credit risk. FICO was founded in 1956 as Fair, Isaac and Company by engineer Bill Fair and mathematician Earl Isaac.

3 Lexicology is the part of linguistics which studies words, their nature and meaning, words' elements, relations between words, word groups and the whole lexicon.

consumers as well as a list of providers recommendations. Upon lexicographic analysis of the feedback of consumer i, the third component of the trust score, L is generated.

2.0 Lexicographic Analysis

Lexicographic analysis is the part of linguistics, which studies words, their nature and meaning, words' elements, relations between words, word groups, and the whole lexicon. Lexicographic inquiry is used to fine tune attribute selection as well as postmortem rating of services. For example, if someone wants to loan an automobile from a provider, we check their credibility by parsing various data from individual's Facebook, Twitter, LinkedIn, and Google+ feeds. A collection of feeds can then be analyzed for their content to gather a measure of trustworthiness. We are able to determine positive and negative sentiment related to the individual's driving ability, trustworthiness, etc. This is accomplished by lexicographic analysis[4]. We created a metric that allows us to determine an individuals choices based on their observed lexicographic feeds. One way to approach the lexicographic measurement is to count the proportion of words that usually have a positive connotation and the proportion of words that have a negative connotation. This is a common analytic strategy for lexicographic analysis, as seen in (Golder and Macy, 2011). Following Golder and Macy (2011), we determine if there is positive or negative sentiment in each feed. Thus, we understand if a sentence is positive, negative, or neutral about an issue and to which grouping these sentiments are affiliated, e.g. car, truck, housing, etc. By investigating a person's feeds; we are able to determine what type of company or product these people would likely select. We are also able to see if they are receptive to suggestions for use of future products. In order to measure positive and negative statements within Twitter, we first need to identify the number of positive and negative words in each feed. We use a simple counting method when checking the positive and negative words[5]. Once we have the word counts, we multiply the negative word count by negative one in order differentiate between the positive and negative statements. Next, we determine the word structure of each feed. We then aggregate the composition of each feed to get an overall emotional measurement of each issue. The emotional responses can range from negative to positive. Thus we are able to explore how individuals *feel* about specific issues. Take the status update for the following example:

Joe is not trustworthy. He cannot be trusted.

For this example, we have two negative words (i.e., not and cannot) and two positive words (i.e. trustworthy and trusted). Therefore, the overall composition of this examples does not reflect well upon Joe. Leading us to believe that this person feels that Joe is not trustworthy giving us a score of -2; only for this example. Therefore, we would see a numerical value of negative one for use of automobiles. Next, we need to sum over all instances where Joe is mentioned or mentions trustworthiness or automobiles. Thereby, giving us Joe's overall trust score in equation 1.

$$L = \sum_{i=1}^{n} t_i \; \forall \; i, L \in [-\infty, \infty] \quad (1)$$

4 Lexicology is the part of linguistics which studies words, their nature and meaning, words' elements, relations between words, word groups and the whole lexicon.

5 Golder and Macy's Twitter study used the lists of positive and negative words that are part of the Linguistic Inquiry and Word Count (LIWC) project. However, we found similar dictionary that's freely available thanks to Neal Caren at University of North Carolina, Chapel Hill.

3.0 Trust Scoring

We compute a simple measure of trust in equation 2.

$$\text{Trust} = K + T + L \quad (2)$$

Apart from trust score value, an individual who has multiple profiles on social networking sites will be assigned a social presence value. The social presence indicates a potential for dissemination of word of mouth information that is a crucial source of PR value for providers.
Social scoring algorithm for an individual i:

1. Social (i) = 0, which is the default value
2. For an individual i, collect social network links to i, E(i).
3. For each social network sn of i, these include connections from Twitter, Facebook, LinkedIn, and Google+.

Therefore, we can compute the social vlue of individual I using equation 3.

$$\text{social}(i, sn) = \text{social}(i, sn) + \rho \sum_{i=0}^{n} \frac{\text{social}(j)}{L(j)} \quad (3)$$

Here n is the number of ties, j is one of i's ties. $\rho$is the propensity coefficient for i's network. Page rank $= 0.85$, which represents the normalized score.

4. The normalized score is computed with equation 4.

$$\text{Social}(i, sn) = \frac{score(A)}{|sn|} \quad (4)$$

4.0 Literature Review

In this section, we will review recent literature on trust based networks and matching games. Cagno and Sciubba (2008) ran laboratory experiments were friendship networks are generated endogenously within an anonymous group. Their experiment builds on two phases in sequence: a network formation game and a trust game. Cagno and Sciubba (2008) found that in those sessions where the trust game is played before the network formation game, the overall level of trust is not significantly different from the one observed in a simple trust game; in those sessions where the trust game is played after the network formation game we find that the overall level of trust is significantly lower than in the simple trust game. Hence surprisingly trust does not increase because of enforced reciprocity and moreover a common social history does affect the level of trust, but in a negative manner. Where network effects matter is in the choice of whom to trust: while we tend to trust less on average those with whom we have already interacted compared to total strangers, past history allows us to select whom to trust relatively more than others. Cagno and Sciubba (2008) show us that there is an importance in how we build trust within a network of individuals. We will see that later work solidifies the understanding of network formation on trust and trustworthiness needed to sustain an online trust based economy.

More recent work in (Di Cagno and Sciubba, 2010) investigated the impact of network formation on trust and trustworthiness. Again they ran laboratory experiments where, in sequence, networks are generated endogenously within an anonymous group and subjects play a trust game. The experimental design includes two main treatments and a baseline: in the baseline subjects play a trust game with no networks being formed, in treatment NT the network building phase precedes the trust game, and in treatment TN the network game is played at the end. This allows us to identify the two main factors through which networks impact on trust and trustworthiness: information accrued to subjects through social interaction (when this occurs first) and reputation (when it follows). Di Cagno and

Sciubba (2010) found that in NT, the overall level of trust is lower but offers are directed to more trustworthy recipients. A common past history matters in determining whom to trust (information value of networks). In TN, continuation play enforces higher levels of trust and trustworthiness (reputation and enforced reciprocity). Profits that subjects make in the trust game are higher in the presence of social interaction, and significantly so when network formation informs the decision of whom to trust. This shows that it is important to have a "matching game" within our program. So that like individuals can have higher levels of trust at the start of the relationship. However, we must also create a network structure that allows to single out more trustworthy individuals so that people have more access to these individuals in the network. (i.e. broadcast these individuals as "users of the week" that show who has the highest ranking or who is the most trustworthy).

(Cannatelli and Antoldi, 2012) describes how the role of network facilitator, played by a third party institution, may substantially contribute to the development of trust among SMEs involved in a strategic alliance. (Cannatelli and Antoldi, 2012) give empirical evidence that is presented by a longitudinal analysis of a case history. The case study focuses on eight international-oriented SMEs located in an industrial district in Northern Italy that built up a formal network called 'I-Style Partners'. Two rounds of in-depth interviews were carried out with firm leaders and facilitator's managers involved in the strategic alliance over a three-year period. Their paper contributes to theory generation suggesting a three-stage process model in which a network facilitator may enhance inter-organizational trust by constituting in turn a substitute of alliance members' perceptions of ability, integrity and benevolence. Thus giving need to a their party to vouch for the trustworthiness of network members. Karlan et al. (2009) builds a theory of trust based on informal contract enforcement in social networks. In their model, network connections between individuals can be used as social collateral to secure informal borrowing. They define network-based trust as the largest amount one agent can borrow from another agent and derive a reduced-form expression for this quantity, which is then use in three applications:

1. Predict that dense networks generate bonding social capital that allows transacting valuable assets, whereas loose networks create bridging social capital that improves access to cheap favors such as information.

2. For job recommendation networks, it is shown that strong ties between employers and trusted recommenders reduce asymmetric information about the quality of job candidates.

3. Using data from Peru, Karlan et al. (2009) show empirically that network-based trust predicts informal borrowing, and we structurally estimate and test our model.

Tams (2012) advances propositions regarding the structure of the relationship between vendor trust and its antecedents as this structure pertains to the relative and complementary effectiveness of trust-building strategies. By understanding how the relationship between vendor trust and its antecedents is structured and why this relationship is structured the way it is, Tams (2012) hope to gain more holistic insights into trust in electronic market transactions and to provide online businesses with a clear recommendation of how to establish trust in an effective and efficient manner. Thus, while past research has made important contributions by uncovering a great number of antecedents to vendor trust, Tams (2012) examines two strategies more in depth: vendor reputation and Web site trust. Drawing

from the literature on trust, the authors propose vendor reputation to be more effective than Web site trust. Tams (2012) propose a small complementary effect between vendor reputation and Web site trust that may help online businesses to generate superior vendor trust.

Cassar and Rigdon (2011) focus on the interaction between network structure, the role of information, and the level of trust and trustworthiness in 3-node networks. They extend the investment game with one Sender and one Receiver to networked versions–one characterized by one Sender and two Receivers ([1S - 2R]) and one characterized by two Senders and one Receiver ([2S - 1R])–under two information conditions, full and partial. Cassar and Rigdon (2011) develop a comparative model of trust for the networked exchange environments and generate two hypotheses:

1. What counts as a signal of trust depends on investment behavior along the other link in the network, and

2. This type of trust can be leveraged under full information, increasing the rate of cooperation on the side of the exchange with multiple traders. The results generally support our hypotheses: trust is comparative and under full information, the [1S - 2R] network shows higher trustworthiness and the [2S - 1R] network displays higher trust. Cassar and Rigdon (2011) Here is a very simple case of a three person network; however, these concepts may be extended to larger networks which we will discuss later on. Which kind of network fosters the diffusion of development-oriented trust? Sabatini (2009) carries out an empirical investigation into the causal relationships connecting four types of social networks (i.e. bonding, bridging, linking, and corporate), and different forms of trust (knowledge-based trust, social trust, trust towards public services and political institutions), in a community of entrepreneurs. Their results suggest that the main factors fostering the diffusion of social trust among entrepreneurs is the establishment of corporate ties through professional associations. Trust in people is positively and significantly correlated also to higher levels of satisfaction and confidence in public services. Participation in voluntary organizations does not appear to increase trust towards strangers. Rather, Sabatini (2009) find evidence of the other way round: interpersonal trust seems to encourage civic engagement. Thus, it is important to create high levels of trust amongst individuals in a network so that there is confidence in the services being provided. Trust is essential to supply chain teams as it has a positive impact on team performance. Long-term relationships in supply chains have also emphasized trust as their key element. Yet traditional models of trust have a limited application in hastily formed networks that are formed on the spot without a long-term component. An example of such hastily formed networks is the humanitarian aid supply network, which consists of a number of individual logisticians from a variety of organizations, coming together to bring relief to a disaster-stricken area. The aim of this paper is, thus, to further the understanding of swift trust in hastily formed networks as a means of improving relief operations in rapid onset disasters. Tatham and Kovacs (2010) present a model of swift trust and conditions are discussed to unearth potential facilitators of swift trust. In the interest a startup Internet firm in the modern age, it is important to create lasting trust quickly. Research into two important control mechanisms for managing the supply chain relationship–contracts and trust–is on the rise. However, our understanding of how they influence innovation in a firm remains rather unclear. Thus, the primary objective of Wang et al. (2011) was to examine the individual and interactive effects of contracts and trust on firms' innovation performance

and the contingent effects of environmental uncertainty on those relationships in China. The empirical results from a survey of Chinese manufacturing firms indicate that there is a positive relationship between trust and firms' innovation performance, an inverted U-shaped relationship between the use of contracts and firms' innovation performance, and that contracts and trust are substitutes. Moreover, Wang et al. (2011) found that environmental uncertainty enhances the effects of trust, but does not influence the impact of contracts on innovation performance. Our network should help to decrease the uncertainty of the users; thus, creating a helpful environment to foster trust between users.

Pan (2010) propose a social learning framework where agents repeatedly take the weighted average of all agents' current opinions in forming their own for the next period. They also update the influence weights that they place on each other. It is proven that both opinions and the influence weights are convergent. In the steady state, opinions reach consensus and influence weights are distributed evenly. Convergence occurs with an extended model as well,which indicates the tremendous influential power possessed by a minority group. Computer simulations of the updating processes provide supportive evidence (Pan, 2010). Here we see validation for our scoring mechanism; given that the way people weigh individuals as well as how social learning for a network of agents tend to converge over time.

Trust is important: several transactions are based on it; unfortunately it is difficult to measure. The recent literature on social capital shows a positive association between this concept and trust. As given that social capital is easier to measure than trust, it is best to analyze the possibility of assessing trust using a measure of social capital. A basic trust game is played in three Western European countries with undergraduate students; a questionnaire measures their level of social capital as time spent within social networks. This measure is stronger and more precise than the ones generally used. Migheli (2012) conducted an experiment that results support the fact that trust can be assessed through social capital, although the presence of a strong geographical effect has to be accounted for. The program used to calculate our score uses all of these factors plus several extras that make the score more robust.

5.0 Conclusions

Network formation on trust and trustworthiness needs to be sustainable for any online trust based economy. It is important to have a matching game within our program in order to match up the most compatible individuals. Similar individuals can have higher levels of trust at the start of the relationship and thus have lasting partnerships. However, we must also create a network structure that allows us to single out more trustworthy individuals in the network; so that people have more access to these individuals in the network. (i.e. broadcast these individuals as users of the week that show who has the highest ranking or who is the most trustworthy) Theory suggests that a three-stage process model in which a network facilitator, i.e. get2know may enhance inter-organizational trust by constituting in turn a substitute of alliance members' perceptions of ability, integrity and benevolence. Thus giving need to a third party to vouch for the trustworthiness of network members. In a network-based trust model we should be able to create a dense network thus generating social capital that allows transacting valuable assets; however, it must also be loose enough network to allow for bridging social capital to occur that improves access to cheap favors such as information. Drawing from the literature on trust, we can propose that reputation is more effective than Web site trust. There is a small complementary

effect between vendor reputation and Web site trust that may help online businesses to generate superior trust. Thus, the kind of network that we develop will foster the diffusion of trust. It is important to create high levels of trust amongst individuals in a network so that there is confidence in the services being provided. By modeling a network to build trust swiftly, we will create conditions that will be later discussed to unearth potential facilitators of swift trust. In the interest a start-up internet companies in the modern age, it is important to create lasting trust quickly. Our network will help to decrease the uncertainty of the users; thus, creating a helpful environment to foster trust between users. Trust is important: several transactions are based on it; unfortunately it is difficult to measure. However, trust can be determined via social capital, although the presence of a strong geographical effect will also be accounted for. Our program calculates the score based on verified factors that measure social capital and thus measures a person's trustworthiness.

References

[1]. Cagno, D. D. and Sciubba, E. (2008). Social networks and trust: not the experimental evidence you may expect.

[2]. Cannatelli, B. and Antoldi, F. (2012). The role of network facilitators in fostering trust within strategic alliances: A longitudinal case study. *Journal of Small Business and Entrepreneurship*, 25(1):19–33.

[3]. Cassar, A. and Rigdon, M. (2011). Trust and trustworthiness in networked exchange. *Games and Economic Behavior*, 71(2):282–303.

[4]. Di Cagno, D. and Sciubba, E. (2010). Trust, trustworthiness and social networks: Playing a trust game when networks are formed in the lab. *Journal of Economic Behavior and Organization*, 75(2):156–167.

[5]. Golder, S. A. and Macy, M. W. (2011). Diurnal and seasonal mood vary with work, sleep, and daylength across diverse cultures. *Science*, 333(6051):1878–1881.

[6]. Karlan, D., Mobius, M., Rosenblat, T., and Szeidl, A. (2009). Trust and social collateral. *Quarterly Journal of Economics*, 124(3):1307–1361.

[7]. Migheli, M. (2012). Assessing trust through social capital? a possible experimental answer. *American Journal of Economics and Sociology*, 71(2):298–327.

[8]. Pan, Z. (2010). Trust, influence, and convergence of behavior in social networks. *Mathematical Social Sciences*, 60(1):69–78.

[9]. Sabatini, F. (2009). The relationship between trust and networks: An exploratory empirical analysis. *Economics Bulletin*, 29(2):661–672.

[10]. Tams, S. (2012). Toward holistic insights into trust in electronic markets: Examining the structure of the relationship between vendor trust and its antecedents. *Information Systems and e-Business Management*, 10(1):149–160.

[11]. Tatham, P. and Kovacs, G. (2010). The application of 'swift trust' to humanitarian logistics. *International Journal of Production Economics*, 126(1):35–45.

[12]. Wang, L., Yeung, J. H. Y., and Zhang, M. (2011). The impact of trust and contract on innovation performance: The moderating role of environmental uncertainty. *International Journal of Production Economics*, 134(1):114–122.

Artificial Intelligence in Clothing Fashion

Haosha Wang[1] and Khaled Rasheed[2]
Institute for Artificial Intelligence
The University of Georgia
Athens, GA, U.S.A. 30602
[1] hswang@uga.edu
[2] khaled@cs.uga.edu

Abstract

"Clothes make the man," said Mark Twain. This article presents a survey of the literature on Artificial Intelligence applications to clothing fashion. An AI-based stylist model is proposed based on fundamental fashion theory and the early work of AI in fashion. This study examines three essential components of a complete styling task as well as previously launched applications and earlier research work. Additionally, the implementation and performance of Neural Networks, Genetic Algorithms, Support Vector Machines and other AI methods used in the fashion domain are discussed in detail. This article explores the focus of previous studies and provides a general overview of the usage of AI techniques in the fashion domain.

Keywords: Artificial Intelligence, Fashion, Clothing Styling.

1. Introduction

There might be a moral argument about whether people should be judged by their apparel. In practice however, few people would consider a person in baggy jeans walking into their first meetings seriously. Dressing properly brings a big ROI (Return of Investment). According to a survey done by OfficeTeam, 93% out of more than 1000 senior managers at companies with 20 or more employees responded that clothing choice affects an employee's chances of promotion (OfficeTeam, 2007). However, keeping track of fashion sense requires significant time and effort, which leads some people to seek help from a professional stylist. Personal stylists can be expensive though and cannot be with clients all the time. This study discusses whether an Artificial Intelligence based computer program could be the new fashion consultant and how it might be executed. There are many benefits associated with using computer programs as future stylists. They could process large amounts of data faster when learning a user's style and memorizing users' feedback. AI stylist programs can also store descriptions of user's items and help users be more organized and efficient.

The goal of this survey study is to explore AI methods in the clothing fashion domain. In the second section, three major components of a full styling task are identified. The third section is a summary of earlier projects and relevant fashion theory. In the methodology section, earlier works in AI implementation and their solutions to each major problem are explored. The last section is a discussion.

2. Related AI-technologies

Every fashion clothing styling task which is completed has 7 steps: picking a theme, deciding on a primary color, mixing and matching clothing pieces, selecting accessories, model or client fitting and finally hair styling and make up.

An AI based computer program focused on modeling centers on solving the following questions:

1. How to represent garments computationally?
2. How to model human stylist behavior?
3. How to detect, track and forecast fashion trends?

Popular AI methods used previously include Fuzzy Logic, Genetic Algorithms, Neural Networks, Decision Trees, Bayesian Networks and Knowledge Based Systems and their variations. This section will briefly outline these AI methods.

A *Bayesian Network* is a probabilistic model that represents variables and their conditional dependencies (Russell & Norvig, 2009). They have been used to infer relationships between previous fashion trends and future trends.

Fuzzy logic is an approach that utilizes uncertainty and approximate reasoning (Eberhart & Shi, 2007). It represents truthfulness and falsehood with degrees and works closer to human brain because it outputs a straightforward like or dislike.

Using *Artificial Neural Networks (ANNs)* is a learning method inspired by animal nervous systems. An ANN maps input to a target output

by adjusting weights (Eberhart & Shi, 2007). This method works well for modeling complex styling tasks with multiple features.

Knowledge-Based Systems are programs that represent knowledge and solve complex problems by reasoning on how knowledge artifacts are related or not related (Eberhart & Shi, 2007). They are used to show the relationships between features in fashion styling.

Decision Trees are tree-structured graphs that represent attributes as internal nodes and outcomes as branches (Kokol, Verlic, & Krizmaric, 2006). They are widely used in human decision-making models.

Genetic Algorithms (GAs) are search techniques that look for approximate or exact solutions to optimization problems. They are guided by a fitness function. Interactive Genetic Algorithms (IGAs) have been used in earlier studies aimed at achieving real-time interaction. The biggest difference between IGAs and regular GAs is that instead of using a fitness function, IGAs assign a fitness value to each individual (Tokumaru, Muranaka, & Imanish, 2003).

3. Contemporary Applications

3.1 Research Design of this Study

In this study, the AI-based fashion applications and articles are classified based on the fundamental definition of fashion. Fashion is a major part of people's daily lives and the fashion market itself is quite large. In the next two sections, current popular computer applications and prior research from the last decade are summarized.

3.2 Popular Applications

There are four major types of applications as shown in Table 1. *Internet Based Human Stylist Consultant Services* put the communication between clients and stylists on the Internet. They improve the flexibility and accessibility of styling work. *Virtual Fitting Systems* fill one of the major gaps between e-commerce and retail stores. The third type of application is *Recommender System Implementations in E-commerce*; for example Amazon recommends new items based on users' browsing histories**.** The last type is *Online Fashion Communities*, such as Polyvore, which provide a platform for people to share, gain and communicate fashion inspiration and shopping information online.

3.3 Earlier Research Projects

Conceptually, fashion can be defined as a two dimensional concept, an object and a behavior process. The form of a "fashion object" can be a

Table 1: Popular Fashion Applications List *All sites listed above were visited on April 15th, 2014

Business Model	Name	Website URL*
Human Stylist Internet-Based Consultant Services		
Recommend new items to mix and match with user's existing items	Topshelf	http://www.topshelfclothes.com/
	MyPrivateStylist	http://www.myprivatestylist.com/
Exclusive, high reputation and experienced stylists offer service to their members	KeatonRow	https://keatonrow.com/
	StyleMeASAP	http://stylemeasap.com/
Human stylists style male customers within clothing collections	TrunkClub	http://www.trunkclub.com/
Virtual fitting system		
Customized virtual avatars for virtual clothes fitting experiences with clothing imagery	Glamstorm	http://glamstorm.com/en
	CovetFashion	http://www.covetfashion.com/
	ChroMirror	(Cheng, et al., 2008)
Recommender system in e-commerce sites		
Recommend items based on user' activities and browsing records	Amazon	http://www.amazon.com/
Recommend new trendy clothing items and search for relevant clothing items based on user's search queries	Google Shopping	http://www.google.com/shopping
Pushes sale information based on user's preferences	Shop It To Me	http://www.shopittome.com/
Online fashion community		
Platforms for users to create, share and look for fashion inspiration	Polyvore	http://www.polyvore.com/cgi/home
	Lyst	http://www.lyst.com/
	StyledOn	http://styledon.com/

specific product or innovative technical features or new membership services. While "fashion process" is a process through which a "fashion object" emerges from its creation to public presentation, trendsetter adapting, majority acceptance/rejection, and replacement of newer object and merge of next trend (Sproles, 1979).

The major focus of this study is to find solutions for the three target problems stated in Section 2. Among the three problems, the first one, garment representation focuses on fashion object. The third one, detection, tracking and forecasting of fashion trend are about fashion process. While the second one, modeling human stylist behavior is a mixture of both.

3.3.1 AI Techniques on "Fashion Objects"

Back in 2000, Genetic Algorithms were used in a fashion design assistant system (Kim & Cho, 2000). Clothing color styling model was proposed in the Virtual Stylist Project in (Tokumaru, Muranaka, & Imanish, 2003). Decision Trees with Genetic Algorithms were used to model individual's clothing in (Kokol, Verlic, & Krizmaric, 2006). Researchers implemented Category Learning and Neural Networks in an intelligent clothing shopping assistant system in 2008 (Cheng & Liu, 2008). Computer Vision techniques with Support Vector Machine (SVM) classifiers were used to discover the semantic correlations between attributes in (Chen, Gallagher, & Girod, 2012).

3.3.2 AI Techniques on "Fashion Process"

Previous studies tried to understand, detect and predict fashion trends and fashion cycles from both the perspectives of theory and application. Two earlier studies focused on predicting clothing color fashion trends. Mello's team developed an expert system that assists the stylist with the proposal of new color trends. Their system implemented a Bayesian Network model stylist proposing process (Mello, Storari, & Valli, 2008). Yu's team compared different AI models for predicting fashion color trends with an expert system (Yu, Hui, & Choi, 2012). There are many interesting models of fashion trends. One model on simplified general fashion cycles was of specific interest. This model has three major factors: base utility, social influence and user boredom (Sarma, Gollapudi, Panigraphy, & Zhang, 2012).

4. AI Methods in Fashion

4.1 Garment Representation

Table 2: AI in Fashion Projects List

Year	Title	Reference
Garment Representation – Fashion Object		
2000	Application of Interactive Genetic Algorithm to Fashion Design	(Kim & Cho, 2000)
2012	Describing Clothing by Semantic Attributes	(Chen, Gallagher, & Girod, 2012)
Human Stylist Behavior Model – Fashion Objects & Fashion Process		
2003	Virtual Stylist Project – Examination of Adapting Clothing Search System to User's Subjectivity with Interactive Genetic Algorithms	(Tokumaru, Muranaka, & Imanish, 2003)
2006	Modeling Teens Clothing Fashion Preferences using Machine Learning	(Kokol, Verlic, & Krizmaric, 2006)
2008	An Intelligent Clothes Search System Based on Fashion Styles	(Cheng & Liu, 2008)
2008	Mobile Fashion Advisor – A Novel Application in Ubiquitous Society	(Cheng & Liu, 2008)
Fashion Trend Detection, Track and Forecasting- Fashion Process		
2010	Application of Machine Learning Techniques For the Forecasting of Fashion Trends	(Mello, Storari, & Valli, 2008)
2012	An Empirical Study of Intelligent Expert Systems on Forecasting of Fashion Color Trend	(Yu, Hui, & Choi, 2012)
2012	Understanding Cyclic Trends in Social Choices	(Sarma, Gollapudi, Panigraphy, & Zhang, 2012)

Figure 1: Describing garments with attributes *
***Websites are visited on April 30th, 2014**

Garments can be described by a set of features as shown in Figure 1. Different pieces have their own unique features. The most basic features are Color, Shape, Print and Fabric (Material). Computer Vision techniques are able to automatically recognize the first three. Color is initially represented in a RGB (Red, Green and Blue) model and converted into an HSI (Hue, Saturation and Intense) model for further computation (Tokumaru, Muranaka, & Imanish, 2003) (Cheng & Liu, 2008). The Outline can easily be extracted with Computer Vision techniques and the print of a garment can be considered in its loudness. The loudness in this case is the frequency of color changes in the garment and the changes in locality on this garment. Fabric is the trickiest one. Even humans would have difficulties recognizing a fabric just by looking at it.

One possible solution for tricky attributes like "fabric" is inspired by the computer vision research project in 2012. They researched stylistic semantic correlations between clothing attributes. For example, Mark Zuckerberg's dressing style contains attributes like "gray/brown" and "t-shirt/outwear". They applied Support Vector Machine classifiers on single attributes to determine how useful these attributes would be in prediction. Then, the system makes predictions based on inference of different attributes' mutual dependency relations in a Conditional Random Field (Chen, Gallagher, & Girod, 2012). With a similar model, systems could learn correlations between attributes and fill in missing values. For instance, a formal dress has a high possibility of being made of silk. Casual style white t-shirts are likely to be made of cotton.

4.2 Computational Styling Task

4.2.1 Color Harmony Evaluation

Color is the very first step in a fashion styling task. In previous research, Matsuda's Color Coordination (MCC) has been used to evaluate the harmony rate between colors (Tokumaru, Muranaka, & Imanish, 2003; Cheng & Liu, 2008). Yutaka Matsuda who had investigated color schemes of female clothes and dress through a questionnaire for 9 years proposed MCC. In MCC there are 80 color schemes (8 hue types and 10 tone types) and a color scheme is harmonious if there are many samples in the system. Each individual has a unique style preference. An ideal intelligent style system should be able to adapt from a standard color scheme to a specific user's personal preference instantaneously. One working solution is to make static color schemes more dynamic.

In VSP, they added linguistic labels ("neutral, "a little", "slightly", "fairly", "very" and "extremely") on the color scheme and linguistic labels are presented as fuzzy sets respectively. Their system adapts itself to a user's preference during its interaction with the user. They also implemented an Interactive Genetic Algorithm so that the system is able to do so in real time. There are three different types of nodes in their IGA: the parameters of four basic senses, the weight of four senses and fuzzy rules. The system displays some dress patterns based on its knowledge and the user rates the similarity to her clothing sensation from 1 to 5. Highly rated individuals are used to generate the next generation individuals (Tokumaru, Muranaka, & Imanish, 2003). The experimental results confirmed that IGA optimized the system and the system was able to adapt itself to specific user preferences.

4.2.2 Shape, Prints and Fabric styling

Identifying shape, print and fabric is the second step in the fashion styling task. Style principles of these attributes are changing all the time. For instance, one of the standard style rules for prints is to wear only one pattern piece at a time. However, in the 2014 spring season, wearing two patterns with similar shades and detailing became very trendy. Moreover, it heavily depends on individual preference. Women with a classic style will not follow this new trend. Many other features also affect the styling task, such as occasion and cultural background. In this scenario, a stylist looks at a number of features from different categories and tries to classify items based on their experience and fashion sense. A Neural Network model is a good approach for a problem like this. In a previous study, an ALCOVE (Attention Learning Covering map) model based on NN was implemented in a clothing match system (Cheng & Liu, 2008). It converted garment physical attributes into garment sensation space in a fuzzy set (Table 3). Four garment sensations are input

Table 3: Garment Physical Attributes and their garment sensation

Major Physical Attributes	Garment Sensation
Color	Warm / Cold
Print	Cheeriness
Shape	Fitness
Fabric	Softness

nodes of the NN. In Cheng and Liu's project, training data were gathered through a questionnaire designed by a fashion stylist. Clothes were categorized into 10 different styles: Sexy, Modest, Sophisticated, Elegant, Luxuriant, Romantic, Girly, Masculine, Sporty and Casual. During training, the NN adjusted weights based on a given target output Neural Network is a state-of-the-art model to classify items with numerous attributes into different classes. The drawback is that they are less interpretable. For instance, updated style rules would be a great data source to keep track of changes in fashion trends, but it is difficult to extract them out of a NN model.

Decision Trees, on the other hand, are very interpretable. Previous research has used Genetic Algorithms and Vector Decision Trees to model user clothing preferences and generated fashion rules (Kokol, Verlic, & Krizmaric, 2006). They implemented a GA in a tree structure where VDT's genotype and phenotype were the same. This actually sped up the fitness computation. The fitness function has 5 major components: average accuracy over all dimensions, accuracy of the whole vector, average performance of classifiers, the factor minimizing the overall fitness bonus score and a linear penalty to avoid overfitting. They trained the model with 60 initial trees with a mutation rate of 0.5%. The model had an accuracy of 85% after 1273 iterations and it showed that such model is able to handle this highly dynamic task (Kokol, Verlic, & Krizmaric, 2006).

4.3 Fashion Trend Tracker

The ideal intelligent stylist program should be very sensitive to fashion trends. Fashion trend forecasting is the key to success in the fashion industry and always a bigger challenge than prediction of other fields. The biggest variable of it

is human value. In the fashion process as shown in Figure 2, new trends get popular slowly before trendsetter adoption (Phase A), reaches to its peak of fast social majority acceptance (Phase B) and declines dramatically thereafter.

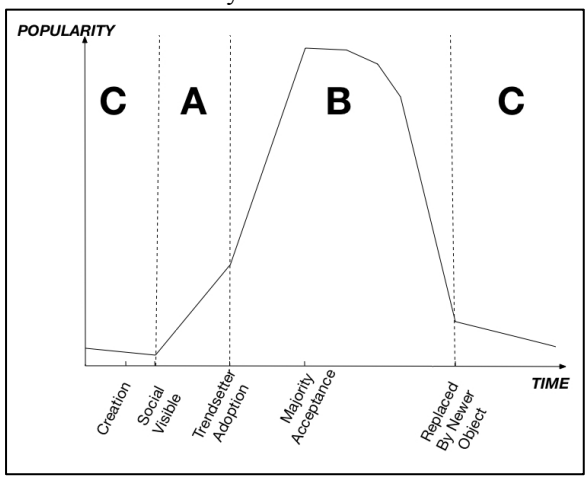


Figure 2: Fashion process or Fashion cycle

Two major reasons for the decline are: 1. Newer trends kick in; 2. People get bored with the trend after the peak. Generally, there are two types of fashion forecasting. The first one focuses on current fashion objects and it does predictions such as fashion color trends. Another type interests in a longer vision, for instance, black and white are never out of date regardless of season or occasion.

Some recent and relatively successful studies have attempted to predict fashion color trends with AI techniques. Dr. Mello and her team have implemented Knowledge Base Systems and Bayesian Networks (Mello, Storari, & Valli, 2008). They modeled a human stylist's new color trend proposal process with a Bayesian Network. Based on the knowledge of past and current color trends, BN classifiers classified color into binary target values, proposed or not proposed based on their probability. The human stylist then confirmed their application performance. In another study, researchers compared the performance of 5 models in fashion color prediction (Yu, Hui, & Choi, 2012) .The 5 models are the statistical models ARIMA (Autoregressive Integrated Moving Average), GM, GNNM (Grey Neural Network Model), Improved GM, ANN, ELM (Extreme Learning Machine). After the comparison, they proposed a hybrid GRA (Grey Relational Analysis)-ELM that achieved high accuracy and was less time consuming.

Researchers proposed a simplified fashion trend model taking three major factors into account: Base Utility, Boredom and Influence (Figure3).

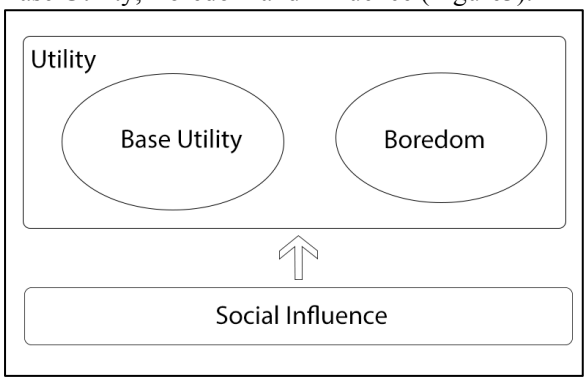


Figure 3: Simplified fashion process model

They modeled individual boredom as a "memory" factor of an object and the parameter of "memory" as the number of times using it in the past. With more times of use, more memories, higher boredom and less utility are left. The goal is to explore how to maximize the user's overall utility of an item. The fashion process is continuous and a solution to this problem is to compute the consumption cycle at a regular time (Sarma, Gollapudi, Panigraphy, & Zhang, 2012). They considered the fashion cycle as an NP-Hard problem and validated it with both a greedy algorithm and double greedy algorithm. In their experiment, they collected data including music, events and the boredom factors from Google Trend. Their experiment showed that a double greedy algorithm has better performance than a greedy algorithm in this model.

5. Concluding Remarks

AI based Stylist programs should consider the following three major components: (1) Visual Garment Representation, (2) Computational Styling and (3) Fashion Trend Tracking. This study has presented various ways to use AI techniques in a fashion stylist computational model.

For the first task, earlier works show that computer vision techniques can extract attribute information from an image and AI techniques such as semantic mapping give the program the ability to deal with tricky attributes, such as fabric. For cloth-piece styling, earlier researchers started off using Color Harmony Evaluation (Tokumaru, Muranaka, & Imanish, 2003) between garment pieces and then taking four basic attributes - color, outline, print and fabric - into consideration. Neural Networks

with category learning techniques have been applied successfully (Cheng & Liu, 2008). However, a completed styling task requires more attributes, such as event, clothes, shoes, accessories, makeup and hairstyling. A stylist is also very personal. Earlier work used GAs and Decision Trees to model fashion personal preferences (Kokol, Verlic, & Krizmaric, 2006). Even though their study is not focused on a completed styling task, their model is a good example of a fashion personal preference model. Change is also central to fashion. Computers have the ability to deal with large amounts of data and the ability to deal with large amounts of data and would be a good assistant for human stylists. Two research projects applied Bayesian Network Neural Networks and Knowledge Base Systems to fashion trend prediction (Mello, Storari, & Valli, 2008) (Yu, Hui, & Choi, 2012). Their experiments indicate that hybrid models and Bayesian Networks both have good performance on predicting color trends in the next season.

Earlier work has shown that AI-based programs have the ability to execute a fashion styling task and fashion can be modeled with a simplified model. However, there is still much work to be done on AI based styling. Models in earlier studies are not comprehensive. For instance, styling task requires more features than color, fabric, shape and outline. Also fashion changes among different cultures and times. For instance, Asian fashion style is different from Parisian.

Bibliography

Blumer, H. (1969). Fashion: From Class differentiation to collective selection. *The sociological Quarterly , 10* (3), 275-291.

Boden, M. (1998). Creativity and Artificial Intelligence. *Artificial Intelligence* , 347-356.

Chen, H., Gallagher, A., & Girod, B. (2012). Describing Clothing by Semantic Attributes. *12th European Conference on Computer Vision* (pp. 609-623). Florece, Italy: Springer Berlin Heidelberg.

Cheng, C.-I., & Liu, D.-m. (2008). An Intelligent Clothies Search System Based on Fashion Styles. *7th International Conference on Machine Learning and Cybernetic.* Kunming.

Cheng, C.-I., & Liu, D.-M. (2008). Mobile Fashion Advisor - A Novel Application in Ubiquitous Society. *International Journal of Smart Home , 2*.

Cheng, C.-M., Ouhyoung, M., Chung, M.-F., Chu, H.-H., Yu, M.-Y., & Chuang, Y.-Y. (2008). ChroMirror: A Real-Time Interactive Mirror for Chromatic and Color-Harmonic Dressing. *CHI.* Florence, Italy.

Eberhart, R., & Shi, Y. (2007). *Computational Intelligence.* Burlington, MA, USA: Elsevier Inc.

Kim, H.-S., & Cho, S.-B. (2000). Application of Interactive Genetic Algorithm to Fashion Design. *Engineering Applications of Artificial Intelligence*, *13*, pp. 635-644.

Kokol, P., Verlic, M., & Krizmaric, M. (2006). Modeling Teens Clothing Fashion Preference Using Machine Learning. *10th WSEAS International Conference on Computers*, (pp. 902-913). Athens, Greece.

Kruschke, J. (1992). ALCOVE: An Exerplar-Based Connectionist Model of Category Learning. *Psychological Review , 99*, 22-44.

Mello, P., Storari, S., & Valli, B. (2008). A Knowledge-Based System for Fashion Trend Forecasting. *21st International Conference on Industiral Engineering and Other Application of Applied intelligent System, IEA/AIE* , (pp. 425-434). Wroctaw, Porland.

Muranaka, N., Tokumaru, M., & Imanish, S. (2003). Virtual Stylist Project - Examniation of Adapting Clothing Search System to User's Subjectivity with Interactive Genetic Algorithms. *CEC 2003 the 2003 Congress on Evulutionary Computation , 2*, 1036-.

OfficeTeam. (2007). *'Clothinges' in on the promotion.* Retrieved from OfficeTeam: http://officeteam.rhi.mediaroom.com/index.php?s=247&item=806

Reynolds, W. (1968). Cars and Clothing: Understanding Fashion Trend. *Journal of Marketing , 32*, 44-49.

Russell, S. J., & Norvig, P. (2009). *Artificial intelligence: a modern approach (3rd edition)*. Prentice Hall.

Sarma, A., Gollapudi, S., Panigraphy, R., & Zhang, L. (2012). Understanding Cyclic Trends in Social Choices. *WSDM*. Seattle, WA,USA.

Sproles, G. (1979). Fashion Theory: a Conceptual Framework. *NA - Advances in Consumer Research. 1*, pp. 463-472. Ann Abor, MI: Scott Ward and Peter Wrigh.

Tokumaru, M., Muranaka, N., & Imanish, S. (2003). Virtual Stylist Project - Examniation of Adapting Clothing Search System to User's Subjectivity with Interactive Genetic Algorithms. *CEC 2003 the 2003 Congress on Evulutionary Computation , 2*, 1036-.

Yu, Y., Hui, C.-L., & Choi, T.-M. (2012). An Empirical Study of Intelligent Expert Systems on Forecasting of Fashion Color Trend. *Expert Systems with Applications*, (pp. 4383-4389).

Implementation to Provide Individual Illuminance and Color Temperature in an Intelligent Lighting System without Chroma Meter

Risa Kawashima[2], Mitsunori Miki[1], Hisanori Ikegami[2], Yohei Azuma[2], and Hiroto Aida[1]
[1]Department of Science and Engineering, Doshisha University, Kyoto, Japan
[2]Graduate School of Science and Engineering, Doshisha University, Kyoto, Japan

Abstract— *The authors have conducted research into an Intelligent Lighting System in order to achieve the illuminance and color temperature levels required by each office worker with low power consumption through the use of a chroma meter. However, as these chroma meters are very expensive, it is not easy to introduce them into offices. For this reason, we have proposed a new method for an Intelligent Lighting System that achieves separate color illuminance and color temperature for the specified location, without using a chroma meter. In verification experiments demonstrating the effectiveness of the proposed method, we have performed comparative experiments between methods using chroma meters and the proposed method. It was possible to achieve the illuminance/color temperature required by each office worker, when the illuminance sensors are separated by more than twice the lighting placement interval.*

Keywords: Office, Lighting control, Intelligent Lighting System, Optimization, Color temperature, Chroma meter, Illuminance sensors

1. Introduction

In recent years, there has been increased attention on improvements in the intellectual productivity, creativity, and comfort of office workers within the office environment [1], [2]. In the research of Boyce et al., it was reported that providing optimal illuminance separately for each worker is effective from the standpoint of improving the lighting environment [3]. Furthermore, when PHILIPS introduced lighting with a color temperature of 17000 K in offices and factories and conducted experiments in relation to this, powers of memory and concentration improved and work efficiency improved [4].

In regard to illuminance, it has been reported that preferences differed according to the individual [5]. Furthermore, in the same way, it is understood that preferences for color temperature also differ according to the individual [6].

For this reason, the authors have concentrated on the lighting environment in offices, and proposed an Intelligent Lighting System that provides illuminance and color temperature based on the individual request of the office worker [7]. With the Intelligent Lighting System, as the office worker sets the required illuminance (target illuminance) and required color temperature (target color temperature) using a chroma meter that can measure the illuminance and color temperature, it is possible to achieve lighting patterns that reach this value with the minimum amount of power.

The effectiveness of the Intelligent Lighting System has already been recognized, and verification experiments have been conducted in multiple offices within the Tokyo Metropolitan District [8], [9]. Due to the fact that chroma meters are expensive, the Intelligent Lighting Systems introduced in actual offices are controlled using only illuminance sensors. Here, illuminance is controlled automatically, and color temperature controls are performed manually for each lighting incidence, so this is considered troublesome for the office workers.

et illuminance for the specified location, without using chroma meters. In this way, it is possible to achieve the illuminance and color temperature required by the office worker without using chroma meters. Furthermore, as the operation of setting the color temperature separately for each type of lighting is unnecessary, the burden on the office worker is reduced. In this study, we construct a system integrating the proposed method, and demonstrate its effectiveness through verification experiments in an environment simulating an actual office.

2. Intelligent Lighting System

2.1 Overview of an Intelligent Lighting System

The Intelligent Lighting System is a lighting system that provides the illuminance and color temperature required by the office worker in the specified location with low power consumption [7]. In this structure, multiple lighting devices capable of light control, multiple chroma meters, a control device, and a wattmeter are connected together in a single network. The structure of the Intelligent Lighting System is shown in Fig.1.

The control device set for each lighting incidence changes the luminous intensity based on the illuminance information and color temperature obtained from the chroma meter,

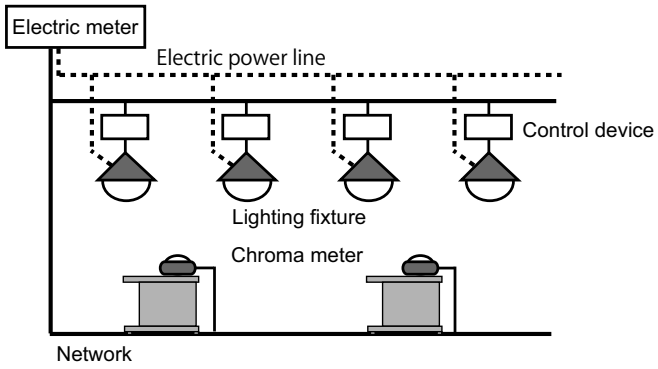


Figure 1: Configuration of Intelligent Lighting System

and the power consumption information obtained from the wattmeter, using the optimization method. Through the repetition of this process, it can achieve the illuminance and color temperature required by the office worker. In addition, the Intelligent Lighting System does not require physical location information from the chroma meter. This is because it learns the influence ratio ((illuminance/luminous intensity) illuminance/luminous intensity influence ratio) of the lighting on the chroma meter from the illuminance information obtained from each of the chroma meters and luminous intensity information obtained from each lighting incidence.

2.2 Control algorithm using regression analysis

Intelligent Lighting System controls use a control algorithm (Adaptive Neighborhood Algorithm using Regression Coefficient: ANA/RC) based on Simulated Annealing (SA) [10]. ANA/RC uses the luminous intensity of each lighting as a design variable and changes the luminous intensity in each lighting per search at random within a range [11] not detectable by the office worker to search for the optimal lighting pattern. The minimum power consumption is then sought under the conditions of illuminance and target illuminance conditions, with the illuminance at target illuminance or more, and color temperature as target color temperature or more. Furthermore, it discovers the illuminance/luminous intensity level of influence applied by each type of lighting to the chroma meter through regression analysis from the change amount of illuminance in the illuminance sensor and the change amount of illuminance luminous intensity. The Intelligent Lighting System, by extracting lighting near the illuminance sensor from the size of the illuminance/luminous intensity level of influence, is able to grasp the positional relationships of the illuminance sensors. By changing the luminous intensity for each lighting incidence based on the illuminance/luminous intensity level of influence, it is possible to swiftly provide the target illuminance and target color temperature for each office worker.

Next, we will describe the objective function used in this algorithm. The objective of the Intelligent Lighting System is to achieve the illuminance and color temperatures required by each office worker and minimize power consumption. The objective functions are formularized as shown in formula (1).

$$f_i = P + \omega_l \times \sum_{j=1}^{n} g_{ij} + \omega_c \times \sum_{j=1}^{n} h_{ij} \quad (1)$$

$$g_{ij} = \begin{cases} 0 & (Ic_j - It_j) \geq 0 \\ R_{ij} \times (Ic_j - It_j)^2 & (Ic_j - It_j) < 0 \end{cases}$$

$$h_{ij} = \begin{cases} 0 & (Cc_j - Ct_j) \geq 0 \\ R_{ij} \times (Cc_j - Ct_j)^2 & (Cc_j - Ct_j) < 0 \end{cases}$$

$$R_{ij} = \begin{cases} r_{ij} & r_{ij} \geq T \\ 0 & r_{ij} < T \end{cases}$$

i:number of lightings, j:number of sensors
ω_l:wight [W/lx^2], ω_c:weight [W/K^2]
P:power consumption [W], Ic:current illuminance [lx]
It:target illuminance [lx], Cc:current color temperature [K]
Ct:target color temperature [K], L:luminace [cd]
T:threshold
r_{ij}:Regression coefficient for illuminance sensor j for illumanation i

The objective function shown in formula (1) is comprised of power consumption P, illuminance restriction g_{ij} and color temperature restriction h_{ij},, and is calculated for each lighting incidence. As the penalty function g_{ij} and h_{ij}, which are restrictive conditions for target illuminance, and the target color temperature for the chroma meter fluctuate more than the regression coefficient r_{ij}, the penalty functions are increasingly minimized the larger the regression coefficient of the lighting. Furthermore, by setting the threshold T for regression coefficient r_{ij}, the target for optimization can be filtered to the lighting, which strongly influences the chroma meter. In this way, the lighting that is distant from the chroma meter is only operated for the purpose of minimizing power consumption.

3. Lighting control method for achieving target color temperature in the specified location without using a chroma meter

3.1 Structure of the proposed method

As described above, due to the fact that the chroma meters are expensive, it is not easy to introduce them into offices. For this reason, this study presents a method of achieving the illumination/color temperature required in the specified locations, using only illuminance sensors. The proposed method extracts lighting close to the illuminance sensors based on the illuminance/luminous intensity influence level, and lights these to the target color temperature. Furthermore,

where the extracted lighting is lighting with a high illuminance/luminous intensity influence level in relation to the illuminance sensor, it is lit to the average mired value for that color temperature. The mired value is a value expressing the reciprocal of the color temperature, and has a proportional relationship with human visual color perception; this makes it possible to express the differences in color temperature between different color temperatures.

3.2 Method of lighting to the specified color temperature

A method for controlling the lighting ratio is used as a way to light each lighting incidence to the specified color temperature. By lighting to colors with two different color temperatures, and changing this lighting ratio, it is possible to achieve color temperatures across a wide range. For this reason, the lighting ratio of the two lighting colors for achieving the target color temperature is sought experimentally, and the relationship between the lighting ratio for neutral white and incandescent lighting and the color temperature is stored within the system as a table. Then, when lighting to the light at the specified color temperature, the light is lit at the ratio obtained from the color temperature table. 1 shows the color temperature table from 2700 - 5400 K.

Table 1: Lighting ratio of neutral and incandescent lightings

Color Temperature [K]	Neutral Lighting [%]	Incandescent Lighting [%]
2700	0	100
3000	21	79
3300	67	33
3600	47	53
3900	57	43
4200	67	33
4500	76	24
4800	84	16
5100	91	9
5400	100	0

3.3 Method of extracting lighting close to illuminance sensors using a lighting layout plan

With the Intelligent Lighting System), when estimating the illuminance/ luminous intensity level of influence, the two lights with the most significant influence on the illuminance sensor can be specified based on the size of the regression coefficient. However, as the regression coefficients of other lights have a large degree of error, there are times when the estimated degree of error may occur [12]. With the proposed method, by using the two lights with the largest regression coefficients and the layout plan, the light closest to the illuminance can be accurately extracted.

Fig.2 is a concept diagram of the method described above. As shown in Fig.2, by placing one illuminance

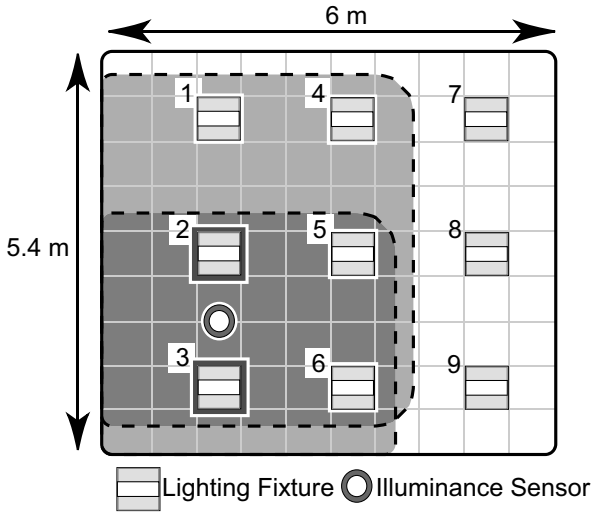


Figure 2: Concept of lighting estimation method

sensor unit, the two lights (lights 2 and 3) with the highest illuminance/luminous intensity influence level can be determined through regression analysis, and these are considered to be adjacent lightings. The lightings that influence this illuminance sensor are estimated to exist near the adjacent lighting. Next, the lighting near the adjacent lighting is extracted using the lighting layout plan. Here, the direct line distance from the adjacent lighting to each lighting incidence is calculated, and the lighting close to either adjacent lighting is sought. As the direct line distance on the diagonal line is the maximum distance between the lighting, distances below this distance can be considered to be close lighting. Therefore, in the case of the environment shown in Fig.2, the lighting group (lighting 2: lighting 1, 3, 4, 5, 6, and lighting 3: lighting 2, 5, 6) with a close straight line distance to the adjacent lighting can be extracted. Thereupon, by taking the intersection of the lighting groups extracted from the lighting layout plan, the closest lighting to any of the adjacent lighting (lighting 5 and 6) can be extracted, and this is the adjacent lighting. The extracted adjacent lighting (lighting 2 and 3) and adjacent lighting (lighting 5 and 6) are the lighting close to the illuminance sensor.

3.4 Proposed control method

The flow of color temperature control in the proposed method is as follows. As a maximum of six lights can influence the illuminance sensor, the upper 4 - 6 lights with the greatest illuminance/luminous intensity to the sensors are lit to the target color temperature..

1) Regression analysis is carried out based on the illuminance and luminous intensity, and the illuminance/ luminous intensity influence level is calculated.
2) The adjacent lighting close to the illuminance sensors are extracted using the illuminance/ luminous intensity level of influence

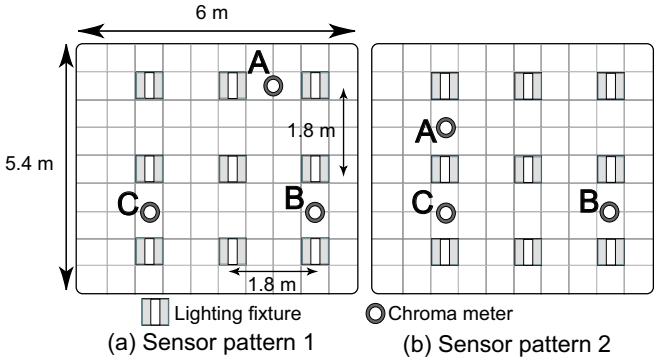


Figure 3: Experiment environment

3) It is judged whether the adjacent lighting influences multiple illuminance sensors
4) Where there is lighting that meets the conditions of item (3), the average mired value is calculated from the target temperature level of the corresponding illuminance sensor, and it is lit to this color temperature.
5) The adjacent lighting is extracted using the lighting layout plan.
6) It is judged whether the adjacent lighting influences multiple illuminance sensors
7) Performs same controls as item (4)
8) Where there is lighting that does not influence any illuminance sensors, the average mired value is calculated from the target color temperature for all illuminance sensors and it is lit to this color temperature

4. Verification experiments

4.1 Experiment outline

Verification experiments were carried out in order to demonstrate the effectiveness of the proposed method. The laboratory is a space measuring 5.4 m $\times$ 6.0 m $\times$ 2.5 m, and during the experiments, a black curtain was erected to avoid disturbance from light reflected from the wall surface. Furthermore, nine SHARP LED lights capable of light control up to 2700 - 5400 K, and 26 - 2040 cd, and three Konica Minolta Co., Ltd chroma meters (CL-200) were set up as seen in Fig.3. In the verification experiment, where the distance between the illuminance sensors was separated and they were adjacent, verification was performed in two kinds of environments to clarify the distance between illumination sensors and the implementability of color temperature. The color temperature information obtained from the color illuminance sensor was only used to clarify the implementability of the color temperature.

Furthermore, a comparison experiment was performed between a method using a chroma meter and the proposed method. When extracting lighting that influences illuminance sensors in the proposed method, we use an illuminance/luminous intensity influence level calculated from 120 seconds of data. This is the learning time adopted by Intelligent Lighting Systems implemented in actual offices.

The initial stage of each lighting incidence is verified using the maximum lit luminous intensity. The color temperature was approximately 3800 K when neutral white, while incandescent lighting used the maximum luminous intensity. Furthermore, the target illuminance and target color temperature for sensors A, B, and C were 500 lx - 3600 K, 350 lx - 3200 K, and 650 lx - 4000 K, respectively.

4.2 Verification when the illuminance sensors are separated

We verified the implementability of illuminance and color temperature when the illuminance sensors are separated by more than 2.8 m, as seen in Fig.3-(a). The histories of illuminance and color temperature both when using the chroma meter and when not using the chroma meter are shown in Fig.4 and 5. In the illuminance history for the proposed method shown in Fig.4-(b), as the change in illumination perceptible to humans is $\pm$ 50 lx [13], we could confirm that the target illuminance for all was satisfied. Furthermore, in the color temperature history for the proposed method shown in Fig.5-(b), it was confirmed that the color temperature for sensor A achieved the target value of within $\pm$ 100 K, and sensors B and C achieved within $\pm$ 150 K. Based on the above results, it was confirmed, in regard to the implementability of color temperature, that the control algorithm integrated into the proposed method could achieve the target color temperature without using a chroma meter.

Next, we verified the light situation for each lighting incidence. The lighting luminous intensity and color temperature in which the illuminance and color temperature were stable lasted 600 seconds, in each method, and is shown in Fig.6 and 7. Fig.6 shows the light distribution diagram for the method using the chroma meter and Fig.7 shows the method not using the chroma meter. The size of the circle in which each light is centered shows the luminance in the vertical downwards direction for each lighting incidence, and as the color of the circle darkens, the color temperature decreases. Furthermore, the color temperature of each lighting incidence is described directly below each incidence.

We confirmed from Fig.6, which shows the light patterns for the method using the chroma meter that the color temperature is different for each lighting incidence when using the chroma meter. This is because a control is performed using a method of optimization based on the information obtained from the chroma meter in order to satisfy the required color temperature. In contrast, in Fig.7 that shows the light patterns in the proposed method, it lights according to the target color temperature of the illuminance sensor corresponding to the adjacent lighting near the Illuminance sensor. Furthermore, we calculated the average mired value for the illuminance sensor corresponding to the adjacent lighting and confirmed that it is lit at that color temperature.

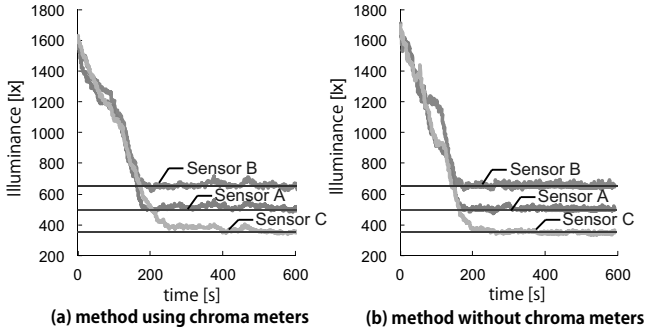


Figure 4: History of illuminance (sensors placed away)

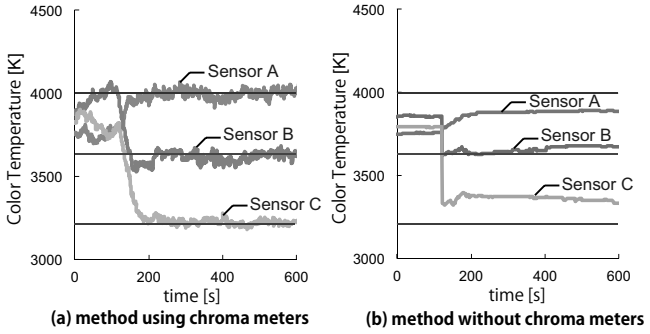


Figure 5: History of color temperature (sensors placed away)

4.3 Verification when the illuminance sensor is neighbor

We verified the various lighting situations where illuminance sensors A and B were located within 1.8 m of each other as shown in Fig.3-(b). Furthermore, the history of illuminance and color temperature for the method where the chroma meter is used and the proposed method where the chroma meter is not used are shown in Fig.8 and 9. The light luminous intensity and color temperature after running for 600 seconds for each method is shown in Fig.10 and 11.

Fig.3-(b), sensor C is separated from the other illuminance sensors, when the illuminance sensors are neighboring. In this way, as in Fig.5, we confirmed that the proposed method can be implemented within $\pm$ 150 K of the target color temperature. Furthermore, in the proposed method for sensor A, it can be implemented within $\pm$ 150 K of the target color temperature, as well. On the other hand, it can be confirmed from Fig.5 that as sensor A is neighboring to sensor B, the sensor A target temperature is interfering with sensor B. As in the proposed method, lighting control is performed without obtaining color temperature information, and the result is that the target color temperature for sensors A and B affect each other.

Based on the above, we confirmed that when the illuminance sensors are adjacent and at a distance of within 1.8 m, it is possible to approach the target color temperatures

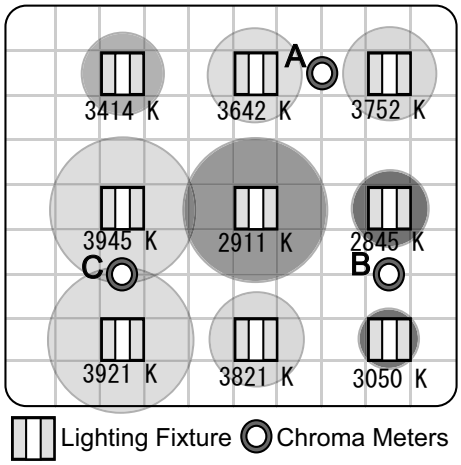


Figure 6: Lighting distribution using chroma meters (sensors placed away)

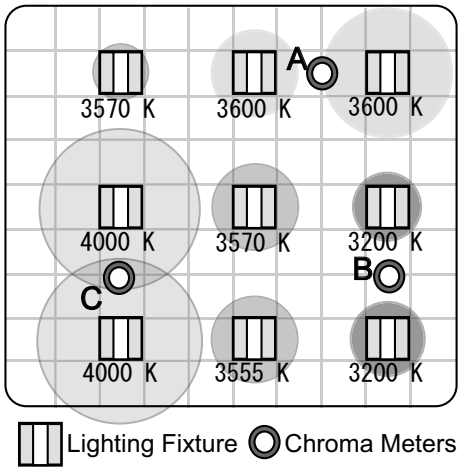


Figure 7: Lighting distribution without chroma meters (sensors placed away)

for each sensor using the proposed method, but that the implementability of the color temperature is poorer than that in the previous section.

5. Conclusion

In this study, we have proposed a method, through the Intelligent Lighting System, of achieving the illuminance and color temperature levels required by each office worker without using a chroma meter. Based on the verification results, we see that when illuminance sensors are located at the same distance as the lighting placement interval, the color temperatures for the neighboring illuminance sensors interfered with each other. In contrast, when the illuminance sensors are located at approximately twice or more the lighting placement intervals, it was possible to achieve the illuminance and color temperature required by the individual office workers. Using this method, the chroma meter is unnecessary and the operation of setting the color temperature of each light separately is also not required. From these points, we

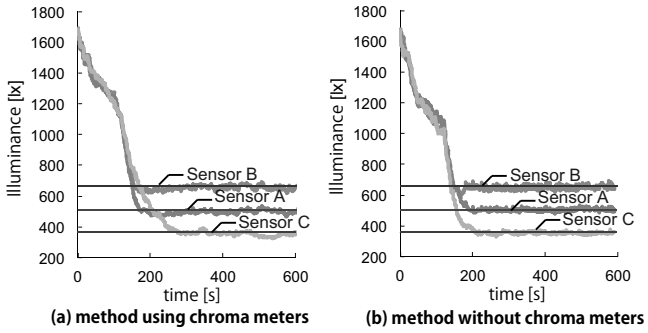


Figure 8: History of illuminance (sensors placed close)

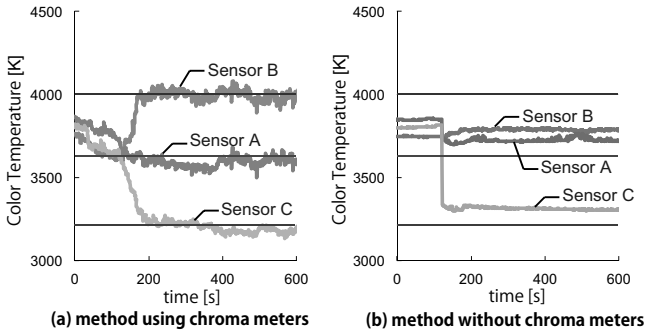


Figure 9: History of color temperature (sensors placed close)

can expect that by integrating the proposed method into an Intelligent Lighting System, individual illuminance and color temperature can be provided to each office worker.

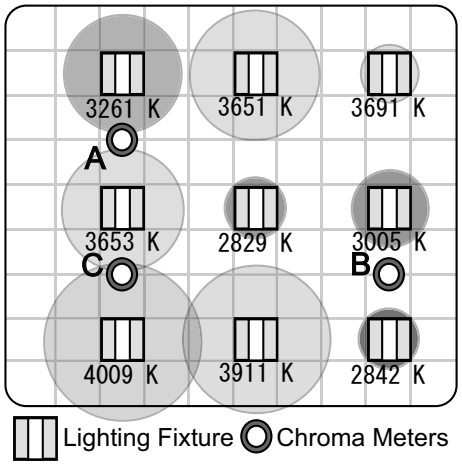


Figure 10: Lighting distribution using chroma meters (sensors placed close)

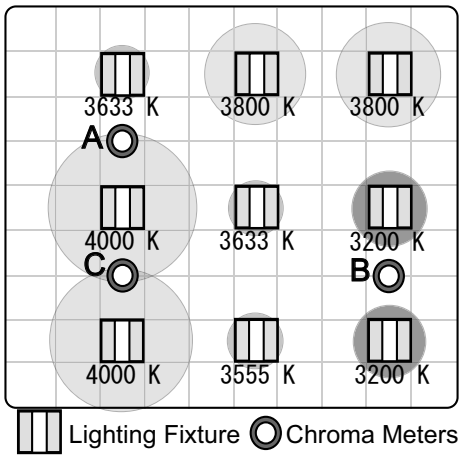


Figure 11: Lighting distribution without chroma meters (sensors placed close)

References

[1] Olli Seppanen, William J. Fisk: A Model to Estimate the Cost-Effectiveness of Improving Office Work through Indoor Environmental Control, Proceedings of ASHRAE, 2005

[2] M. J. Mendell, and G. A. Heath: Do indoor pollutants and thermal conditions in schools influence student performance? A critical review of the literature, Indoor Air, Vol.15, No.1, pp.27-52, 2005

[3] Peter R. Boyce, Neil H. Eklund, S. Noel Simpson: Individual Lighting Control: Task Performance, Mood, and Illuminance, JOURNAL of the Illuminating Engineering Society, pp.131-142, Winter 2000

[4] PHILIPS. ActiViva club. http://www.lighting.philips.co.jp/

[5] Hajimu Nakamura, Yoshinori Karasawa, Tadashi Kuwata, Hidemi Kashiwagi, Isao Maeda, Mitsuo Yokoyama, Mineo Suzuki, Mari Miyata, Psychological influence of illuminance and color temperature on atmosphere, Proceeding of Annual Conference of The Illuminating Engineering Institute of Japan, Vol.29, pp.265, 1996

[6] Y.Taniguchi, M.Miki, T.Hiroyasu, M.Yoshimi, Preferred illuminance and color temperature in creative works, 2011 IEEE International Conference on Systems, Man, and Cybernetics (SMC), pp.3255-3260, 2011

[7] Maiko Ashibe, Mitsunori Miki, and Tomoyuki Hiroy- asu. Distributed optimization algorithm for lighting color control using chroma sensors. pp. 174-178, 2008.

[8] S.Inoue, MITSUBISHI ESATE COMPANY Ltd,Towards ”the City of the Future, http://www.jetro.org/documents/green innov/Shigeru Inoue Presentation.pdf

[9] Fumiya Kaku, Mitsunori Miki, Tomoyuki Hiroyasu, Masato Yoshimi, Shingo Tanaka, Takeshi Nishida, Naoto Kida, Masatoshi Akita, Junichi Tanisawa, Tatsuo Nishimoto, “Construction of intelligent lighting system providing desired illuminance distributions in actual office environment,” Artifical Intelligence and Soft Computing, vol. 6114, pp. 451-460, 2010.

[10] Shingo Tanaka, Mitsunori Miki, Tomoyuki Hiroyasu, and Mitsuharu Yoshikata. An evolutional optimization algorithm to provide individual illuminance in work- places. pp. 941-947, 2009.

[11] T.Shikakura, H.Morikawa, and Y.Nakamura, “Research on the Perception of Lighting Fluctuation in a Luminous Offices Environment, Journal of the Illuminating Engineering Institute of Japan”, Vol.85 5, pp.346-351, 2001

[12] M.Miki, Y.Azuma, H.Ikegami, An Extraction of Influential Lightings for Illuminance Sensors and Lighting Off Mechanism in An Intelligent Lighting System, The 15th International Conference on Artificial Intelligence 2013, 2013

[13] T.Shikakura, H.Morikawa, and Y.Nakamura,Research on the Perception of Lighting Fluctuation in a Luminous Offices Environment,Journal of the Illuminating Engineering Institute of Japan, Vol.85 5, pp.346-351, 2001

NeuroSNP: Tool to Filter SNPs in Whole Genomic DNA

B. Zonovelli[1], **C. C. H. Borges**[1], **and W. A. Arbex**[2]
[1]Federal University of Juiz de Fora, Juiz de Fora, MG, Brazil
[2]Brazilian Agricultural Research Corporation, Juiz de Fora, MG, Brazil

Abstract— *The classification task the punctual differences found when comparing the genomes of two distinct individuals is complex. In draft genome this difference can be 7-8x higher than a final version itself genome. This difference is a noise, this can be in the input or output label. The presence this noise difficult the process of the classification the differences how being single nucleotide polymorphisms (SNPs) or not. The SNPs are specific differences between different pairs of aligned sequences and consist of the most common type of genetic variation. To task of the classification was used the technique of the neural network with excellent results for whole genome the different genomes..*

Keywords: Bioinformatics, Genomic DNA, SNP Filtering, Machine Learning, Computational Intelligence, Neural Network

1. Introduction

In this work we discuss the capacity the neural network to identify correctly the single nucleotide polymorphisms (SNPs), distinguishing them from mismatch that can be error. The SNP that are specific differences between different pairs of aligned sequences and consist of the most common type of genetic variation [5]. SNPs can cause functional or phenotypical alterations, which, in turn, can result in evolution or biochemical consequences on individuals where the SNPs are manifested.

It is known that, at each stage destined to DNA sequencing an error can be introduced, even in small portions, introducing noise which could lead to erroneous identification of an SNP. To solve this problem, filters have been constructed, linked or not of software the alignment and mapping of sequences, which are used in the assembly of the genome of a particular organism. In this scenario, there is the software MAQ (Mapping and Assembly with Quality) [13]. This program aims at mapping and assembly of complete genomes sequenced by NGS platforms, in addition to having a SNPs filter coupled.

2. Background

With the growing advancement of genome sequencing platforms, the need arises for computational models capable of analyzing, effectively the large volume of available data. The correct identification of SNPs is an important step for their use in other studies, however, for better identifying can be necessary a filtering process. The filtering of SNPs in data from next generation platforms presents as an area of research where there is a need for new developments. Specifically, filters based on strategies at machine learning and computational intelligence, which basically are not exploited. The main difficulty in the correct identification of SNPs reside in the noise present in the input and the output, ie, the values used for the training of machine learning tools contains noise, same after its application.

2.1 next-generation sequencing

In the late 70's, was developed two classical DNA sequencing methods, the chemical degradation method or procedure of [18] and the method enzymatic degradation or Sanger procedure [24]. Such techniques employ chemist process to identify and determine the order of nitrogenous bases in the DNA of an organism. But because of the ease of interpretation of the data derived from the method developed by Frederick Sanger, his technique was widely used by stakeholders in DNA sequencing. However, the high cost and low efficiency inherent in this method has become a limiting factor for projects aimed at large-scale genomic sequencing [4].

From 2005, the sequencing technologies have undergone a considerable advance, reducing costs and increasing capacity for sequencing. Today, the new sequencing platforms known as next-generation sequencing (NGS), became effective options for routine use in sequencing projects and ressequencing individual genomes [25], [10]. These platforms are a powerful alternative to the detection of variations between the target genome and the reference, to the studies of structural and functional genomics [17], [19]. Are able to generate information of millions of reads in a single run [27], [4].

The difficulty in defining the presence of noise in the filtering process of the SNPs is inherent in the generation process of data, from sequencing until assembly process. Draft genome assembly, in general, provide versions, known as drafts before the final version, as with the human genome project [6]. The draft genome contain more errors than the final version, so that the use of a filter may be of great importance. To assess the impact of noise, and the ability of machine learning tools for circumvents this work was being undertaken on the draft genome of an animal of the Fleckvieh breed, using as reference the bovine genome bosTau4.0 [12].

2.2 Noise Identification

The task of completing assembly of a genome, pass by the process of sequencing, alignment and assembly of reads. In each of these stages is there the insertion of noise, which in the discovery phase hinder the correct identification of a SNP. At this stage any difference between the sequences is an mismatches, however some of these differences are SNPs and the other no. Despite being computationally a difference, define when mismatches is or not a SNP is a complex task, this definition is at the discretion of the filtering step. Regarding the NGS platforms is known that errors are introduced in the range 0.1% of 1% [9].

By aligning two sequences with the reference genome to generate consensus, the software alignment and assembly can identifies a mismatches in the first positions of the fragment. However, this may be the best alignment for the fragment data. This situation typically occurs when used reads are short, which is useful when is used the NGS platforms. The mismatches generated by this alignment may be the result of an error in the sequencing step, or a SNP. The mismatches can be generated by an alignment error resulting from sequencing error [16].

3. Methods

3.1 Reassembly

The processes of discovery and filtering are executed always after assembly of the genome, so the need to perform this phase of the project. A project of assembling a genome can be extensive. So, to make it possible to follow the steps of discovery and filtering of SNPs, the genomes of two distinct species were reassembled using the MAQ software. The stage of discovery generates the necessary file for the study of false positives. The SNPs filter of the MAQ software, SNPfilter, and the filter implemented in this article use this file to perform the filtering step.

The genome is the an animal of the species Bos taurus, Fleckvieh breed, which was sequenced using NGS [8]. The reassembled genome was sequenced using the Solexa Genome Analyzer II platform, generating 24 giga bases of sequence with size 36bp mate-pair after trimming, resulting in an assembly with 7.4x average depth. Was used as reference the assembly bosTau4.0 the bovine genome, sequenced by Baylor College of Medicine and made available by the University of California at Santa Cruz [12]. Being found 6,599,143 SNPs in the discovery and 2,162,709 after filtering step.

3.2 Neural Network

For the filtering work was chosen the technique of artificial neural networks that comprise a computational resource often used for the solution of various problems, including bioinformatics and biological problems [26], [11], [7], [22], [15], [3]. However, in research carried no were references found to the use of neural networks to filter SNPs in complete genomic DNA, sequenced in NGS platforms. The artificial neural network was the technique of computational intelligence chosen, because her classification ability is one of its main characteristics and can therefore be used in the assembly of a filter, which is nothing more than a classifier.

3.2.1 Resilient Network

The concept of resilience, or resilient, can be defined as something or someone with the ability to adapt to an unexpected situation, so flexible. Applying this concept to the network, a resilient network has the ability to adapt in the best way, the data presented. Resilient networks do not require that the rate of learning to be informed, because it is updated by the learning algorithm developed. Thus solving one of the main difficulties presented, which is the definition of the learning rate [1]. The algorithm used to build a resilient network was (**R**esiliente back**prop**agation) [23]. The parameters used by the algorithm, by definition of the authors, are started with: $\Delta_0 = 0.1$; $\Delta_{max} = 50.0$; $\Delta_{min} = 1e^{-6}$; $\eta_+ = 1.2$; $\eta_- = 0.5$.

3.3 Filter Implementation Using Neural Networks

Classification models supervised for filtering SNPs are not yet explored in the literature. Possible reasons are the difficulty of having a reliable database for both false positives how to for SNPs proven to obtain the assumption of generalization. Thus, any attempt to use supervised classification for filtering SNPs must necessarily pass through the definition of a strategy for the construction of training base and / or determination of the class of instances, these strategies aim to minimize the noise present in the training set and / or testing. To mount the databases three strategies were defined: i)Using a pre-filter for determining the classes; ii) construction of specific bases to maximize the power of generalization of supervised classification tool; iii) construction of specific bases using some rules of pre-filtering. In order that each of the strategy generated a different model.

It is noteworthy that both strategies suffer from problems relating to noise in determining the class (or false positive SNPs) of each candidate. Thus, we seek to evaluate the potential of one of the models of neural networks to circumvent this feature of the problem of detecting SNPs. The library Fast Artificial Neural Network (FANN) [20] was used for the encoding of the model. The library allows the creation of a network using a number of different programming languages, the C language was chosen for this work. The process of assembling a computational model for filtering SNPs can be described in the following steps:

1) Riding a dataset for training and testing;
2) Training several networks with the given set of data;
3) Analysis of the results obtained with the training and choose the best network;

4) The filter program, reads the selected network and filters SNPs;
5) Analysis of the results provided by the filter;
6) When necessary, redo the process.

The definition of the structure of the network, passes by the choice of activation function that best adapts to the problem. Among the various activation functions available, four were chosen. The logistic sigmoid due to its extensive use. The Gaussian function to be of general use. The sigmoid Elliot function can have a lower mathematical complexity, so that it is expected to be faster than the sigmoid logistic, and the Elliot symmetrical.

The topology used consisted of a network with ten neurons in the input layer, a hidden layer with twenty neurons, this value was chosen randomly. The output layer with one neuron, initially binary, classifying SNPs in 0 or 1, simulating the behavior of the MAQ software filter. For the training algorithm was used resilient backpropagation, making it not necessary to define several learning rates. The data set used was reassembled genome of the Bos Taurus. The network was implemented following the code patterns informed at the manual FANN [20]. Is presented, then the first model.

3.3.1 First Model

The first model based on machine learning to be presented for filtering SNPs is part of of the line the classification with noise, where basically the framework of the classification of SNPs is presented polluted by noise. This noise is introduced by a pre-classification needed to be used for classification of candidate SNPs. Logically, this pre-classification is failures, otherwise would not require an adjustment by means of a strategy of machine learning.

In this work, it is natural to use the filtering achieved by the logical expressions of the MAQ filter as a first assessment of SNPs and false positives. Thus, one can form a training base with the output of the instances being defined by MAQ output filter. The noise of the instances that are misclassified by the filter and thus hinder the process of learning the strategy used in this case neural networks.

The expectation is that the use of additional variables not used in MAQ filter as well as the potential of a neural network with one or more inner layers represent nonlinear functions can generate a classification hypothesis that can be properly generalize the filtering minimizing the effect classroom noise levels.

The dataset used was extracted from the output files of the MAQ software. The first file is derived from the discovery stage and the second stage of the filter. In these steps the reassembled genome Bos Taurus was used, then the first file had $\approx$ 7 millions the SNPs and the second $\approx$ 2 millions. The file used for automatic training of the neural network by FANN library has a specific format. Therefore, we developed a PHP script that scans the file first randomly selecting 4,000 SNPs per chromosome for the training set and 2,000 for the test set. If the SNPs are present in the second file it was indicated as 1 or how 0 otherwise. Getting this set so as SNPs 1 and 0 as error.

The setting values of 4,000 and 2,000 for samples SNPs training and testing. Were chosen after initial tests with several different values. The values tested were from 2/3 and 1/3 of the total, up to only 100 and 50 SNPs for chromosome, and the value of 4,000 and 2,000 were very close to the results obtained by the values of 2/3 and 1/3 total.

The first model was the basis for defining the best network structure for the development of the work, the resilient backpropagation algorithm is efficient, and the activation function ELLIOT was the one that achieved the best results, with errors next to 0.6%. However, this first strategy considered MAQ as correct, and suffered from the presence of noise in the data set used for training. The expectation is that the generation of new datasets, based on knowledge gained on the functioning of alignment and assembly programs, reducing this noise occurs.

The filter the MAQ software is binary, classifying SNPs as 0 or 1, ie, true or false positive. Thus the characteristic function is a threshold function, However, the implemented and trained network achieved better results with the sigmoid function, that ranked SNPs in the range [0, 1], this characteristic has been exploited generating an important feature of the filter, defined here as a restriction. Were defined three constraints, LOW, MEDIUM and HIGH. The restriction LOW allows any SNPs is classified by the network, if it has a different value than 0. MEDIUM constraint behaves like the threshold function, ie all SNPs rated value above 0.5 is considered true. HIGH constraint, only ranked SNPs with a value of 1.

3.4 Second Model

Although the first model has shown interesting results with respect to the classification process using polluted classes, initial experiments indicated that the neural network did not provide an generalization ability to obtain a better determination of SNPs and false positives by regarding determined MAQ reference filter.

However, it has been the target of a machine learning tool with proper training can be competitive compared to traditional filters. In this second model, we intend to enhance the generalization capability of machine learning model by attempting to diminish the influence of the noise arising from the pre-filtering used.

Thus, we seek to replace the use of MAQ filter on pre-filtering for "rules" more stringent for determining SNPs and especially false positives. The new form of filtering, in this way, will be less sensitive to borderline cases. Thus, these rules shall establish a training base with greater definition mainly of false positives. It is hoped that in this way, has

become benefits in the process generalization with greater ease in learning and discrimination of new instances.

The determination of the rules for generating the class of basic training, like any filter, will be subject to noise. The expectation is that they will become stricter in the detection of false positives that, for this particular problem, is the most data. Thus, we expect a better reflection of the results, with the generalization by providing a more accurate filtering.

The data sets used were designed based on two rules. The first rule defines a group of SNPs with high confidence, and the second group with low trust, where trust is understood by how much an mismatches could be considered a SNP. The objective is to define a SNP with high confidence as being true, and a SNP with low confidence as a mistake. SNPs that are not in either group will be classified by the network. Based on the above tests, the same topology has been defined for the second model, using the sigmoid activation function Elliot, with time constant of 0.5. Thus, as in the previous step, the genome reassembled Bos Taurus was used.

3.4.1 Generation of Data Sets

The first rule for the generation of the data set used in training is the choice of SNPs with high confidence. The rule was set after parameter analysis and genome study. The parameters are the twelve columns of the output file MAQ software. The first two columns identify the SNPs, therefore, are not used in the filters. The other 10 have diverse information, four of which inform the nucleotides present in the reference genome and consensus genome, so these columns are not considered when selecting high confidence SNPs. The mean value between the second and third best call was not used because it is a feature of the MAQ software information. The value of hit was not used because, according [13], this variable can create doubt when the filter therefore is among the parameters presented to the neural network, however, but was not considered in the selection of SNPs for mounting data set.

The choice of SNPs with high trust followed the following criteria: depth greater than or equal to 6 (the bovine genome has an average depth of 6.98 so the choice of SNPs that are near to or above); Phred-like greater than or equal to 20; quality and mapping quality on the flank of 6 greater than or equal to 50, this value was also used by [14] in their work. Therefore, 429.078 SNPs were found in the total pool, originating from the discovery of the software before MAQ filter which satisfy these criteria file.

The construction of the group of SNPs with low confidence used the same parameters in the group with high confidence. The criterion for the determination of SNPs low confidence consists in the withdrawal of the total data set, SNPs having at least one parameter equals 0, which satisfy the criterion 1,821,527 SNPs.

The data set consists of a mounted file workout with 116,000 entries and a test file with 58,000, consisting in a balanced manner, ie, deriving half of the dataset with high trust and the other half of the set with low confidence. Moreover, we used the same number of SNPs for each of the 29 chromosomes present in bovine genome studied.

3.5 Third Model

In this third model, it is intended, as in the second, improve the generalization of the machine learning model. The difference between the second and third model is the rule for selecting SNPs with high confidence that this model is less restrictive. The difference is also in the non-consideration of SNPs that have a null parameter. The data set was constructed based on two rules that are described below. As in previous cases, the same topology defined for the first and second models will be used. Also as in the previous step, the genome reassembled Bos Taurus was used.

3.5.1 Generation of Data Sets

The choice of SNPs with high trust followed the following criteria: greater than or equal to 6 deep; Phred-like greater than or equal to 20; mapping quality greater than or equal to 40 and greater quality in the flank or equal to 20. We notice that the values used are the same as the MAQ filter, except for the depth value.

SNPs with low trust, do not have different criteria for the second and third model. That is, the same SNPs were considered to be of low confidence used in the assembly of the sets of data from both models, the second and third.

The two new sets of data have different contents, however, were set identically. Each dataset consists of a training file with 116,000 entries and a test file with 58,000. As the second model bases are balanced with half the originating data set with high reliability and half of the assembly with low confidence. Remains the same representation bases for the 29 chromosomes present in the bovine genome studied.

3.6 Training Second and Third Models

Both models were trained, each with its particular set of data. Ten held trainings, selecting the best for the construction of training and testing graph. Unlike the first model graph, you can see a small increase in test curve, while the curve of workout keeps reducing. This phenomenon is indicative of stopping the training process. Elliot function, converged quickly to a great result, as expected.

The algorithm implemented for training saves the network with their synaptic weights, when the error in the test is less than the previous error, this time indicated by the blue arrow. Thus, the parameters of the neural network with the best performance is stored for later use. Even if the algorithm does not stop training, the network is stored one that had the lowest error on the test.

The results in the training phase of the three models using different databases, indicate a very different behavior between them. It is not trivial to identify which model

provides higher quality results. Intends to apply neural networks generated for each model in complete genomes can thus get a better indication regarding the quality of the bases used in the generation of neural networks. The following shows the construction of the filter that uses neural networks trained for subsequent application to whole genomes.

3.6.1 Implementing NeuroSNP

After completion of the training step, the next step was the development of the filter itself. The filter consists of an algorithm that reads one trained network and filter the SNPs of the source file, generating a new file with the SNPs that passed the filter. With all the initial tests completed, the filter based on computational intelligence techniques, called NeuroSNP, was completed and the call itself now requires four parameters, explained in the table 1

Table 1: Parameters NeuroSNP

Parameter	Description
-n	Output file of training the network. This file contains the structure of the trained network
-o	Output file, the SNPs considered positive are saved in this file.
-d	Source file of the SNP by default and the output file MAQ Software.
-r	Restriction (0 - LOW, 1 - HIGH, 2 - MEDIUM).

The NeuroSNP gets the table parameters 1 at the time of your call. The first action of the filter is to reassemble the neural network with its weights. The parameter $-n$, contains the path to the output of the network training phase, and this file used to reassemble the network. Then the filter starts reading the file with SNPs, parameter $-d$. Each SNPs contained in the file has 12 columns with their identification and characteristics of assembly and alignment, then the filter reads the columns and informs from data contained in the network. The network returns an output value if the returned value satisfies the constraint informed, parameter $-r$, then the SNPs with your data and stored in the output file, parameter $-o$.

4. Results

When these tests were performed, contained in NCBI 13,704,221 SNPs submitted and 3,003 valid for animals breed Bos Taurus, last accessed 02/2013. are extracted the nucleotide sequences of the dbSNP, used for locally building the database used by software BLAST. The database contains all SNPs from NCBI to breed animals Bos Taurus.

To build the database, the BLAST uses, were assembled reads with 120bp in size, where the polymorphic variation present in each of SNPs generated a new reads, with the SNP at position 60. To analyze the results, only the alignments without gaps or mismatechs were accepted with 100% similarity, and always in the direction $5' \rightarrow 3'$ and size of 120bp. To compare different filters was used statistical measure known as odds ratio (OR) [2].

4.1 Results Obtained by the First Model

The first model used four different activation functions with three time constants. However, only the best structure of the first model was chosen. The structure chosen uses the Elliot activation function and time constant equal to 0.5. The results of the first model are shown in Table 2. The selected networks in this model, the NeuroSNP1.A and NeuroSNP1.B, obtained the following training errors: 0.003560 e 0.011363.

When analyzing the table 2 it is possible to see that the value 5.3746 obtained by calculating the OR for SNPfilter was only exceeded by the value of 6.0669 NeuroSNP1.B with HIGH restriction. However, the MEDIUM and LOW restrictions, have lower values OR (4.8398 and 3.4714), indicating that the NeuroSNP1.B was less efficient. For the increasing number of filtered SNPs or sample, did not generate an equal increase in the number of valid alignments. The same behavior is observed for the NeuroSNP1.A, with a value of 5.0302 OR with HIGH restricted, and lower values for the MEDIUM and LOW restrictions (4.3026 and 3.5126).

Table 2: Comparison between Model First and SNPfilter.

	SNPs	alignments	OR	CI
MAQ	6,599,143	2,162,709	-	-
SNPfilter	2,174,341 (32.95%)	1,573,706 (72.77%)	5.3746	5.3565 - 5.3929
		NeuroSNP1.A		
HIGH	1,878,258 (28.46%)	1,334,174 (61.69%)	5.0302	5.0124 - 5.0480
MEDIUM	2,243,455 (34.00%)	1,519,172 (70.24%)	4.3026	4.2887 - 4.3166
LOW	2,725,354 (41.30%)	1,720,551 (79.56%)	3.5126	3.5022 - 3.5229
		NeuroSNP1.B		
HIGH	1,557,915 (23.61%)	1,164,256 (53.83%)	6.0669	6.0429 - 6.0910
MEDIUM	2,001,787 (30.33%)	1,405,903 (65.01%)	4.8398	4.8232 - 4.8565
LOW	2,809,366 (42.57%)	1,765,863 (81.65%)	3.4714	3.4613 - 3.4815

Another factor to be observed is the CI, that in all cases remained low, almost zero if the value is rounded to one decimal place only. The small CI indicates that the OR calculated is accurate, being extremely significant.

Thus, using this first model only was superior to SNPfilter when used HIGH restriction, the value obtained by OR NeuroSNP1.A is close to the SNPfilter, and the NeuroSNP1.B is superior. Therefore, the network trained using the class of each instance as the output SNPfilter not shows promise for the classification of mismatches. However, the expectation is that neural networks are able to perform this task satisfactorily, training with the most promising bases being only necessary.

4.2 Results Obtained by the Second Model

The second model was trained with a set of data assembled from the rules given in section 3.4.1. The table 3 shows the comparative results between the second model and SNPfilter. The selected networks obtained the following errors: 0.000646 for NeuroSNP2.A and 0.000791 for the NeuroSNP2.B. As you can see the two values are very close. Analyzing table 3, is possible to see that both networks can classify the mismatches very efficiently, obtaining higher

ORs then of SNPfilter values in the three restrictions, and primarily values near ORs between the restrictions, indicating that the second model is stable.

As can be observed, the correct classification of mismatches is a difficult task because two networks with next errors have very different final results. The search space traversed by the network in the optimization of the error may have many minimum local , possibly close to the global minimum, thus generating networks with low errors, but with variations in the classification stage. Another hypothesis is that the proximity between the candidates is large, making two networks, with errors very close, may have different behaviors for the same dataset. For this analysis it suffices to observe the sample size, which has a moderate variation in NeuroSNP2.A, and A larger variation for NeuroSNP2.B. As in the first model, the second model networks have small CIs, showing that OR calculated is accurately.

Table 3: Comparison between SNPfilter and the Second Model.

	SNPs	alignments	OR	CI
MAQ	6,599,143	2,162,709	-	-
SNPfilter	2,174,341 (32.95%)	1,573,706 (72.77%)	5.3746	5.3565 - 5.3929
		NeuroSNP2.A		
HIGH	209,875 (03.18%)	164,320 (07.60%)	7.3993	7.3220 - 7.4774
MEDIUM	398,005 (06.03%)	308,975 (14.29%)	7.1191	7.0649 - 7.1736
LOW	658,551 (09.98%)	507,243 (23.45%)	6.8769	6.8359 - 6.9180
		NeuroSNP2.B		
HIGH	81,797 (01.24%)	61,480 (02.84%)	6.2074	6.1092 - 6.3072
MEDIUM	408,590 (06.19%)	314,781 (14.55%)	6.8834	6.8321 - 6.9350
LOW	1,143,865 (17.33%)	853,942 (39.48%)	6.0420	6.0148 - 6.0694

The use of the second model proved to be more promising than the first. It is noteworthy, also, that the second model features superior results to those obtained using the SNPfilter . However, as the error observed in the two networks is next, determining which network is best for classification stage is not a trivial task. Case the need has a sample more controlled and with larger information NeuroSNP2.A is better, and in the case of a smaller sample that keep the amount of information present at NeuroSNP2.B appears more promising.

4.3 Results Obtained by the Third Model

The third model, was trained with a base mounted on the rules presented in section 3.5.1. The table 4 shows the comparative results between the neural networks in relation to SNPfilter. After training, the following error values were obtained: 0.002003 for NeuroSNP3.A and 0.002167 for a NeuroSNP3.B. As you can see, again the two values are very close.

When analyzing the table 4 notices a very behavior close between this model and the first, regarding classification ability of SNPs. Both models are little informative as indicated by the value of OR that oscillates with the increase in sample size. It is important to note that despite having a higher OR SNPfilter that the restriction on the HIGH the NeuroSNP3.A (6.9419), and in the restriction MEDIUM the NeuroSNP3.B (5.8454), the values of the ORs do not have a pattern, showing that the network is not being efficient in the classification process. However, it has behavior similar to the second model in relation to variation in the size of the sample. The CIs obtained in the networks of the third model, as in previous models, were small, showing that the calculated ORs are extremely accurate.

Table 4: Comparison between SNPfilter and networks of the Third Model.

	SNPs	alignments	OR	CI
MAQ	6,599,143	2,162,709	-	-
SNPfilter	2,174,341 (32.95%)	1,573,706 (72.77%)	5.3746	5.3565 - 5.3929
		NeuroSNP3.A		
HIGH	545,227 (08.26%)	420,862 (19.46%)	6.9419	6.8967 - 6.9874
MEDIUM	952,373 (14.43%)	660,245 (30.53%)	4.6363	4.6148 - 4.6579
LOW	2,740,161 (41.52%)	1,715,832 (79.34%)	3.4361	3.4261 - 3.4463
		NeuroSNP3.B		
HIGH	75,242 (01.14%)	46,722 (02.16%)	3.3605	3.3111 - 3.4107
MEDIUM	680,492 (10.31%)	503,720 (23.29%)	5.8454	5.8124 - 5.8785
LOW	1,938,537 (29.38%)	1,362,622 (63.01%)	4.8535	4.8366 - 4.8704

Either way, the supervised classification proved to be a useful tool in the complex task of detecting SNPs. Expectations regarding the universalization of its use in different genomes ie, for which the neural network was not specifically trained, will be evaluated in the experiments following.

5. Conclusion

Computational experiments clearly indicated the potential of the presented learning tool for the detection of SNPs. Their use alone or in conjunction with traditional filters is presented as an alternative for robust determination of SNPs in different genomes. The use of OR showed that the application of the filter increases the chance of finding a positive alignment of SNPs within the sample, and the expectation that this increase reflects the reduction of false positives.

Logically, the construction of training base can be improved mainly in two directions: by defining more specific rules for determining priority of false positives, and the use of SNPs biologically proven for building class of true positives. In any case, the supervised classification proved to be a useful tool in the complex task of detecting SNPs.

This work was presented and developed a computational strategy based on computational intelligence and machine learning, with ability to filter SNPs from whole genomic DNA (NeuroSNP). In the construction process NeuroSNP, Three different models did analyzed each one compared with the reference filter MAQ software, namely SNPfilter. In the genomes evaluated, the NeuroSNP managed to similar or superior results to the MAQ filter.

Acknowledgement

The authors thanks to reviewers who gave useful comments, and would like to express thanks to the National

Research Center of Dairy Cattle (Embrapa Dairy Cattle) of Brazilian Agricultural Research Corporation (Embrapa) for providing database and the provision of the necessary infraestructure to conduct this work; to the Graduate Program in Computational Modeling of Federal University of Juiz de Fora (UFJF) for the academic support; and to the Coordination for the improvement of Higher Level Personnel (CAPES) and the State of Minas Gerais Research Support Agency (FAPEMIG) for the financial support for the accomplishment of this paper.

References

[1] Basheer, I., Hajmeer, M., 2000. Artificial neural networks: fundamentals, computing, design, and application. Journal of Microbiological Methods 43 (1), 3 – 31, neural Computting in Micrbiology.

[2] Bland, J. M., Altman, D. G., 2000. The odds ratio. British Medical Journal 320 (7247), 1468.

[3] Bridges, M., Heron, E. A., OĎushlaine, C., Segurado, R., Morris, D., Corvin, A., Gill, M., Pinto, C., (ISC), T. I. S. C., 05 2011. Genetic classification of populations using supervised learning. PLoS ONE 6 (5), e14802.

[4] Chen, F., Dong, M., Ge, M., Zhu, L., Ren, L., Liu, G., Mu, R., 2013. The history and advances of reversible terminators used in new generations of sequencing technology. Genomics, Proteomics & Bioinformatics 11 (1), 34 – 40.

[5] Consortium, I. H., 2003. The international hapmap project. Nature 426 (6968), 789 – 96.

[6] Consortium, I. H. G. S., 2001. Initial sequencing and analysis of the human genome. Nature 409 (6822), 860–921.

[7] Curtis, D., 2007. Comparison of artificial neural network analysis with other multimarker methods for detecting genetic association. BMC Genetics 8 (1), 49.

[8] Eck, S., Benet-Pages, A., Flisikowski, K., Meitinger, T., Fries, R., Strom, T., 2009. Whole genome sequencing of a single bos taurus animal for single nucleotide polymorphism discovery. Genome Biology 10 (8), R82.

[9] GLENN, T. C., 2011. Field guide to next-generation dna sequencers. Molecular Ecology Resources 11 (5), 759–769.

[10] Gupta, P. K., 2008. Single-molecule DNA sequencing technologies for future genomics research. Trends in Biotechnology 26 (11), 602 – 611.

[11] Heidema, A. G., Boer, J., Nagelkerke, N., Mariman, E., van der A, D., Feskens, E., 2006. The challenge for genetic epidemiologists: how to analyze large numbers of snps in relation to complex diseases. BMC Genetics 7 (1), 23.

[12] HGSC, B. C. o. M., 2007. Ucsc, genome bioinformatics.

[13] Li, H., Ruan, J., Durbin, R., 2008. Mapping short dna sequencing reads and calling variants using mapping quality scores. Genome Research 18 (11), 1851–1858.

[14] Liu, Q., Guo, Y., Li, J., Long, J., Zhang, B., Shyr, Y., 2012. Steps to ensure accuracy in genotype and snp calling from illumina sequencing data. BMC Genomics 13 (Suppl 8), S8.

[15] Long, N., Gianola, D., Rosa, G., Weigel, K., Avendano, S., 2009. Comparison of classification methods for detecting associations between snps and chick mortality. Genetics Selection Evolution 41 (1), 18.

[16] Malhis, N., Jones, S. J. M., 2010. High quality snp calling using illumina data at shallow coverage. Bioinformatics 26 (8), 1029–1035.

[17] Mardis, E. R., 2008. The impact of next-generation sequencing technology on genetics. Trends in Genetics 24 (3), 133 – 141.

[18] Maxam, A. M., Gilbert, W., 1977. A new method for sequencing dna. Proceedings of the National Academy of Sciences of the United States of America 74 (2), 560–4.

[19] Morozova, O., Marra, M. A., 2008. Applications of next-generation sequencing technologies in functional genomics. Genomics 92 (5), 255–264.

[20] Nissen, S., 2005. Neural networks made simple. Software 2.0 magazine (2), 14–19.

[21] Ossowski, S., Schneeberger, K., Clark, R. M., Lanz, C., Warthmann, N., Weigel, D., 2008. Sequencing of natural strains of arabidopsis thaliana with short reads. Genome Research 18 (12), 2024–2033.

[22] Ren, L., Wang, W.-P., Gao, Y.-Z., Yu, X.-W., Xie, H.-P., 2009. Typing snp based on the near-infrared spectroscopy and artificial neural network. Spectrochimica Acta Part A: Molecular and Biomolecular Spectroscopy 73 (1), 106 – 111.

[23] Riedmiller, M., Braun, H., 1993. A direct adaptive method for faster backpropagation learning: the rprop algorithm. IEEE International Conference on Neural Networks 1 (3), 586–591.

[24] Sanger, F., Nicklen, S., Coulson, A. R., 1977. Dna sequencing with chain-terminating inhibitors. Proceedings of the National Academy of Sciences of the United States of America 74 (12), 5463–5467.

[25] Service, R. F., 2006. The race for the $1000 genome. Science 311 (5767), 1544–1546.

[26] Tomita, Y., Tomida, S., Hasegawa, Y., Suzuki, Y., Shirakawa, T., Kobayashi, T., Honda, H., 2004. Artificial neural network approach for selection of susceptible single nucleotide polymorphisms and construction of prediction model on childhood allergic asthma. BMC Bioinformatics 5 (1), 120.

[27] Zhang, J., Chiodini, R., Badr, A., Zhang, G., 2011. The impact of next-generation sequencing on genomics. Journal of Genetics and Genomics 38 (3), 95–109.

Representation and efficient algorithms for the study of cell signaling pathways

A. Doncescu[1], **P. Siegel**[2], **T. Le**[1]
[1]LAAS-CNRS, University of Toulouse, Toulouse, France
[2]Aix Marseille Université, CNRS, LIF UMR 7279, 13288, Marseille, France

Abstract— *Artificial Intelligence and constraint programing find in the formal description of cell's interactions a field of inquiry well suited. Particularly, the questions concerning the representation of incomplete biological knowledge and explanation to some biological effects could be well formulated in A.I. frame. Basically, A.I. plays with small examples. In systems biology the algorithmic complexity is too important to be handled using traditional approaches. This article tries to answer to the question "What is the simplest representation for signalling pathways, reflecting reasonable complexity and tractable algorithms?"*

Keywords: Cell signalling pathways, knowledge representation, non-monotonic logics, default logic, abduction, tractable algorithms.

1. Introduction

For biologists, it is possible to represent a cell and its evolution by a graph that describes the interactions between proteins/genes. This representation can includes mathematical properties as connectivity; presence of positive and negative loops which is related to a main property of genetic regulatory networks. Biochemical reactions are very often a series of time steps instead of one elementary action. Therefore, one direction research in system biology is to capture or to describe the series of steps called pathways by metabolic engineering. All reactions that allow the transformation of one initial molecule to a final one constitute metabolic pathways. Each compound that participates in different metabolic pathways is grouped under the term metabolite.

The study of gene networks poses problems well identified and studied in Artificial Intelligence over the last thirty years. Indeed, the description of network is not complete: biological experiments provide a number of protein interactions but certainly not all of them. On the other hand the conditions and sometimes the difficulties of the experiments involves these data are not always accurate. Some data may be very wrong and must be corrected or revised in the future. Finally the information coming from different sources and experiences can be contradictory. It is the goal of different logics, and particularly non-monotonic logics, to handle this kinds of situation. Afterwards this interaction maps should be validated by biological experiments. Of course, these experiments are time consuming and expensive, but less than an exhaustive experiment.

We focus on four main problems: handling the conflicts which can occur in the gene representation, completing in-silico the gene network, knowledge discovery on these networks and the practical handling complexity of algorithms.

In this paper we present algorithms and formalism in the simplest manner to solve and process problems related to gene networks.

Our approach is based on default logic allowing to handle the incomplete information, and on abductive reasoning to complete the missing information from the gene network. But, even limited to propositional calculus, the theoretical complexity for default logic and abduction revolves around $\sum_2^p$ which is totally unacceptable for real problems. The last part is dedicated to a new language of representation, which seems to be the key to algorithm complexity handling.

2. Signaling Pathway

Nowadays, bioinformatics represents the key field to explain the functionality of life science. To analyze a biological system it is necessary to find out new mathematical models allowing to explain the evolution of the system in a dynamic context or to deal in a simple manner with the complex situations where the human experience overtakes mathematical reasoning. The study of systems biology trough First Order Logic aims at improving the understanding metabolite-proteins interactions by finding out new possible interactions which add consistency to the knowledge. The p53/Mdm2 complex represents the best-studied relationship between a tumor suppressor gene which functions as a transcription factor and an oncogene. Figure 1 given and studied by [1] and [14] is an elementary pathway of interactions around p53 and mdm2. This example explains in a very simple manner the interaction cancer-p53/Mdm2, represented on the form of causality graph, using Default Logic and Abduction.

Through different mechanisms, the ultraviolet UV drives the cell to cancer. This is represented by an arrow : $UV \rightarrow cancer$. On the other hand the UV activate the protein $p53$, $(UV \rightarrow P53)$. This protein activates a protein A $(P53 \rightarrow A)$) and A blocks cancer ($A \dashv cancer$). But, in some conditions, $Mdm2$ binds protein $p53$ and the obtained complex, activates B and B blocks A. We also have C blocks B. We will uses X thereafter.

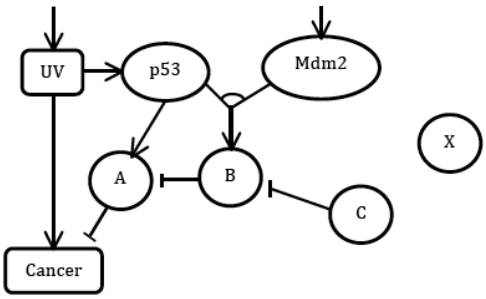
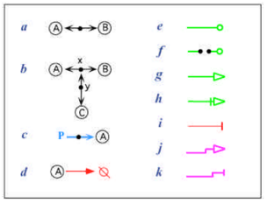


Fig. 1: The Simplified Model of Double Strand Break.



Symbols :
(a) Proteins A and B could link, the knot represents the binding A:B ;
(b) Multimolecular Complex binding ;
(c) Covalent Modification of A ;
(d) Degradation of A ;
(e) Enzymatic stimulation in transcription ;
(f) Autophosphorylation
(g) General Stimulation;
(h) Necessity ;
(i) Inhibition ;
(j) Activation of transcription ;
(k) Inhibition of transcription

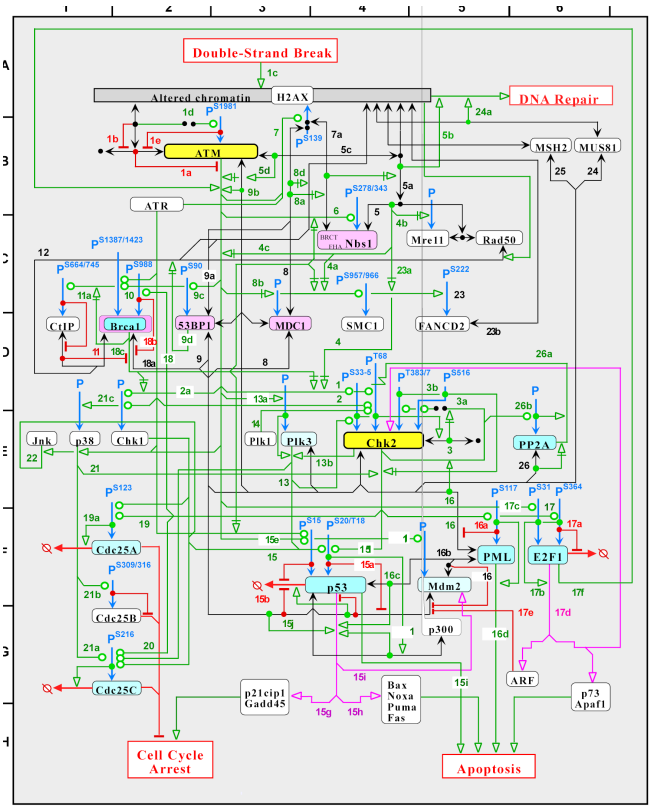


Fig. 2: DNA Double Strand Break Map.

2.1 DNA Double Strand Break Map

A pathway may include thousands of genes, which will pose problems of complexity. We have worked from years on the DNA double strand break map introduced by Pommier and all [16] (Fig. 2). This map is very complicated and it is very difficult to complete it due to the great number of experiments necessary. Therefore, the only issue is to find out the main pathway when a gene is knockout.

3. Logic representation

3.1 Language representation

In our model genes and proteins are viewed as the same object (the genes produce proteins). To describe interactions between proteins it is possible to use a language of classical logic (propositional or first order logic). We can say, for example $stimulation(UV)$ to say that the cell is subjected to ultraviolet or $GlassScreen \rightarrow \neg stimulation(UV)$ to say that a glass screen protects against ultraviolet.

In the context of cell pathways, a predicate can be an action on one or more protein. For example :

$product(P), trigger(P,R), binding(P,Q,R), block\text{-}bindind(P,Q),$
$stimulation(P), phosphorylation(P)$
$dissociation, transcription\text{-}activating..$

The predicate can also represent properties on protein concentration :

$concentration(P, > 1000),$
$concentrationIncrase(P)$
$concentrationDiscrase(P)..$

A formula is :

$product(p\text{-}s15\text{-}p53\text{-}mdm2) \wedge product(p\text{-}chk1)$
$\rightarrow phosphorylation(p\text{-}chk1, p\text{-}s15\text{-}p53\text{-}mdm2)$

and T represents time written as :

$stimulate(dsb, dna, T) \rightarrow product(altered\text{-}dna, T+1)$

Using a simple logic formalism can express much of what biologists are needed to represent. We are in a logical framework, so it is possible to represent a lot of information in a very natural way. The compromise in the case of using the entire first order language, is incompleteness and the combinatorial explosion of complexity algorithms. It is therefore essential to reduce the expressivity power of the language by often using of a propositional representation, if it is possible from biological view point.

Accurate quantification of protein concentration is very important in the case of determining a binding constant or measure enzyme kinetics. But even if something more qualitative is considered, having a good idea of how much protein you have will enable you to compare results from one experiment to the next and from one protein to the next. Therefore, the protein concentration is rarely precise and to represent a change in concentration predicates such as $increased$ or $decrease$ are introduced in our model.

3.2 From conflict to inconsistency and causality

The Representation of Signaling Pathways uses classical first-order logic formulas. This representation is not adequate due to the conflict between the different rules

and therefore inconsistency. The principal problem comes from simultaneously activations and blocking. For example, the Signaling Pathway represented in the Figure 1 we can represent that the UV gives cancer ($UV \rightarrow cancer$) by the clause $trigger(UV, cancer)$; but in the same time we can consider that UV blocks cancer ($UV \dashv cancer$) using the predicate $block(UV, cancer)$ [1] [6]. In this case A and UV active cancer simultaneously which is not possible and therefore the inconsistency of the model.

In fact, this is a very simple form of causality. A lot of research has been done to represent causality. It is possible to use symbolic or numeric formalisms. Even if Bayesian approaches or probabilistic logics [19] seem to be very attractive approach, it is impossible here to go around all these works. We will simply try to describe and use a form of causality (the simplest possible) sufficient for the application to Biological Systems.

The inferences of classical logic $A \rightarrow B$ or $A \vdash B$ are fully described formally, with all the "good" logic properties (tautology, not contradiction, transitivity, contraposition..). But the causality cannot be seen as a classical logic relation.

In a first approach, the first properties that we want to give can be expressed naturally:

(1) If A triggers B and A is true, then B is true.
(2) If A blocks B and A is true, then B is false.

Depending on the context, true can mean the known, certain, believed, proved... The first idea is to express these laws in classical logic by axioms:

$trigger(A, B) \wedge A \rightarrow B$
$block(A, B) \wedge A \rightarrow \neg B$

They can also be weakly expressed by inference rules:

$trigger(A, B), A \vdash B$
$block(A, B), A \vdash \neg B$

But these two formulations are problematic when there is conflict. If for example we have a set of four formulas $F = \{A, B, trigger(A, C), blocks(B, C)\}$, we will in the two approaches above infer from F, B and $\neg B$. This is inconsistent. To solve such conflicts, we can try to use methods inspired by constraint programming, such as the use of negation by failure in Prolog or Solar. It is also possible to use a non-monotonic logic.

The first method, negation by failure, poses many theoretical and technical problems if you go further as the simple cases. These problems are often solved by adding properties to the formal system, properties that pose other problems. Therefore, we will use a classical non-monotonic formalism, the default logic [18].

4. Default logic and Causality

To resolve the conflicts seen above, the intuitive idea is to weaken the formulation of rules :

(1) If A triggers B, if A is true and it is possible/non-contradictory that B, then B is true.

(2) If A blocks B, if A is true and it is possible/non-contradictory that B is false then B is false.

The question which we want to answer is *What is formally "possible" and "not contradictory"* ?

This question began to arise in artificial intelligence forty years ago. In this type of reasoning, one has to reason with incomplete information, uncertain and subject to revision and sometimes false. On the other hand we must often choose between several possible conclusions contradictory. The non-monotonic logics have been studied to solve this type of problems.

4.1 Non-monotonic logics

The "classical" logics, as first order logic or modals logics are monotonic : if it adds information or a formula E' to a formula E, everything which was deduced from E will be deducted from $E \cup E'$. In other words, anything which is deduced from informations will always be true if we had new information. This monotonicity property will generate incompleteness or revisable informations. Indeed, in this case it will be common to invalidate previously established conclusions when new information is added or changed.

A non-monotonic logic allows to eliminate the monotony property of the classical logic: the conclusions could be revised with the addition of new knowledge.

4.2 Default logic

Of course, the problem of non-monotonicity generates many logics, but the most used is Default Logic [18]. Default logic could be seen as an improvement and a generalization of the negation by failure in $Prolog$. It is also a generalization of ASP formalisms which appeared later [13]. Default logic formalizes the default reasoning: conclusions can be made, in the absence of opposite proof.

A default logic is defined by $\Delta = (D, W), W$ is a set of facts (formulae from the propositional logic or the first order logic) and with D, a set of defaults (inference rules with specific content, which handle the uncertainty). A default is an expression of the form:

$$d = \frac{A(X) : B(X)}{C(X)}, \tag{1}$$

where $A(X), B(X)$ and $C(X)$ are formulae and X is a set of variables. $A(X)$ is the prerequisite, $B(X)$ is the justification and $C(X)$ is the consequent. Intuitively, the default d means: if $A(X)$ is true, if it is possible that $B(X)$ is true, then $C(X)$ is true. If $B(X) = C(X)$, the default is normal. In this paper a default is semi-normal its justifications entail its conclusion. The normal default means "Normally, the A are B". In this paper a default is semi-normal if its justification entail its conclusion.

The goal of default logic is to find extensions of a default theory $\Delta = \{W, D\}$. Simplifying, an extension E is a consistent set of formulas obtained by adding, under

condition, to W a maximal set of consequents of D. For us, an extension can for example, represent a subgraph without conflict, of the gene network. Formaly E is an extension of Δ if and only if $E = \cup_{i=0}^{\infty} E_{i+1}$ with:

1) $E_0 = W$

2) $E_{i+1} = Th(E_i) \cup \{C / (\frac{A : B}{C}) \in D, A \in E_i, \neg B \notin E\}$

where $Th(E_i)$ is the set of theorem obtained in a monotonic way from E_i.

It is important to notice that E appears in the definition of E_{i+1}. We need to verify that $\neg B \notin E$. So, we need to know E to find E_i. In a general case an extension is a fixed point which give a lot of problems. But, if we work with normal or seminormal defaults, the definition of an extension is changed: we need only to verify that $\neg B \notin E_i$ and the algorithms are much simpler and effective. For our case study, we just use normal or seminormal defaults.

4.3 Default logic for Signaling Pathways

4.3.1 Representation using Default Logic

We showed that the problem of activation and inhibition is traduced in logic as:

(1) If A triggers B, if A is true and it is possible that B, then B is true.

(2) If A blocks B, if A is true and it is possible that B is false then B is false.

In default logic, these rules can be represented by a set $D = \{d1, d2\}$ of two normal defaults :

$$d1 = \frac{trigger(A, B) \wedge A : B}{B}$$

$$d1 = \frac{trigger(A, B) \wedge A : \neg B}{\neg B}$$

$trigger(A, B)$ et A are both $TRUE$, so it is written as a set $W = \{trigger(A, B), A\}$ containing two classical logic formula. Therefore, the information is represented by the default theory $\Delta = \{W, D\}$

4.3.2 Extensions and choice of extension

If the extensions Δ are calculated for the previous biological situation, 2 extensions are obtained. Each extension is represented by a subgraph without $conflict$. The extensions $E1$ and $E2$ are:

$E1 = \{trigger(A, B), A, B\}$ if $d1$ is used.
$E2 = \{trigger(A, B), A, \neg B\}$ if $d2$ is used.

$E1$ contains B and $E2$ contains $\neg B$. From biological interpretation $E1$ mean B is active and $E2$ is blocked.

The conflict is solved, but it is not possible to rank the extensions: B is active or blocked? In fact this will really depend on the context. For biologists, some times the positive interactions are preferred to negatives. Another possibility is to use probabilistic or statistical methods or to weight each extension based on the evaluation of the knowledge. From an algorithmic viewpoint the ranking of extension could also be evaluated during the calculation of the extensions and even the off-line ranking could be preferred.

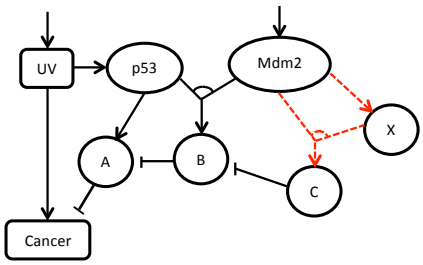


Fig. 3: mdm2 binds X.

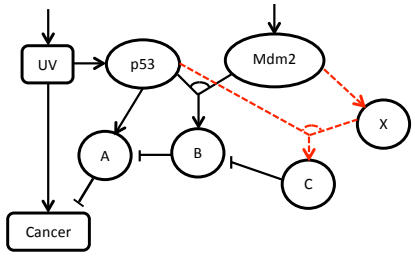


Fig. 4: p53 binds X.

5. Completing the Signaling Pathways by Default Abduction

In section 2 figure 1, we have introduced a simple example which sums up the question *"How to block cancer by preventing B?"*. One analytical solution is to activate a protein C, which blocks B and to consider that a protein X is candidate for binding with $p53$ or $mdm2$. This simple reasoning reflects the biologists approach. Therefore, in this case it is possible to set-up some experiments which prove this hypothesis. It is well known that experiments are time consuming and expensive. It is obviously that in the case of big data it is necessary to automize the process taking into account all available information. In this case we can consider how it is possible to find out, in-silico, a molecule (a future drug) which has the chance to act effectively by blocking some pathways. In terms of A.I. the drug discovery is a problem of abduction.

Classical logic uses the deduction $F \vdash R$ that says that a result R is inferred from an information (formula) F. Abduction generalizes deduction. The information F is incomplete and abduction amounts to adding to F a set of hypotheses H such that $F \wedge H \vdash R$ and $F \wedge H$ is consistent.

Abduction is very important in AI. The trouble comes with implementation of the algorithms. Abduction algorithms are far too high computational complexity. Even limited to

propositional calculus, the theoretical complexity revolves around $\sum_2^p$ which is totally unacceptable when we go beyond small the examples. Many theoretical studies have been done on the complexity of the abduction and research sub-language of propositional calculus where complexity is reduced. These sub-languages most often cover the Horn clauses and renaming. But even here the complexity is too great for even, more or less, NP-complete. Conversely, existing polynomial classes provide only a low power of expression on issues to be addressed. On the other hand, for many real applications, experience shows that it is not necessary to use the full expressive power of logic.

In the case of Signaling Pathways, abduction is used mainly to search missing interactions. The predicate $trigger(\alpha, X)$ is used inside the default with variables $\frac{:trigger(\alpha,X)}{trigger(\alpha,X}$. Here α is a variables which could be instansiated by any protein. This can be generalized to all predicates. By calculating the extensions which contains the predicate $block(cancer)$ we obtained 8 extensions but only 2 contains $p53$ and $mdm2$ (Figures 3 and 4)

6. Logic Representation of Pathway to Reduce Computational Complexity

We present the outline of a language dedicated to discovery of biological interactions answering these requests. This formalism uses the default logic and also has a dynamic approach by considering time as a succession of events. The syntax is described in the next section.

6.1 Clauses

Here $product(P53)$ means that the protein p53 increases in concentration and $\neg product(P53)$ means that it is not possible to determine if the p53 concentration is increasing. The dynamic of the system can be, specified by $concentration(p53, 100, T)$ which means the concentration of $p53$ at the time T is equal to 100 a.u. And $\neg concentration(P53, sup(200), T+2)$ says that at the time $T + 2$, the concentration of $p53$ is not greater than 200 a.u.

The simplest formulas are the clauses. Formally, a clause is a disjunction (or a list) of literals $l_1 \vee .. \vee l_n$. A Horn clause is a clause with a maximum of one positive literal. The clauses a and $\neg b \vee \neg c \vee d$ and $\neg b \vee \neg c$ are Horn clauses. And $a \vee b$ is not one. For the rest we use Horn clauses which are interesting for two reasons.

First using Horn clauses is a natural way to represent knowledge. The formula $a \wedge b \wedge c \wedge d \rightarrow d$ is equivalent to the Horn clause $\neg a \vee \neg b \vee \neg c \vee d$. In the same time the formula $\neg(a \wedge b)$ (a and be cannot be True in the same time) is equivalent to the negative Horn clause $\neg a \vee \neg b$.

The second advantage of Horn clauses, fundamental here, is that their use drastically reduces computational complexity. Indeed, any logical formula can be rewritten as a set of clauses, so complexity problems may arise in terms of clauses. For propositional calculus the fundamental problem is whether a set of clauses is consistent or not. This is the problem SAT which is NP-complete. Otherwise all known algorithms are exponential in the worst case. On the other hand, if all clauses are Horn causes, algorithms can be linear proportional to the size of the data. For genes pathways, the use of Horn clauses provides practically usable algorithms .

Obviously Horn clauses can not represent all formulas. In particular $a \vee b$ is not a Horn clause. But in practice, this type of positive disjunctive information is quite rare. We have not really found it for the gene networks that we studied. If there are, most of the time you can use renaming techniques to solve the problem. Finally, if nothing works and it is impossible to use only Horn clauses, there are techniques to limit the combinatorial explosion. For example use strong backdoors, managing mutual exclusion and cardinality, recognition of symmetries. Here we are in the topic of practice solving NP-complete problems.

A logician may be offend by the foregoing. But for our experiments on cell signaling pathways, it was not necessary to leave the framework of Horn clauses. In fact the use of Horn clauses and default logic had been studied for dynamical application [20]. In that case, default logic is used to simulate decisions of a commander on a submarine in wartime. This is a problem of incomplete information and the decisions have to be done in real-time. Using Horn clauses, defaults logic, mutual exclusion and simple management of the temporal aspect, it was possible to simulate several hours of fighting in a few seconds. In that simulation program the abduction was not used but it is one of the first proof of default logic Horn clauses used to solve real problems.

6.2 Language syntax

A rule is a triplet $(< type >, < corps >, < weight >)$.

• <type> can take 2 values : $hard$ or def. If the value is $hard$ the rule is an hard-rule and represents an Horn clause, which is sure and non-revisable. If the value is def the rule represents a normal default.

• <weight> weights the rule. These weights will make it possible to choose between the different extensions proposed by the algorithm.

• <corps> is a couple (L, R). The left element L is a set of literals $(l_1, ..l_n)$ perhaps empty. This set is identified to $l_1 \wedge .. \wedge l_n$. The right element R is either a single literal or empty. If the rule is hard, the couple (L, R) represents the formula $L \rightarrow R$. If the rule is a default, the couple represents a normal default $\frac{L:R}{R}$ An increased attention is done to these two cases.

Hard Rules

A hard rule (L, R) represents the formula $L \rightarrow R$ where L is a conjunction of literals and R a literal. How we decided to restrict our algorithm to Horn clauses all literals of L are positive. The literal R can be positive or negative. Here we have two special cases. :

1) L is empty. Therefore the rule represents a positive or negative unary clause. The unary clauses are elementary sources of information. They did not contain variables, they are ground clauses. This allows the decidability of the algorithm. However the other clauses can contain variables, leave the pure propositional calculus.

2) R is empty. For this empty-consequence, the rule $L \rightarrow \emptyset$ is equivalent to $\neg L$ equivalent to a negative clause. For example, we can use such a clause to represent a mutual exclusion "It is impossible to trigger and to block a protein at the same time".

Default Rules

If the rule (L, R) is a default, then it represents a normal default, the prerequisite is L, and R is the justification and also in the same time the consequent. If the prerequisite is empty, the default is without justification. By definition of the defaults it is impossible to have an empty consequence. Contrary of the hard rules the prerequiste R can contains negative literals.

6.3 Cell Signaling Pathway Representation

We have worked on the bibliographic data of the response to DSBs translated on a map of molecular interactions Figure 2 [16]. A draw back of this map is that it is very difficult to add a new interaction or protein without full reassessment. In particular the management of conflicts is very difficult. So we worked on the translation of this map into our language. Initial results have translated this map and tested some algorithms [10], [11].

Today, the map is translated by 206 rules in a very natural way, without having to "tweak" the predicates or the rules. The rules are expressed in the syntax above. These rules can be hard rules or defaults. With our syntax it is very simple to change the nature of the rules to test different configurations. We can calculate the extensions in a very short time. We never needed to use non Horn clauses. This reinforces our opinion that it is possible to use a nonmonotonic logic and also abduction and also time, on real applications.

6.4 Rule Examples

In the context of cell pathways, a predicate can be an action on one or more protein. In section 3.1 we saw examples of predicates.

We give here some examples of rules written for our example :

$hard : stimulate(dsb, dna)$

that is an elementary fact (a ground unary clause) who says that dsb stimulates ADN.

$def : stimulate(dsb, dna) \rightarrow product(altered\text{-}dna)$

that is default rule "Generally when dsb stimulates DNA, altered DNA is produced.

$hard : product(p\text{-}atm\text{-}atm\text{-}bound)$
$\rightarrow \neg product(atm\text{-}atm)$

that is a negative clause.

Using a simple logic formalism can express much of what biologists are needed to represent.

7. Implantation of Algorithms

The algorithm is written in SWI Prolog. A rule :
$(< type >, < corps >, < weight >)$
is represented by a unary Prolog clause:
$rule(< type >, < corps >, < weight >).$

Therefore, the rules and the algorithm are in the same Prolog program, which is very practical. Another advantage to use Prolog is that the unification, the backtracking and the lists are well optimized. Of course Prolog is interpreted, so it is slower than compiled languages (but not that much). In the other hand Prolog programs are short and simple, which saves a lot of time to test programs and heuristics.

This algorithm calculates the extensions. As the clauses are Horn clauses and as the defaults are normal, the research tree is optimized. Particularly it is easy to calculate extensions without duplication (we do not calculate several times the same extension). For algorithms, we can also use a weak form of negation as failure.

For initial tests, given by the map of the entire network of Pommier [16], we can calculate all extensions in a short time. For example with most of the rules by default, there are two extensions. The calculation takes 500000 LIPS and 0.4 seconds of CPU time on MacBook. The temporal aspect of gene networks has been tested for small examples, but the scaling has not yet been done. For the abduction, it is almost the same.

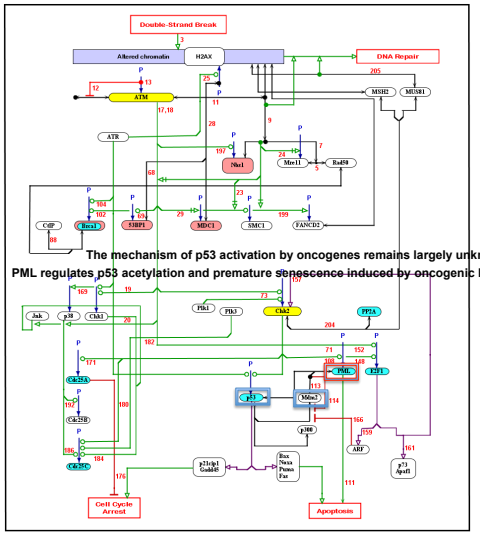


Fig. 5: DNA double strand break generated automatically by using the most relevant extension.

8. Results

Basically, many researchers are trying to complete the Signaling Map. In our approach the map is simplified which is very useful for biological experiments. Introducing time in defaults (the prerequisite considered at time t and the conclusion at time $t+1$), we obtained a simplified map of Pommier. The most interesting result is the identification of the molecule "X" from figure 1 as be PML which regulates p53 acetylation and premature senescence induced by oncogenic Ras.

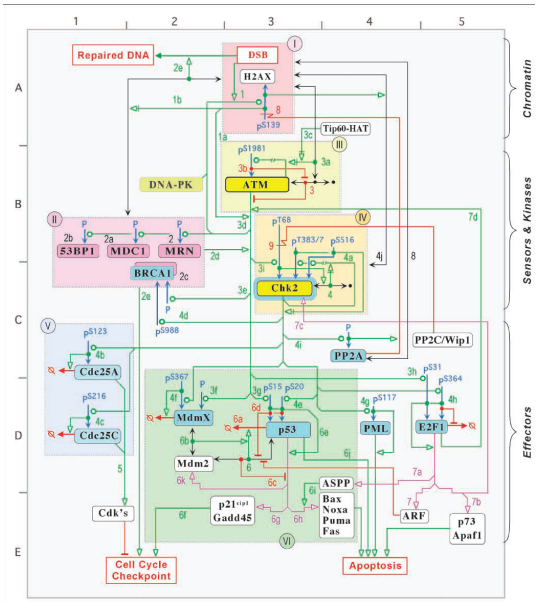


Fig. 6: The Molecular Interaction Map and Rationale for Chk2 build-up "manually" by Pommier [17] using biological cause-effect graph.

At first we tested the notion of production fields and a consequence finding algorithm for producing clauses [4], [15]. We tested also the SOLAR language that uses production field and this algorithm. The results are mixed in the case of "big" examples. We also looked at the ASP formalism [13]. Again the impression is mixed. Indeed ASP deal mainly with normal defaults without prerequisites. Getting all the power representation of defaults with prerequisite is possible by rewriting techniques. But you lose a a lot of clarity and also efficiency.

By generating automatically the DNA double strand breaks map (Figure 5) we noticed that the protein p73 is not directly involved in activation of apoptosis as in the case of Pommier map (Figure 6). This result obtained from incomplete knowledge constitutes a theory formation framework for "Knowledge Discovery" using Default Logic.

9. Conclusion.

The A.I. challenge is to explain new phenomena using automatic causal discovery. In this paper we introduced a formalism able to infer signaling pathway by using defaults approach and abductive reasoning.Therefore the originality of the work is given by the capacity of the proposed logical model to find out the consistency of Pommier's map by efficient algorithms.

References

[1] N. Tran, C. Baral, *Hypothesizing and reasoning about signaling networks.* Journal of Applied Logic, 7, 253-274,2007.

[2] CH. Bassing, FW Alt. *H2AX may function as an anchor to hold broken chromosomal DNA ends in close proximity.* Cell Cycle 2004; 3:149-53.

[3] J. Bartkova, Z. Horejsi,et al., *DNA damage response as a candidature anticancer barrier in early human tumorigenesis.* Nature 2005; 434:864-70.

[4] J.M. Boi, E. Innocenti, A. Rauzy, P. Siegel, *Production Fields : A New approach to Deduction Problems and two Algorithms for Propositional Calculus.* Revue d'Intelligence Artificielle, 25(3) : 235-255, 1992.

[5] G. Bossu, P. Siegel. *Saturation, Nonmonotonic Reasoning and the Closed World Assumption.* Artificial Intelligence, 25(1) :13-63, 1985.

[6] A. Doncescu, Y. Yamamoto, K. Inoue, *Biological systems analysis using Inductive Logic Programming.* Proc. of the 21st International Conference on Advanced Information Networking and Applications (AINA 2007), pages 690-695, IEEE Computer Society, 2007.

[7] A. Doncescu, K. Inoue, Y. Yamamoto, *Knowledge-based discovery in systems biology using CF-induction. New Trends in Applied Artificial Intelligence.* Proc. 20th International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems (IEA / AIE 2007), Lecture Notes in Artificial Intelligence, volume 4570, pages 395-404, Springer, 2007.

[8] A. Doncescu, J. Weisman, G. Richard, G. Roux, *Characterization of bio-chemical signals by inductive programming.* Knowledge Based Systems, 15 (1), 129-137, 2002.

[9] A. Doncescu, T. Le, P. Siegel, *Default Logic for Diagnostic of Discrete Time Systems.* Proc. BWCCA-2013 - 8th International Conference on Broadband and Wireless Computing, Communication and Applications p. 488-493, Compiegne, France, Oct 2012

[10] A. Doncescu, P. Siegel, *The Logic of Hypothesis Generation in Kinetic Modeling of System Biology,* Proc. 23rd IEEE International Conference on Tools with Artificial Intelligence, p. 927-929, Boca Raton, Florida, USA, Nov. 2012

[11] A. Doncescu, T. Le, P. Siegel, *Utilization of Default Logic for Analyzing a Metabolic System in Discrete Time.* Proc.13th International Conference on Computational Science and Its Applications, ICCSA 2013, p. 130-136, Ho Chi Min, Vietnam, June 2013.

[12] D. Kayser, F. Levy, *Modeling symbolic causal reasoning,* Intellecta 2004, 1, 38, pp 291-232, 2004

[13] E. Giunchiglia, J. Lee, V. Lifschitz, N. McCain, H. Turner, *Nonmonotonic causal theories* Artificial Intelligence, No. 1-2 vol.153 pp.49-104, 2004.

[14] K. Inoue,A. Doncescu, H. Nabeshima. *Completing causal networks by meta-level abduction.* Machine Learning, 91 (2) :239-277, 2013.

[15] H. Nabeshima, K. Iwanuma, K. Inoue, O. Ray. *SOLAR: An automated deduction system for Finding consequence.* AI Commun, 23 (2-3): 183-203 (2010)

[16] Pommier Y. and all. *Targeting Chk2 Kinase : Molecular Interaction Map and Therapetic Rationale.* Current pharmacy design, 11(22):2855-72, 2005.

[17] Pommier Y. and all. *Chk2 Molecular Interaction Map and Rationale for Chk2 Inhibitors* Clin Cancer Res. 2006 May 1;12(9):2657-61..

[18] R. Reiter *A Logic for Default Reasoning.* Art. Int. 13(1-2): 81-132 (1980).

[19] T Sato, Y Kameya. *PRISM: a language for symbolic-statistical modeling.* International Joint Conference on Artificial Intelligence 15, 1330-1339.

[20] I. Toulgouat, P. Siegel, Y. Lacroix, J. Botto *Operator decision modeling : application to a scenario involving two submarines* Proc. COMPIT10, Gubbio, April 2010 *and* Proc. 13th NMR10, Toronto, Canada May 2010

Methods of the Arabic Manuscripts Digitization[1]

Prof. Oleg Redkin, Dr. Olga Bernikova
Department of Asian and African Studies, Laboratory for Analysis and Modeling of the Social Processes,
St. Petersburg State University, St. Petersburg, Russia

Abstract *The mediaeval Arabic manuscripts are not only valuable artifacts but they also represent one of the major sources of scholar information in the field of Oriental Studies. This paper discusses the methods of Arabic Manuscripts Digitization. Over the last fifteen years a lot of Arabic manuscript digitization projects have been carried out. Digital collections of the manuscripts based on Arabic script are represented in the collections of libraries worldwide, including on-line databases. Nevertheless, these collections are restricted by their functionality: technology of metadata integration relies on the human made characteristics. While a possibility of automatic metadata introduction would facilitate the task of manuscript processing, at the same time it allows automatic quantitative and quality analysis of the manuscripts. This paper analyses different methods for classifying and retrieving historical Arabic handwritten documents and suggests the most efficient methods of their digitization.*

Keywords: manuscript, digitization, Arabic

1 Introduction

Mediaeval Arabic manuscripts are not only valuable artifacts but they represent one of the major sources of scholar information in the field of Oriental Studies as well Although Arabic manuscripts have always remained in the focus of the scholars' attention, for a long period of time the methods of their description and investigation have been almost unchanged and based not only on researcher's experience, qualification and knowledge, but on a subjunctive opinion as well.

The description of these manuscripts has a long history and, as a rule, includes a collection of data on the history of their origin, content and characteristics of the physical state of the document.

Recent decades have witnessed the spread of the digital processing, retrieval, storage and transmission of information which, in its turn, has allowed new methods of data processing in Arabic, and opened new opportunities for scholars. Thus the digitalization of Arabic handwriting heritage has completely revolutionized this process and provides creation of electronic on-line catalogs, the digitization of the scanned images and, to some extent, optical character recognition (OCR).

2 The term "digitization"

In the historical perspective "digitizing of a document" meant a creative surrogate, an alternative carrier intended to be preserved [2]. Today there are several different interpretations of the term. Simplified understanding of the first approach is digitizing as making images: computer processing of Arabic manuscripts limited to their scanning and recording received in *.bmp, *.jpg, *.ipeg or other types of files on the media or posting them on sites of other academic institutions. The second approach lies in the field of text recognition, i.e. digitizing that includes scanning and optical character recognition as a minimum. This solution is quite difficult in case of Arabic manuscripts. There is another interpretation of the term "digitizing" which we refer to a historical document. Digitizing a huge amount of manuscripts requires a sophisticated information system that established relations between data (digital images) and metadata.

Metadata is a "structured piece of information that describes, explains, locates, or otherwise makes it easier to retrieve, use, or manage an information resource" [3]. As a minimum, metadata should conform to the Shareable Metadata Guidelines for libraries. Digitization enhances access to the artifact as its image can be seen on the web by users all over the world. Besides, it can be sent for offline viewing using a higher resolution of uncompressed master file.

3 Previous experiences

Over the last fifteen years a lot of Arabic manuscripts digitization projects were carried out. Digital collection of manuscripts based on Arabic script is represented in the collections of libraries worldwide, including on-line databases [4]. For example, Princeton University Library and the Free University, Berlin, in conjunction with the Imam Zayd ibn Ali Cultural Foundation (IZbACF) in Sanaa, Yemen [5] implemented the collection that is a part of the Princeton Digital Library of Islamic Manuscripts [6].

[1] The authors acknowledge Saint-Petersburg State University for a research grant 2.37.175.2014.

One of the best manuscripts database is presented by the Welcome Arabic Manuscript Cataloguing Partnership (WAMCP) that combines the efforts of the Welcome Library (London), Bibliotheca Alexandrina (Alexandria, Egypt) and Department of Digital Humanities (King's College, London) [7]. The interface of the website is user-friendly. Both manuscript and metadata can be compared side by side. All indexed fields are searchable. A positive issue is the additional possibility to search in full text as well as in one of the following fields only: incipit, colophon of the manuscript, colophon of work, table of contents, notes and provenance. At the same time it is worth noticing the lack of detailed information about the amount of the digitized manuscripts and their most common topics which are primarily related to medicine.

Significantly, the system allows searching via the full text search facility. We can hardly check the correctness of this kind of searching technology. Today automatic recognition of handwritten words remains a challenging task. To a certain extent it is caused by the peculiarities of the Arabic text (it was described in details in our previous works, see [8] and [9]). The problem of recognition of handwritten documents, especially manuscripts, which include the individual characteristics of the authors' handwriting, is even more complex, not to mention the extra noises – notes of the scribes, defects of the written material, and lacunae and gaps in the text, notes and additions to the original text. All this makes the correct identification of the Arabic written texts extremely difficult. One of the experiments in this field was carried out in terms of the developing a new database with handwritten Arabic town/village names [10]. It proves the fact that error rate of handwritten recognition systems are much better for applications with a restricted lexicon of words. That is why the searching mechanism presented in Welcome Arabic Manuscripts Online requires clarification: is it really based on the full-text manuscript search or it relies on restricted parts of the retyped manuscript? We assume that the searching mechanism relies only on those parts of manuscripts that have printed form. Our methods of manuscript digitizing and classification strongly differ from the technologies used in WAMCP, as they are concentrated on different tasks. WAMCP used manual processing of the metadata manuscripts while our experiment is targeted at the basic automatic manuscript processing. Our search for similar available solutions confirmed the uniqueness of statements of problems.
Nevertheless, a group of scholars from the Cairo University investigated similar issues. They presented results for historical document classification of old Arabic manuscripts using texture analysis and a segmentation free approach. The main purpose of their project was "to discriminate between historical documents of different writing styles to three different ages: Contemporary (Modern) Age, Ottoman Age and Mamluk Age" [11].

4 Specificity of Arabic Manuscript Digitization

Currently most of the manuscript centers and libraries focus on the creation of a digital copy of the manuscripts which in fact is the formation of databases either within the library network, or on-line supplied with search engines, rather than processing of the special features of a certain manuscript. The related and additional information is downloaded manually, and criteria of its selection depend on the operator's subjunctive opinion.

For unification of the manuscript digitization the UMass Amherst Libraries Guidelines for Digitization were developed by members of the Digital Creation and Preservation Group for the application of all library digitization projects. These guidelines are designed to provide digital project managers with a set of minimum specifications for preservation-quality digitization of printed text, manuscripts, photographs, slides, rare books, sheet music, graphic arts, and maps. They provide a baseline for creating digital images that are of sufficient quality for long-term preservation [12].

Traditionally so called "subject metadata" include manuscript number, title, author, origin, organization, commentary, commentator, language of material, script, complete/incomplete, folio number, script, subject, bundle number, folio number, pages, missing portion, illustrations, condition, catalogue source, remarks, manuscript date, manuscript length, manuscript width. As well as the information about its origin, incipit, explicit, physical details, which include information about the script, color of ink, quality and completeness of the manuscript itself, number of lines on the page, columns, binding, loose quires etc.

The majority of the existing databases rely upon manual processing of the manuscript with its following integration into the software environment. At the same time existing technological solutions allow for the automatic description of a number of metadata characteristics. As a result there are two methods of manuscript analysis: the first of them is based upon the results of subjective perception and the second one is based on objective quantitative and qualitative indicators, which are presented in digital form.

The objective data usually refer only to the number and size of the pages, type of the binding, paper quality, number of lines per page, etc. More detailed characteristics, for example those associated with the so-called "rhythm" of the handwriting, with few exceptions still remain outside the view of researchers.

Meanwhile, the more detailed analysis of the formal data may provide additional information that can draw conclusions about the time and the place of the manuscript origin.

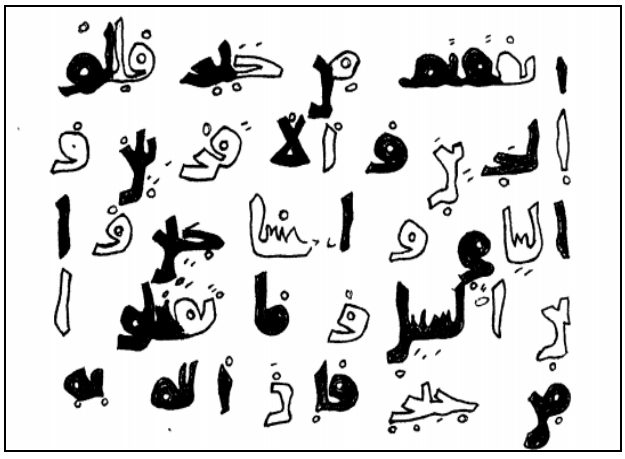
On the contrary, the computer analysis of the text may enable automatic determination of the whole range of characteristics such as:

- the type of the script;
- proportions between the size of the fields and text spacing between words and long words;
- the ratio between the height and width of the characters, diacritics location, tilt of the handwriting;
- the width of the lines of the characters can determine the degree of pressure, and hence the written tool type;
- the presence of different types of handwriting in one manuscript;
- the ratio between the text and the gaps;
- to identify particular color palette, illustrations, inventory marks and seals, inscriptions on the sidelines, the color characteristics of ink, the use of color in the text of the manuscript;
- to define ratio between the parameters of the text and the features of its and orientation of its fragments, the shape and the type of the manuscript layout.

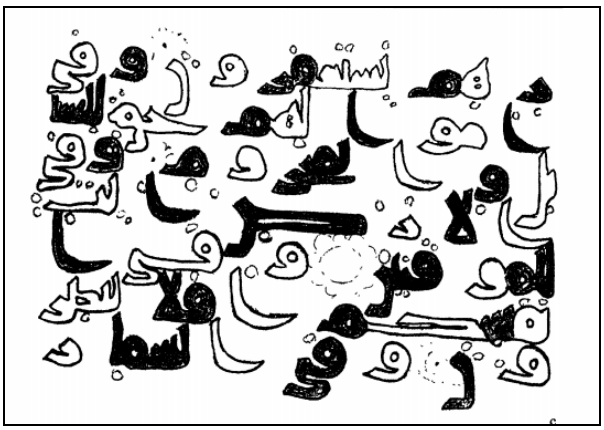
Due to the objective assessment of each manuscript the text may be correlated with a particular school of manuscript, in fact such a correlation is possible when the information is available on the manuscript scribes.

In this case the traditional methods of the classification of the script are insufficient since there are also "hybrid" versions of the characters (eg. Maghreb typeface of) ف fā ') / ق (ḳāf)).

It is the computer analysis which can provide scale and objective systematization of ideas about spellings lower and upper detail view, closed) م mīm) and open) ب bā') inner elements of the characters, diacritics, vowels, ligatures.



Picture 1a. Types of manuscripts which are found different in diagonal effects [13].



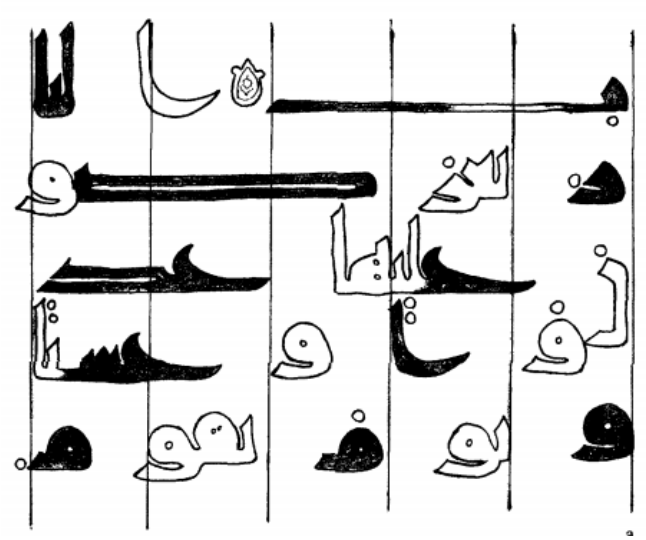
Picture 1b. Types of manuscripts which are found different in diagonal effects [13].

The traditional classification of Arabic handwriting scripts (naskh, kufi, maghribi, nasta'liq etc.) which is based on subjective visual analysis and evaluation remains unchanged, and it is rather arbitrary ('number of points along the lines'), and does not reflect the whole reality of Arabic tradition of handwriting.

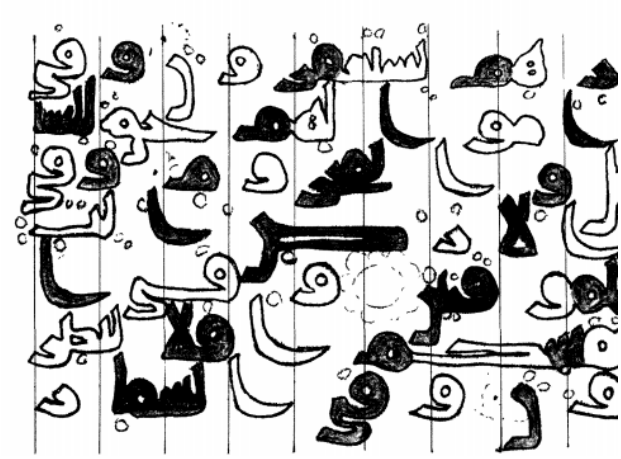
Computer analysis would allow going far beyond the approximate classification of Arabic handwriting. It will help to classify handwritten texts based on objective (digital) characteristics, for example, based on such indicators as the differences in the degree of roundness of letters in Naskh and Kufic texts. As a result, the whole range of the script variants may be divided into several groups (clusters) depending on the type of characters, their density, etc.

It is known that texts which belong to different handwritten schools have different writing of certain letters, as well as the differences in the location and orientation of the fragments of the text, text density (space occupied by individual letters, the number of letters falling on the page), variety of extra elements, such as illumination, colors. Thus, the task is to identify certain patterns and proportions of the text rather than to implement linear and vertical segmentation, as it is customary in the OCR.

Diachronically the basic parameters of the Arabic manuscripts were developing (i.e. size of the pages, type of binding, ink color, its type) depending on the manufacturer, written tools, local technologies and existing manuscript tradition (line width, spacing, types of word wraps, peculiarities of graphics, text illuminations etc.). Since all these parameters are multifaceted and include a lot of data, they cannot be systemized and described as a whole without the application of computer technologies.



Picture 2a. Division of the text into different segments [13].



Picture 2b. Division of the text into different segments [13].

5 Databases and digital 'passport' of the manuscript

The spectrum and combination of digital indicators of this kind are unique for each manuscript, and may be considered as manuscript digital ID.

Each manuscript has a unique set of special features and characteristics (types of script, variations of color, filigrees, chainlines, various types of paper, etc.) which may be compared with fingerprints, and which may be processed with the help of computer technologies. Thus, digital processing can determine the exact number of lines in the text, the angle at which the text is placed between the grid lines (laid paper / chainlines) on a specific page, as well as throughout the manuscript as a whole. In this case the task is to determine which features are relevant to a certain manuscript and, finally, to build hierarchical system of these features. The data obtained will allow formation of databases and developing advanced search engines.

Characteristics of the manuscript can be divided into those which 'lie on a surface' (explicit), and those that require further analysis (implicit).

The language of the manuscript, handwritten text color, manuscript size, number of pages, paper color, text orientation (landscape / non-album) etc. are among these explicit characteristics. Among the implicit characteristics are the number of characters, the 'inner rhythmic of the text', location and orientation of its elements, use of words and expressions, etc.

Digital 'passport' forms a manuscript's feature vector which may be interpreted as a point in a metric space. Having a set of feature vectors it is possible to perform a clustering (grouping) of manuscripts. On the one hand manuscript clustering allows to build a classifier to automatically determine the writing style, era and even the possible authorship. On the other hand it allows to extract an essential data for analysis of relationships between manuscripts and for tracking the dynamics of writing style evolution. It is known that clustering methods on the basis of randomized learning theory [14, 15] provide an appropriate results for different kind of data sets with noises and uncertainties. Such useful properties seem to be very promising for Arabic manuscripts processing.

6 Conclusion

The set of indicators specific to each individual manuscript is unique, and allows us to classify it and assign to a specific point in time and the spatial coordinates. Creating of such digital manuscript "passport" will allow its comparative analysis in comparison with other manuscripts, to implement their classification, to draw conclusions about its authorship, as well as to define whether it is an autographed manuscript or a copy, attribute it to a certain handwritten school or copyist, and also to help in its dating. Such kind of digital "passport" will facilitate the process of cataloging manuscripts, preparing them for further scholar investigation.

7 References

[1] Gacek A. "Some Remarks on the Cataloguing of Arabic Manuscripts"; British Society for Middle Eastern Studies, Vol. 10, No. 2 (1983). Pp. 173-179.

[2] Zdeněk U. "Digitization is not only making images: manuscript studies and digital processing of manuscripts"; Knygotyra. 2008. 51. Pp. 148-160.

[3] Understanding Metadata. NISO, 2004. Pp. 1-20.

[4]http://guides.lib.umich.edu/islamicmsstudies/onlinecollections

[5] http://wamcp.bibalex.org/about-us, 03/03/2014/

[6] http://pudl.princeton.edu/objects/9s1616928

[7] http://pudl.princeton.edu/collections/pudl0032

[8] Redkin O.I., Bernikova O.A. "Problems of the Arabic OCR: New Attitudes"; Proceedings of the 2013 International Conference on Artificial Intelligence. Las Vegas, USA, 2013. Pp. 777-782.

[9] Redkin O.I., Bernikova O.A. "On the Optical Character Recognition and Machine Translation Technology in Arabic"; Proceedings of the 2012 International Conference on Artificial Intelligence. Las Vegas, USA, 2011. Pp. 861-867.

[10] Pechwitz M., S. Snoussi Maddouri, Märgner V., Ellouze N., Amiri H., "IFN/ENIT-Database of Handwritten Arabic Words"; 7th Colloque International Francophone sur l'Ecrit et le Document , CIFED 2002, Oct. 21-23, 2002, Hammamet, Tunis, (2002). Pp. 1-8.

[11] Ahmad M. Abd Al-Aziz, M.Gheith, Ayman F. Sayed Recognition for old Arabic manuscripts using spatial gray level dependence (SGLD); Egyptian Informatics Journal, 2011, 12. Pp. 37-43.

[12] Banach M., Shelburne B., Shepherd K., Rubenstein A. Guidelines for Digitization, Digital Creation and Preservation Working Group. 2010-2011.

[13] Polosin V. "Manuscripts of Ibn Muqla's Calligraphical School (the problem of their identification); Oriental Written Sources, The Institute of Oriental Manuscripts. 2004. Pp. 160-177.

[14] Avros R., Granichin O., Shalymov D., Volkovich Z., Weber G. "Randomized Algorithm of Finding the True Number of Clusters Based on Chebychev Polynomial Approximation", Data Mining Foundations and Intelligent Paradigms, Vol. 1: Clustering, Association and Classification. Springer-Verlag, 2012.

[15] Granichin O., Shalymov D., Avros R., Volkovich Z., "A randomized algorithm for estimating the number of clusters", Automation and remote control, 2011, V. 72(4) Pp. 754-765.

A Top-down Perception Approach for Object recognition

C. Bernay-Angeletti[1], **C. Aynaud**[1], **R. Chapuis**[1] **and R. Aufrère**
[1] Institut Pascal UMR 6602 UBP / CNRS, Campus des Cézeaux 63177 AUBIERE Cedex, FRANCE
[2] LIMOS UMR 6158 UBP/ CNRS, Campus des Cézeaux 63177 AUBIERE Cedex, FRANCE

Abstract— *In this article we present a multi-purpose generic top-down algorithm. It is designed here for object recognition and is based on a progressive focusing.*

The object is assumed as constituted with several parts. The recognition is made by the detection of theses parts. The initial values and the relationships between each part are grouped in a state vector with its a priori knowledge mean value and its covariance matrix.

The algorithm is to estimate this vector recursively using the detection of the parts and following an objective relying on the best accuracy / reliability of the recognition.

The selection of the best part to detect is achieved by a Bayesian network. The state update is done thanks to a Kalman filter. This allows to focus in an optimal way the detection of each part and, in case of detection failure, to dynamically decide which hypothesis has to be tried.

Keywords: top-down approach, focus, accuracy, reliability, parts component

1. Introduction

Object recognition in an image is not an easy task: information may be ambiguous, noisy or not relevant, the object model may also be not well defined, etc. Many methods try to achieve object recognition. Even if it is not clear how our brain works [8], the usual methods can be split in two: example based approaches and structural approaches.

In the first category, besides historic methods relying on prior segmentation, the global appearance of an object is more often learnt. The structure of the object is not explicitly used: a database is created using several thousands examples of images. After this training stage, the image is scanned in position and scale [12] to accomplish the recognition. This stage is more often done using kernel techniques with SVM or cascade classifiers [3], [15]. To reduce the computation time implied by the scanning process, some techniques use focus indication like colours, shadows, regions of interest (ROI), or reduce the number of parameters[12] or the number of scales for spatial resolution. This first category can be seen as *bottom-up* methods since no model drives the recognition.

The second type of approaches needs an explicit decomposition of the object into several parts. This decomposition can be given by the user. These approaches like visual grammars [9], [16] or deformable part-based models [6], [10] provide an elegant framework for object detection. As an advantage, these methods explicitly model structural variations [9]. This allow therefore to use the knowledge we can have about the relationships between the object parts. This second category can be seen as *top-down* methods since the model can explicitly be used to drive the recognition.

Despite this, the first category has nowadays outperformed the second one thanks to huge databases for common life usual objects. This is mainly due to the fact that even if it seems easy to decompose for example a pedestrian in sub parts (legs, shoulders, arms, head, ...), the global shape appearance is simpler to detect having a huge database in memory.

Nevertheless we believe it is important to keep on the top-down approaches because humans behaviours relies more or less on these approaches [8] or, according to Craik [4] :"If the organism carries a "small-scale model" of external reality and of its own possible actions within its head, it is able to try out various alternatives, conclude which is the best of them, react to future situations before they arise, utilize the knowledge of past events in dealing with the present and future, and in every way to react in a much fuller, safer and more competent manner to the emergencies which face it." So human are not passive with perception [1], [13] and they anticipate and correct their behaviour to realize a specific task.

In this work we propose an extension of the method presented in [2] that was also used for robot localisation [14].

We assume the object to recognise is composed with several parts. We suppose having specific detectors for each part and so we can define couples of part-detectors. The object model is described by a vector (called *state vector*) with an initial mean value and an associated covariance matrix (representing the accuracy) and a confidence in this prior initialisation (the probability that the object parameters are in the range of the space defined by the state vector and the covariance).

Since each part has also its own parameters that are linked to the state vector by *observation* functions, the goal of the algorithm is to recursively estimate this state vector according to observations made by the part detectors.

Our algorithm is top-down based (even if we plan to join a bottom-up step later). The question is: which part should be detected in first or what is the couple part-detector the more relevant ? A discrete Bayesian network is used to select this couple. Bayesian networks are very well adapted to this task since they can infer both reliability (confidence in the current state recognition) and an optimal hypothesis pruning after a detection failure.

After this selection phase, the selected detector will try to find the associated part. The state vector and its covariance matrix are updated thanks to a Kalman filter. More accurate the state becomes, smaller will be the search zone for the next step. This will improve drastically the signal noise ratio and the computation time.

At each recognition step the algorithm provides i) the current state estimation (Kalman filter) and ii) the confidence in the recognition (Bayesian network).

In the next sections we describe the main principles of the algorithm and we show in results that the algorithm steps are in accordance with what we expected and that different objects can be recognized with the same principle.

2. Proposed Top-down approach

2.1 State of the art

Deformable part models provide an elegant framework for object detection but on difficult datasets, they are often outperformed by global approaches such as bags of features [11]. Even if some works [6] have shown that robustness can be comparable to the state of the art, the object recognition remains slow (more than one second per image). Some approaches [5] proposed constellation models to locate distinctive object parts and determine constraints on the spatial arrangement. The main drawbacks of these approaches are that spatial models typically could not handle significant deformations like big rotations or occlusions.

Grammar based models generalized part models, they are richer as they can model structural variations but suffer of the same computation time problems. The robustness from occlusions or rotations depends on the number of possible parts : having more views implies a bigger and-or-graph so a more complex representation. The variability that can be represented by a grammar remains however interesting and can be seen as an objective to achieve. To reach the real time, we will use strategies that have given good results like the focus in particular zone. We are inspired by the cascade detection. Here the cascade will not be fixed but computed by the algorithm to select at each iteration the best detector. Then the next detector will be only applied in a smaller zone given by the result of this previous detection. Like in cascade detection, the more detections succeed, the more confident we have in our detection and if the detection fails the hypothesis can be cancelled so we will look somewhere else. This detection order is really important for the time optimisation. In [9], the computation time is much better if for detecting a bicycle the algorithm begin by the detection of the wheels. We want to automate this choice. This selection also allows us to add other detectors without changing the structure. As the algorithm computes a selection criterion for each couple part-detector, add a detector just means compute one more selection criterion for each new couple part-detector.

In fact our algorithm proposes a general framework because we can add all detectors and parts we want. If a new really fast and reliable detector for detecting something is created, it can be easily integrated either by using its results for an existing part or by creating a new part. So it can be seen like a tool to fuse different detectors data. Furthermore, this fusion is not limited to image, it can be used for localization purpose.

Moreover, the algorithm is able to provide after each iteration the confidence it has in the current recognition state. This allow to keep on the process and to focus on other parts or to quickly prun the current hypothesis.

2.2 Object Modelling

2.2.1 Object State

An object is represented by a vector $\underline{X}$ of n parameters. So, recognize this object consists in estimating this parameters vector $\underline{X}$. For this, we will do a recursive recognition. We define l the recognition level (or step). For a given level l, the recognition of an object is characterized by $\mathbf{E}_l$ such as $\mathbf{E}_l = \{\underline{X}_l, \mathbf{Cx}_l, P(O_l)\}$ with:

- $\underline{X}_l$ is the estimation of $\underline{X}$ at recognition level l.
- $\mathbf{Cx}_l$: its covariance matrix.
- $P(O_l)$ is the confidence in the recognition level l. It characterizes the integrity of the estimation at this level. In fact, it is the probability that the real parameters vector $\underline{X}$ follows the supposed gaussian law defined by $\underline{X}_l$ and $\mathbf{Cx}_l$ or in other words, that $\underline{X}$ (true position) is really in the space defined by $\underline{X}_l$ and $\mathbf{Cx}_l$.

The objective of recognition will consist in determining the optimal state $\mathbf{E}_l$ i.e to find the state that minimizes the inaccuracy on $\underline{X}$ (so for which the trace of $\mathbf{Cx}_l$ is minimum) and that maximizes $P(O_l)$.

2.2.2 Object Parts

An object is composed of M parts. Each part i is itself represented by a vector $\underline{\chi}_i$ of np_i parameters : $\underline{\chi}_i$, a covariance matrix Σ_i and a probability of part existence $P(pE_i)$. This probability is used to represent some optionnal parts like glasses for a face (all faces will not have glasses). How an object is decomposed into parts is not the purpose of this article. We assume that all usefull parts are known.

We also suppose to have a function $\mathbf{f}_i$ which links the part vector Σ_i of the part i with the state vector $\underline{X}$ of our object.

$$\underline{\chi}_i = \mathbf{f}_i(\underline{X})$$

So, parts of a same object are only connected by the full state object $\underline{X}$. As previously, doing a gaussian hypothesis, the covariance matrix Σ_i is given by:

$$\Sigma_i = \mathbf{J}_{f_i} \mathbf{Cx} \mathbf{J}_{f_i}^T$$

where $\mathbf{J}_{f_i}$ is jacobian matrix of the function f_i. $\underline{\chi}_i$ represents the dispersion of the part parameters knowing the estimation $\underline{X}_l$ of the state parameters. We define the state of a part i by

$$\eta_i = \{\underline{\chi}_i, \Sigma_i, P(pE_i)\}$$

Besides, this state η_i is dependant of the current object state. We define:

- η_{i_l}: the part state estimation depending of the object state recognition at step l
- $\eta_{i_{l+1}}$: the part state estimation after a detection of this part.
- $\mathbf{E}_{l+1}$: the object state after the part detection.

2.3 Overall principle

The object recognition consists here in estimating the unknown vector $\underline{X}$. In this algorithm, we want to have a good accuracy and a good confidence for our estimation.

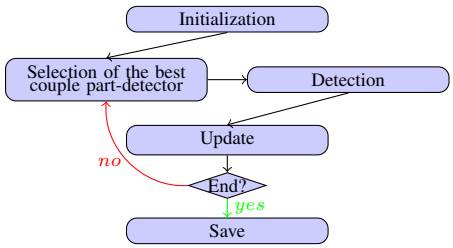


Fig. 1: Our simplify algorithm global scheme.

The figure 1 describes our algorithm principle. So there are different steps.

2.3.1 Initialization

The objective of this step is to define all the a priori values. In particular, we suppose that $\mathbf{E}_0 = \{\underline{X}_0, \mathbf{Cx}_0, P(O_0)\}$ is known: we have an initial mean value, an initial covariance and an initial confidence about the object. Note that if we have very poor information, we just will have a gigantic covariance and put $P(O_0)$ at 0.5 (we have no a priori at all about the presence of the object in the image). It is in this step where the initial existence probabilities $P(pE_{i_0})$ for all parts i are defined. These probabilities can evolve during following iterations. It is in this phase where all the a priori knowing about the object is provided to our algorithm.

2.3.2 Part Selection

It is sequential: a part is detected, the state is updated, another part is detected... But we want to choose efficiently the part to detect at a given level l and with which detector, so what is the best couple part-detector? Here, a selection criterion will be computed for each existing couple part-detector and the best one will be chosen. This criterion will take into account the accuracy that a couple part-detector can give if the detection successes, the probability of the detection success and the mean time for realizing the detection. See section 2.4 for details.

2.3.3 Detection

Once the couple is chosen, the detector is executed in the associated Region of Interest(ROI). The detection can be a success: at least an acceptable feature for the part is found. In this case, the detector returns the estimation of the parameters features $\underline{\hat{\chi}}$. The part state vector $\underline{\chi}_i$ is updated and its covariance Σ_i too. See section 2.5.

2.3.4 Update

First, the confidence in our estimate will be updated. If we note l the current iteration of the detection, we can say that $P(O_l|d) \geq P(O_l)$: we are more confident in our previous state because the detection successes and that $P(O_l|\bar{d}) \leq P(O_l)$: we are less confident in our state recognition because the detection failed. To compute these probabilities a bayesian network is used. If the detection succeeded, the part has been instanciated and so we can update the state vector $\underline{X}_l$. For this, an Extended Kalman Filter is used. The confidence of this new state $\underline{X}_{l+1}$ is computed too. See section 2.6.

2.3.5 End

The algorithm ends when a criterion is reached. The algorithm will give the last estimation of the state: so the last state vector with its associated covariance and confidence. $P(O_l)$ will provide the confidence we can have in this recognition. See section 2.7.

In next sections, these differents steps will be detailed.

2.4 Part Selection

2.4.1 Principle

The objective here is to carefully select the best couple detector-part. For that, we define a goal in terms of accuracy ($\mathbf{C}_{goal}$) and confidence (P_{goal}). Then choose the best couple detector-part can be seen like the choice that will lead the more efficiently towards the objective. Some couples are unattractive because: they don't improve our estimation or because the detector is not reliable and/or will take too many time to find features.

2.4.2 Selection criterion

To select the most relevant couple part-detector, a selection criterion is defined for each couple. So the couple with the best criterion will be chosen. Criterion choice is defined with respect to several constraints:

- if there is no new information the couple should not be selected. No information means that the couple has already be chosen in a previous iteration and that the conditions of detection are the same.
- if the feature associated to the couple part-detector has very little chance of being detected, the criterion for this couple will be low.
- it is better to take a detector which is faster even it is a little less accurate because the other detector will be use in a smaller zone if the first one finds its feature.
- if between two couples part-detector, the only difference is that the first is more accurate than the second, the criterion of the first must be higher.

Figure 2 shows an illustration of this principle. We are searching for a polygone $ABCD$ made of four segments and we have only a segment detector. Assume the vertical right side named AB has been detected. For sides BC and AD, we have some information about their length and their angles with AB thanks to $\underline{X}_l$ and $\mathbf{Cx}_l$. We want to know

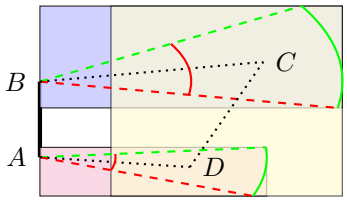


Fig. 2: The polygone we are searching for. In black it is the side we have already detected. In green and red it is the border or acceptable segments: in full, it is limits given by angles and in dashed it is limits given by lengths. In dotted point, it is the median value expected for the three sides. The full rectangle represent the zone of search for each side.

if we should begin searching BC, CD or AD. If we find CD, all the polygone parameters will be known but it is the one with the bigger zone and with the more risk to choose a bad segment. AD and BC are more reliable because one of their extremities is fixed. As AD has the smaller zone and the detector is the same, AD will be chosen.
Finally, the selection criterion C_s used with respect to previous constraints is the following:

$$C_s = P(moveforward)\frac{d}{t} \quad (1)$$

It can be seen like the convergence speed towards the objective.

- t is simply the time that will be taken by the detector.
- d is the distance between the covariance matrix $\mathbf{Cx}_l$ at the current step l and the predicted covariance matrix if the detection succeeds $\mathbf{Cx}_{l+1|l,d}$ at the next step $l+1$ related to the objective covariance $\mathbf{C}_{goal}$. Assume $\mathbf{C}_{goal}$ is a diagonal matrix (if it is not, the basis has to be changed to compute the criterion).

$$d = \sqrt{\sum_i [\frac{\sqrt{\mathbf{Cx}_l(i,i)} - \sqrt{\mathbf{Cx}_{l+1|l,d}(i,i)}}{cObj(i,i)}]^2} \quad (2)$$

  The objective is seen as the scale of measure which allows to put on the same plan advances on every axis. For a covariance matrix the square root of the diagonal coefficients represent the bound max on each axis. So, on one axis, our distance represents the advance of this bound max with regard to our accuracy objective. Figure 3, shows an example in a 2D space.
- $P(moveforward)$ represents the probability that the recognition really progresses which means the probability we made a good detection (if the feature chosen is the one we are looking for) and this detection is new (to avoid risk of surconvergence in the Kalman filter). It is computed using a bayesian network.

2.4.3 Bayesian Networks

They are usually used to formalize the knowledge in an uncertain environment and are therefore quite appropriate in this case. In 4, the structure of our Bayesian network is shown. It is the same structure for all the couple part-detector.

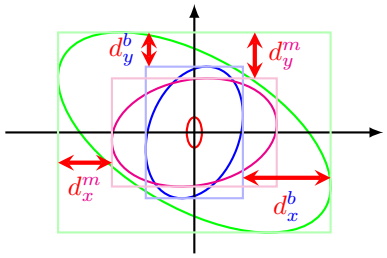


Fig. 3: Distance principle. In red is the ellipse objective. In green is the current ellipse covariance. Two examples ellipses are represented in blue en magenta. The distance gains on each axis are represented by the red arrows for the two example couples. The distance d for each curve can be calculated thanks to the two distances (d_y and d_x) and the value of objective on each axis.

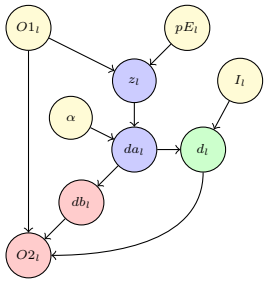


Fig. 4: Our bayesian network. Yellow nodes are entry nodes (their probabilities are known). Pink nodes are nodes we are looking for. Green node is the observation node: after detection it will become an evidence. Blue nodes are intern nodes.

Events modeled in this bayesian network are:

- $O1_l$: confidence of localization before update
- pE_l: the part we are looking for exists.
- z_l: observability of a landmark with selected detector.
- da_l: the landmark we are looking for has been detected.
- d_l: at least a landmark has been detected.
- db_l: a landmark has been detected and data association is true.
- I_l: this couple part-detector has never been used before.
- $alpha$: the probability that the detector doesn't see a landmark whereas it is present.
- $O2_l$: confidence of localization after update

With all the events, it is possible to construct a bayesian network which can take into account the detectors performances, the parts occlusion[1], the ambiguity, the size of search area[2].

2.4.4 Focus

The focus is the step where we compute for each couple part-detector, the part state η_{i_l} (see 2.2). This part state is used to compute a detection zone Zi_l. This one is simply a rectangular zone that include all features in the space defined by $\underline{\chi}_i$ and Σ_i. Figure 2 shows an example of detection zone. Focus is needed by the criterion selection: the time t depends on the size zone, the predicted covariance matrix $\mathbf{Cx}_{l+1|l,d}$ is

[1]the observability will be not the same if it is hidden or not.
[2]the bigger the zone is, better are the chances to find others candidates.

computed assuming the feature found is $\underline{\chi}_i$, and probabilities in the bayesian network depend on this focus too. Besides its utility for criterion selection, focus has multiple advantages. First, time computation is improved because the detector is rarely used in the whole image (in general just at the beginning). Second, it increases the detector performance by improving the signal to noise ratio. If the state part is really restricted (Σ_i has low values on its diagonal), the probability to detect outliers is tiny.
Now the detection zone Zi_l is defined, we can detail the detection step.

2.5 Detection

2.5.1 Principle

For each part, one or more detectors j are enabled (but all will return the same type of features). Detectors will return all features in their detection zone (for example, a segment detector will return all segments found). Detecting a part consist in finding an adequate feature to estimate the parameters $\underline{\chi}_i$, so to see if among all the features found, there is at least one that can be acceptable. If there is many candidates, it has to choose between them.

2.5.2 Characteristics

The detectors are not perfect so a detector j is characterized by:

- α_j: the probability it doesn't see a feature even if it is present in image. It is the percentage of false negative.
- β_j: the probability it see a feature that not exists. It is the percentage of false positive.
- Σ_j: is the detector covariance, it represents the accuracy of the detector.

2.5.3 Selection of feature

The feature $\underline{\chi}_f$ we are searching for, should also follow a gaussian law centered in $\underline{\chi}_i$ and with a covariance of Σ_i. So an feature estimate by the detector j in $\underline{\hat{\chi}}_f$ is admissible for a part i if it is in the space defined by $\underline{\chi}_i$ and$\Sigma_{i,j}$ where $\Sigma_{i,j} = \Sigma_i + \Sigma_j$. The figure 5 gives an example. to know if

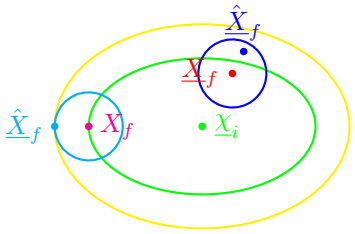


Fig. 5: Detection error. In red, it is the real point we are looking for. In green, it is the part estimation of where should be this point. In blue, it is the point found by the detector (instead of the red). So the blue point is not in the ellipse defined by the part because of the detector error. But this point is in the yellow ellipse that represent the covariance defined by the sum of the two other covariances. In cyan and magenta, it is another example with the real point that is really at the limits.

a feature $\underline{\hat{\chi}}_f$ found with the detector j, is candidate for a part i, we compute a Mahalanobis distance:

$$d_{i,j,f} = \sqrt{(\underline{\hat{\chi}}_f - \underline{\chi}_i)^T \Sigma_{i,j}^{-1} (\underline{\hat{\chi}}_f - \underline{\chi}_i)} \quad (3)$$

The feature is candidate if $d_{i,j,f} < S$ where S is a fixed threshold. Usually, we take $S = 2$. If there are several candidates, the chosen feature is the one with the lower Mahalanobis distance.

2.5.4 Part Update

After detection, the part state is updated.

- detection failed : $\eta_{i_{l+1}} = \{\underline{\chi}_{i_l}, \Sigma_{i_l}, P(pE_{i_{l+1}})\}$ where $P(pE_{i_{l+1}}) = P(pE_{i_l}|\bar{d})$.
- detection succeeded : $\eta_{i_{l+1}} = \{\underline{\hat{\chi}}, \Sigma_j, P(pE_{i_{l+1}})\}$ where $P(pE_{i_{l+1}}) = P(pE_{i_l}|d)$. We consider the parameters of the part are the one of the chosen feature. As the accuracy of the features parameters is that of the detector, we consider the new accuracy of the parameters part is also the one of the detector.

All probabilities are computed using our bayesian network with inference (the event d_l is now known). The probabilities of some other nodes like db_l are also changed because now we exactly know the number of possible choices whereas in the selection step it was an estimation.
Now the part is updated, we have to update the object state.

2.6 Update

Like for the part update, there are two cases : if the detection succeeded or not.

2.6.1 Success

$\mathbf{E}_{l+1} = \{\underline{X}_{l+1}, \mathbf{Cx}_{l+1}, P(O_{l+1})\}$ where $P(O_{l+1}) = P(O2_l|d)$ is computed using always our bayesian network with inference. For $\underline{X}_{l+1}$ and $\mathbf{Cx}_{l+1}$, an Extended Kalman Filter is used. For the prediction step, the object stays at the same place during all recognition process so $\underline{X}_{l+1|l} = \underline{X}_l$ and $\mathbf{Cx}_{l+1|l} = \mathbf{Cx}_l$. With the previous notation, the Kalman equations can be wrote as follows :
$\underline{Y}_l = \underline{\hat{\chi}}_f - \underline{\chi}_{i_l}$
$\underline{S}_l = \mathbf{J}_{f_{i_l}} \mathbf{Cx}_{l+1|l} \mathbf{J}_{f_{i_l}}^T + \Sigma_{i_{l+1}} = \mathbf{J}_{f_{i_l}} \mathbf{Cx}_l \mathbf{J}_{f_{i_l}}^T + \Sigma_j$
$K_l = \mathbf{Cx}_{l+1|l} \mathbf{J}_{f_{i_l}}^T \underline{S}_l^{-1}$
$\underline{X}_{l+1} = \underline{X}_{l+1|l} + K_l \underline{Y}_l$
$\mathbf{Cx}_{l+1} = (Id - K_l \mathbf{J}_{f_{i_l}}) \mathbf{Cx}_{l+1|l}$ where Id is the identity matrix of the adequate size.

2.6.2 Fail

When the detection failed, that can be for several reasons :

- a) the part does not exist : $P(\overline{pE_l})$.
- b) the feature searched cannot be observed: $P(\overline{obs}) = P(\overline{z_l}|O1_l \ \& \ pE_l)$.
- c) the detector has not found the feature but it is present. $P(\alpha_l)$. It is an error of the detector.
- d) the estimate is wrong: $P(\overline{O1_l})$

If we are in the case d), that can mean that we have made an error in previous recognition steps (the association between

a part and a feature was not the good). So, in this cas, we want to cancel this bad association and to start again from this step. In all other cases, the non detection has another reason and so we can continue with our current hypothesis.
No return: in this case, we simply have $\mathbf{E}_{l+1} = \{\underline{X}_l, \mathbf{Cx}_l, P(O_{l+1})\}$ where $P(O_{l+1}) = P(O2_l|\bar{d})$. As the detection failed, we have no more information about the $\underline{X}_l$ and $\mathbf{Cx}_l$, that why they stay the same.
Return: in this case, we have $\mathbf{E}_{l+1} = \{\underline{X}_m, \mathbf{Cx}_m, P(O_{l+1})\}$. m is the indice previous the last multiple success detection or if there wasn't any ($m = 0$): we return to the initial value. Indeed, we have to return before the bad association so before the last success detection where the associated part has more than one possible candidate. For the probability, it is more complicated: we don't want to have the same probability as before. Image we begin with $P(O_0) = 0.5$ so we don't know if the object is in the image. If all detections failed, we can say at the end that it is very likely not present. So $P(O_0)$ should be low. In fact, we use independant bayesian network similar to the one shown in figure 4 whereas we should chain all this bayesian network like the figure 6 shows. We have always $P(O2_l) = P(O1_{l+1})$.

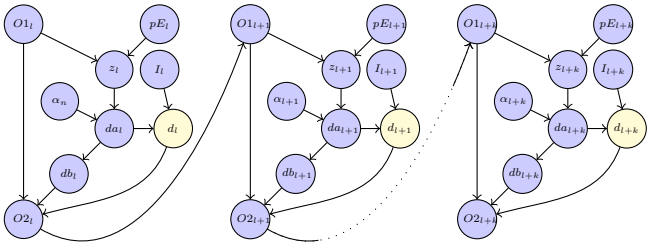


Fig. 6: Chain of Bayesian Networks. Yellow nodes are evidence.

Using directly this big chain will be too much computationnal so we do the return step by step using some calculation tricks. We replace two bayesian networks chain by the one represented in figure 7. The node Cal_l is an artificial node that is built so that $P(O1_l|Cal_l \ \& \ \hat{d}_l) = P(O1_l|\hat{d}_l, \hat{d_{l+1}})$ where $\hat{d}_k = d_k$ or $\hat{d}_k = \bar{d}_k$ according to the situation at this step.

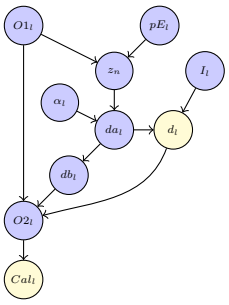


Fig. 7: Our bayesian network for return. Yellow nodes are evidence.

Now, the object state is updated, we will test if it is the end of recognition or if we should try a new detection.

2.7 End Criterion

There are three cases that make the progam ends.

- The first is the better case: we have found a lot of parts and our detectors are enough accurate and reliable. We can reach the goals fixed in term of precision and confidence : $P(O_l) \geqslant P_{goal}$ et $\forall i, \mathbf{Cx}(i,i) \geqslant \mathbf{C}_{goal}(i,i)$ (if the objective matrix is diagonal).
- The second case is the opposite: we couldn't find anything coherent with the studied object so we can say it is not present in the image if $P(O_l) \leqslant 1 - P_{goal}$
- The third case is between the two, we found something but not enough to reach to our objective but we have done all we can (for each part we have used every possible detectors) and so it is not possible to have better: objectives are too ambitious, detectors are not enough reliable or not enough accurate.

3. Results

The objective of this section is not to present a powerfull application which surpasses all the state of the art in object recognition. Here, we want to validate all the steps we presented, show the algorithm behaviour on an application and show that it does what we expected.

3.1 Road Sign

We choose to make a first application to detect a speed limit road sign. We define only three parts for this object: a red blob, a big circle and a little circle. We use two detectors. The first one is a fast but inaccurate detector of red blob. The second one is a slow but very accurate and reliable circle detector. Note as we search circles, we can find the road sign only when it is not too much disturbed and so there is well circles in image and no ellipses. Because of the impact of time, it will always begin by the red blob part.

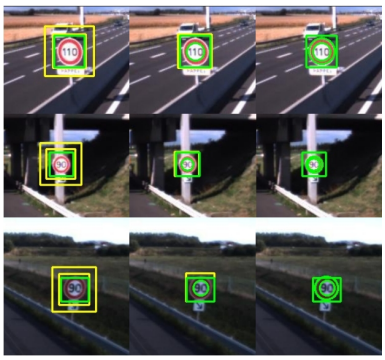


Fig. 8: Some focus examples for a road sign. In yellow it is the detection zone and in green it is the parts already detected. In the first column, as the detector of red returns really unaccurate zone, the detection zone for circle is bigger than the red zone found. In the second column, the detection for the last circle (the big one) is smaller than in the first column : as the little circle has been detected and its detector is more accurate, the zone has been reduced. The last column shows all parts found.

Figure 8 shows three differents images with a road sign. The initial covariance is so that the first detector searches in the whole images (not shown here). Each row of this figure is for a different initial image. The first column is after the detection of the red blob. The second column is after the detection of the first circle (which is the little circle). In the last column, the road sign is found. Thanks to the focusing, the zone sizes decrease with the successive detections.

3.2 Face Detection

Here, the objective is to illustrate that our algorithm is generic and can be easily used for others applications with other detectors. To do that, we choose a face detection application. We decomposed this objects into three parts: a full face part, a right eye part and a left eye part. To detect these parts, we use the eye detector of OpenCv, the face detector of OpenCv and the face detector of dlib [7]. Figure 9 shows some detection examples. The parameters given are so that the OpenCv face detector is chosen in first by our algorithm (it is faster than the dlib one), after it is the dlib detectors and in end eyes parts are searched.

Fig. 9: Some face detections. The first colum is the results with the dlib face detector. The second colum is the results with the OpenCv detectors (in pink: the faces, in cyan: the eyes). In last column, it is the result of our algorithm: in pink it is the final face found, in green the different parts found, in blue the parts not found.

The first row shows a success example for the three algorithms. The second row is a good illustration of the improvement due to the focus principle: in the whole image eyes are not detected but as the zone is restricted, in our algorithm eyes are found. The third row is another example of this improvement: the dlib doesn't find face in the whole image but used in the restricted zone computed thanks to the previous detection, it finally finds the face.

4. Conclusion

We present in this article a general algorithm for object recognition. The object is assumed to be constructed with several parts and the recognition is made by detecting this one part at a time. In an initialization phase, all the a priori knowledge is given to the algorithm: the initial state of our object but also the existence probabilities of parts and the characteristics of detectors (speed detection and reliability). At each iteration, the algorithm is able to choose what is the best couple part-detector and to launch the detection only for this couple. Add a detector or a part at an object is easy and will just add the computation of one more selection criterion. Detectors return all features found in the ROI computed to be optimal and our algorithm is able to choose the one that correspond the more likely to the part (to decrease the numbers of bad association). Even, if bad associations are done, our bayesian network allows to cancel the bad hypothesis and to continue the recognition trying some others hypothesis until the success or the failure of the recognition.

A road sign detection application illustrates the process of our algorithm. A face detection application shows how this process can increase existing detectors by combining them. In future works, we will developp a more complex application such as it uses the same principle and has a good robustness with good computation time.

References

[1] R. Bajcsy. Active perception. *Proceedings of the IEEE*, 76(8):966–1005, Aug 1988.

[2] Roland Chapuis, Frederic Chausse, and Noel Trujillo. Progressive focusing: A top down attentional vision system. In *Advances in Visual Computing*, pages 468–477. Springer, 2008.

[3] Yu-Ting Chen and Chu-Song Chen. Fast human detection using a novel boosted cascading structure with meta stages. *IEEE Transactions on Image Processing*, 17(8):1452–1464, August 2008.

[4] K Craik. The nature of explanation. Cambridge, 1943.

[5] Li Fe-Fei, Robert Fergus, and Pietro Perona. A bayesian approach to unsupervised one-shot learning of object categories. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, page 1134–1141. IEEE, 2003.

[6] P.F. Felzenszwalb, R.B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(9):1627 –1645, sept. 2010.

[7] Davis E. King. Dlib-ml: A machine learning toolkit. *Journal of Machine Learning Research*, 10:1755–1758, 2009.

[8] Jürgen Kornmeier, Christine Maira Hein, and Michael Bach. Multistable perception: when bottom-up and top-down coincide. *Brain and cognition*, 69(1):138–147, 2009.

[9] Liang Lin, Tianfu Wu, Jake Porway, and Zijian Xu. A stochastic graph grammar for compositional object representation and recognition. pages 1297–1307, July 2009.

[10] Anuj Mohan, Constantine Papageorgiou, and Tomaso Poggio. Example-based object detection in images by components. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 23(4):349–361, 2001.

[11] Loris Nanni and Alessandra Lumini. Heterogeneous bag-of-features for object/scene recognition. *Applied Soft Computing*, 13(4):2171–2178, April 2013.

[12] Constantine P. Papageorgiou, Michael Oren, and Tomaso Poggio. A general framework for object detection. In *Proceedings of the Sixth International Conference on Computer Vision*, ICCV '98, pages 555–, Washington, DC, USA, 1998. IEEE Computer Society.

[13] Giovanni Pezzulo. Coordinating with the future: The anticipatory nature of representation. *Minds and Machines*, 18(2):179–225, June 2008.

[14] Cédric Tessier, Christophe Debain, Roland Chapuis, and Frédéric Chausse. Map aided localization and vehicle guidance using an active landmark search. *International Journal on Information Fusion*, S1566-2535(09)00085-2, September 2009.

[15] Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, page I–511. IEEE, 2001.

[16] Song-chun Zhu and David Mumford. A stochastic grammar of images. page 259–362, 2006.

Estimation of Illuminance/Luminance Influence Factor in Intelligent Lighting System Using Operation Log Data

Yuki Sakakibara[2], Mitsunori Miki[1], Hisanori Ikegami[2],Hiroto Aida[1]
[1]Graduate School of Science and Engineering, Doshisha University, Kyoto, Japan
[2]Department of Science and Engineering, Doshisha University, Kyoto, Japan

Abstract— *We have proposed the intelligent lighting system which provides the required illuminance to user. We introduce this system to a real office. In office, we measure illuminance/luminance Influence factor, and use this value to control the intelligent lighting system. Illuminance/luminance Influence factor needs update because this value changes office environment, e.g. deterioration of lighting fixture, installation of partition. In order to measure this value, we need to turn on and off the lighting fixture one by one. But we can't measure illuminance/luminance influence factor because it might be uncomfortable to office worker. In this paper, we propose method to update illuminance/luminance influence factor using log data of intelligent lighting system without influence measurement. As a result of verification, it was confirmed that this propose method is effective to update Illuminance/luminance influence factor.*

Keywords: Lighting Control, optimization, Saving energy

1. Introduction

In recent years, research has been widely conducted on the impact of the office environment on the productivity of employees, and it has been reported that improving an office environment will lead to improvements in the intellectual productivity of employees[1][2][3]. In particular, according to research that focuses on the lighting environment as one of the aspects of an office environment, providing each employee with the optimum amount of brightness (illuminance) for work is an effective means of improving the office environment[4]. Moreover, the amount of energy consumed in office buildings is increasing year by year, and this increase in energy consumption has become a problem. The power consumed by lighting in office buildings accounts for approximately 40% of the total amount of power consumed[6], and thus reducing the power consumed by lighting will lead to a reduction in the amount of power consumed by an office. With that in mind, the authors are conducting research and development on a lighting system (hereafter, Intelligent Lighting System)[5] that realizes individual illuminance for the purpose of improving the intellectual productivity of employees in an office environment and conserving the amount of energy consumed by offices [9]. The Intelligent Lighting System was tested and verified in a laboratory, after which a prototype system was introduced in multiple offices where demonstration testing was conducted for the purpose of achieving practical applications, and the usefulness thereof was verified. The Intelligent Lighting System is configured by connecting multiple light controllable light fixtures with built-in microprocessors, multiple illuminance meters, and power sensors to a network. By using the optimum technique based on information obtained from the illuminance meters and the power sensors, each light achieves a lighting pattern that both satisfies the target illuminance set by each user and reduces power consumption. With the Intelligent Lighting System, the impact that the luminance of each lighting unit has on the illuminance of each meter (hereafter, illuminance/luminance influence factor) is used to implement lighting control. The Intelligent Lighting System introduced in actual offices measures the illuminance/luminance influence at the time of system introduction (hereafter, illuminance/luminance influence factor measurement method), and the value thereof is used to control the lighting. This method measures the actual illuminance/luminance influence factor that each light has on each illuminance meter by turning off all the lights in the targeted environment and then turning on one light at a time.

2. Intelligent Lighting System

2.1 Overview of an Intelligent Lighting System

The intelligent lighting system, as indicated in Figure 1, is composed of lights equipped with microprocessors, illuminance sensors, and electrical power meters, with each element connected via a network.

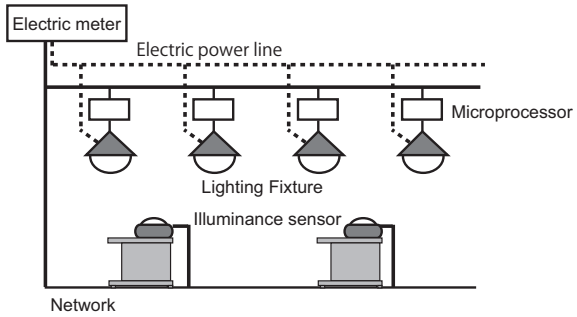


Figure 1: The construction of a intelligent lighting system

Individual users set the illuminance constraint on the illuminance sensors. At this time, each light repeats autonomous changes in luminance to converge to an optimum ligthing pattern. Also, with the intelligent lighting system, positional information for the lights and illuminance sensors is unnecessary. This is because the lights learn the factor of influence to the illuminance sensors, based on illuminance data sent from illuminance sensors. In this fashion, each user's target illuminance can be provided rapidly.

The most significant feature of the intelligent lighting system is that no component exists for integrated control of the whole system; each light is controlled autonomously. For this reason, the system has a high degree of fault tolerance, making it highly reliable even for large-scale offices.

2.2 Control of the Intelligent Lighting System

In the intelligent lighting system, the algorithm where Simulated Annealing (SA) is improved for lighting control (Adaptive Neighborhood Algorithm using Regression Coefficient: ANA/RC) is used to control luminance intensity for each lighting fixture[5][7].

It is possible with ANA/RC to provide the target illuminance with minimum power consumption by making luminance intensity for lighting fixtures the design variable and by using the difference between the current illuminance and target illuminance as well as power consumption as objective functions[7]. Furthermore, by learning the influence of each lighting fixture on each illuminance sensor using the regression analysis and by changing the luminance intensity depending on the results, it is possible to promptly change to the optimal luminance intensity. This algorithm is effective to solve the problem which the objective function is near monomodal function and changes in real time.

And it is possible to cut power consumption about 50 % in actual office[6][9].

2.3 Illuminance/luminance Influence Factor

As explained in the preceding section, the light control algorithm (ANA/RC) determines the next appropriate luminance and endeavors to improve convergence speed by computing the regression coefficient (illuminance/luminance influence factor) from the amount of change in luminance and the amount of change in illuminance[7]. Therefore, understanding the accurate value for the illuminance/luminance influence factor is important for the control of each light by the Intelligent Lighting System. However, because the luminance of each light is randomly changed, the possibility of correlation temporarily occurring with the change in the luminance of each light increases as the number of lights is increased. When this occurs, a phenomenon is observed in the Intelligent Lighting System introduced in the actual offices wherein the correlation between the change in luminance of fundamentally irrelevant lights and the change in illuminance measured with an illuminance meter temporarily increases, and an erroneous illuminance/luminance influence factor is obtained. At this time, phenomena such as a temporary brightening of unnecessary lights occurs. This type of phenomenon is temporary and has almost no impact on the illuminance required by each employee or on overall power consumption. However, when the Intelligent Lighting System was introduced in actual offices, a questionnaire given to the employees therein indicated that the employees were slightly concerned about the phenomenon. In order to improve this, in offices with no movement of the illuminance meter, we proposed a technique to provide the users with more stable lighting control by measuring the actual illuminance/luminance influence factor when the Intelligent Lighting System was introduced and storing that value in a database rather than dynamically estimating the illuminance/luminance influence factor. The technique for measuring the actual illuminance/luminance influence factor computes the illuminance/luminance influence factor from the relationship expression shown by Equation (2) by causing one light at a time to turn on when the Intelligent Lighting System is introduced. This technique is called the illuminance/luminance influence factor measurement method. The illuminance/luminance influence factor is a number that is obtained by combining the perspective [far and near] relationship of the lighting and the illuminance meter, reflection from the wall surfaces, light shielding by partitions, and such, and quantifying the impact of the luminance of a single light on the illuminance of a single meter.

2.4 Issues in Actual Offices

We are conducting demonstration testing of the Intelligent Lighting System in multiple actual offices in the Tokyo area in Japan. In the actual offices in which the Intelligent Lighting System was introduced, there are many fixed seats and no movement of the illuminance meters, and therefore, the illuminance/luminance influence factor computed using the illuminance/luuminance influence factor measurement method described in the preceding section is used to control the lighting. However, the illuminance/luminance influence factor varies depending on the following types of changes in the lighting environment.

When the above-described changes in the lighting environment occur in an actual office, the illuminance/luminance influence factor must be re-measured and updated using the illuminance/luminance influence factor measurement method. However, measuring the illuminance/luminance influence factor in an actual office creates problems for employees trying to work in the office because the lights are repeatedly switched on and off. Therefore, implementing this type of measurement is generally not allowed in many cases. Moreover, implementing measurement tests during the night when employees are not present is also not allowed from

the viewpoint of security. Thus, as a technique to estimate the illuminance/luminance influence factor for this research, a technique is proposed that is based on mathematical programming using operation log data for the Intelligent Lighting System.

3. Estimating the Illuminance/Luminance Influence Factor Using Operation Log Data

A technique for estimating the illuminance/luminance influence factor using operation log data for the Intelligent Lighting System is proposed. By using this technique, the illuminance/luminance influence factor can be updated without interrupting the work going on in the office, and the illuminance/luminance influence factor can be estimated in accordance with changes in the lighting environment without incurring any cost or labor to measure the actual illuminance/luminance influence factor. An example of log data used in the estimation technique is shown in the Figure 2, and as shown therein, the luminance value for the number of lights therein and the illuminance value for the number of illuminance meters are used to estimate the illuminance/luminance influence factor.

Time	luminance (light 1)	luminance (light 2)	measured_illuminance (sensor A)	target_illuminance (sensor A)
9:00	900	900	618	300
9:01	927	918	614	300
9:02	927	918	610	300
9:03	917	899	598	300
9:04	917	899	595	300
9:05	880	854	576	300
9:06	906	802	573	300
9:07	906	802	573	300
9:08	951	761	573	300
9:09	912	715	557	300
9:10	912	715	553	300
9:11	912	715	551	300
9:12	912	715	550	300
9:13	875	700	532	300
9:14	866	672	521	300
9:15	857	644	510	300

Figure 2: Log data of a intelligent lighting system

The relational expression of Equation (1) shows the relationship between the illuminance value obtained from each illuminance meter and the luminance value for the multiple lights. Therefore, if there is a significant difference between the luminance value for each light in the operation log data and the illuminance value of each illuminance meter, the relational expression of Equation (1) can be used to compute the illuminance/luminance influence factor by setting up a system of equations for the illuminance/luminance influence factor.

$$I_i = \sum_{j=1}^{m} R_{i,j} L_j \tag{1}$$

I_x : Illuminance, L_y : Luminance,
$R_{i,j}$ ：Illuminance/Luminance influence factor
m : number of Lighting fixture

However, in some cases, the luminance value in the operation log data and the illuminance value do not vary. In addition, a margin of error exists also in the observed illuminance and luminance values. From these points, the illuminance/luminance influence factor is computed by solving an optimization problem to minimize the margin of error rather by than solving a system of equations. The optimization problem finds the objective function expressed by the square of the difference of the estimated illuminance and measured illuminance for each data series when the luminance of the lighting and the illuminance of the meters at the same time in the operation log data are used as a single data series. The summation for the data series of the objective function thereof is minimized as a single objective function, and the illuminance/luminance influence factor is estimated. The objective function is shown in Equation (2).

$$min : F = \sum_{i=1}^{d} f_i(R_{j,k}) \tag{2}$$

$$f_i(R_{j,k}) = \sum_{j=1}^{n} (E_{i,j} - I_{i,j})^2$$

$$E_{i,j} = \sum_{k=1}^{m} R_{j,k} L_{i,k}$$

d ：number of Data, m : number of Lighting fixtures, n : number of sensors, L : luminance intensity in an operation log data, I ：illuminance intensity in an operation log data, E : estimate illuminance, $R_{j,k}$: illuminance/luminance influence factor(design value)

The optimization problem for the illuminance/luminance influence factor estimation shown in Equation (2) is nonlinear, and therefore the method of steepest descent is used here as the mathematical programming technique.

4. Testing Overview

In this chapter,I examine the usefulness of the proposed method by carried out the following experiment.

- Determine if estimation of the illuminance/luminance influence factor is possible using the proposed technique. (Accuracy Verification Test)
- Determine if the illuminance/luminance influence factor can be updated in accordance with changes to the lighting environment. (Lighting Environment Variation Test)

In the accuracy verification test, the illuminance (true value) computed from the actual illuminance/luminance influence factor measured using the illuminance/luminance

influence factor measurement method and the illuminance/luminance influence factor estimated by the proposed technique were used to compute the illuminance (estimated value), and with that illuminance (estimated value) as the evaluation target, the usefulness of the proposed technique was verified. In the lighting environment variation test, partitions were installed in order to simulate changes in the lighting environment while the Intelligent Lighting System was in operation. When partitions are installed, the illuminance/luminance influence factor of the luminance of the lights on the illuminance of the meters changes, and as a result, the lighting pattern of the lights worsens. In this case, we verified whether or not an optimum lighting pattern can be realized by using the log data after installation of the partitions to update the illuminance/luminance influence factor.

4.1 Testing Enviroment

A plan view of the model environment used in the verification testing is shown in the Figure 3. In this testing, three illuminance meters and 15 white fluorescent lamps were used. In addition, work by employees in an actual environment was simulated, the illuminance meters were installed 0.7m above the floor at positions with 1.9m from the illuminance meters to the ceiling, and testing was conducted. For the light fixtures, white fluorescent, light controllable lamps (FHP45EN) from Panasonic with luminance ranging from a minimum lighting luminance of (30% lighted: 300 cd) to a maximum lighting luminance of (100% lighted: 1000 cd) were used, and for the illuminance sensors, general ANA-F11 type Class A digital illuminance meters were used. Estimation of the illuminance/luminance influence factor in an actual environment was simulated, and the illuminance/luminance influence factor was estimated using 60 minutes of log data in which the employees changed the target illuminance as shown in the Table 1, or more specifically, using 1800 steps of operation log data with 2 seconds per step. The history of illuminance used in the estimation is also shown in the Figure 4.

Table 1: The history of target illuminance

time [min]	sensor A [lx]	sensor B[lx]	sensor C [lx]
0-10	300	500	700
10-20	500	500	700
20-30	300	400	400
30-40	600	500	700
40-50	300	500	700
50-60	300	300	700

4.2 Test Results and Considerations(Accuracy Verification Test)

The illuminance/luminance influence factor was estimated using the proposed technique. In order to clarify the test results here, measurements equal to or greater than 0.05

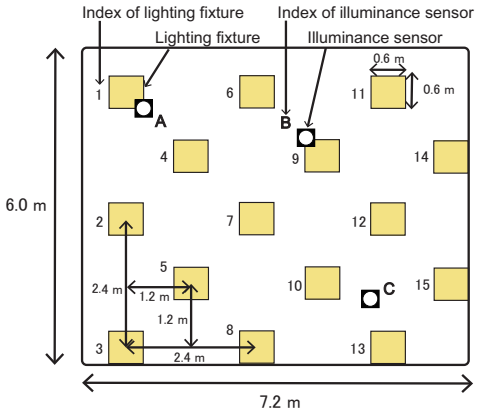


Figure 3: Experiment enviroment(acurracy vertification of experiment)

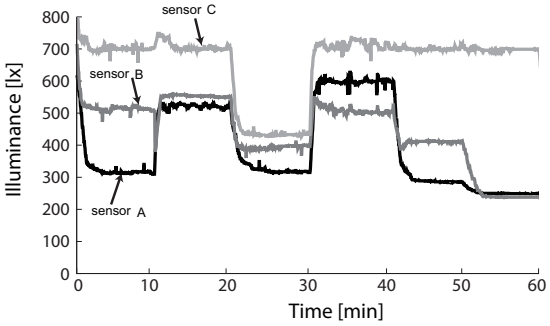
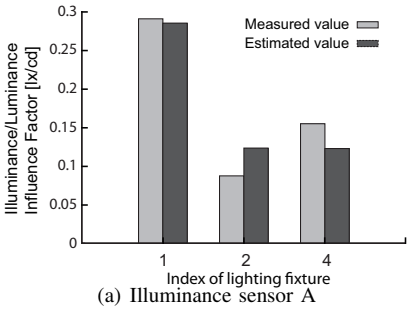


Figure 4: The history of measured illuminance

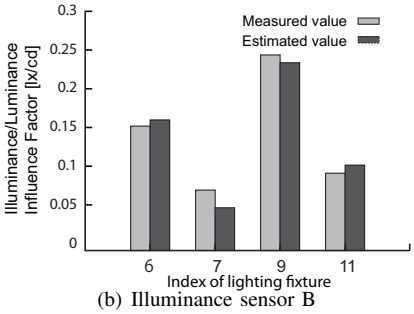
lx/cd for the illuminance/luminance influence factor with respect to each illuminance meter are shown in the figure. Moreover, estimations were achieved with the same accuracy for illuminance/luminance influence factors of less than 0.05 lx/cd.

As also described in section 2.4, the illuminance/luminance influence factor is a quantification of the amount of impact that each light has on each illuminance meter. Therefore, the extent of change that actually occurs in the illuminance when each light is lighted, must be verified. With that in mind, next we used the illuminance/luminance influence factor estimated by the technique proposed in this research to evaluate the proposed technique by computing the illuminance margin of error.

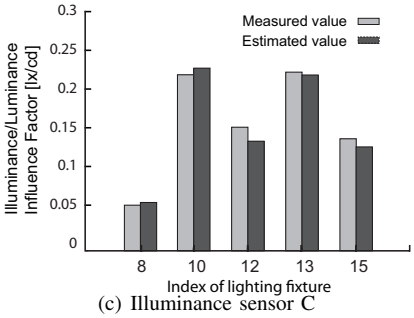
The illuminance margin of error is the error between the illuminance (true value) computed by multiplying the actual measured illuminance/luminance influence factor by the luminance and the illuminance (estimated value) calculated by multiplying the luminance by the illuminance/luminance influence factor estimated by the proposed technique. The proposed technique was verified by preparing a distribution of the illuminance margin of error using a histogram. In order to verify the illuminance margin of error through



(a) Illuminance sensor A

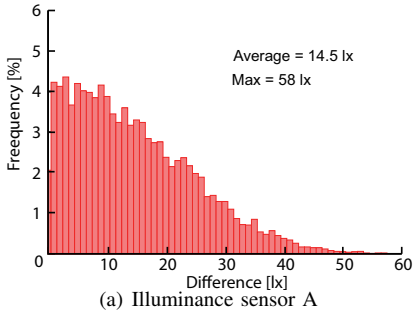


(b) Illuminance sensor B

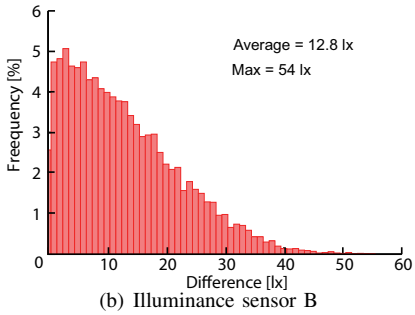


(c) Illuminance sensor C

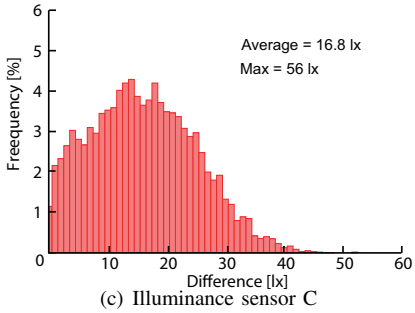
Figure 5: Estimated value of Influence Coefficient and measured value of Influence Coefficient



(a) Illuminance sensor A



(b) Illuminance sensor B



(c) Illuminance sensor C

Figure 6: Difference of illuminance

various lighting patterns, the lighting luminance of the lights in the test environment was randomly prepared, and the data thereof was used for the luminance that was used to compute the histogram. The figure shows a histogram of the illuminance margin of error.

From the figure, the average illuminance margin of error is below 20 lx. This illuminance margin of error is a level that is not perceivable by people working in an office[8], and one could thus argue that the proposed technique is useful in an actual environment. Moreover, the occurrence of the maximum illuminance margin of error of 58 lx could be attributed to the following causes.

- Error due to the calibration curve with regards to the luminance and the signal value.
- Change in the illuminance value due to the temperature of the fluorescent lamps.

First, consideration is given to a calibration curve error. The Intelligent Lighting System controls the design variable as luminance. However, in order to cause a light to turn on with the luminance thereof, a pulse changes the duty ratio in 256 steps based on Pulse Width Modulation (PWM) and transmits a signal value. In the calculation of the signal value from the luminance value, conversion is based on the relationship (calibration curve) between the luminance measured in advance and the signal value. The calibration curve measures the relationship between luminance and the signal value using an illuminance meter, but with typical illuminance meters, a margin of error exists in the measured values, and the illuminance meter (ANA-F11) used in the testing also generated a margin of error of $\pm$ 4%. Therefore, because the calibration curve also contains the margin of error of the illuminance meter, it is thought that this is a factor that caused the illuminance margin of error to occur. Next, consideration is given to the characteristics of the fluorescent lamps. It was found that with the fluorescent lamps used in this test, a maximum margin of error of 160 cd was generated in the lighting luminance of the lights due to temperature. Moreover, it is conceivable that a margin

of error was generated in the luminance of the lights due to individual differences in the light fixtures as well and to the state of deterioration of each light. It is thought that the margin of error occurred due to these factors.

4.3 Test Results and Considerations (Lighting Environment Variation Test)

In the lighting environment variation test, a verification was conducted to determine whether or not the lighting pattern of the lights could be improved by appropriately updating the influence factor according to changes in the lighting environment. As shown in the figure, the test environment was the same as the environment used in the accuracy verification test, and to simulate changes in the lighting environment, partitions were installed below the illuminance meters C. A plan view of model environment used in Lighting Environment Variation Test is shoen Figure 7.

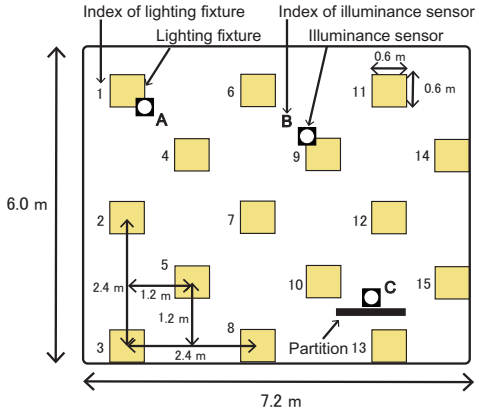


Figure 7: Experiment enviroment (Lighting Environment Variation Test)

The lighting pattern used to realize the target illuminance for the lights after the partitions were installed is shown in the Figure 8. The size of the circle indicates the strength of the lighted luminance of the lights. Moreover, the target illuminance values of the illuminance meters A, B, and C were 400 lx, 500 lx, and 600 lx respectively.

The illuminance/luminance influence factor changes when partitions are installed. As a result, it was found that the light 13 is strongly lighted because the illuminance/luminance influence factor is not updated even though the illuminance/luminance influence factor that the light 13 has on the illuminance meter C decreases. Turning on lights that are not necessary for employees in an office is a factor that has a pejorative effect on energy conservation, and thus this is not an optimum lighting pattern in the Intelligent Lighting System. Next, consideration was given to employees working in an office, and similar to the accuracy verification test, a 1-hour

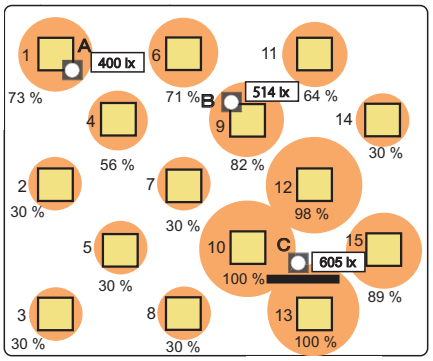


Figure 8: The distribution of luminance(before update illuminance/luminance influence factor)

illuminance convergence test as shown in the table was conducted. Also, based on the obtained operation log data, the illuminance/luminance influence factor was updated using the proposed technique. The lighting pattern that realizes the target illuminance using the updated illuminance/luminance influence factor is shown in the Figure 9.

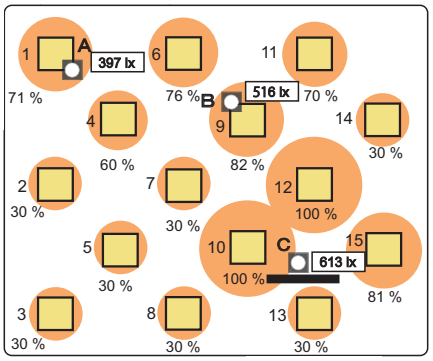


Figure 9: The distribution of luminance(after update illuminance/luminance influence factor)

It was found that by updating the illuminance/luminance influence factor in accordance with changes in the lighting environment, the light 13 in the figure dims compared to before the update. Through this, it became possible to improve the lighting distribution of each light. The figure also shows a comparison of power consumption before and after the update. A reduction in power consumption of about 35 W was possible by improving the lighting pattern.

From these information, one could argue that the proposed technique is useful in the Intelligent Lighting System.

Table 2: Compare of electoronic power

	electric power [W]
before update	607.9
after update	571.1

5. Conclusion

This research addressed the issue of not being able to measure the actual illuminance/luminance influence factor in an actual office and proposed a technique for updating the illuminance/luminance influence factor using operation log data for the Intelligent Lighting System. By using this technique, the illuminance/luminance influence factor is updated from operation log data, and as a result, the illuminance/luminance influence factor can be updated without interrupting the work going on in an office. Moreover, the illuminance/luminance influence factor can be estimated in accordance with changes in the lighting environment without incurring any cost or labor to measure the illuminance/luminance influence factor. Thus, light control tests were conducted at this university, and the log data thereby obtained was used to verify the usefulness of the proposed technique. From the test results, it was found that the illuminance/luminance influence factor can be updated in accordance with changes to the lighting environment, and that the illuminance required by employees in an office can be realized with an optimum lighting pattern. Moreover, improving the lighting pattern also enabled a reduction in power consumption. We believe that by updating the illuminance/luminance influence factor using the proposed technique for the Intelligent Lighting System that was installed in actual offices, a more optimal lighting pattern can be realized for employees working in offices, and that this will contribute significantly to the widespread use of the Intelligent Lighting System.

References

[1] N.Nishihara,S.Tanabe,Subjective experiment on productivity under Moderately hot Enviroment, J.Environ. AIJ,No.568, pp.33-39, 2003

[2] M.Miki, and T.Kawaoka, Design of intelligent artifacts:a fundamental aspects, Proc. JSME International Symposium on Optimization and Innovative Design(OPID97), pp.1701-1707, September 1997

[3] Olli Seppanen, William J. Fisk: A Model to Estimate the Cost-Effectiveness of Improving Office Work through Indoor Environmental Control, Proceedings of ASHRAE, 2005

[4] Peter R. Boyce, Neil H. Eklund, S. Noel Simpson, Individual Lighting Control: Task Performance, Mood, and Illuminance, Journal of the Illuminating Engineering Society, pp.131-142, Winter 2000

[5] M.Miki,T.Hiroyasu,K.Imazato,Proposal for an intelligent lighting system,and verification of control method effectiveness, Proc IEEE CIS, pp.520-525, 2004

[6] M. Miki, T. Hiroyasu, and K.Imazato, Proposal for an intelligent lighting system, and verification of control method effectiveness, Proc. IEEE CIS, pp.520-525, 2004

[7] S.Tanaka,M.Miki,T.Hiroyasu,M.Yoshikata,An Evolutional Optimization Algorithm to Provide Individual Illuminance in Workplaces, Proc IEEE Int Conf Syst Man Cybern, Vol.2, pp.941-947, 2009

[8] T.Shikakura, H.Morikawa, and Y.Nakamura, Research on the Perception of Lighting Fluctuation in a Luminous Offices Environment, Journal of the Illuminating Engineering Institute of Japan, Vol.85 5, pp.346-351, 2001

[9] M. Miki, F. Kaku, T. Hiroyasu, M. Yoshimi, S. Tanaka, J. Tanisawa, and T. Nishimoto, Construction of Intelligent Lighting System Providing desired Illuminance Distributions in Actual Office Enviroment, Journal of the Institute of Electronics of Japan, Vol.J94-D, pp.637-645, 2011

Heuristic applied to the Euclidean Steiner tree problem with node-depth-degree encoding

Marcos Antônio Almeida de Oliveira[1], Telma Woerle de Lima[1], Les R. Foulds[1], Anderson Da Silva Soares[1], Alexandre Claudio Botazzo Delbem[2], and Clarimar Jose Coelho[3]
[1]Intitute of Informatics, Federal University of Goiás, Goiânia, GO, Brazil
[2]Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, SP, Brazil
[3]Departamento de Computação, Pontifícia Universidade Católica de Goiás, Goiânia, GO, Brazil

***Abstract*—In this paper a variation of the Beasley [1] algorithm for the Euclidean Steiner tree problem (ESTP) is presented. This variation uses the Node-Depth-Degree Encoding (NDDE), which requires an average time of $O(\sqrt{n})$ in operations to generate and manipulate spanning forests. For spanning tree problems, this representation has linear time complexity when applied to network design problems with evolutionary algorithms. Computational results are given for test cases available in [2] involving instances up to 50 vertices. These results indicate that NDDE can be used by other techniques besides evolutionary algorithms with good performance. Furthermore, it shows that the proposed algorithm has advantages in the solution found.**

Keywords: Euclidean Steiner Tree Problem, Heuristic Algorithm, Node-Depth-Degree Encoding

I. Introduction

Network design problems (NDPs) are present in several areas of science and engineering [3], [4]. A fundamental NDP is the Steiner tree problem [5]. This problem is one of the oldest in mathematical optimization [6]. Several NDPs and real applications can be represented as generalizations of the Steiner tree problem [3].

The problem of computing minimum Steiner trees is NP-complete [7] as well as the Euclidean Steiner tree problem (ESTP) [5], [8], which is one of its versions. In the case of NP-complete problems it is impossible to find an exact efficient algorithm due to its exponential complexity [5]. Exponential algorithms that can be used in practice are rare [5]. Although exact algorithms provide optimal solutions to the problem [9], [10], [11], it is appropriate to use heuristic algorithms[5]. Many heuristic algorithms have been developed for the problem, including the algorithm of Chang (1972)[12], the Thompson's method of minimum evolution (1973)[13], a hierarchical genetic algorithm of Barreiros (2003)[14], a recent randomized Delaunay triangulation heuristic of Laarhoven and Ohlmann (2010) [15], among others [16], [17], [1], [18], [19].

Most heuristic algorithms for ESTP perform a series of iterative improvements using the MST as a starting solution [20]. The first heuristic algorithms for ESTP proposed by Chang (1972) and Thompson (1973) are based on the iterative insertion of Steiner points in an MST[10], [13], [12]. Beasley (1992) [1] presents an heuristics for ESTP based on finding Steiner optimal solutions to the subgraphs of the Minimum Spanning Tree (MST) of the total number of vertices.

Besides the use of heuristic algorithms, another factor that can improve the performance of the algorithms is the representation used to encode solutions. Several optimization problems can be encoded by a variety of different representations, and the performance can vary greatly depending on the encoding used [23]. An adequate representation should at least be able to encode all possible solutions of an optimization problem [23]. One of the representations that have shown relevant results for encoding trees in graphs is the Node-Depth-Degree encoding (NDDE) proposed by Delbem et al. (2012) [22].

In general, the time complexity of operations for generating spanning trees for an effective representation is $O(n \log n)$[23]. NDDE requires an average time $O(\sqrt{n})$ in operations to generate and manipulate spanning forests. This efficiency boosts its use in heuristics developed for PPRs.

This paper proposes the use of a representation for spanning trees and forests. The possibility of applying the NDDE for the ESTP offers a contribution to several real-world applications that require more effective solutions. The NDDE is used to construct operators that implement the steps of the heuristic algorithm proposed in [1] for ESTP. Computational results are given for test cases available in [2] involving instances up to 50 vertices.

The next section gives a brief description of the Euclidean Steiner tree problem. Section III introduces the NDDE. Section IV presents the Beasley's heuristic algorithm and the algorithm proposed as a variation of this approach. Section V presents the obtained results. Final considerations are presented in section VI.

II. Euclidian Steiner tree problem (ESTP)

The Steiner tree problem consists in to find the minimum network that interconnects n points, by the addition of

auxiliary points to minimize the total cost [24], [1], [19]. An extra vertex that is added to a tree in order to reduce its cost is called Steiner point [24]. Details on the Steiner problem can be found in [9], [24], [10], [11], [6].

The minimum connection network is called Steiner minimal tree (SMT). SMT consists by set N with the n given points and set S with the s Steiner points added [24]. A T network connection is called Steiner tree if it satisfies three conditions: T is a tree; any two edges of T meet at an angle s of $120\,^{\circ}$; a Steiner point cannot be of degree 1 or 2 [10].

The topology of a tree defines the connection array of points of N and S. A minimum tree for a given topology is called relatively minimal. A topology is full if the number of Steiner points is $s=n-2$ [24]. A tree is named full Steiner tree (FST) when it's the relatively minimal tree for a given full topology [24].

The ESTP is the problem of finding the minimum length of an Euclidean tree, covering a set of fixed points in the plane, with the addition of Steiner points [19], [1]. This problem is defined to the Euclidean plane [11]. For two points $a(x_1, y_1)$ and $b(x_2, y_2)$ the Euclidian distance is:

$$d(a,b) = O(\sqrt{((x_1 - x_2)^2 + (y_1 - y_2)^2)}) \qquad (1)$$

Consider V a set of n points in the Euclidian plane. The solution to the ESTP is obtained through the minimum spanning tree (MST) of $V \cup S$ in which S is a set of Steiner points [1].

III. Node-Depth-Degree Encoding (NDDE)

The Node-Depth-Degree Encoding (NDDE) proposed in [22], is based on the concepts of vertex, depth and degree of a vertex in an undirected graph. The way of representing trees is composed of linear lists of triples (vertex, depth, degree). The order in which the triples are arranged in the list defines the edges that make up the tree. A vertex is always linked to the first preceding vertex with lower depth [22]. This order can be obtained by a depth search on the tree from a vertex chosen as the root [22].

Figure 1 illustrates an example of a spanning tree of a graph with 9 vertices. This spanning tree is represented by the NDDE presented in Table I.

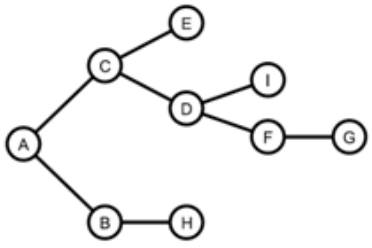


Figure 1: Example of a spanning tree with 9 vertex.

NDDE has two operators to generate new forests from an existing one. These operators have been applied in evolutionary algorithms with satisfactory results for network design problems [22]. This justifies the possibility of using NDDE in other techniques, such as heuristics for the ESTP.

Table I: Linear list containing in each column the NDDE triple representing the spanning tree presented in Figure 1.

Position	0	1	2	3	5	5	6	7	8
Vertex	A	B	H	C	D	F	G	I	E
Depth	0	1	2	1	2	3	4	3	2
Degree	2	2	1	3	3	2	1	1	1

IV. Proposed approach

Proposed approach consists in to develop the algorithm proposed in [1] with NDDE. In order to achieve this propose we need to construct operators to implement the steps of the heuristic algorithm of [1]. In the following paragraphs we present the heuristic algorithm of Beasley (1992), and modifications proposed in this paper.

A. Algorithm proposed by Beasley (1992)

Beasley (1992) [1] presents an heuristics for ESTP and the Rectilinear Steiner Tree Problem (RSTP). This heuristics is based on finding Steiner optimal solutions to the subgraphs of the Minimum Spanning Tree (MST) of the total number of vertices. The heuristic considers all subgraphs of the MST that contain four vertices, find an optimal Steiner tree for each subgraph. After it adds to set V the Steiner vertices obtained for these subgraphs [1].

This heuristic consists, basically, of the following steps. Consider V a set of vertices maintained by the algorithm. V is initialized from the given points and in the end of the algorithm is a set containing the initial points plus the Steiner points whose MST is the solution offered by the heuristic. Algorithm 1 presents the steps performed by the heuristic.

Algorithm 1: Algorithm of Beasley (1992)

1) Consider: $T(K)$ the cost of connecting K vertices through its MST; $T_S(K)$ the cost of connecting the same K vertices, through its Steiner minimal tree (SMT); V_0 the set of initial vertices and t the iteration counter. Configure $V = V_0$ and $t = 0$.
2) Find the MST of V. Attribute $t = t + 1$;
3) Enumerate the set L being all the subtrees of four vertices K in the MST.
4) For each $K \in L$ calculate the reduction of cost $R(K)$ that occurs when the vertices of K are connected via SMT.
5) If $max[R(K)|\ K \in L]=0$ stop.
6) Be M an empty set of vertices with each new iteration. Order in a decreasing way according to $R(K)$ the sets of vertices $K \in L$. In this ordered set, for each K, if $|M \cap K|=0$ and $R(K)>0$ do:
   a) Add the Steiner points associated with V.
   b) Set $M=M \cup K$

7) Find the MST of V and remove any vertex of V which represents a Steiner point with a degree less than or equal to two.
8) For each Steiner point with degree 3, move this vertex to its optimum location. Repeat this process until no cost reduction os possible.
9) Find the MST of V and for each subtree (V* of cost C*) that has the appropriate topology for FST:
   a) Apply the algorithm of Hwang [25] to find the FST of V*.
   b) If the cost of the FST is smaller than C* move the Steiner vertices to the locations given by the FST.
   c) If the cost of the FST is greater than C* try a cost reduction by removing Steiner vertices that are increasing the cost of V*.
10) Go back to step (2).

For more details on this algorithm see [1]. Beasley (1992) [1] shows computational results which indicates that this heuristic provides better quality solutions compared to other heuristics [12], [16]. The improvement of using this heuristic is typically 3 to 4% [1].

B. Algorithm with NDDE

We proposed a modification on Beasley algorithm [1] that consists in using the NDDE as a graph tree data structure. Another difference is that our algorithm does not implement step 9 from the original heuristic. The main purpose, of this implementation, is to demonstrate the use of NDDE to solve the ESTP without evolutionary algorithms. Another goal is to indicate the advantages of NDDE to implement some steps of the based heuristic.

NDDE is used to represent the tree topology. A graph structure represents the locations of the vertices in the set V defined for the Beasley heuristics. For this problem, we need to add three other informations to each node in NDDE. The Steiner value indicates if the node represents a Steiner point; the value parentIndex indicates the position of the parent node in the tree hierarchy; and childIndex indicates the position of the last child in the tree hierarchy.

The Steiner value contributes, in some operations applied only to Steiner points, in identifying these points. The parentIndex and childIndex columns help identifying the edges of a given node. One of the edges connects the node to parentIndex, the other should be between the position of that node and childIndex. For example, at one Steiner point, its degree should be 3 and the edges connecting this node should be in the positions parentIndex, node index + 1, and childIndex. Thus, these connections can be discovered without the need to iterate through the NDDE. We will describe below some operators implemented using the NDDE.

1) Sets query operator (SQO): The SQO described in the Algorithm 2 was used to implement step 3 of the Algorithm 1. The Algorithm 2 has as input the MST represented by NDDE, and returns the set L.

Algorithm 2: Sets query operator

1) Considering a, b, c and d, the four elements of the set K that will be added to L. For each MST node represented by the NDDE, starting from the last and going up to the first node of the tree, do:
   a) Consider a as the current node;
   b) If the degree of a is greater than or equal to three, get b, c and d by browsing the MST from the parentIndex column, add the new K to L;
   c) If the degree is greater than or equal to 1, configure b with the parentIndex of a, and try to find subtrees rooted at b such that c is the child of b and d is the child of c; Add all possible K to L;
   d) If the degree of a is greater than or equal to three, get all possible star type subtrees, in which b may be the parentIndex of a or any of its children, and c and d are children of a; Add all possible K to L;

2) Node Removal Operator (NRO): The NRO described in the Algorithm 3 was used to implement step 7 of the Algorithm 1. In this step the Steiner nodes added that has degree less than 3 must to be removed from the tree. The Algorithm 3 has as input the MST represented by the NDDE and the set V returns the new MST.

Algorithm 3: Node Removal Operator

1) While it is possible to remove nodes:
   a) for each node i of MST, if the node represents a Steiner point with degree lower than three:
      i) Remove the vertex of the set V;
      ii) Move the nodes of the subtree rooted at i to the immediately preceding position, decreasing the depth of each node by 1;
      iii) If i does not have a child, decrease 1 degree from the parent of i;
      iv) Move all subsequent nodes that are not subtree of i to the immediately preceding position on the list;
      v) Update the parentIndex and childIndex of the nodes after i.

One of the contributions to the algorithm efficiency attributed to the use of NDDE is the operator SQO. Subtrees of four nodes are found in three possible ways. The first easily finds the subtrees by browsing the parentIndex value. In the second and third cases the NDDE limits the search of subtrees on the list until the index indicated by the childIndex of the node. The NDDE is essential to keep indexes parentIndex and childIndex updated, and also to identify when it is possible to perform each of these operations.

With the NDDE the update operation of parentIndex and childIndex indexes can be easily done through an iteration that traverses the tree and gets these indexes from a dynamic

vector that stores the last indexes of each depth. Thus, for every node of the current iteration parentIndex can be found from the last index of the previous depth, at the same time this element, corresponding to parentIndex, will have the index of the current iteration node as childIndex.

In the NRO operator the node is easily removed, dragging the elements to the previous position from the first element subsequent to the removed one. The NDDE maintains its structure with only three exceptions: In case the element removed is a leaf, the parent node must have its degree reduced by one unit; the element that composes a subtree of the removed node should have their degrees decreased by one unit in order to become subtree of the parent of the node removed; The parentIndex and childIndex indexes must be updated. These two operators already indicate some advantages in using NDDE.

V. RESULTS

The algorithm presented in this paper was implemented in C language. The tests were run on an Intel (R) Core (TM) i3-2120 CPU @ 3.30 GHz with 4GB of RAM and Operating System Windows 7.

Test cases were the same used by Beasley (1992) available at [2]. Each test case consists of 15 different instances of *n* points in the Euclidean plane with two dimensions. These *n* points, represented by the pair *(x, y)*, have values ranging from 0 to 1, accurate to 7 decimal places.

Follows a comparison between the results of the algorithm of Beasley (1992) [1] and the proposed approach. The results show the minimum, average, and maximum values of the different graphs for each number of nodes. Table II shows the number of iterations until the algorithm stops. We can observe that the algorithm with the NDDE expends a high number of iterations in average and maximum than Beasley algorithms. Probably, this result is a consequence of the removal of step 9.

Table II: Comparison of the number of iterations.

Number of iterations.						
	Beasley			Algorithm with NDDE		
N	Min.	Avg.	Max.	Min.	Avg.	Max.
10	2.000	3.000	4.000	2.000	6.933	40.000
20	3.000	3.600	5.000	3.000	14.200	35.000
30	3.000	3.533	5.000	3.000	10.000	21.000
40	3.000	5.133	21.000	4.000	12.600	35.000
50	3.000	4.067	7.000	3.000	12.333	23.000

Table III shows the percentage of reduction achieved given by

$$100(T(V_0)\text{-}T(V_F))/T(V_0) \quad (2)$$

, where $T(V_0)$ is the cost of the minimum spanning tree and $T(V_F)$ is the cost of the final solution. Table III also present the percentage of reduction between the MST and the optimum SMT. We can observe that the proposed algorithm provides solutions very close to the optimal solution and the solutions obtained by the Beasley algorithm.

Table III: Comparison of the percentage of reduction in relation to the MST.

Percentage of reduction.							
	Beasley			Algorithm with NDDE			Optimum SMT
N	Min.	Avg.	Max.	Min.	Avg.	Max.	
10	0.477	3.138	6.168	0.477	3.140	6.168	6.168
20	1.389	3.015	4.737	1.384	2.958	4.367	4.758
30	2.059	2.868	4.752	2.059	2.841	4.639	4.964
40	1.669	3.024	4.174	1.643	3.023	4.135	4.407
50	2.138	2.841	3.620	2.053	2.806	3.687	3.786

Table IV presents the number of Steiner points add by the algorithms. We can observe that both algorithms added the same number of Steiner points in most of the cases tested. It shows that the removal of step 9 did not effectively affect the number of Steiner points added.

Table IV: Comparison between the number of Steiner points.

Number of Steiner Vertices.						
	Beasley			Algorithm with NDDE		
N	Min.	Avg.	Max.	Min.	Avg.	Max.
10	2.000	3.400	6.000	2.000	3.400	6.000
20	6.000	7.667	10.000	6.000	7.933	10.000
30	9.000	11.667	16.000	9.000	11.800	15.000
40	13.000	15.733	18.000	13.000	15.733	18.000
50	16.000	19.333	22.000	16.000	19.533	22.000

VI. CONCLUSIONS

In this work we show that is possible to apply the NDDE to the Euclidean Steiner tree problem with favorable results. We want to highlight that the use of NDDE reduces the running time of the algorithm. In Beasley approach, the test were executed in a supercomputer and in our approach we use a personal computer and still can reach the results in some seconds. In addition to this new class of problems, it was also found that the NDDE can be applied to other heuristic techniques. Up until now, it had only been applied to Evolutionary Algorithms.

VII. ACKNOWLEDGMENT

Thanks the financial support from Fundação de Amparo à Pesquisa do Estado de Goiás (FAPEG).

REFERENCES

[1] J. E. Beasley, "A heuristic for Euclidean and rectilinear Steiner problems," *Eur. J. Oper. Res.*, vol. 58, pp. 284-292, 1992.

[2] J. E. Beasley, "OR-Library: distributing test problems by electronic mail," *Eur. J. Oper. Res.*, vol. 41, pp. 1069-1072, 1990.

[3] J. Smith, *Generalized Steiner network problems in engineering design*. In: Design optimization., Ed. Orlando, United States of America: Academic Press, Inc, 1985.

[4] A. Delbem, A. de Carvalho, and N. Bretas, "Main chain representation for evolutionary algorithms applied to distribution system reconfiguration," *Power Syst. IEEE Trans.*, vol. 20, pp. 425-436, 2005.

[5] Michael R. Garey, and David S. Johnson, *Computers and Intractability; A Guide to the Theory of NP-Completeness* Ed. New York, USA: W. H. Freeman & Co., 1990.

[6] E. N. Gordeev, and O. G. Tarastsov, "The Steiner problem: A survey," *Discrete Math. Appl.*, vol. 3, pp. 339-364, 1993.

[7] M. R. Garey, R. L. Graham, and D. S. Johnson, "The Complexity of Computing Steiner Minimal Trees," *SIAM Journal on Applied Mathematics*, vol. 32, pp. 835-859, 1977.

[8] M. Garey, and D. Johnson, "The rectilinear Steiner tree problem is NP-complete," *SIAM Journal on Applied Mathematics*, vol. 32, pp. 826-834, 1977.

[9] Z. A. Melzak, "On the problem of Steiner," *Canadian Mathematical Bulletin*, vol. 4, no. 2, pp. 143-148, 1961.

[10] F. Hwang, and D. Richards, "Steiner tree problems," *Networks*, vol. 22, pp. 55-89, 1992.

[11] F. K. Hwang, D. S. Richards, and P. Winter, *The Steiner Tree Problem* Ed. Netherlands: Elsevier Science Publishers B. V., 1992.

[12] S. Chang, "The generation of minimal trees with a Steiner topology," *Journal of the ACM JACM*, vol. 19, no. 4, pp. 699-711, 1972.

[13] E. A. Thompson, "The method of minimum evolution," *Annals of human genetics*, vol. 36, no. 3, pp. 333-40, 1973.

[14] J. Barreiros, "An hierarchic genetic algorithm for computing (near) optimal Euclidean Steiner trees," in *Proceedings of the Genetic and Evolutionary Computation Conference*, 2003, p. 8.

[15] J. W. Laarhoven, and J. W. Ohlmann, "A randomized Delaunay triangulation heuristic for the Euclidean Steiner tree problem in Rd," *Journal of Heuristics*, vol. 17, no. 4, pp. 353-372, 2010.

[16] J. M. Smith, D. T. Lee, and J. S. Liebman, "An O(n log n) heuristic for steiner minimal tree problems on the euclidean metric," *Networks*, vol. 11, no. 1, pp. 23-39, 1981.

[17] D. Du, and Y. Zhang, "On better heuristics for Steiner minimum trees," *Mathematical Programming*, vol. 57, no. 1-3, pp. 193-202, 1992.

[18] J. E. Beasley, and F. Goffinet, "A delaunay triangulation-based heuristic for the euclidean steiner problem," *Networks*, vol. 24, no. 4, pp. 215-224, 1994.

[19] D. Dreyer, and M. Overton, "Two heuristics for the Euclidean Steiner tree problem," *Journal of Global Optimization*, pp. 1-14, 1998.

[20] M. Zachariasen, "Algorithms for Plane Steiner Tree Problems," PHD Thesis, University of Copenhagen, Copenhagen, Denmark, March 1998.

[21] S. Fortune, *Voronoi diagrams and Delaunay triangulations*. In: Handbook of discrete and computational geometry., Ed. United States of America: CRC Press, Inc., 1977.

[22] Alexandre C.B. Delbem, Telma Woerle de Lima, and Guilherme P. Telles, "Efficient forest data structure for evolutionary algorithms applied to network design," *IEEE Transactions on Evolutionary Computation*, vol. 16, no. 6, pp. 829-846, 2012.

[23] Franz Rothlauf, *Representations for genetic and evolutionary algorithms*, 2nd ed., Ed. Netherlands: Springer-Verlag, 2006.

[24] E. N. Gilbert, and H. O. Pollak, "Steiner Minimal Trees," *SIAM Journal on Applied Mathematics*, vol. 16, no. 1, pp. 1-29, 1968.

[25] F. Hwang, "A linear time algorithm for full Steiner trees," *Operations Research Letters*, vol. 4, no. 5, pp. 235-237, 1986.

Reducing the Number of Times Lighting Control is required to reach Illuminance Convergence in the Intelligent Lighting System

Shohei MATSUSHITA[1], **Sho KUWAJIMA**[1], **Mitsunori MIKI**[2], **Hisanori IKEGAMI**[1], **and Hiroto AIDA**[2]
[1]Graduate School of Science and Engineering, Doshisha University
[2]Department of Science and Engineering, Doshisha University,
1-3 Tataramiyakodani, Kyotanabe-shi, Kyoto, 610-0394 Japan

Abstract— *We are conducting research of an Intelligent Lighting System to realize personal illuminance for each employee in an office. The effectiveness of the Intelligent Lighting System has been recognized, and we are currently conducting demonstration experiment of the system in multiple actual offices. However, when introducing the Intelligent Lighting System in large-scale offices, as the number of lighting fixtures and illuminance sensors that configure the system increases, the time required for lighting control increases, which is problematic. Thus, in this paper, we propose a technique to reduce the number of times that lighting control is implemented in order to solve the problems associated with expanding the introduction scale of the Intelligent Lighting System. Herein, we present the results of our verification experiment and a comparison with a conventional Intelligent Lighting System, and demonstrate that the number of times lighting control is used until illuminance convergence is reached can be significantly reduced.*

Keywords: Lighting, Intelligent Lighting System, illuminance simulation, daylight, optimization, illuminance sensor

1. Introduction

In recent years, continuing research into the office environment has identified that the office environment has a major influence on workers[1], [2]. Previous research has reported that improving the office environment can increase workers' intellectual productivity and comfort. With regard to the lighting environment, it has also been reported that providing each worker's desired brightness can raise intellectual productivity[3].

However, at present, the standard lighting design of Japanese offices features a desktop illuminance of 750 lx or greater in Japan. Consequently, this cannot be considered a lighting environment suited to each worker. Furthermore, it is also believed that desired illuminance differs by race and culture.

For all these reasons, we have been researching into an intelligent lighting system in order to provide individual illuminance environments in our laboratory[4], [5]. An intelligent lighting system provides each user's desired illuminance, and also gives energy saving.

The effectiveness of the Intelligent Lighting System has already been recognized, and demonstration experiment thereof is being conducted in multiple actual offices in the Tokyo area. In those offices, the necessary illuminance is being successfully provided in the necessary locations, and a significant level of power has been saved. Now that these results from demonstration experiment have been recognized, next we are examining the introduction of the Intelligent Lighting System in large-scale office environments. However, as the introduction scale of the Intelligent Lighting System is expanded, a dramatic increase in the number of luminaires and illuminance sensors that configure the Intelligent Lighting System also occurs. Moreover, with the increase in the number of devices configuring the Intelligent Lighting System, an issue also arises in which the time required for a single implementation of lighting control and illuminance acquisition also increases.

Therefore, in order to resolve the problems associated with expanding the introduction scale of the Intelligent Lighting System, we propose a technique to reduce the number of times that lighting control is implemented until illuminance convergence is achieved. By reducing the number of times that lighting control is implemented, even if the time required for a single implementation of lighting control and illuminance acquisition increases, the illuminance required by employees in an office (the target illuminance) can be provided at a speed equivalent to or faster than that of the conventional technique.

2. Intelligent Lighting System

2.1 Construction of Intelligent Lighting System

An intelligent lighting system realizes an illuminance level desired by the user while minimizing energy consumption by changing the luminous intensity of lightings. The intelligent lighting system, as indicated in FIG. 1, is composed of lights equipped with microprocessors, portable illuminance sensors, and electrical power meter, with each element connected via a network. Control devices installed in the lightings continuously vary the luminance using an optimization method on the basis of the illuminance information and the power consumption information. It is thereby possible to

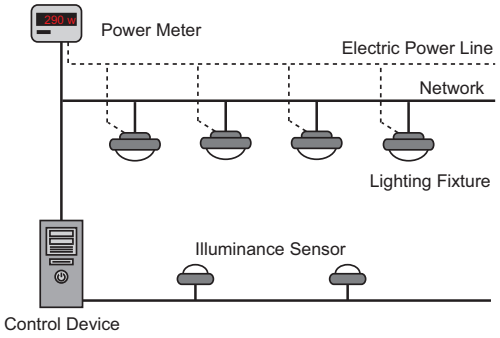


Figure 1: Configuration of Intelligent Lighting System

achieve the illuminance desired by a user with low power consumption.

2.2 Control Algorithm of Intelligent Lighting System

In Intelligent Lighting System, the algorithm where Simulated Annealing (SA) is improved for lighting control (Adaptive Neighborhood Algorithm using Regression Coefficient: ANA/RC) is used to control luminance intensity for each lighting fixture[4], [6].

It is possible with ANA/RC to provide the target illuminance with minimum power consumption by making luminance intensity for lighting fixtures the design variable and by using the difference between the current illuminance and target illuminance as well as power consumption as objective functions. Furthermore, by learning the influence of each lighting fixture on each illuminance sensor using the regression analysis and by changing the luminance intensity depending on the results, it is possible to promptly change to the optimal luminance intensity. This algorithm is effective to solve the problem which the objective function is near monomodal function and changes in real time. The objective function is indicated in the Equation (1).

$$f_i = P + w \times \sum_{j=1}^{n} (I_j - I_j^*)^2 \quad (1)$$

i : lightng ID, j : illuminance sensor ID
P : power consumption [W], w : weight
n : number of illuminance sensors
I : current illuminance [lx], I^* : target illuminance [lx]

As indicate in the Equation (1), the objective function f consists of power consumption and illuminance constraint. Also, changing weighting factor w enables changes in the order of priority for electrical energy and illuminance constraint. The illuminance constraint brings current illuminance to target illuminance or greater, as indicated by formula.

2.3 Issues Associated with Expanding the Intelligent Lighting System Introduction Scale

The Intelligent Lighting System that is currently used operates in an office environment with about 50 lights and 40 illuminance sensors. In an office environment of this size, target illuminance is realized for each office employee and illuminance convergence is implemented by repeating optimization 20 to 50 times with optimization lasting two seconds per time in the Intelligent Lighting System currently being utilized. However, when the Intelligent Lighting System is introduced in a large-scale office environment, the number of luminaires and illuminance sensors that configure the Intelligent Lighting System increase dramatically. Moreover, as the number of devices that configure the Intelligent Lighting System increases, the time required for a single implementation of lighting control and illuminance acquisition also increases. As will be discussed in section II-B, in a large-scale office environment that we envision having 1,000 lights and 600 illuminance sensors, it is understood that the time required for lighting control and illuminance acquisition increases to about seven seconds. With the control of the Intelligent Lighting System now being used, lighting control and illuminance acquisition are performed twice per a single time of optimization, and thus, if the time required for a single implementation of lighting control and illuminance acquisition is increased, the result will be a significant increase in the time required to reach illuminance convergence, which is problematic.

3. Reducing the Number of Times Lighting Control is used until Illuminance Convergence is reached in an Environment with No Daylight

3.1 Lighting Control Algorithm to Realize Personal Illuminance in an Environment with No Daylight

A technique to reduce the number of times that lighting control is implemented until illuminance convergence is reached in an environment with no daylight has been proposed. With the proposed technique, power information and illuminance information are estimated, and the estimated values are used to repeat optimization, and through this, the system searches for the optimum lighting pattern without acquiring information from the power meter or illuminance sensors. Moreover, by reflecting the lighting pattern that was searched for by computer in the lighting of the actual environment, the target illuminance for each office employee is realized with lighting control implemented a single time. With the Intelligent Lighting System currently being used, illuminance sensors are placed on the top of the desks of each office employee, and the illuminance on the desktop

surface when lighting is changed is acquired by the illuminance sensors to search for the optimum lighting pattern. In contrast, a benefit of the proposed technique is that illuminance information is estimated by computer, so there is no need to install illuminance sensors.

3.2 Formulation of the Objective Function

With the proposed technique, the objective function is evaluated, and optimization is repeated by formulating the power consumption and the constraint conditions with respect to illuminance using the luminance of each light.

Because the luminance of the lights and power consumption are in a relationship that can be approximated with a linear equation, power consumption can be expressed as in Equation (2).

$$P = \sum_{i=1}^{n} P_i \qquad (2)$$
$$P_i = f(L_i) = \alpha L_i + \beta$$

P : power consumption [W], i : lighting ID
n : number of lights
P_i : power consumption [W] by light i
L_i : luminance [cd] by light i
α : coefficient [W/cd], β : constant [W]

The coefficient α and the constant term β in Equation (2) are values that are inherent to each lighting type. Thus, preliminary tests are necessary in order to determine the main coefficient α and the constant term β for each type of light that is used. Furthermore, power consumption can be expressed by the luminance of the lights by using Equation (2).

The constraint condition with respect for illuminance can be determined from the difference in the current illuminance and the target illuminance. The target illuminance is a constant number that is set by each office employee, and thus in order to formulate the constraint condition with regard to illuminance, the current illuminance must be calculated from the luminance of the lights.

Here, a proportional relationship exists in the relationship between the luminance of each light and the illuminance obtained from the illuminance sensors, and that relationship can be expressed as in Equation (3). In addition, as long as there are no changes to the lighting environment, the R of Equation (3) can be considered to be a constant number. Hereinafter, this R is referred to as the illuminance/luminance influence factor.

$$I_j = \sum_{i=1}^{N} (R_{ij} \times L_i) \qquad (3)$$

i : lighting ID, j : illuminance sensor ID
N : number of lights
I_j : illuminance [lx] at the spot of the illuminance sensor
L_i : luminance [cd] of the light i
R : illuminance/luminance influence factor [lx/cd]

By using Equation (3), if the illuminance/luminance influence factor R is known, the current illuminance on the surface of the desktops of each office employee can be calculated from the luminance of each light, and the constraint condition with respect to illuminance can be expressed by the luminance of the lights.

By formulating the power consumption and the constraint conditions with respect to illuminance using the luminance of the lights in this manner, the objective function can be evaluated by computer without acquiring information from a power meter or illuminance sensors, and optimization can be repeated.

3.3 Method for Measuring the Illuminance/Luminance Influence Factor

Typical offices have a large number of fixed seats, and movement by office employees is rare. Therefore, in this research, the seating locations of office employees are assumed to be fixed. As discussed in section 3.2, if there are no changes in the lighting environment, the illuminance/luminance influence factor R can be considered to be a constant number. Therefore, with this technique, by positioning illuminance sensors on the surfaces of the desktops of each office employee in the experiment environment and then turning one light at a time on and off, the illuminance/luminance influence factor R can be measured based on the amount of illuminance change measured by the illuminance sensors and on the amount of luminance change of the lights that occurs at that time. By measuring the illuminance/luminance influence factor R in advance while the position of the illuminance sensors is known, the illuminance on the surface of the desktops of each office employee in an actual working environment can be estimated without taking into consideration various parameters such as light flux, the maintenance rate, the light distribution curve, and reflections from walls in a room.

3.4 Control Flow for the Proposed Technique in an Environment with No Daylight

The flow of control for the proposed technique in an environment with no daylight is shown below.

1) Set the target illuminance for each office employee.
2) Repeat optimization by computer and search for the optimum lighting pattern to realize the target illuminance for each office employee.
3) Reflect the lighting pattern that has been found by the search in the lighting of the actual environment.

With item 2), optimization is repeatedly implemented by computer using ANA/RC, an adaptive neighborhood algorithm that was discussed previously, to search for the optimum lighting pattern. By searching for the optimum

lighting pattern by computer and then reflecting that pattern in the lighting of the actual environment in this manner, the target illuminance level for each office employee can be realized with lighting control implemented a single time.

Note that with the proposed technique, the time required to search for the optimum solution by computer depends on the performance of the computer. When a search is performed for the optimum solution using a typical PC with a clock frequency of about 2.6 GHz, the time required to repeat optimization 100 times has been confirmed to be about 50 ms. In this study, in order to simplify comparisons with the conventional technique, the time required for one lighting control cycle was set to two seconds, which is the same as the conventional technique.

4. Effectiveness Verification Experiment in an Environment with No Daylight

4.1 Experiment Overview and the Experiment Environment

Illuminance convergence experiment was performed using an Intelligent Lighting System with the proposed technique and a system that is currently being used. Moreover, the illuminance histories and number of times lighting control was used were compared to verify the effectiveness of the proposed technique. Note that with the proposed technique, there is no need to install illuminance sensors; however, in order to confirm the feasibility of the target illuminations for each office employee, in this test, illuminance sensors were installed.

The experiment environment is shown in FIG. 2. In this test, 15 lights and 3 illuminance sensors A, B, and C were used, and the target illuminance for each illuminance sensor was set to 450, 600, and 750 lx respectively. Note that the vertical distance between the floor and the lights was 2.6 m, and the vertical distance between the floor and the desktop surface was 0.7 m. Because the illuminance sensors were positioned on the top surface of the desks, the vertical distance between the lights and the illuminance sensors was 1.9 m. An environment with no daylight was created by blocking the window in FIG. 2 with a movable wall surface, and the experiment was implemented therein.

4.2 Illuminance Convergence Experiment Results in an Environment with No Daylight

Using the proposed technique and the conventional technique, illuminance convergence experiment was conducted in an environment with no daylight. The illuminance history of the Intelligent Lighting System for the proposed technique is shown in FIG. 3, and the illuminance history with the conventional technique is shown in FIG. 4. The horizontal axis represents time, and the vertical axis represents illuminance. Note that the plot points in FIGS. 3 and 4 represent the points in time when lighting control was actually implemented.

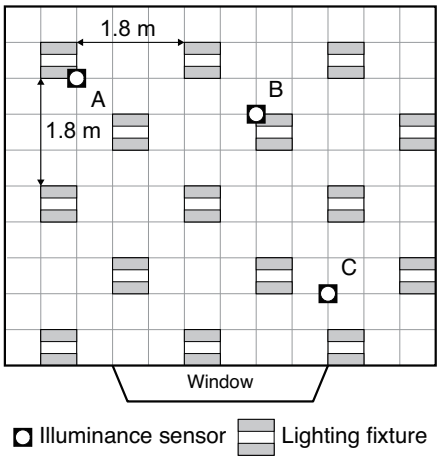


Figure 2: Experiment environment

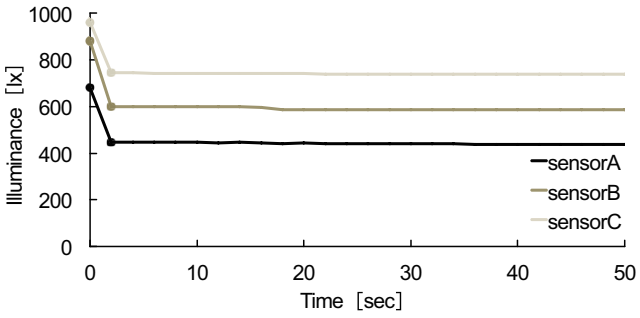


Figure 3: Illuminance history(proposal method)

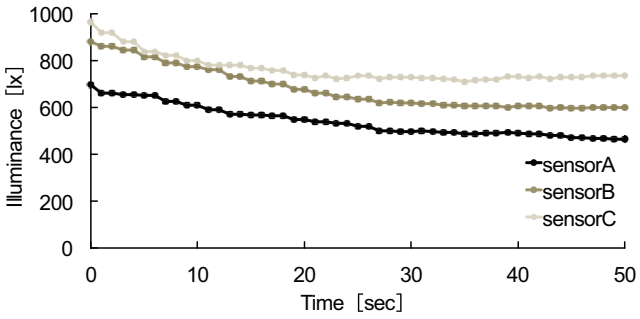


Figure 4: Illuminance history(conventional method)

From FIGS. 3 and 4, we can see that, when compared to the illuminance convergence with the conventional technique (in which lighting control was implemented about 30 times), with the proposed technique, illuminance convergence was achieved with lighting control implemented only one time. Based on this result, we confirmed that by using the proposed technique in an environment with no daylight, the target illuminance level for each office employee can be realized

without installing illuminance sensors on the tops of desks. Furthermore, because the proposed technique achieved illuminance convergence with lighting control implemented only one time, target illuminance levels for each office employee can be realized at a convergence speed equal to, or faster than, the conventional technique even if the introduction scale of the Intelligent Lighting System is expanded and the time required for lighting control is increased.

5. Reducing the Number of Times Lighting Control is required Until Illuminance Convergence is Reached in an Environment with Daylight

5.1 Lighting Control Algorithm to Realize Personal Illuminance in an Environment with Daylight

If the aforementioned technique is used in an environment with daylight, the impact of the daylight cannot be taken into consideration, and personal illuminance for each office employee cannot be realized. Thus, a technique has been proposed to realize personal illuminance in an environment with daylight and reduce the number of times that lighting control is implemented until illuminance convergence is reached. This is done by installing illuminance sensors on the top surfaces of the desks of each office employee to acquire the illuminance on the top surface of each desk.

5.2 Estimating Daylight Illuminance

The illuminance acquired by illuminance sensors in an environment with daylight can be expressed by Equation (4).

$$I_i = I_i^{(L)} + I_i^{(E)} \tag{4}$$

i : illuminance sensor ID
I : illuminance acquired from the illumiannce sensor
$I^{(L)}$: illuminance from the lights
$I^{(E)}$: illuminance from daylight

Moreover, from Equation (4), the daylight illuminance at the location of an illuminance sensor can be expressed by Equation (5).

$$I_i^{(E)} = I_i - I_i^{(L)} \tag{5}$$

i : illuminance sensor ID
I : illuminance acquired from the illumiannce sensor
$I^{(L)}$: illuminance from the lights
$I^{(E)}$: illuminance from daylight

Here, the illuminance $I^{(L)}$ for the illuminance of the lights on the spot where the illuminance sensor is located can be calculated using Equation (3), and thus the illuminance $I^{(E)}$ for the illuminance from daylight on the spot where the illuminance sensor is located can be estimated from Equation (5).

5.3 Flow of Control for the Proposed Technique in an Environment with Daylight

In an environment with daylight, a value that is calculated using Equation (6) is used as the target illuminance used in optimization by computer.

$$I_i^{**} = I_i^{*} - I_i^{(E)} \tag{6}$$

i : illuminance sensor ID
I^{**} : target illuminance used by a computer
I^{*} : target illuminance set by an office employee
$I^{(E)}$: illuminance from daylight

Because the daylight illuminance $I^{(E)}$ on the spot where the illuminance sensor is located can be estimated using Equation (5), the target illuminance used by the computer can be calculated from Equation (6). Moreover, a lighting pattern that considers the impact of daylight can be sought for by repeating optimization such that the target illuminance used by the computer is realized.

The flow of control with the proposed technique in an environment with daylight is shown below.

1) Set the target illuminance for each office employee.
2) Acquire the illuminance on the desktop surface of each employee.
3) Compute the target illuminance for use by a computer from Equation (6).
4) Repeat optimization by computer, consider the impact of daylight, and search for an optimum light pattern that realizes the target illuminance for each office employee.
5) Reflect the lighting pattern found from the search in the lighting of the actual environment.

Note that constant changing of daylight is conceivable, and thus, changes to daylight are supported by repeatedly implementing items 2) through 5).

6. Effectiveness Verification Experiment in an Environment with Daylight

6.1 Experiment Overview and the Experiment Environment

Illuminance convergence experiment was performed using the proposed technique in an environment with daylight to verify whether or not the proposed technique can support changes in daylight.

In the experiment, the movable wall surface that was used to block the window in section 4.1 of FIG. 2 was removed to open the window area and create an environment with daylight. Note that experiment was then performed with all other conditions being the same as those in section 4.1.

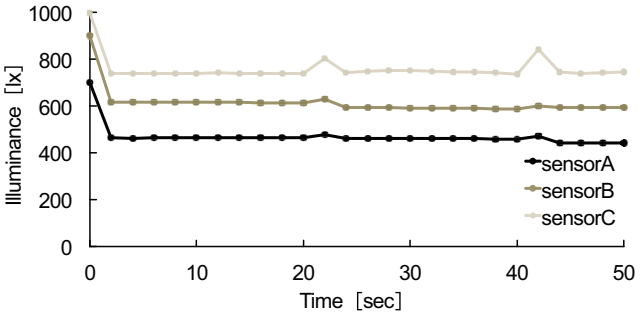


Figure 5: Illuminance history(proposal method)

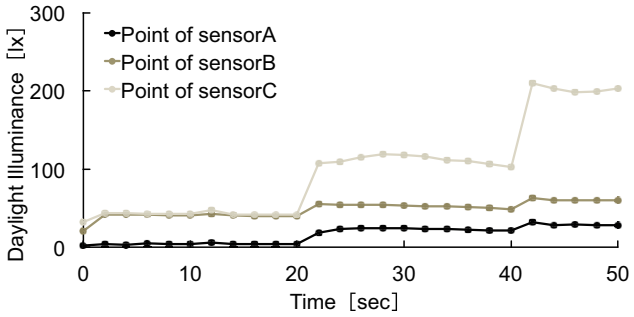


Figure 6: Daylight illuminance history

Moreover, experiment was also conducted by changing the angles of the blinds installed on the window surface both 20 seconds and 40 seconds after the experiment began so that the daylight changed significantly.

6.2 Illuminance Convergence Experiment Results in an Environment with Daylight

The illuminance history for the proposed technique in an environment with daylight is shown in FIG. 5, and the estimated daylight illuminance history at the time of experiment is shown in FIG. 6. The horizontal axis represents time, and the vertical axis represents illuminance. Note that the plot points in FIGS. 5 and 6 represent the points in time at which lighting control was actually implemented.

The values for daylight illuminance in FIG. 6 are values that were found using Equation (5). From this figure, we can see that the estimated daylight illuminance varies significantly at 20 seconds and 40 seconds after the illuminance convergence experiment was started. Moreover, we can also see that with the changes in daylight illuminance, the acquired illuminance similarly increased temporarily at 10 seconds and 20 seconds after starting the illuminance convergence experiment of FIG. 5. However, we also found that after the acquired illuminance of the illuminance sensors increased, convergence to the target illuminance occurred once again with the implementation of the next lighting control.

Based on the above results, we confirmed that the proposed technique realizes illuminance convergence with respect to changes in daylight when lighting control is implemented only two times.

Note that, because the proposed technique assumes operation in a large-scale office environment, consideration must be given to the time required for lighting control and acquiring illuminance in a large-scale environment. From preliminary experiment results, we confirmed that about seven seconds were required for a single implementation of lighting control and illuminance acquisition for a case that assumes the large-scale office environment to be an environment with 1,000 lights and 600 illuminance sensors. We also confirmed that, even in an environment with daylight, with the proposed technique, illuminance convergence is achieved with lighting control implemented two times after a change in daylight occurs. Therefore, even if the proposed technique is used in a large-scale environment, such as one having 1,000 lights and 600 illuminance sensors, illuminance convergence can be realized at a convergence speed equivalent to or faster than that of conventional techniques.

7. Conclusions

In this study, we proposed a technique to reduce the number of times that lighting control is implemented by reflecting an optimum lighting pattern that was found through a computer search. When illuminance convergence tests were conducted using the proposed technique in an environment with no daylight, we confirmed that illuminance convergence can be realized by implementing lighting control one time and without having to install illuminance sensors on the tops of the desks of each office employee.

Meanwhile, when illuminance convergence tests were conducted using the proposed technique in an environment having daylight, we confirmed that illuminance convergence can be realized by implementing lighting control two times with respect to a change in the daylight.

Based on the above results, for cases in which the illuminance/luminance influence factor R can be measured in advance, we confirmed that the proposed technique is effective at reducing the number of times that lighting control must be implemented until illuminance convergence is reached. By using the proposed technique, we believe that the target illuminance for each office employee can be realized in a convergence time that is equivalent to or faster than the conventional technique even if the introduction scale of the Intelligent Lighting System is expanded.

References

[1] Olli Seppanen, William J.Fisk, “A Model to Estimate the Cost-Effectiveness of Improving Office Work through Indoor Environmental Control", Proceedings of ASHRAE, (2005).

[2] M.J.Mendell and G.A.Heath, “Do indoor pollutants and thermal conditions in schools influence student performance? A critical review of the literature", Indoor Air, **15**[1], 27-52 (2005).
[3] Peter R.Boyce, Neil H.Eklund and S.Noel Simpson, “Individual Lighting Control: Task Performance, Mood, and Illuminance", Journal of the Illuminating Engineering Society, 131-142 (2000).
[4] M.Miki, T.Hiroyasu and K.Imazato, “Proposal for an intelligent lighting system, and verification of control method effectiveness", Proc. IEEE CIS, **1**, 520-525 (2004).
[5] M.Miki, K.Imazato and M.Yonezawa, “Intelligent lighting control using correlation coefficient between luminance anc illuminance", Proc.IASTED Intelligent Systems and Control, **497**[078], 31-36 (2005).
[6] S.Tanaka, M.Miki, T.Hiroyasu, M.Yoshikata, “An Evolutional Optimization Algorithm to Provide Individual Illuminance in Workplaces", Proc IEEE Int Conf Syst Man Cybern, **2**, 941-947 (2009).

An Automatic Speech Recognition for the Filipino Language using the HTK System

John Lorenzo Bautista, and Yoon-Joong Kim
Department of Computer Engineering, Hanbat National University, Daejeon, South Korea

Abstract - *This paper presents the development of a Filipino speech recognition using the HTK System tools. The system was trained from a subset of the Filipino Speech Corpus developed by the DSP Laboratory of the University of the Philippines-Diliman. The speech corpus was both used in training and testing the system by estimating the parameters for phonetic hmm-based (Hidden-Markov Model) acoustic models. Experiments on different mixture-weights were incorporated in the study. The phoneme-level word-based recognition of a 5-state HMM resulted an average accuracy rate of 80.13 for a single-Gaussian mixture model, 81.13 after implementing a phoneme-alignment, and 87.29 for the increased Gaussian-mixture weight model. The highest accuracy rate of 88.70% was obtained from a 5-state model with 6 Gaussian mixtures.*

Keywords: Speech Recognition; Filipino Language; Acoustic Model; Hidden Markov Model; HTK System;

1 Introduction

Speech is the most effective means of natural communication between humans. This is one of the very first skills humans acquire by learning through active interaction with the environment by mimicking sounds and voices. During the past years, computer scientists and engineers have been eager to incorporate this skill through intelligent machineries to provide a better and effective means of communication between humans and machine.

Developing an Automatic Speech Recognition system (ASR) fills the gap between a more natural means of communication between humans and intelligent systems. ASR systems were designed to recognize speech data and process them into text via natural language processing. These systems have been effectively integrated to the English language, however, only a few study related on the Filipino Language were realized[1]. A cross-training approach was one mean of implementing an appropriate language adaptation for the Filipino language using languages such as English and Indonesian [1][2]. Other studies use a Phonetically-balanced wordlist to train phonetic models for the Filipino Language [3].

In 2002, a Filipino Speech Corpus (FSC) was created by the Digital Signal Processing Laboratory of the University of the Philippines-Diliman [4]. This allowed multiple studies to be conducted relating to the speech recognition of the Filipino Language [5].

In this study, a Hidden-Markov Model (HMM) based ASR system will be used to train and recognize Filipino words using the HTK system[20] by training and testing for a phoneme-level speech recognition system using a subset of the Filipino Speech Corpus

This paper is organized as follows: Section 2 describes the Filipino language as well as the phoneme set used in this paper. Section 3 provides an introduction to the Filipino Speech Corpus (FSC) by the DSP Laboratory of the University of the Philippines. Section 4 shows the Data preparation done for this paper, while Section 5 provides information on the ASR system and the methodology for training data using the HTK System. Section 6 describes the results obtained from the experiments, which is discussed on Section 7 with the conclusions and future works that will be associated with this study.

2 The Filipino Language

The Filipino language is the lingua-franca of the Philippines and is largely used by around 22 million native speakers [7]. The Filipino language is prestige register of the Tagalog Language [8]. Tagalog varies more or less likely from places to places, with distinctive regional dialects noted among the provinces of Bataan, Bantangas, Bulacan, Manila, Tanay, and Tayabas, with the Bulacan dialect considered as the purest, while the Manila dialect is commonly treated as the standard of the Filipino pronunciation[9].

The Filipino language has evolved rapidly within the past few decades. Between 1930s and the mid 1970's a system of syllabication for the alphabet called the abakada was used to represent the native sounds of the language [10].

In a 2013, a new guideline for the language's orthography was published by the Komisyon sa Wikang Filipino (Commission for the Filipino Language)[11]. This guideline emphasized the use of a new set of Filipino alphabet and its phonetic counterpart:

Consonants

b, c, d, f, g, h, j, k, l, m, n, ng, p, q, r, s, t ,v, w, x, y, z

Vowels

a, e, i, o, u

This modern alphabet set was the basis for the phoneme set used in this study. The presentation was patterned on the phonetic code used in the TIMIT database [12], with 35 phonemes consisting of 14 vowels (including diphthongs) and 21 consonants. The phonetic table of the designed phoneme sets is shown as follows:

TABLE I
A PHONETIC CODE FOR THE FILIPINO LANGUAGE USED IN THIS STUDY MODELED AFTER TIMIT

Symbol	Example Word	Syllabication	Transcription
ah	b**A**boy	/ba-boy/	b **ah** b oy
ao	k**O**nti	/kon-ti/	k **ao** n t iy
aw	sig**AW**	/si-gaw/	s iy g **aw**
ax	t**A**nda	/tan-da/	t **ax** n d ah
ay	bant**AY**	/ban-tay/	b ax n t **ay**
b	**B**ihis	/bi-his/	**b** iy h ih s
ch	**CH**okoleyt	/cho-ko-leyt/	**ch** oh k oh l ey t
d	**D**ilaw	/di-law/	**d** iy l aw
eh	**E**spiritu	/es-pi-ri-tu/	**eh** s p iy r iy t uh
ey	r**EY**na	/rey-na/	r **ey** n ah
f	**F**iesta	/fi-yes-ta/	**f** iy y eh s tah
g	**G**ulay	/gu-lay/	**g** uh l ay
h	**H**ayop	/ha-yop/	**h** ah y ao p
ih	**I**ntay	/in-tay/	**ih** n t ay
iw	sis**IW**	/si-siw/	s iy s **iw**
iy	**I**bon	/i-bon/	**iy** b ao n
j	**J**eepney	/jip-ni/	**j** ih p n iy
k	**K**amay	/ka-may/	**k** ah m ay
l	**L**arawan	/la-ra-wan/	**l** ah r ah w ax n
m	**M**ata	/ma-ta/	**m** ah t ah
n	**N**ais	/na-is/	**n** ah ih s
ng	**NG**ayon	/nga-yon/	**ng** ah y ao n
oh	**O**ras	/o-ras/	**oh** r ax s
ow	epis**OW**d	/e-pi-sowd/	eh p iy s **ow** d
oy	tul**OY**	/tu-loy/	t uh l **oy**
p	**P**ula	/pu-la/	**p** uh l ah
r	**R**egalo	/re-ga-lo/	**r** eh g ah l oh
s	**S**ipa	/si-pa/	**s** iy p ah
t	**T**ao	/ta-o/	**t** ah oh
ts	**TS**inelas	/tsi-ne-las/	**ts** iy n eh l ax s
uh	**U**bo	/u-bo/	**uh** b oh
v	**V**igan	/vi-gan/	**v** iy g ax n
w	**W**ika	/wi-ka/	**w** iy k ah
y	**Y**aman	/ya-man/	**y** ah m ax n
z	**Z**igzag	/zig-zag/	**z** ih g **z** ax g

3 The Filipino Speech Corpus

The Filipino Speech Corpus (FSC) was created by the Digital Signal Processing Laboratory (DSP Lab) of the UP Electrical and Electronics Engineering Institute with the help of the UP Department of Linguistics in 2002 [2]. The corpus contains more than 100 hours of read Filipino text prompts and spontaneous Filipino speech. The corpus was recorded by 50 female and 50 male speakers from different parts of the Philippines using high-fidelity audio recording equipment. A variation to the FSC was created in 2009 that included syllabic utterances (bi-phones). This recent version is used in this study.

4 Data Preparation

In this section, we will discuss the data preparations done for the resources used in the speech recognition system including the development of a lexicon and pre-processing of the speech data.

4.1 Lexicon Development

An 18,000 unique word lexicon was developed by the Intelligent Information and Signal Processing Laboratory (IISPL) of Hanbat National University. This lexicon is labeled as "Hanbat Filipino Lexicon" was based on words gathered from several wordlists on the internet including entries from a medium-sized tri-lingual dictionary [14]. The lexicon consists of the word entry id, its corresponding written format, and the associated pronunciation as follows :

SIKAPING [SIKAPING] s iy k ah p ih ng
SIKAT-adj [SIKAT] s iy k ax t
SIKAT-nn [SIKAT] s ih k ax t
…

Homographs with different pronunciations were associated with an additional token for distinguishing the differences between the words while homophones with the same pronunciation were treated as one.

This lexicon was used to convert the text transcriptions from the FSC to its phonetic transcription patterned on the TIMIT phoneme set, as discussed in section 2.

A total of 1913 Out-of-Vocabulary (OOV) words were found from the FSC after processing with the Hanbat Lexicon (including 431 bi-phone utterances). These OOV words were manually transcribed and were included into to a separate lexicon.

4.2 Preprocessing the Speech Data

The FSC contain speech data recorded at 44.1 kHz for each recording session. These data were recorded continuously and are saved in one file (waveform format). This became inconvenient for the researchers to process the data especially with the unintelligible utterances and noises that were included in the data.

The data were segmented using the program Transcriber [15] based on the specified transcription files from FSC. The data were then pre-processed by removing the unintelligible utterances and noises, as well as reducing the silence parts of each sentence start and ending, programmatically. Mistakes in the transcriptions as well as incorrect timing were fixed in this phase.

Only a subset of the FSC were used in the training and testing of the system (prior to the 2009 variation) which includes 20 male and 20 female speakers from the total population of 50 male and 50 female speakers.

5 Building the Acoustic Model using HTK

The HTK system is based on Hidden Markov Models (HMM) [20]. HMMs are probabilistic models used for modeling stochastic sequence with underlying finite state structure. This model is widely used to model speech utterances in contemporary speech recognition development and is used by multiple ASR adaptations in languages such as Vietnamese, Indian, Polish, etc. [16][17][18].

In a phoneme based ASR, the HMM would represent the set of phonemes of the language, with each phoneme associated with a single HMM model [19]. A phoneme model (w) - denoted by HMM parameters (lambda) - is presented with a sequence of observations (sigma) to recognize a phoneme with the highest likelihood given:

$$w^* = \arg\max_w P(w|W)P(\sigma|\lambda_w) \quad (1)$$

where:

w = phoneme
W = total phoneme set
σ = observation values
λ_w = HMM model for phoneme w

In which the model λ_w visits different states x_t to generate an output o_t given time *t.* The doubly stochastic processes allows the state transitions and output generation depending on the actual state. The model is defined with a set of N reachable states $S = \{s_1, s_2, \ldots, s_N\}$, and a set of M possible output symbols $V = \{v_1, v_2, \ldots, v_M\}$. A set of three possible measures, *A, B,* and π is required to specify an HMM model λ as { *A, B,* π } where :

$A = \{a_{ij}\}$ as the state transition probability
$B = \{bj(k)\}$ as the observation probability
$\pi = \{\pi_i\}$ as the initial state distribution

5.1 Acoustic Processing

The speech data were encoded into Mel-frequency cepstral coefficient (MFCC) vectors. This vector values represents the speech signals' power spectrum (frequency content of the signal over time). Each vector values is the peak of the power spectrum called a formant frequency.

To compute for the format frequency, a series of signal processing are implemented to the speech signal. A Discrete Fourier Transform (DFT) is used to compute the frequency information in the time-domain, and is processed using a Fast-Fourier Transform (FFT) to contain real point values for the magnitude and phase information from the time domain.

Features from these values were extracted using Mel-cepstral filter banks, where in the central frequencies and bandwidth were scaled, and the FFT values were filtered based on the each Mel filter bank. These coefficients correlate to the formant frequencies of the speech signal. All these processes were obtain using the HTK tool HCopy which converts the speech file to its MFCC values.

The MFCC values computed includes the First Derivative and Second Derivative of the 13 MFCC values (12 MFCC + 1 zeroth Coefficient), totaling to 39 Vector Values.

The coding parameters include the use of 13 MFCC Features (12 MFC Coefficients + 1 Zeroth Coefficient/Energy), MFCC Derivatives, and MFCC Acceleration values. A Pre-emphasis coefficient of 0.97 is for the pre-processing, while a Hamming windowing is used in the experiments with a window size of 25ms per window. 26 Filterbank channels are used with 22 Cepstral Lifters.

5.2 Prototyping the HMM Models

The speech data were trained in a phoneme-based HMM model in 4, 5, 6, and 7 states for each phoneme, with the first and last states consisting of non-emitting start and end states. An HMM topology for a 6-state

An initial model (prototypes) for each n-state models were initialized for the training. The HTK tool *HCompV* was used to compute for the global means and variances for all Gaussians used in the HMM. These values are used for the prototype model that used in the embedded training of the model. The embedded re-estimation uses a Baum-Welch Re-estimation algorithm to compute for the new hmm parameters

using the HTK tool *HRest*. Each n-state models were estimated for 5 iterations before being aligning the phonemes in the data.

A total of 37 models are used consisting of the 35 phonetic models as well as 2 silence models (a silence model "SIL" for long silences, and a short pause model "SP" for shorted silences or breaks).

A phoneme alignment is done to fit the right phoneme utterances for word that have more than one pronunciation. Alignment provides a new set of speech transcription for better training of the phoneme models. This phoneme alignment is done using the HTK tool *HVite*. Once the phonemes were aligned, each n-state models were re-estimated for an additional 15 iterations.

5.3 Refining the Phonetic Models

A single phonetic model (monophone model) has some problems in terms of co-articulations of two consecutive phonemes. This co-articulation problem could be addressed by increasing the phoneme models from monophones to triphone models. However, the data used in this study does not contain enough training data to provide accurate triphone-based acoustic models. To improve the monophone models, a fine-tuning of the models are used to refine the system by increasing the Gaussian mixtures. Initially, for the first 20 iterations, only one Gaussian mixture is used.

To refine the monophone models, an additional 12 iterations for each n-state models are performed while increasing the Gaussian mixture weights by increments of 2 for every two consecutive iterations.

6 Testing and Results

The performance of the ASR's trained acoustic models were tested with a 500 word isolated speech data. These data were tested for each iterations of HMM parameters for iterations 6 to 32. Results of the testing are shown in Table 2, with the highest accuracy rate of **81.52%** for the 5-state model in a single Gaussian mixture.

TABLE II
MAXIMUM RECOGNITION RATE FOR EACH N-STATE MODEL FOR THE 20 RE-ESTIMATION OF THE HMM MODELS

States	Recognition Accuracy Rate (%)
4	77.22
5	81.52
6	77.07
7	62.50
Average	74.58

The results imply that the 5-state model provides the acoustic model representation for the phoneme-sets based on the training data used. The average recognition rate of the n-state models of this study is 74.58%.

TABLE III
ACCURACY RATE FOR TESTS BEFORE PHONEME ALIGNMENT (ITERATIONS 6 TO 10)

Iterations	Number of States			
	4	5	6	7
6	74.74	80.32	75.45	63.31
7	75.78	81.12	76.83	64.27
8	76.78	81.85	77.68	64.81
9	77.63	82.61	78.46	65.18
10	75.15	79.65	75.8	62.43
Average	76.15	80.13	76.03	61.91

Table 3 shows the results of the testing data based on the HMM parameters estimated for iterations 6 to 10, without phoneme alignment. Table 4 on the other hand shows the results based on the HMM parameters estimated for iterations 11 to 20 after phoneme alignment. Results for the experimental test shows that the highest recognition rate was achieved at the 9th iteration without phoneme alignment.

TABLE IV
ACCURACY RATE FOR TESTS AFTER PHONEME ALIGNMENT (ITERATIONS 11 TO 20)

Iterations	Number of States			
	4	5	6	7
11	75.55	79.70	75.89	62.00
12	76.03	80.03	75.89	61.91
13	76.37	80.26	76.11	61.85
14	76.63	80.51	76.22	61.89
15	76.81	80.70	76.43	62.07
16	76.96	81.05	76.58	62.15
17	77.06	81.15	76.71	62.33
18	77.14	81.27	76.89	62.41
19	77.24	81.43	77.05	62.51
20	77.22	81.52	77.07	62.50
Average	77.02	81.13	76.74	62.30

The increase in recognition rate based on the number of re-estimations for each n-state models for iterations

6 to 20 using a single Gaussian mixture weight are shown in Figure 2.

TABLE V
ACCURACY RATES FOR EACH N-STATE
MODELS AFTER INCREASING THE GAUSSIAN MIXTURE WEIGHTS

Mixtures	Re-estimations	Number of States			
		4	5	6	7
2	21	78.78	82.74	78.51	64.06
	22	81.79	84.80	80.60	68.03
4	23	84.78	85.80	81.37	70.20
	24	87.69	87.29	82.95	73.32
6	25	88.65	88.27	83.92	74.89
	26	88.62	88.70	83.82	75.23
8	27	88.48	88.51	84.09	75.38
	28	87.72	88.21	84.59	75.00
10	29	87.81	88.26	84.35	74.72
	30	88.28	88.20	84.23	74.43
12	31	88.86	88.44	84.02	74.51
	32	88.86	88.26	84.96	74.42
Average		86.69	87.29	83.12	72.85

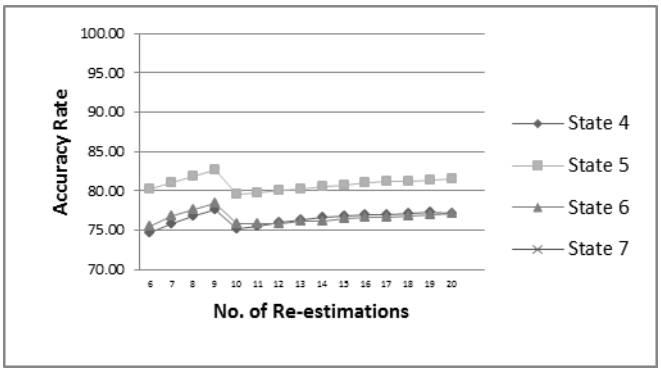


Figure 2. Accuracy Rate for each re-estimation with a single Gaussian mixture

After Increasing the Gaussian Mixture Weights for the n-state models, the testing data were again tested against the HMM parameters for the iterations 21 to 32. Accuracy Rates for the mixture systems are shown in Table 5.

Table 5 shows the results based on the increased Gaussian mixtures models. Results for the experimental test shows that the highest recognition rate was achieved at 6 Gaussian mixtures for a 5-state model with an accuracy rate of **88.70%.**

A graphical representation of the recognition rates for the mixture systems (iterations 21 to 32) are shown in Figure 3.

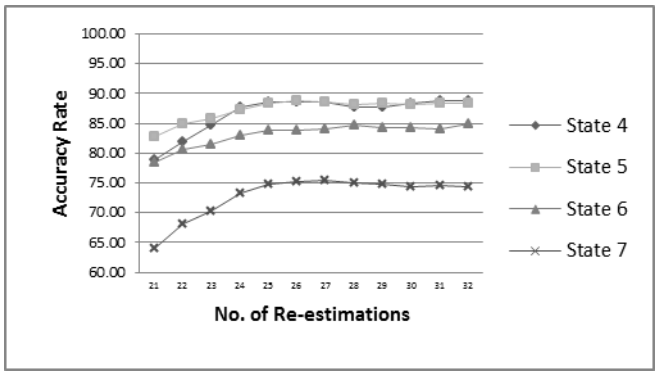


Figure 3. Accuracy Rate for each re-estimation with increased Gaussian Mixture Weights

7 Conclusion and Future Works

In this study, we have presented a speech recognition based on HTK for the Filipino Language using the Filipino Speech Corpus (FSC) by the University of the Philippines-Diliman. A subset of the FSC was taken to be trained using the HTK System tools with recordings by 20 male and 20 female speakers. The data were pre-processed and cleaned by removing non-intelligible sounds and noises, as well as reducing the silences in the wave file. The transcriptions were converted to its phonetic transcription using an 18,000 word lexicon developed by IISPL in Hanbat National University. The data were trained at 4, 5, 6, and 7-state HMM models. The experiments included a single-Gaussian mixture model training, incorporating phoneme-alignment, and increased Gaussian mixture weights.

Results from the experiment showed that a 5-state HMM model is the best n-state model based on the experiments, with an average accuracy rate of **80.13** for a single-Gaussian mixture model, **81.13** after implementing a phoneme-alignment, and **87.29** for the increased Gaussian-mixture weight model. While the highest accuracy rate of **88.70%** for a 5-state model with 6 Gaussian mixtures.

4 and 5 state models are recommended for applications for speech recognition based on the experiments performed in this study, with the lesser n-state model (4-state) to be more robust and faster compared to higher n-state models. A significant drop to the accuracy rates were also noticed for the 6 and 7-state models as compared to the 5-state model. A significant increase in the accuracy rate after increasing the Gaussian mixtures noted. However, increasing the mixture weights would also affect the robustness of the recognition system because of the increased number of computations performed compared to a single Gaussian mixture model.

For future research, the training data will be increased with the full FSC data which will be tested with a tri-phone based HMM model as well as incorporating a language model to improve the recognition capability of the system with

sentences, and increasing the developed Filipino lexicon into a larger word entry count. As of the moment, a Filipino language model is underworks and is expected to be included on the future research experiments.

8 Acknowledgment

The researchers would like to thank the Digital Signal Processing Laboratory (DSP Lab) of University of the Philippines-Diliman for providing the Filipino Speech Corpus (FSC).

9 References

[1] Sagum, R., Ensomo, R., Tan, E., Guevara, R.C., "Phoneme Alignment of Filipino Speech Corpus", *Conference on Convergent Technologies for Asia-Pacific Region, TENCON-2003*, October 2003

[2] Sakti S., Isotani, R., Kawai H., and Nakamura, S.," The Use of Indonesian Speech Corpora for Developing a Filipino Continuous Speech Recognition System*",* 2010

A. Fajardo, Y. Kim, "Development of Fillipino Phonetically-balanced Words and Test using Hidden Markov Model", *Proc. International Conference on Artificial Intelligence,* pp. 22-25, United States of America, July 2013

[3] Guevara R., Co, M., Espina, E., Garcia, I., Tan, E., Ensomo R., Sagum R., "Development of a Filipino Speech Corpus", *3rd National ECE Conference, Philippines,* November 2002

[4] Guevara R., et. al., "Filipino Databases and their Applications to Educational Institutions", *1st Philippine Conference Workshop on Mother Tongue-Based Multilingual Education*, February 2010

[5] Rara, K., Cristobal, E.R., de Leon, F., Zafra, G., Clarin, C., Guevara R., "Towards the Standardization of the Filipino Language: Focus on the Vowels of English Loan Words", *2009 International Symposium on Multimedia and Communication Technology (ISMAC 2009),*Bangkok, Thailand, January 2009

[6] http://wika.pbworks.com/w/page/8021671/Kasaysayan, "Ebolusyong ng Alpabetong Filipino", Retrieved 2014

[7] Nolasco, R., "Ang Filipino at Tagalog, Hindi Ganoong Kasimple" (*Filipino and Tagalog, it's not that simple)*, Komisyon sa Wikang Filipino, 2007

[8] Schachter, P., Otanes, Fe., "Tagalog Reference Grammar", *University of California Press,* (1972)

[9] http://wika.pbworks.com/w/page/8021671/Kasaysayan, "*Ebolusyong ng Alpabetong Filipino*", Retrieved 2012

[10] Almario, V., "Bagong Gabay sa Ortograpiya ng Wikang Filipino" (*New Guidelines for Orthography in the Filipino Language)*, Komisyon sa Wikang Filipino, 2013

[11] Garofolo, J., et. al., "TIMIT Acoustic-Phonetic Continuous Speech Corpus", *Linguistic Data Consortium*, Philadelphia, 1993

[12] Lazaro R., Policarpio L., Guevara R., "Incorporating Duration and Intonation Models in Filipino Speech Synthesis," *2009 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC 2009)*, October 2009

[13] S. Calderon, "Diccionario Ingles-Español-Tagalog", Manila, Philippines, 2012

[14] Barras C., et. al, "Transcriber: a Free Tool for Segmenting, Labeling, and Transcribing Speech", *First International Conference on Language Resources and Evaluation (LREC)*, May 1998

[15] Nguyen, H., et. al., "Automatic Speech Recognition for Vietnamese Using HTK System", *International Conference on Computing and Communication Technologies, Research, Innovation, and Vision for the Future (RIVF)*, Hanoi, November 2010

[16] Al-Qatab, B., et. al, "Arabic Speech Recognition using Hidden Markov Model Toolkit (HTK)", *International Symposium in Information Technology (ITSim)*, Kuala Lumpur, June 2010

[17] Zi'olko, B., et. al., "Application of HTK to the Polish Language", *International Conference on Audio, Language and Image Processing ICALIP 2008*, Shanghai, July 2008

[18] Rabiner, L. R. and Juang, B. H., "*Fundamentals of Speech Recognition"*, Englewood Cliffs, NJ, Prentice Hall, 1993

[19] S. Young, "Hidden Markov Model Toolkit: Design and Philosophy", *CUED/F-INENG/TR.152*, Cambridge University Engineering Department, September 2012

An Intelligent Lighting System using a Smartphone as an Illuminance Sensor

Sho Kuwajima[1], **Shohei Matsushita**[1], **Mitsunori Miki**[2], **Hisanori Ikegami**[1], **Yohei Azuma**[1], **and Hiroto Aida**[2]
[1]Graduate School of Science and Engineering, Doshisha University, Kyoto, Japan
[2]Department of Science and Engineering, Doshisha University, Kyoto, Japan

Abstract— *The authors have engaged in research and development of an Intelligent Lighting System which realize individual illuminance required by workers with minimum power consumption. As an Intelligent Lighting System use illuminance sensors for illuminance control, smartphones with a built-in illuminance sensor may be used for this purpose. This can reduce the initial cost for introducing an Intelligent Lighting System as well as realize easier maintenance. For that purpose, in this study, we examined the possibilities of an Intelligent Lighting System using built-in illuminance sensors in smartphones. Performance verification experiments concerning smartphone's built-in illuminance sensors showed that their resolutions are so low that their measurements differ from the actual illuminance values. Based on this result, we propose methods to realize individual illuminance control using the smartphone as an illuminance sensor. A verification experiment to confirm the effectiveness of the system incorporating the proposed method demonstrated that it can realize the individual illuminance control.*

Keywords: Lighting Control, Illuminance, Optimization, Smartphone, Position Estimation, Energy Conservation

1. Introduction

In recent years, as the improvement of energy efficiency has become a topic of broad discussion, there has been a drive toward energy saving in office buildings. Since lighting accounts for about 20% of the total power consumption in office buildings, improving the lighting environment can bring about a significant power saving and thus a great contribution to energy conservation [1], [2], [3]. Meanwhile, there have been many researches concerning intellectual productivity, creativity and comfort [4], [5]. Improving office lighting environments leads to the improvement of worker productivity. Above all, it has been reported that realizing brightness (illuminance) optimal for the work individually for each worker is effective for improving office environments [6].

Against this backdrop, the authors focused on lighting environments and proposed an Intelligent Lighting System which individually realizes illuminance levels required by individual workers while saving power consumption [7], [8]. The effectiveness of Intelligent Lighting System has already been recognized and verification experiments have been conducted at real offices in Tokyo and Fukuoka. These experiments verified that the system can individually realize illuminance levels preferred by workers while cutting down on energy consumption.

In Intelligent Lighting System, an illuminance sensor is set on each worker's desktop; based on the illuminance measurements by these sensors, the brightness (luminance) of the office lighting fixtures are controlled by an optimization method. These systems currently use expensive illuminance sensors manufactured to order, but it may be possible to replace them with smartphones now widely owned by people. This is expected to reduce the initial cost for introducing an Intelligent Lighting System while realizing easier maintenance. Hence, this study examines the possibilities of an Intelligent Lighting System using smartphones as illuminance sensors.

2. Intelligent Lighting System

2.1 Configuration of Intelligent Lighting System

An Intelligent Lighting System is a system to realize required illuminance at the position where an illuminance sensor is installed with minimum power consumption [7], [8]. As shown in Fig.1, it consists of lighting fixtures, control devices, illuminance sensors, a power meter and a network to connect them.

The control device accompanying each lighting fixture varies the luminance to an extent unnoticed by workers using an optimization method based on illuminance and power consumption data. By repeating this, the system realizes the illuminance required by each worker with saving power consumption.

2.2 Illumination Control Algorithm

An Intelligent Lighting System solves an optimization problem using an Adaptive Neighborhood Algorithm using Regression Coefficient (ANA/RC) based on Simulated Annealing (SA) [7], [9] in an autonomous distributed style, in which the luminance of each lighting fixture is the design parameter, the target illuminance of each lighting fixture

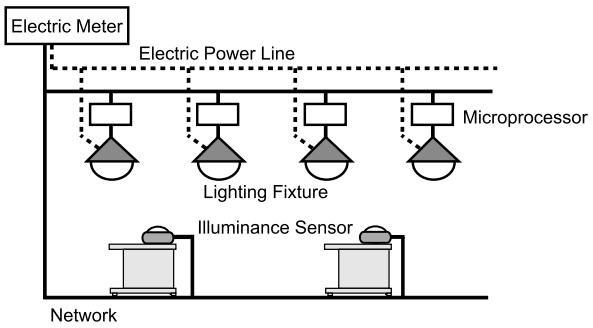


Fig. 1: The construction of an Intelligent Lighting System

is the constraint and the total power consumption of the lighting is the objective function. In short, it randomly varies the luminance of each lighting fixture in each search to an extent unnoticeable by workers [10] to search an optimum lighting pattern. The ANA/RC estimates the level of influence (referred to as "illuminance/luminance influence factor") on each illuminance sensor through regression analysis concerning the illuminance change and luminance change during search for a solution, and gives an orientation to the random change in luminance corresponding to the illuminance/luminance influence factor.

In this way, the system operates those lighting fixtures yielding strong influence on each illuminance sensor to realize the illuminance required by each worker at minimum power consumption, while operating the other lightings to minimize power consumption. Eq. (1) is the objective function used in this algorithm.

$$f_i = P + \omega \times \sum_{j=1}^{n} g_{ij} \quad (1)$$

$$g_{ij} = \begin{cases} 0 & (Ic_j - It_j) \geq 0 \\ R_{ij} \times (Ic_j - It_j)^2 & (Ic_j - It_j) < 0 \end{cases} \quad (2)$$

$$R_{ij} = \begin{cases} r_{ij} & r_{ij} \geq T \\ 0 & r_{ij} < T \end{cases}$$

i: Number of lightings, j: Number of sensors
P: power consumption[W], ω: weight[W/lx]
Ic: current illuminance[lx], It:target illuminance[lx]
r: illuminance/luminance influence factor（regression coefficient）, T:threshold value

The objective function was derived from amount of electric power P and illuminance constraint g_{ij} . Also, changing weighting factor w enables changes in the order of priority for electrical energy and illuminance constraint. The illuminance constraint is decided so that a difference between current illuminance and target illuminance within a threshold, as indicated by Eq. (2).

2.3 Illuminance Sensor in the Intelligent Lighting System

Illuminance sensors used in Intelligent Lighting Systems are broadly divided into wired types and wireless types. Wired illuminance sensors can be powered via PoE, but necessitate power cables under the floor. This is not practical in an environment where illuminance sensor positions may change. On the other hand, wireless illuminance sensors, requiring such considerations as battery replacement or charge but not cables, are a practical solution for offices having frequent sensor relocations due to layout changes. Moreover, the illuminance sensors currently used by Intelligent Lighting Systems are expensive order-made products, which has hindered the popularization of Intelligent Lighting Systems.

Meanwhile, smartphones and tablets (hereinafter mentioned as "smartphones") have become ubiquitous in recent years, and the illuminance sensors built into these smartphones may be used as illuminance sensors in Intelligent Lighting Systems. Hence, this study proposes a method to control Intelligent Lighting Systems using smartphones as illuminance sensors.

3. Built-in illuminance sensors in smartphones

3.1 Advantages of using smartphones

In recent years, smartphones such as Android products or iPhones have prevailed as sophisticated multi-function mobile phones. Since smartphones have a built-in illuminance sensor for screen brightness control, they may be utilized as illuminance sensors for Intelligent Lighting System. Not only that they may function as illuminance sensors, the use of commercial general purpose products will also increase the ease of maintenance while reducing initial system construction cost; they may also provide a touch-panel user interface.

However, since illuminance sensors built into smartphones are intended primarily for screen brightness control, whether they function properly in an Intelligent Lighting System needs to be checked through performance verification.

3.2 Performance verification experiment for built-in illuminance sensors

To verify the performance of illuminance sensors built into smartphones (hereinafter referred to as "built-in illuminance sensors"), their measurements were compared with the actual measurement values of an external illuminance meter. In the experiment, only one lighting fixture right above point A shown in Fig.2 was turned on. Then illuminance meters and smartphones (including tablets) were placed on 70 cm high desktops at points A to E. For the experiment, lighting fixtures with a light control in 256 steps were used.

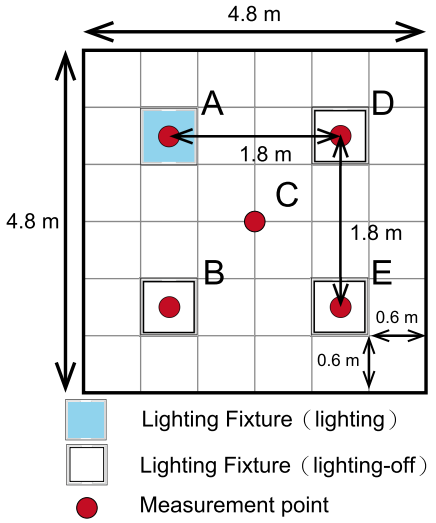


Fig. 2: Experimental environment for performance verification (top view)

At each of these steps, the brightness at each point was measured using a built-in illuminance sensor and an external illuminance meter. The illuminance meter used was ANA-F11 (Tokyo Koden: JIS C 1609-2 compliant), and the four smartphone models listed in the left column of Table 1.

Fig.3 compares the values from the built-in illuminance sensors of the four models and measurements by an external illuminance meter. Also, Table 1 shows the resolution of the built-in illuminance sensor as obtained from this experiment. In Fig.3, the horizontal axis shows the measurements from the external illuminance meter, while the vertical axis shows the values from each built-in illuminance sensor. Only at Point A, values up to 400 lx were plotted both for the vertical and horizontal axes, while values up to 200 lx were plotted for others. For the purpose of clarity, the figure shows the number of plots in summary. The data from point D is not shown because the result was similar to that at point B.

From Fig.3 and Table 1, the values from built-in illuminance sensors of ARROWS and XOOM are were relatively of high resolutions and a linearity was confirmed between them and the external illuminance meter measurements, but the values from the XOOM built-in illuminance sensor at position A (vertically and directly below the lighting fixture) were larger than those from the external illuminance meter. On the other hand, the values from AQUOS and INFOBAR showed greater gaps from external illuminance meter measurements, aside from low resolutions.

To sum up, the measurements from a built-in illuminance sensor of a smartphone significantly differ from the measurements from an external illuminance meter. For this reason, we will examine their effects on the control of an Intelligent Lighting System using built-in illuminance sensors of smartphones.

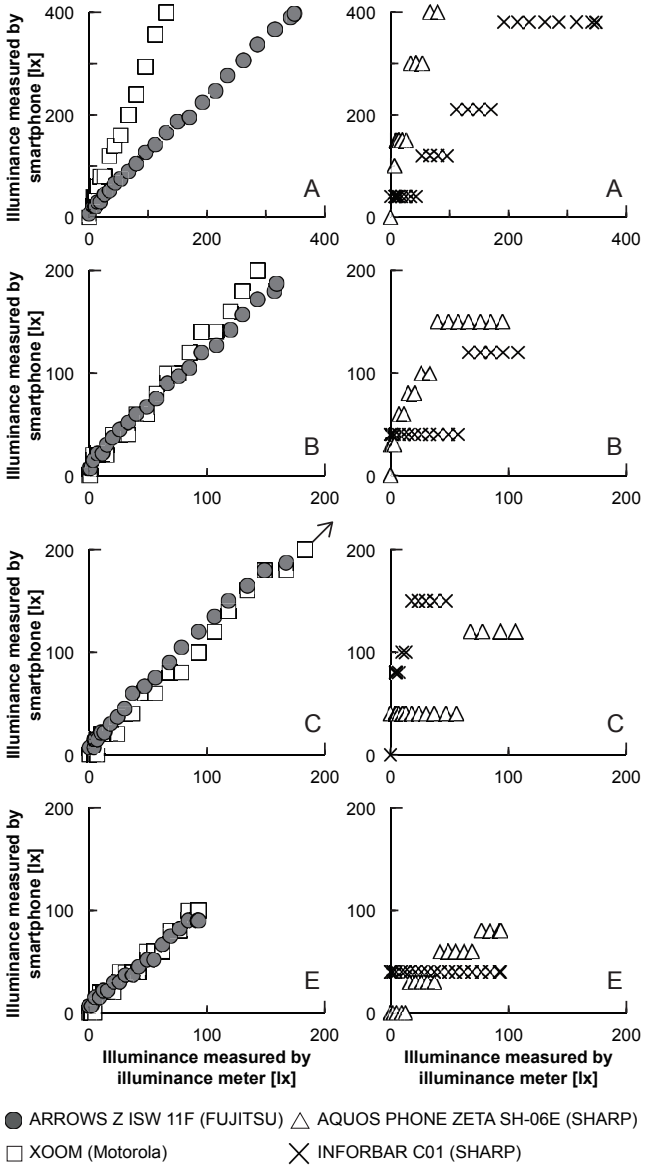


Fig. 3: The values from the illuminance sensor built in the smartphone

3.3 Considerations on the performance of built-in illuminance sensors

First, we will look at the resolutions of built-in illuminance sensors. It was demonstrated that the built-in illuminance sensors of the smartphones used in the experiment all had a resolution lower than that of the external illuminance meter. Meanwhile, as mentioned in section 2.2, an Intelligent Lighting System estimates the illuminance/luminance influence factor of each lighting fixture on an illuminance sensor from a regression analysis of the lighting's luminance change and the sensor's illuminance change, and use that estimate in an illuminance control algorithm. In the process, the luminance of the lighting fixture is varied by such a small extent

unnoticeable to workers that illuminance sensors of a low resolution may not be able to show the change in illuminance measurements in response to the microscopic change in luminance. Since the illuminance/luminance influence factor cannot be correctly estimated in such cases, convergence into an optimum lighting pattern is not easy. This gives rise to the need of a new method to estimate illuminance/luminance influence factor which can work properly with built-in illuminance sensors.

Next, let us think about the gap between built-in illuminance sensor measurements and the actual illuminance values. An Intelligent Lighting System sets the brightness required by each worker as a target illuminance value for each illuminance sensor, then controls the luminance of lighting fixtures so that the target will be met. However, when smartphone's built-in illuminance sensors are used, there occurs a gap between their measurements and the actual illuminance values, necessitating some idea to solve this gap.

These are considered to be the challenges for using smartphone's built-in illuminance sensors for an Intelligent Lighting System.

4. Individual illuminance control using built-in illuminance sensors of smartphones

4.1 Determining illuminance/luminance influence factors based on a position estimation method

Here we discuss a new method to estimate illuminance/luminance influence factors, which we mentioned as a challenge in section 3.3. This method determines an illuminance/luminance influence factor by estimating the approximate position of a smartphone. Note that the illuminance/luminance influence factor used in this method does not take into account the effects of light reflected by walls nor degradation of lighting fixtures. Since the system controls illuminance by varying the luminance of lighting fixtures, this study will utilize the change in the luminance of lighting fixtures for estimating smartphone positions.

From section 3.2, it is confirmed that, although smartphone's built-in illuminance sensors have low resolutions, the output values can be raised by intensifying light above a certain extent. Therefore, a binary search approach is applied to estimate a smartphone position. The concept is shown in Fig.4(1) to (4). The control flow for the position estimation is shown below.

1) From the initial status, divide all lighting fixtures in the room into two groups
2) For each of the divided groups, change the luminance of lighting fixtures uniformly by a degree unnoticeable to workers [10], to cause an illuminance change greater than the resolution of the built-in illuminance sensor
3) The built-in sensor on the smartphone measures the changes in illuminance values caused in step 2)
4) Choose the group showing a greater change in illuminance as measured in step 3), and divide it again into two groups
5) Return to step 2)

In case lighting fixtures are arranged in an odd number of rows as in the case of Fig.4(3), the central row is regarded as belonging to both groups for the purpose of control. In case a smartphone is positioned around the middle of the two groups, there may be only little difference between the values of change obtained from the built-in illuminance sensor. In that case, the adjacent ends of the two groups are combined into a group as shown in Fig.4(5) for the purpose of search.

Since the lighting fixtures significantly affecting the built-in illuminance sensor are basically the four around it, when the scope of search is reduced to the size of 2 lines $\times$ 2 rows, the search is deemed to reach an end. The search results are classified into patterns A to C shown in Fig.4(6). The smartphone is positioned around the center of four lighting fixtures in pattern A, around the middle of two lighting fixtures in pattern B and right below a lighting fixture in pattern C. Since pattern B and pattern C result in less than four lighting fixtures, also other nearby lighting fixtures are defined as "lighting fixtures close to the smartphone" as shown in Fig.4(7). A group resulting in 2 lines $\times$ 1 row is treated in the same way as pattern B.

Next, from the result of position estimation, the illuminance/luminance influence factor is determined. For this purpose, a preliminary experiment was conducted in which accurate illuminance/luminance influence factors were calculated for each of the patterns A to C shown in Fig.4(7) using an external illuminance meter, and the resulting values were associated with those lighting fixtures which were judged to be close to the smartphone according to the position estimation results. In this way, the lighting fixtures picked up through the position estimation process can be controlled by an algorithm described in section 2.2.

Table 1: The resolution of the illuminance sensor built in the smartphone

Smartphone's model	Resolution [lx]
ARROWS Z ISW 11F	7 - 8
XOOM	20 - 22
AQUOS PHONE ZETA SH-06E	50 - 500
INFORBAR C01	80 - 170

4.2 Solution to a difference between a built-in illuminance sensor measurement and the actual illuminance value

This section discusses how to solve the problem of a difference between built-in illuminance sensor measurements

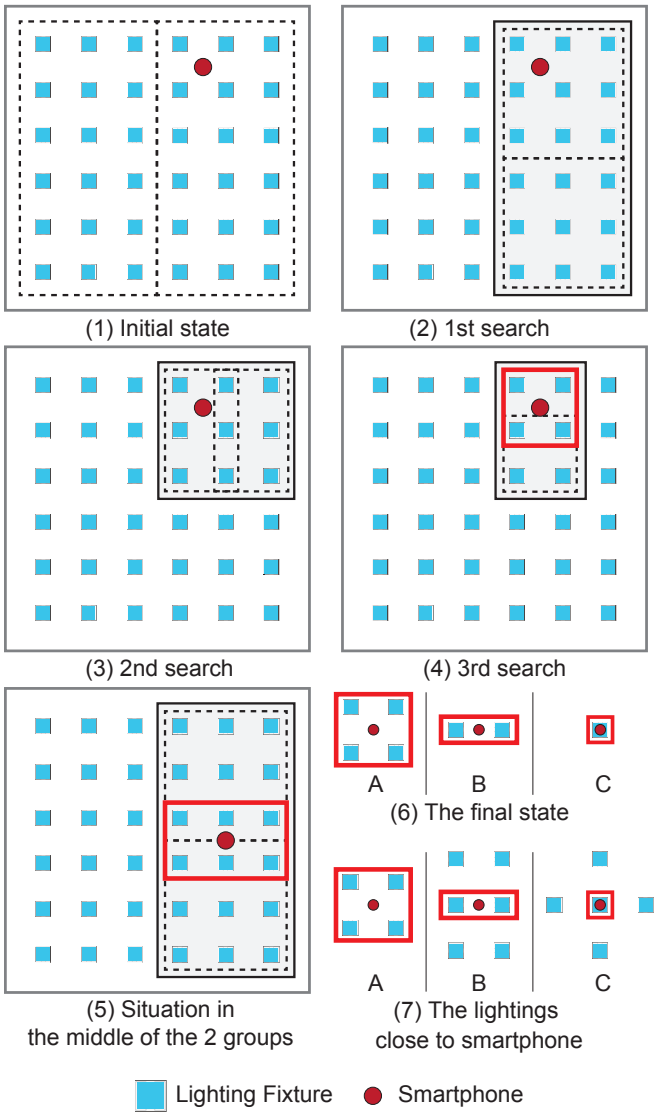


Fig. 4: The concept of proposal position estimation

and the actual illuminance values. As a first step, for the purpose of this study, we propose an approach to solve the problems stemming from this difference without correcting the built-in illuminance sensor measurements, provided that a single type of smartphones of a relatively high resolution are used.

In an Intelligent Lighting System, a target illuminance is set by the user (worker) designating a preferred brightness level as an illuminance value. However, designating an illuminance value is just one of the ways for an individual to set a preferred illuminance level: what a worker essentially requires is not illuminance as an absolute value. For this, a target illuminance may be configured in a relative approach, in which the user chooses whether to raise or lower illuminance from the current level. In this approach, each time the worker raises or lowers the illuminance setting

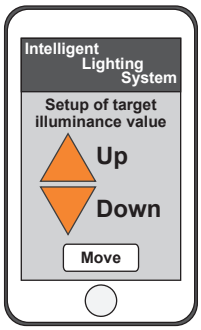


Fig. 5: The concept of operation screen for smartphones

in relative terms, the target illuminance value held inside the system is changed, and each lighting fixture controls its luminance to bring the built-in illuminance sensor measurement as close as possible to the target value. If the built-in illuminance sensor measurement is in a linear relation with the actual illuminance value, it is not necessary to show the sensor measurement to the worker: the worker can realize a preferred illuminance simply by increasing or lowering the target illuminance relative to the current setting. A sample user interface on a smartphone in this approach is shown in Fig.5.

5. Verification experiment

5.1 Experiment summary and conditions

A verification experiment was conducted for an Intelligent Lighting System using smartphones based on the proposed method. In the experiment, 3 smartphones and 9 lighting fixtures were set up as shown in Fig.6 in a room without natural light. Based on section 3.2, ARROWS Z ISW 11F was selected as a smartphone model for this purpose, the type with the highest-performance built-in illuminance sensor. As lighting fixtures, white LED lamps manufactured by Panasonic Corporation (controllable between 20% and 100%, maximum lighting luminance is 1040 cd). The lighting fixtures were installed at 1.8 m intervals, which is a distance commonly used in typical office environments.

In this experimental environment, as shown in Table 2, the target illuminance value preset on each smartphone was changed at the point of 650 seconds after the start of the experiment. The initial luminance setting of each lighting fixture was 100%. Under these conditions, each smartphone estimates lighting positions, determines illuminance/luminance influence factors and starts a search for an optimum lighting pattern.

5.2 Experiment result

Fig.7(a) shows the history of values obtained from the built-in illuminance sensors of the each smartphone. The chart shows that the three smartphones completed estimating the lighting positions and illuminance/luminance influence

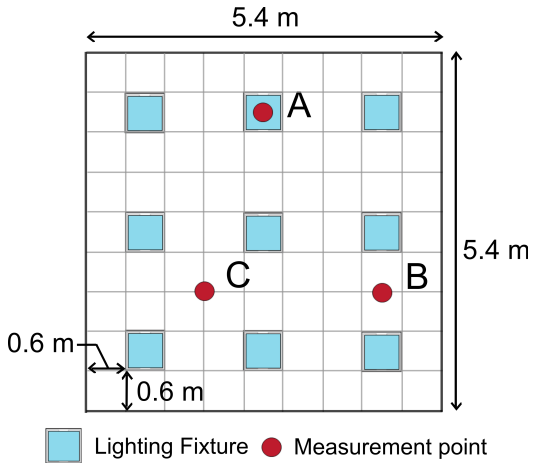


Fig. 6: Experimental environment

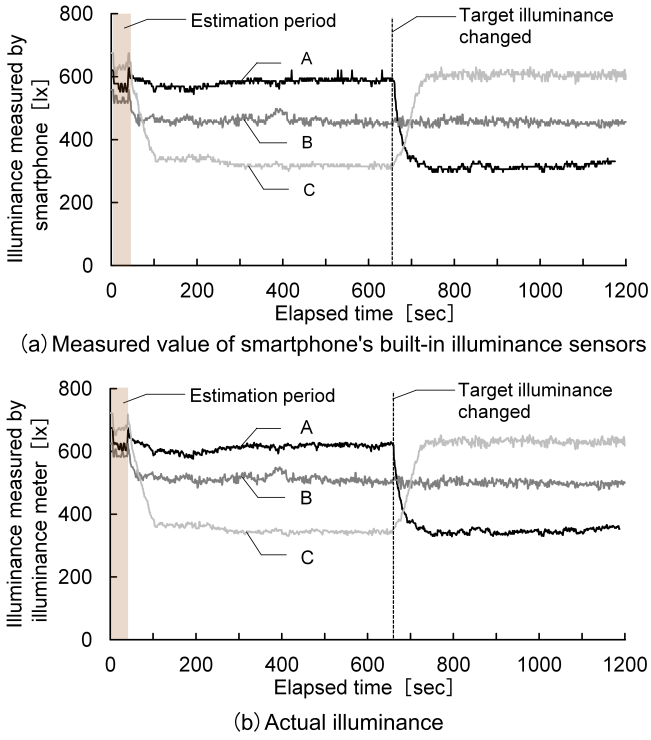


Fig. 7: The history of illuminance data

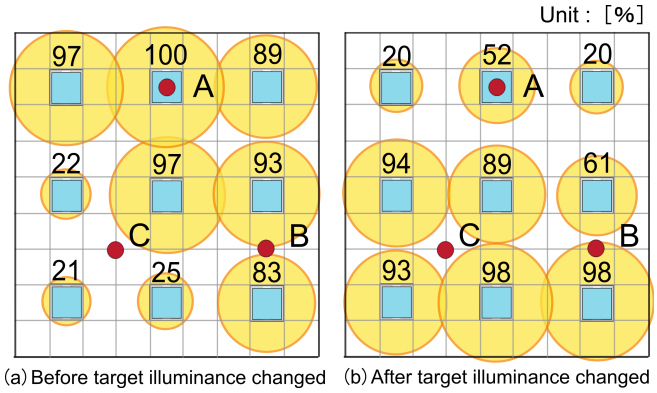


Fig. 8: The status of lightings

factors in about 50 seconds, and the illuminance sensor measurements converged to the target value in about the next 50 seconds.

Fig.8(a) shows the status of lighting at the point of 500 seconds from Fig.7(a). This figure shows the percentage of luminance of each lighting fixture to its maximum luminance, with the size of the circle representing the luminance percentage. The figure shows that the lights with a large influence on the built-in illuminance sensors light up strongly, while those inessential for realizing the illuminance targets set on the smartphones light up at a minimum luminance level. The fact that convergence to the target illuminance was realized in this lighting status confirms that the position estimation by smartphones was effective for controlling each light. Next, Fig.7(b) shows the history of actual illuminance measured by an illuminance meter at the same points. This chart shows that the built-in illuminance sensor measurements and the actual illuminance values were controlled at equivalent levels, confirming that using built-in illuminance sensors can also maintain actual illuminance at a constant level.

Furthermore, the target illuminance settings on smartphones A and C were changed at the point of 650 seconds after the experiment start. Fig.7(a) shows that the measurements by smartphones converged to the target illuminance levels in about 70 seconds from the target change, which confirms that using smartphones can also cope with changes to the target luminance.

Fig.8(b) shows the status of lighting at the point of 1,100 seconds from Fig.7(a). This figure shows the percentage of illuminance of each lighting fixture. Just like the lighting status before change shown in Fig.8(a), it shows that lighting fixtures with strong influence on the built-in illuminance sensors are lighted strongly. Furthermore, the lights which had been lit strongly before the change to the target illuminance are now lowered after the luminance targets were changed on smartphones. These results demonstrate that an Intelligent Lighting System using smartphones as illuminance sensors can respond to changes to target illuminance and realize an appropriate individual lighting control.

Table 2: Setup of the target illuminance values

	A [lx]	B [lx]	C [lx]
First target illuminance	600	450	300
Second target illuminance	300	450	600

6. Conclusions

By using the proposed method, smartphones of a single model can substitute for illuminance sensors used in an Intelligent Lighting System, provided that their built-in illuminance sensors have a high resolution. A verification experiment demonstrated that the actual illuminance can be maintained at a constant level even when smartphone's built-in illuminance sensors are used. Moreover, even though the built-in sensor measurements differ from actual illuminance values, the approach proposed in section 4.2 can realize illuminance levels preferred by the worker.

In future, we plan to study such topics as individual illuminance control in an environment with multiple smartphone models, or position estimation in a shorter time period.

References

[1] Francis Rubinstein, Michael Siminovitch and Rudolph Verderber, "Fifty percent energy saving with automatic lighting controls", IEEE Industry Applications Society, vol29, pp.768-773, 1993

[2] P.J.Littlefair, "Predicting lighting energy use under daylight linked lighting controls", Building Research and Information, vol.26, no.4, pp.208-220, 1998

[3] D.H.W.Li and J.C.Lam, "An investigation of daylighting performance and energy saving in a daylight corridor", Energy and Buildings, vol.35, no.4, pp.365-373, 2003

[4] Olli Seppanen, William J.Fisk, "A Model to Estimate the Cost-Effectiveness of Improving Office Work through Indoor Environmental Control", Proceedings of ASHRAE, 2005

[5] M.J.Mendell and G.A.Heath, "Do indoor pollutants and thermal conditions in schools influence student performance? A critical review of the literature", Indoor Air, vol.15, no.1, pp.27-52 ,2005

[6] P. R. Boyce, N. H. Eklund and S. N. Simpson, "Individual Lighting Control: Task Performance", Mood and Illuminance JOURNAL of the Illuminating Engineering Society, pp.131-142, 2000

[7] M. Miki, T. Hiroyasu and K. Imazato, "Proposal for an intelligent lighting system and verification of control method effectiveness", Proc IEEE CIS, pp.520-525, 2004

[8] M.Miki, K.Imazato and M.Yonezawa, "Intelligent lighting control using correlation coefficient between luminance anc illuminance", Proc.IASTED Intelligent Systems and Control, vol.497, no.078, pp.31-36, 2005

[9] S.Tanaka, M.Miki, T.Hiroyasu, M.Yoshikata, "An Evolutional Optimization Algorithm to Provide Individual Illuminance in Workplaces", Proc IEEE Int Conf Syst Man Cybern, vol.2, pp.941-947, 2009

[10] T.Shikakura, H.Morikawa and Y.Nakamura, "Research on the Perception of Lighting Fluctuation in a Luminous Offices Environment", Journal of the Illuminating Engineering Institute of Japan, vol.85, no.5, pp.346-351, 2001（in Japanese）

The Prediction and Classification of Family Type by using Power Consumption Model over Smart Grid

Jihyun Kim[1], **Howon Kim**[1]
[1]Department of Computer Engineering, Pusan National University,
Busan, Kyung Sang Province, Republic of Korea

Abstract - *In recent years, the introduction of smart grid which enables bidirectional communication between the supplier and the user is under way actively in electric power system field. If smart grid environment is set up, it would be possible to create new services or provide existing services efficiently. In this paper, we analyze the power consumption data and predict a family type of household by using machine learning techniques. For prediction, we construct a power consumption model of household and make a simulator based on the model. We find out an optimum machine learning algorithm and features for predicting a family type of household. Also, we show that our study can be utilized in the customized home services or marketing services of the companies.*

Keywords: Smart Grid, Power Consumption Model, Machine Learning

1 Introduction

Smart grid is a next generation grid which enables two-way communication between the customers and electric energy providers[1]. With an effective use of energy is receiving worldwide attention, the interest about smart grid is also increasing. In the United States, EPRI(Electric Power Research Institute) has suggested a novel power system through the intelligrid program since the early 2000s[2]. The investment in smart grid was accelerated as a part of economic stimulus due to the unprecedented global financial crisis in 2008. The United States government has invested about ten billion dollors in order to modernize the grid and transmission infrastructure through the Grid 2030 project[3]. In Europe, they prosecute the expansion of smart grid for invigoration the power transaction between member countries. With EU directives, about 80% households must install smart meter until 2020 and all households have to do upto 2022[4]. China is the second largest investing country for smart grid after the United States. They have carried forward smart grid project in an optimal load distribution and reinforcement of the power transmission system.

Like this, smart grid is a mainstream. And its environment will be constructed. Then it is possible to create novel services such as overuse alarm service of the electric power or the power usage prediction service by using data acquired from smart grid. Therefore extracting the meaningful information from the measured data is very important. It can be a driving force to smart gird service. In this paper, we analyze the electric power consumption data and utilize it to classify the family types. We propose the power consumption model and find that which feature and machine learning algorithms are proper for classification. Also, because of the difficulty in collecting power consumption data with family type, we developed the simulator based on the proposed model. In section 2, we look into related works and describe the power consumption model in section 3. We explain the result of experiment in section 4. In the last section, we make a conclusion.

2 Related Works

Many studies about collecting the meaningful data from power consumption data of a household are under way. In order to prevent a blackout by the unexpected electric power demand, Adrian Albert conducted a study on the relation between the human's daily schedule and the electric power usage[5]. He classified the patterns of the power use at the household by using k-Means clustering. Honglian Fei suggested the method for distinguishing the heatpump use by analyzing a power consumption data of a specific household[6]. He used the electric power data from 300,000 household and the sales records of heatpump sales company. For classification, he applied BSVM to data. Also, he presented the possibility that his study can be utilized for marketing of the heatpump sales companies. Asma Dachraoui predicted the demand of electric power based on MODL[7] by utilizing the data which is acquired from 6000 households who are living in Ireland during 500 days[8]. He focused on the increase of electric power demand at a specific time. And his study enabled that a power company can adjust an electric power price.

All studies which are described in this section have something in common. They all use unlabeled data. The unlabeled data means that there are no information about the data. However, we have to train a classifier using labeled data in order to classify family types. Therefore, we designed the power consumption model.

3 The Power Consumption Model

We designed the power consumption model of household. Based on the model, we can define family types and measure an electric energy consumption by developing a simulator. Our model consists of two sub-models. The former is a scheduling

model which determines when the appliances will be used depending on a basic life pattern of the household. The latter is a power generating model which generates an electric energy following the usage schedule of the appliances.

3.1 The Scheduling Model

In the scheduling model, we define an operation schedule of the appliances in the house. The scheduling model defines family types in general range in order to decide the schedules.

Family Organization. Even though the sizes of family are same between households, the power consumption patterns can be different according to family structure. For example, there are two households which consist of 3 members. The first family members are the parents(office workers) and the son(kindergartener). The other family members are the parents(office worker, housewife) and the son(university student). The life patterns of these two families are naturally different. And the power consumption patterns are also different as well. Therefore we define seven types of family member. Table 1 shows the family member types and the number of ID. We organize the family by a combination of members.

Table 1 Family member types

Types	ID
Kindergartener	1
Elementary School Student	2
Middle School Student	3
High School Student	4
University Student	5
Office Worker	6
Housewife	7

Appliances. We assorted appliances and devices which consuming an electric power in many households. Tables 2 shows the list of them and their information. We distinguished two types of the appliances. Type 1 is the appliances which consume the power in proportion to the operation time. Type 2 is the appliances which is operated automatically. Power means the power consumption per minute.

Table 2 The information of appliances

Appliances	ID	Types	Power(Watt)
Refrigerator	1	2	100
Air conditioner	2	1	1300
Kimchi fridge	3	2	130
Washing machine	4	1	130
Toaster	5	1	720
Dishwasher	6	1	1850
Coffee maker	7	1	1270
Vacuum cleaner	8	1	1900
Hair dryer	9	1	1237
Electric oven	10	1	1500
Microwave	11	1	1250
Air cleaner	12	2	40
Fan	13	1	55
TV	14	1	120
Steam iron	15	1	600
Printer	16	1	430
Wireless AP	17	2	6
Telephone	18	1	1
Audio	19	1	40
Electric Kettle	20	1	2000
Rice cooker	21	1	500
1st-bathroom lamp	22	1	10.8
2nd-bathroom lamp	23	1	10.8
1st-room lamp	24	1	54
2nd-room lamp	25	1	36
3rd-room lamp	26	1	18
4th-room lamp	27	1	18
Living-room lamp	28	1	80
Computer	29	1	170
Kitchen lamp	30	1	36

Scheduling Model. Let n is the number of family member types. Let U is a set and $u_i(1 \le i \le n)$ is an element which represents the type of family member. Let m is the number of family members and define $member_j(1 \le j \le m)$ as a symbol of a family member. u_i becomes an attribute of $member_j$. $individual\ basis_j =< t^w_{u_i}, t^o_{u_i}, t^r_{u_i}, t^b_{u_i} >$ is a tuple. The elements mean times of a basic life pattern(get up, go to work, come home from work, go to bed). Our daily life schedule is almost same during week days but not the same exactly. So we use parameters which is subordinate to family members such as working overtime or going to business trip if a family member is u_6(office worker). Let k is the number of the parameters. $parameter_j =< op^1_j, op^2_j, \dots, op^k_j >$ is a tuple. Therefore the schedule of $member_j$ is

$$member_j = individual\ basis_j + parameter_j \quad (1)$$

In this way, we can organize a family by following equation,

$$family = \sum_{j=1}^{m} member_j \quad (2)$$

Let w is the number of appliance types. Let A is a set of appliances and $a_j(1 \le j \le w)$ is an element which represents the type of a specific appliance. $app_j =< a_j, t_j, d_j >$ is a tuple of appliance. It composed of an appliance type a_j, a start time of use t_j and duration d_j. Let v is the number of appliances in the house. The schedule of all appliance of a specific household is

$$home = \sum_{j=1}^{v} app_j \quad (3)$$

Finally, we can write a whole schedule by following equation.

$$S = Schedule(family, home) \quad (4)$$

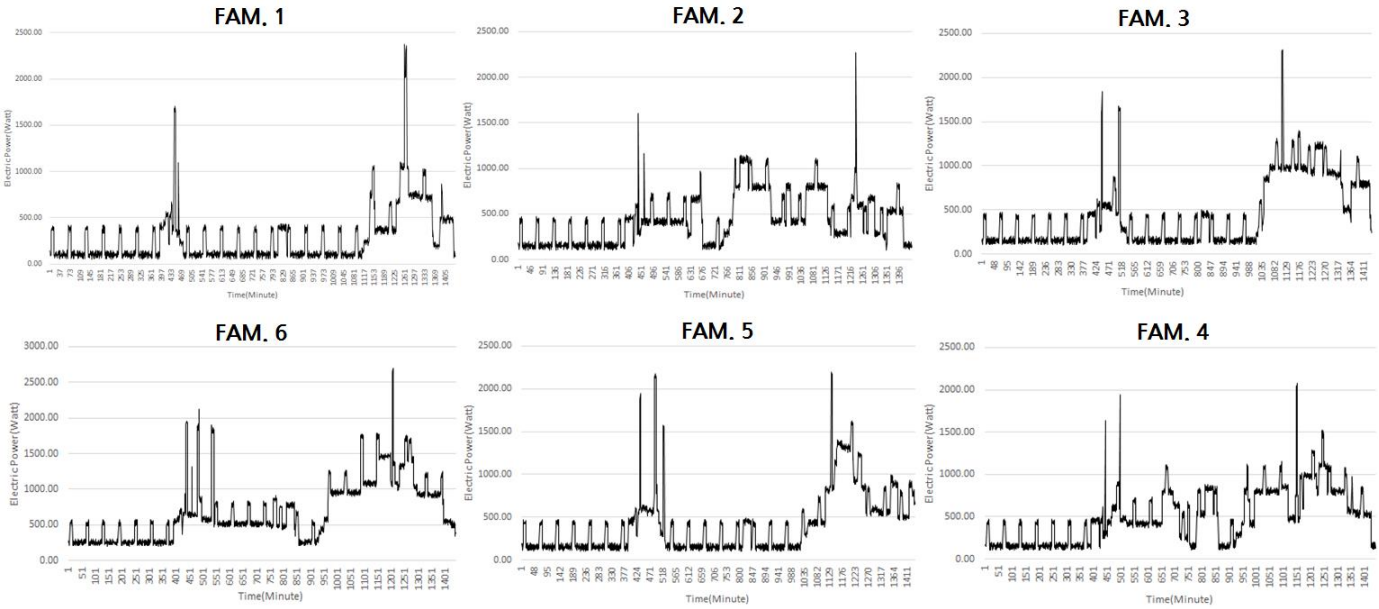


Figure 1 Patterns of Power Consumption : The patterns are listed in order clockwise from upper left by following table 3. The x-axis means minutes(0~1440) and the y-axis means electric power(Watt)

3.2 Power Generating Model

Using schedule *S*, power generator generates a power consumption every minute. All appliances can be scheduled many times in a day. We call it sub-schedule. Let *s* is the number of sub-schedules of app_j. We define (t_s^n, t_f^n) for each sub-schedule. t_s^n is a start time of use and t_f^n is an end time where $(1 \le n \le s)$. $p_{app_j}(i)$ is a power on *i-th* minute and it can be decided as follows

$$p_{app_j}(i) = \begin{cases} Consumption\ power, & (if\ t_s^n \le i \le t_f^n) \\ Standby\ power, & otherwise \end{cases}$$

$$, (1 \le t_s^n \le i \le t_f^n \le 1440) \quad (5)$$

Then the schedule of $app_j$is

$$P_{app_j} =< p_{app_j}(1), \dots, p_{app_j}(i), \dots, p_{app_j}(1440) > \quad (6)$$

Let *m* is a measurement period, *k* is a specific day and *v* is the number of appliances as we described in section 3.1, then *k-th* day power consumption is

$$P_{Day_k} = \sum_{j=1}^{v} P_{app_j}, (1 \le \mathrm{k} \le \mathrm{m}) \quad (7)$$

The power consumption during *m* can be written as follows

$$P_m = P_{Day_1} | \dots | P_{Day_k} | \dots | P_{Day_m} \quad (8)$$

3.3 Simulation

We developed a simulator by following power consumption model. Table 3 shows the family types and member composition. The numbers of structure of family column are IDs which defined in table 1. For example, FAM. 1 is comprised of two office workers. They can be a couple or father and son or mother and son and so on. We can infer various types of family. In this way, we defined 6 types of family to test the simulator.

Table 3 The types of family

Family Types	Structure of Family
FAM. 1	6, 6
FAM. 2	6, 7
FAM. 3	2, 6, 6
FAM. 4	2, 6, 7
FAM. 5	2, 3, 6, 6
FAM. 6	2, 3, 6, 7

Figure 1 shows the patterns of power consumption for each family type during a day. The x-axis means minutes(0~1440) and the y-axis means electric power(Watt). We can notice that even though the number of family members is same, the patterns can be different. For instance, because two members of FAM. 1 are office worker, electric power usage is low during the daytime. In the case of FAM. 2, one of FAM. 2 members is a housewife compared to F AM. 1. Therefore the electric power usage is higher than FAM. 1 during the daytime. This is the major reason that we classify the family types depending on the structure of family.

4 Experiment

The purpose of our experiment is to find an optimum feature and machine learning algorithm for classifying the family types. Figure 2 shows the flow of our experiment. Before generating schedules, we define the family specification. After that we generate schedules and the electric power consumption data by simulator. And then, we extract the features from data. Based on them, we train the classifiers and classify the family types.

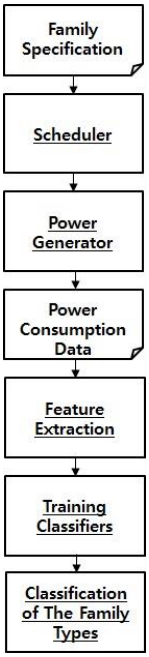


Figure 2 The performance testing flow

4.1 Data Sets

We generate the electric power consumption data during one year as a training data for each types of family listed in Table 3. In addition, we generate data for one month as a test data. In order to increase realism, we distinguished the weekdays and weekend. In addition, the schedule is randomly changed within a certain ratio whenever it is generated. Also, we added a season factor to the simulator. Although the structures of family are same, the pattern can be different depending on the each family member's schedule. In case this kind of situation, we generate five different schedules for each family type.

4.2 Feature Extraction

Feature extraction is very important in order to acquire the information from raw data. The performance of classification depends on the feature. We extract four features. Table 4 shows the features and their descriptions.

Table 4 The features and descriptions

Features	Description
F1	An average of power consumption per an hour
F2	Maximum and minimum power consumption per an hour
F3	An average power consumption of a day
F4	A standard deviation per an hour

Let P(i) is an average power consumption of *i-th* hour and x(j) is a power consumption of *j-th* minute. Then we can make a numerical formula of F1.

$$\mathrm{P(i)} = \frac{\sum_{j=1}^{60} x\times\{(60\times i)+j\}}{60}, (i = 0, 1, \dots, 23) \qquad (9)$$

Let Max(i) and Min(i) are maximum and minimum power consumption within *i-th* hour. And function max finds the biggest number and min finds the smallest number. F2 can be decided as follows.

$$\mathrm{Max(i)} = \max[x \times \{(60 \times i) + j\}]$$

$$\mathrm{Min(i)} = \min[x \times \{(60 \times i) + j\}] \qquad (10)$$

$$(\mathrm{i} = 0, 1, \dots, 23, \mathrm{j} = 1, 2, \dots, 60)$$

Let D(k) is an average power consumption of *k-th* day, then F3 is as in the following.

$$\mathrm{D(k)} = \frac{\sum_{j=1}^{1440} x(j)}{1440}, (k = 1,2, \dots) \qquad (11)$$

Let σ(i) is a standard deviation per an hour, then F4 is

$$\sigma(\mathrm{i}) = \sqrt{\frac{\sum_{j=1}^{60} x^2\times\{(60\times i)+j\}}{60} - P(i)^2}, (i = 0, 1, \dots, 23) \qquad (12)$$

4.3 Performance on Classification

We use five different machine learning algorithm. They are Tree(C4.5), Neural network, SVM(Support Vector Machine), k-NN(k-Nearest Neighbor) and Bagging(Ensemble). Each classifier is trained by using the data which were generated during one year for each family types. The test data which were generated during one month will be used to measure the error rate of the classification. The error rate is a ratio of misclassification. We compare the error rate of each classifier. Table 5 to 9 show the error rates of the family classification for each feature and machine learning algorithm. In case of using kNN algorithm, we can get the best performance when we extract the data by an average of power consumption per an hour. In Table 6, extracting maximum and minimum power consumption per an hour from data is good for classification. In Table 7, the result is the same with Table 6. F1 and F2 have the same error rate in Table 8. In the last table, choosing F2 brings a good result as well.

Table 5 The error rate of family classification by using kNN

Features	Error rate(%)
F1	23.33
F2	26.67
F3	36.67
F4	56.67

Table 6 The error rate of family classification by using Tree

Features	Error rate(%)
F1	16.67
F2	10.00
F3	36.67
F4	36.67

Table 7 The error rate of family classification by using SVM

Features	Error rate(%)
F1	30.00
F2	23.33
F3	66.67
F4	36.67

Table 8 The error rate of family classification by using Bagging

Features	Error rate(%)
F1	16.67
F2	16.67
F3	36.67
F4	36.67

Table 9 The error rate of family classification by using Neural Network

Features	Error rate(%)
F1	43.33
F2	23.33
F3	33.33
F4	46.67

We calculated the average error rates for each algorithm and feature. In Figure 3, we can notice that Tree algorithm has the lowest error rate among others. In Figure 4, the average error rates of F2 is the lowest. As a result, we can conclude that F2 and Tree are the optimum feature and algorithm for classifying the family types.

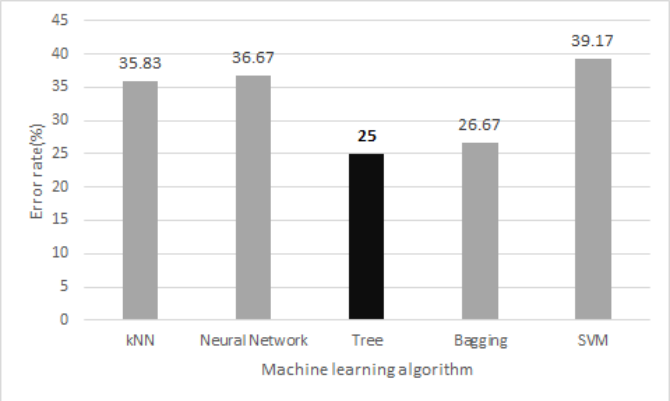


Figure 3 The Average Error Rates for each Algorithm

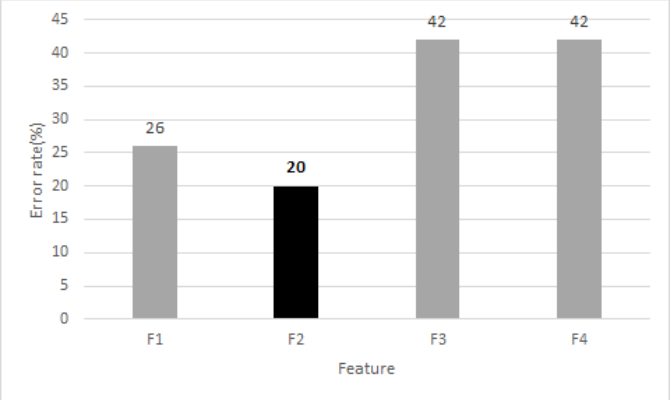


Figure 4 The Average Error Rates for each Feature

5 Conclusion

We proposed the electric power consumption model. Based on our model, we developed the simulator for generating the electric power consumption data of various family types. We experimented which algorithm and feature are proper for classification. We found out Tree and Maximum and minimum power consumption per an hour are optimum algorithm and feature. For realizing the complete smart grid environment, revitalization of the related services must be accompanied. For this, the data from smart grid infrastructure should be provided to service suppliers within preserving the privacy of users. We expect that our study can play an important role for offering the customized service in the future. So far, we only used the electric power data and season factor. However, if we utilize external factors such as a family member's income level or characteristics of residential zone, we could raise accuracy. Therefore, we will focus on applying the compositive factors to our model for an accurate classification. Also, we will progress our study to predict the power usage at the household in order to contribute the efficient electric power use.

ACKNOWLEDGMENT

This work was supported by the Industrial Strategic Technology Development Program (No.10043907, Development of high performance IoT device and Open

Platform with Intelligent Software) funded by the Ministry of Science, ICT & Future Planning (MSIP, Korea).

REFERENCES

[1] Farhangi, Hassan. "The path of the smart grid." Power and Energy Magazine, IEEE 8.1, pp.18-28, 2010.

[2] IntelliGrid: www.epri-intelligrid.com.

[3] ″GRID 2030: A National Vision for Electricity's Second 100 Years." United States Department of Energy, Office of Electric Transmission and Distribution, July 2003.

[4] Faruqui, Ahmad, Dan Harris, and Ryan Hledik. "Unlocking the€ 53 billion savings from smart meters in the EU: How increasing the adoption of dynamic tariffs could make or break the EU's smart grid investment." Energy Policy38.10, pp.6222-6231, 2010.

[5] Albert, Adrian, et al. "Drivers of variability in energy consumption." Submitted for review to European Conference on Machine Learning, 2013.

[6] Fei, Hongliang, et al. "Heat pump detection from coarse grained smart meter data with positive and unlabeled learning." Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2013.

[7] Boullé, Marc. "MODL: A Bayes optimal discretization method for continuous attributes." Machine learning 65.1, pp. 131-165, 2006.

[8] Dachraoui, Asma, Alexis Bondu, and Antoine Cornuejols. "Early Classification of Individual Electricity Consumptions.", RealStream, 2013.

An Extraction Method of Influential Lightings for Illuminance Sensors on An Intelligent Lighting System in Large Office

Hisanori Ikegami[2], Sho Kuwajima[2], Mitsunori Miki[1] and Hiroto Aida[1]
[1]Department of Science and Engineering, Doshisha University, Kyoto, Japan
[2]Graduate School of Science and Engineering, Doshisha University, Kyoto, Japan

Abstract— *We research and develop an intelligent lighting system to improve office workers comfort and to reduce the power consumption. We have introduced the intelligent lighting system to realize individual lighting environments into real office environments. According to target illuminance values, we reduce the power consumption drastically. We are considering to introduced the intelligent lighting system to the larger office. On the other hand, we have proposed the effective lighting control algorithm Adaptive Neighborhood Algorithm using Regression Coefficient:ANA/RC. In this method, to learn the positional relation of lightings and illuminance sensors using regression coefficient, luminous intensity is capable of appropriately changing. In the future, to introduce the intelligent lighting system to the larger office, we have to verify increasing learning time of the positional relation and accuracy of lighting extraction. The study proposes these verification and a new lighting extraction method without regression coefficient. We show that the proposed method is superior to previous method in the learning time and accuracy of lighting extraction.*

Keywords: Lighting Control, Large Office, Regression Coefficient

1. Introduction

In recent years, continuing research into the office environment has identified that the office environment has a major influence on workers. Previous research has reported that improving the office environment can increase workers' intellectual productivity and comfort[1], [2]. With regard to the lighting environment, it has also been reported that providing each worker's desired brightness can raise intellectual productivity[3]. However, at present, the standard lighting design of Japanese offices features a desktop illuminance of 750 lx or greater in Japan. Consequently, this cannot be considered a lighting environment suited to each worker. Furthermore, it is also believed that desired illuminance differs by race and culture. For all these reasons, we have been researching into an intelligent lighting system in order to provide individual illuminance environments in our laboratory[4], [5]. An intelligent lighting system provides each user's desired illuminance, and also gives energy saving.

Past applications of Intelligent Lighting Systems include a case using 35 lighting fixtures. Another example of an Intelligent Lighting System has been introduced at Kayabacho Green Building (Mitsubishi Estate Co., Ltd) completed in May 2013, which optimally controls lighting for the whole 7th and 8th floors, using 50 lighting fixtures per floor. Owing to their effectiveness, Intelligent Lighting Systems are now being considered for applications of even larger scales.

Current Intelligent Lighting Systems use a control algorithm called "ANA/RC" for optimizing their lighting patterns, which is based on the hill-climbing method. ANA/RC varies the luminance of each lighting appropriately for the level of influence by the lighting's luminance on illuminance sensor measurements (hereinafter called the illuminance/luminance influence factor). Here, by lowering or turning off those lighting fixtures which little affect illuminance sensor measurements, these systems can realize a high standard of energy efficiency, while their energy efficiency largely depends on how accurately the system can learn illuminance/ luminance influence factors. To dynamically learn illuminance/ luminance influence factors, ANA/RC varies the luminance of each lighting minutely to the extent unnoticeable to human eyes and performs regression analysis based on the luminance change and illuminance change. However, while it learns illuminance/ luminance influence factors by regression analysis, temporary correlation phenomenon between random numbers may occur, to cause a regression coefficient of a lighting fixture distant from an illuminance sensor to be assessed too highly, inhibiting the system to realize an energy efficient lighting pattern. To solve this problem for improving the system's accuracy in learning illuminance/ luminance influence factors by regression analysis, referring to the fact that relatively accurate regression coefficients are available for two lighting fixtures closest to an illuminance sensor, we developed a method to detect a group of lighting fixtures close to an illuminance sensor, using these two lighting fixtures and a lighting layout map (hereinafter referred to as "MAP method"). This method, as mentioned earlier, determines illuminance/luminance influence factors with reference to influence ranks. Using the MAP method enables the system to identify lighting fixtures close to illuminance sensors with much higher accuracy compared to a pre-existing method of ANA/RC. However, with a greater number of lighting

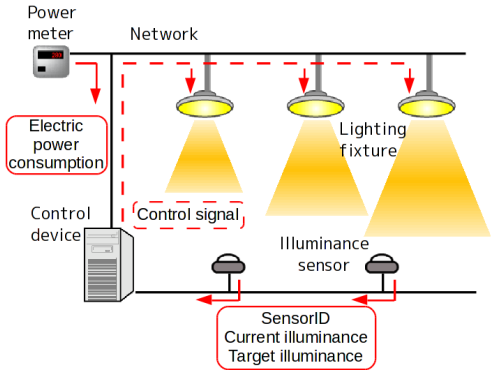


Fig. 1: Configuration of the Intelligent Lighting System

fixtures, the system takes a longer time for regression analysis as well as for identifying two lighting fixtures close to an illuminance sensor.

By simulating lighting configurations with a minimum of 16 to a maximum of 400 lighting fixtures, this study examines the time required for detecting lighting fixtures close to illuminance sensors based on regression analysis, as well as the accuracy of detection results, and indicates problems in methods based on regression analysis. Furthermore, to identify lighting fixtures close to an illuminance sensor even in configurations comprising many lighting fixtures, a new method without using regression analysis will be proposed. It will be shown that the proposed method can identify lighting fixtures close to an illuminance sensor more correctly and quickly than regression analysis-based methods.

2. Intelligent lighting system

2.1 Overview of the intelligent lighting system

The intelligent lighting system, as indicated in Fig. 1, is composed of lights equipped with microprocessors, portable illuminance sensors, and electrical power meters, with each element connected via a network.Individual users set the illuminance constraint on the illuminance sensors. At this time, each light repeats autonomous changes in luminance to converge to an optimum ligthing pattern. Also, with the intelligent lighting system, positional information for the lights and illuminance sensors is unnecessary. This is because the lights learn the factor of influence to the illuminance sensors, based on illuminance data sent from illuminance sensors. In this fashion, each user's target illuminance can be provided rapidly.

2.2 Lighting control algorithm

The hill-climbing method is an algorithm where the optimal solution is derived by generating the solution of the next step based on the solution of the current step. Solutions are accepted based on the changes in the objective function value, and the transition process is repeated until an optimal solution is derived. Control is performed by taking the lightness of lighting (the luminous intensity) to be the design variable, and taking the sum of the difference between current and target illuminance and electric power consumption to be the objective function. Furthermore, in ANA/RC, differences in lightness that a light exerts on the illuminance meter are learned by regression analysis, and luminous intensity is appropriately changed in response to the degree of exertion[4], [6], [7]. By using this process, solutions can be quickly derived. Hereafter, the lightness difference a light exerts on an illuminance meter will be called 'influence'.

Intelligent lighting system aims to adjust the illuminance to equal or greater than the target illuminance for the location where the sensors are installed, and autonomously finds the lighting intensity to minimize the amount of electrical power used for lighting. This illuminance must be formulated as an objective function. The objective function for each lamp is shown in Eq. 1.

$$
\begin{aligned}
f_i &= P + w\sum_{j=1}^{n} g_{ij} \quad i = 1, 2, 3, ..., m \qquad (1)\\
g_{ij} &= \begin{cases} 0 & (Lc_j - Lt_j) \geq 0 \\ R_{ij}(Lc_j - Lt_j)^2 & (Lc_j - Lt_j) < 0 \end{cases}\\
R_{ij} &= \begin{cases} r_{ij} & r_{ij} \geq Threshold \\ 0 & r_{ij} < Threshold \end{cases}
\end{aligned}
$$

f:Objective function, i:light number, m:Number of light
n:Number of illuminance sensors, w:Weighting factor
P:Amount of consumed electrical power
Lc:Current illuminance, Lt:Target illuminance
r:Correlation coefficient, $Threshold$:Threshold value

Making the brightness for each lamp the design variable, we aim to minimize the f in Eq.1. f_i consists of the amount of consumed power *P*, and g_{ij}, which is derived by multiplying by the correlation coefficient r_{ij} by the difference between the current illuminance L_c and the target illuminance L_t entered by the user. The correlation coefficient r_{ij} accounts for the change in luminance for the lamp *i* and the change in illuminance for the illuminance sensor *j*. If the correlation is less than or equal to the threshold value, it is multiplied by 0. g_{ij} is added only if the current illuminance has fallen below the target illuminance. Thus, the accuracy to which the target illuminance can be met is improved by narrowing down the optimization target for the sensor with the highest correlation, that is, for the sensor which is located nearby. Also, g_i is multiplied by a weight *w*, and the value of this weight *w* determines whether priority is given to minimizing, either the constraint conditions on the target illuminance, or the amount of consumed power.

2.3 Issues in large-scale lighting configurations

The effectiveness of Intelligent Lighting Systems for increasing workers' comfort and reducing power consumption has already been recognized, and verification experiments are underway at several offices in Tokyo. To date, there have been applications using 35 lighting fixtures and 50 lighting fixtures, and applications of even larger scales are under consideration. Applications to large-scale lighting configurations will require considerations on many factors such as increased physical wiring to lighting fixtures and illuminance sensors, delay in the acquisition of illuminance data from illuminance sensors, delay in the transmission of control signals to lighting fixtures, or applicability of the lighting control algorithm. This study, will examine the applicability of conventional methods using regression analysis (the ANA/RC and MAP methods) to large scale lighting configurations, particularly concerning the detection of influential lighting fixtures for each illuminance sensor: here, we will examine the time required for accurate detection of influential lighting fixtures, and indicate problems in the detection of influential lighting fixtures by regression analysis.

3. Examination of the accuracy detecting illuminance sensors in large office

3.1 Experiment summary

Experiments simulating nine types of lighting arrangements (4 rows x 4 lines, 6 rows x 6 lines ... and 14 rows x 14 lines) were conducted. Out of the simulated configurations, Fig. 2 shows the arrangements of 4 rows x 4 lines and 6 rows x 6 lines. In all lighting configurations simulated here, the illuminance sensor was positioned right under the lighting in the third row and third line. To calculate illuminance in the simulation, a point-by-point method was used. A point-by-point method is a simple method of calculating illuminance at an illuminated point from the luminance of the light source based on the distance between the light source and the illuminated point. In order to sufficiently meet illuminance requirements by workers, four to six lights need to be chosen for each illuminance sensor. For this, in the experiment, the detection of nearby lighting fixtures by ANA/RC was considered to be successful when four or more lights out of the nine closest to the illuminance sensor show the highest regression coefficients. In the MAP method, if the two lights closest to the sensor are identified, then four to six lights can be selected based on the lighting layout map. Hence, for the MAP method, the detection of influential lighting fixtures was deemed successful if the regression coefficients for at least two out of the nine closest lights were highest.

3.2 Examination of the time required for detecting in previous method

We measured the time required to successfully detect lighting fixtures close to illuminance sensors in configurations with a small number of lighting fixtures and a large number of lighting fixtures. In conditions assuming these configurations, 100 times of simulation experiments were conducted. Figures 4 and 5 show the time required for learning regression coefficients, and success rates in the detection of influential lighting fixtures in 4x 4 and 10x 10 lighting conditions respectively.

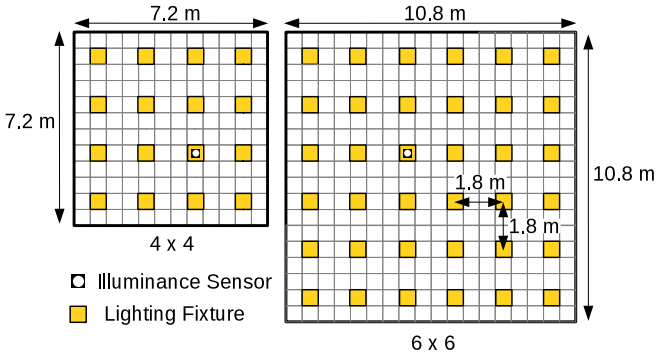


Fig. 2: A part of the experiment environment(top view)

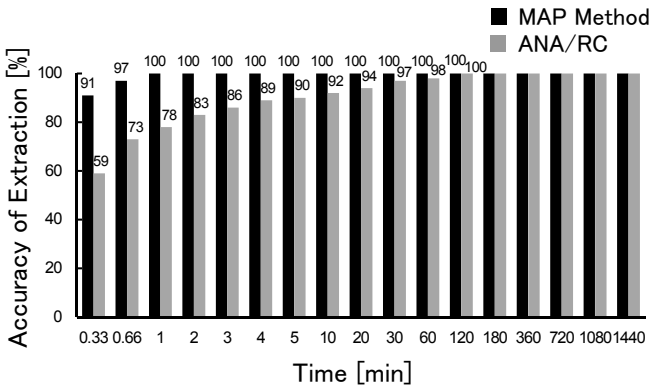


Fig. 3: Time for Accuracy Extraction (4×4 Lights)

Fig. 3 shows that in a small lighting configuration, the MAP method requires only one minute (30 steps) for detecting lighting fixtures close to the illuminance sensor. We can see from Fig. 4, however, that even in the MAP method, the nearby lighting fixture detection accuracy does not reach 100% before 12 hours (21600 steps). When a single regression analysis is to be conducted, the larger the number of lighting fixtures is, the longer it will take to calculate regression coefficients correctly even for the two lighting fixtures closest to the illuminance sensor. When a multiple regression analysis is to be conducted, the number of the data needed for a regression analysis increases as the number of lighting fixtures (the number of explanatory variables) increases: because at least as many steps as the number of lighting fixtures (explanatory variables) are needed until a regression analysis is available, it is impractical for a large scale lighting configuration. In order to prevent discomfort

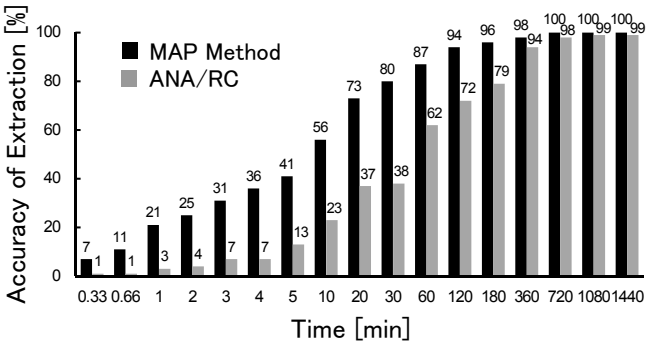


Fig. 4: Time for Accuracy Extraction (10×10 Lights)

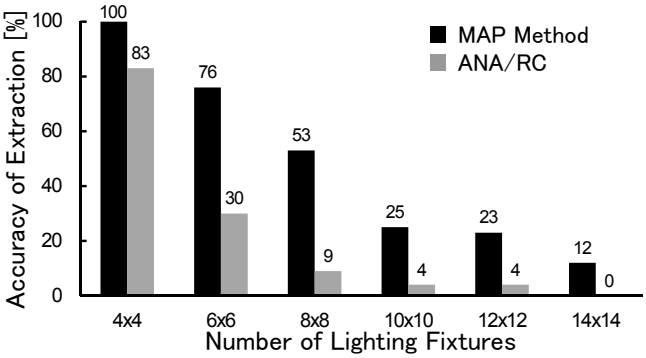


Fig. 5: Extraction Accuracy and the Number of Lighting Fixtures

or mistrust of the system in workers, it must be able to quickly detect influential lighting fixtures and realize the target illuminance. When introducing an Intelligent Lighting System in a real office, we first determine the time required for accurate detection of influential lighting fixtures while preventing discomfort among workers, through repeated experiments and dialogues with users at the real office, before configuring the system's regression-analysis based learning time. To realize an application in a real office configuration, quick and accurate detection of influential lighting is essential.

3.3 Verifying the accuracy of short-time detection of nearby lighting fixtures

Simulation experiments in nine lighting conditions were conducted by repeating 100 tests each, using a regression-analysis-based learning time in 60 steps (about two minutes), and the accuracies of the detection of lighting fixtures close to an illuminance sensor were compared between different conditions. The learning time of 60 steps (about two minutes) is based on the setting adopted by the Intelligent Lighting System at Futako Tamagawa Catalyst BA in August 2012. The success rate in the detection of influential lighting fixtures in different lighting conditions is shown in Fig. 5.

As you can see from Fig. 5, as the number of lighting fixtures increases, the success rate goes down. Although the regression coefficient for an illuminance sensor close to a lighting fixture can be obtained with a relatively close gap from the true value, a regression coefficient for an illuminance sensor distant from a lighting fixture is often far from the true value, sometimes exceeding the coefficient for a sensor close to the lighting: ANA/RC calculates a regression coefficient by randomly varying the luminance of lighting fixtures by minute scales, but when illuminance change on an illuminance sensor happens to be similar to luminance change on lighting fixtures distant from the sensor, the regression coefficient for the lighting distant from the sensor may be calculated higher than it really is. The probability of this phenomenon is higher when the number of lighting fixtures is greater, and that is considered to be the cause of the decline of the detection success rate with the increase in the number of lighting fixtures. In the case of the Intelligent Lighting System installed at Futako Tamagawa Catalyst BA using MAP method, the system prevents wrong detection by trying the learning process again if the two lights detected as most influential are found distant from the sensor on the lighting layout map. Specifically, in the case of a configuration with 36 (6 x 6) lights in Fig. 5, if the initial accuracy is 76%, another two learning trials (taking about 6 minutes including the initial learning) can boost the success rate to 98.6%[1]. In a configuration having not more than 50 lighting fixtures, such as the current office applications of an Intelligent Lighting System, detecting influential lighting fixtures has been possible with a practical accuracy by finding wrong detections using the MAP method. However, in a configuration with 10 x10 lights, the success rate remains as low as 57.8% after another two learning trials.

From these results, an accurate identification of influential lightings takes much time even with a MAP method; the larger the number of lighting fixtures is, the more difficult it is to accurately identify lighting fixtures close to an illuminance sensor in a short time. Now it is essential to establish a method which can identify lighting fixtures near an illuminance sensor within a learning time as short as in the existing real office applications of Intelligent Lighting Systems, even when the number of lighting fixtures increases.

4. Identification of lighting by scanning illuminance sensor coordinates

Here, we propose a method without using a regression coefficient to identify lighting fixtures close to an illuminance sensor in a large-scale lighting configuration. In this method, lights are divided into row groups and line

[1] p+(1-p)*p+(1-p)*(1-p)*p (when the success probability is p)

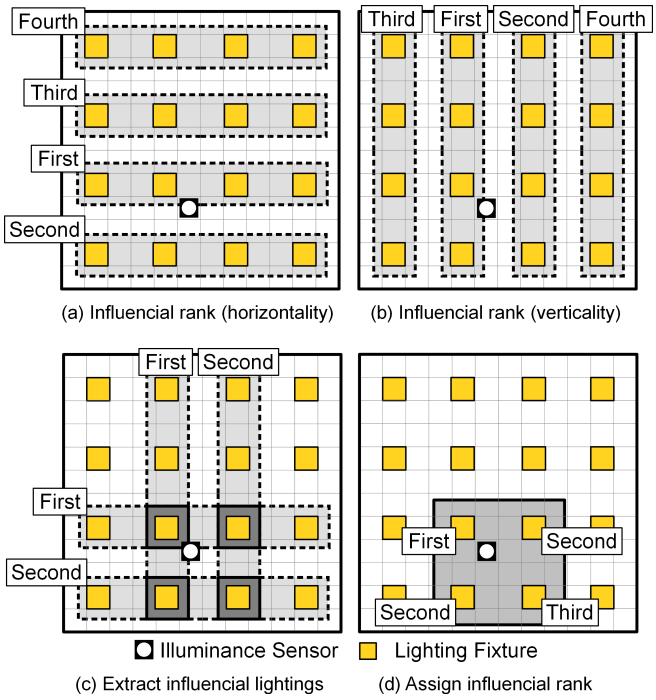


Fig. 6: Illuminance sensor detecting method on the coordinate

groups, and the luminance settings of these groups are varied by degrees unnoticeable to human eyes one group after another[8]. The system varies the luminance of lights in each group to a certain degree, and considers the lighting group causing the greatest variation in sensor illuminance to be the group closest to the illuminance sensor. By finding an intersection of the groups causing the greatest sensor illuminance change, the system identifies the lighting fixtures close to the illuminance sensor. Fig. 6 shows the concept of this method. Also, the flow of the identification of nearby lighting fixtures is shown below.

1) All lights are divided into row groups and line groups
2) The system obtains the current illuminance on each illuminance sensor
3) The system changes the luminance of all lights in a lighting group by ΔL which is the same value for all groups.
4) The system acquires the illuminance data from all illuminance sensors and calculates illuminance changes.
5) For all lighting groups, steps from (2) to (4) are repeated.
6) Based on the illuminance changes calculated in step (4), take an intersection of the lighting groups of the greatest illuminance change.
7) With reference to the illuminance change of the group a detected lighting belongs to, give it a rank of influence on the illuminance sensor.

Concerning step (3), based on the minimum noticeable change ratio[8], the system varies the luminance by magnitudes unnoticeable to human eyes. Because the value may depend on the lighting arrangement, distance between the ceiling and the desk surface and the lighting's radiation characteristics, âĹĘL needs to be determined through experiments in a real configuration. Concerning step (7), as shown in Fig. 6-(c) and Fig. 6-(d), for example, the lighting picked as an intersection of the largest illuminance change row group and line group is ranked the first, whereas the lighting picked as an intersection of the largest illuminance change row group and the second-largest illuminance change line group is ranked the second. Using this way, lights can be ranked in terms of distance from each illuminance sensor. By determining illuminance/ luminance influence factors based on the ranking of lights close to each illuminance sensor, power consumption can be saved as much as by an Intelligent Lighting System already introduced in real office configurations. Furthermore, the time required for picking nearby lighting fixtures in this method is for (the number of lights per row + the number of lights per line) steps regardless of the total number of illuminance sensors. Hence, even in the case of a configuration with 30 x30 lights, influential lighting fixtures can be picked in only 60 steps (or about 2 minutes). Since the time required for picking nearby lighting fixtures is in a linear relation with the number of lighting fixtures, this method is more suitable for large scale lighting configurations than conventional methods which are subject to a non-linear increase in the time required for accurately picking nearby lighting fixtures.

On the other hand, real-office applications need to take into consideration the effect of factors other than the luminance of the lighting, such as people's shadows or change in natural light, on the illuminance detected by a sensor. Since the illuminance given by the luminance of lighting fixtures is in inverse relation to the second power of the distance, the illuminance change given by an illuminance sensor by each lighting group is larger in groups closer to the illuminance sensor, and smaller in groups farther from the sensor. The illuminance change from the effects of factors other than the lighting such as human shadows and natural light can be detected by comparing illuminance changes between adjacent lighting groups. Furthermore, concerning step (3), the search time may be even shortened by simultaneously changing the luminance of two groups of lighting fixtures which are distant enough to be mutually free from interference. This method has already been used in an Intelligent Lighting System installed at Futako Tamagawa Catalyst BA, which is now in the process of disturbance detection and verification of search time reduction

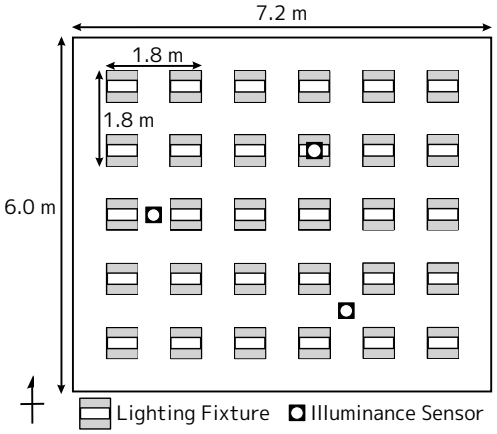


Fig. 7: Experiment Environment(top view)

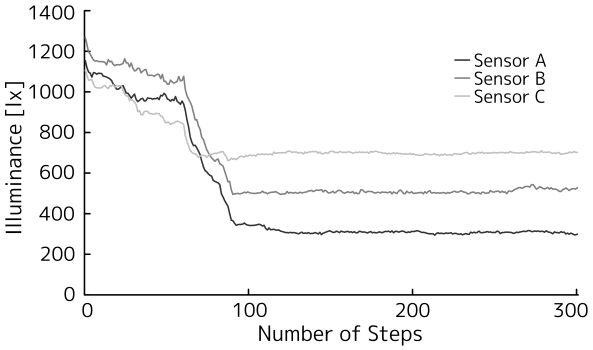


Fig. 8: Illuminance History in ANA/RC

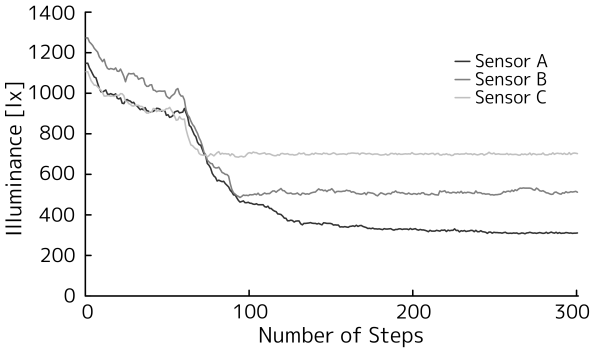


Fig. 9: Illuminance History in MAP Method

5. Illuminance convergence and lighting pattern verification experiment

A verification experiment was conducted to demonstrate the effectiveness of an Intelligent Lighting System incorporating the proposed method. In the experiment, a small number of illuminance sensors were used to make it easier to verify whether lighting fixtures in appropriate positions are turned on. For the experiment, a configuration using 30 lighting fixtures and three illuminance sensors as shown in Fig. 8 was used. The experiment was conducted in a space measuring 7.2m x6.0m x1.9m, equipped with cool white fluorescent lamps (FHP45EN) made by Panasonic Corporation variable between 30% and 100%.

An illuminance convergence experiment was conducted in an experimental configuration shown in Fig. 7. Concerning the ANA/RC and MAP method, 60 steps (about 2 minutes) were provided for the system's earning by regression analysis. This is the same as the learning time used in existing Intelligent Lighting Systems adopted in real offices. Between a method using regression analysis and the proposed method, the illuminance histories and lighting statuses of each lighting were compared. Fig. 9 shows the illuminance history of each illuminance sensor with ANA/RC; Fig. 10 shows the illuminance history of each illuminance sensor with a MAP method; and Fig. 11 shows the illuminance history of each illuminance sensor with the proposed method. For information, each lighting control step takes about two seconds. Fig. 12 shows the status of lighting after 300 steps in each method.

As one can see from Fig. 8 and Fig. 9, there is no significant difference in illuminance history between the methods which detect influential lighting fixtures based on regression analysis. However, as Fig. 11 shows, some lighting fixtures distant from the illuminance sensor are turned on in the ANA/RC method while only those near the sensor are turned on in the MAP method, realizing an efficient lighting pattern in terms of energy consumption. The proposed method also realizes an energy efficient lighting pattern like the MAP method. Next, as you can see from Fig. 10, the proposed method has completed the detection of influential lighting fixtures in eleven steps, which is faster than in methods based on regression analysis. If the illuminance/ luminance influence factor is to be learned in ten steps (about 20 seconds) in a convergence experiment using a conventional method, according to Fig. 5 in Section 3.2, the detection success rate will be 7% in the MAP method and 1% in the ANA/RC method. Unless influential lighting fixtures are successfully detected, it will never be easy to realize a target illuminance nor energy-efficient lighting patterns. From these results, it has been demonstrated that the proposed method is able to detect lighting fixtures close to an illuminance sensor more quickly and accurately than conventional methods.

With a method using regression analysis, the system can learn the illuminance/ luminance influence factor while controlling convergence to the target illuminance. On the other hand, with the proposed method, the system cannot simultaneously pursue the two processes of learning the illuminance/ luminance influence factor and controlling convergence to the target illuminance. However, as one can see from Fig. 10, the illuminance variation in the learning period for detecting influential lighting fixtures is so small as to be unnoticeable to workers [8] that the short period of the learning phase without illuminance convergence control is

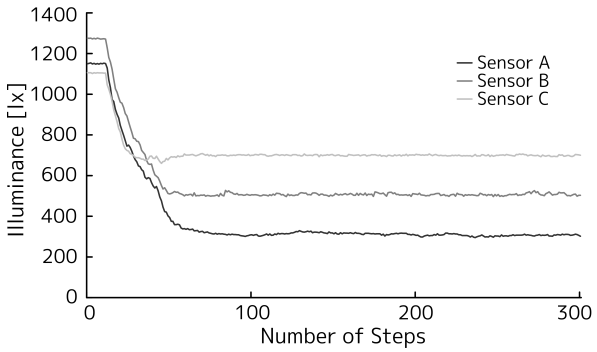


Fig. 10: Illuminance History in Proposed Method

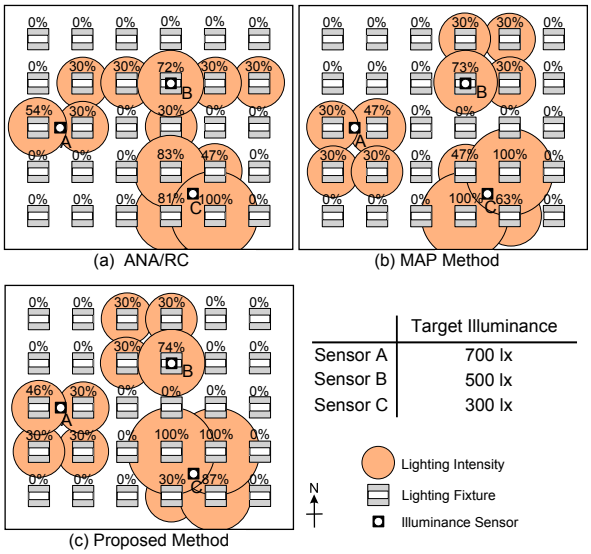


Fig. 11: Status of Lightings (300 steps)

considered to cause no problem.

For information, if the proposed method is used in a configuration with 10 x10lights, influential lighting can be detected in 20 steps (about 40 seconds). For comparison, Fig. 12 shows the accuracy of detection in other methods in an illuminance/ luminance influence factor learning time of 20 steps in a configuration with 10 x10lights.

Because it is not easy to reproduce by simulation all disturbance factors plausible in a real office configuration, the detection accuracies shown in Fig. 13 are experimental values from simulations. Currently, a verification experiment concerning disturbance detection and search time reduction is underway with the Intelligent Lighting System installed at Futako Tamagawa Catalyst BA. Furthermore, we plan to build an Intelligent Lighting System involving worker movements with more than 50 lighting fixtures, which will be experimented in another verification experiment at a real large-scale office

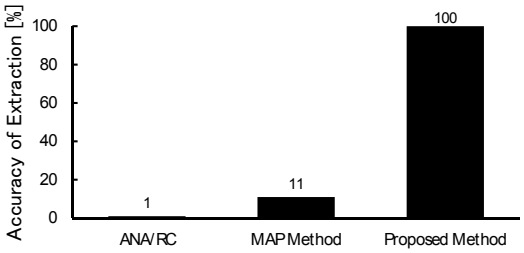


Fig. 12: Extraction Accuracy of Influencial Lightings for an Illuminance Sensor (10×10 Lights)

6. Conclusion

In this study, we demonstrated that detecting influential lighting fixtures close to an illuminance sensor by regression analysis requires much time for accurate lighting detection when the number of lighting fixtures is greater. In view of this fact, in order to quickly detect lighting fixtures close to an illuminance sensor even in a configuration with many lighting fixtures, we developed a new method not relying on regression analysis. The proposed method divides lighting fixtures into groups, and detects lighting close to an illuminance sensor based on the luminance change in each lighting group and the illuminance change on the illuminance sensor. Verification experiments have demonstrated that the proposed method can shorten the time required for detecting lighting fixtures compared to regression analysis methods, and enables correct detection of nearby lighting fixtures.

References

[1] Olli Seppanen, William J. Fisk: A Model to Estimate the Cost-Effectiveness of Improving Office Work through Indoor Environmental Control, Proceedings of ASHRAE, 2005

[2] M. J. Mendell, and G. A. Heath: Do indoor pollutants and thermal conditions in schools influence student performance A critical review of the literature, Indoor Air, Vol.15, No.1, pp.27-52, 2005

[3] Peter R. Boyce, Neil H. Eklund, S. Noel Simpson, Individual Lighting Control: Task Performance, Mood and Illuminance JOURNAL of the Illuminating Engineering Society, pp.131-142, 2000

[4] Miki M, Hiroyasu T, Imazato K, Proposal for an intelligent lighting system, and verification of control method effectiveness, Proc IEEE CIS, pp. 520 - 525,2004

[5] M.Miki, An Intelligent Lighting System and the Consortium for Smart Office Environment, Journal of Japanese Society for Artif icial Intelligence, Vol.85 No.5, pp.346-351, 2001

[6] K.Ono,M.Miki,Y.Motoi,Autonomous Distributed Optimization Algorithm for Intelligent Lighting System, The transactions of the Institute of Electrical Engineers of Japan. C, A publication of Electronics, Information and System Society Vol.130 No.5, pp.750-757, 2010

[7] S.Tanaka,M.Miki,T.Hiroyasu,M.Yoshikata,An Evolutional Optimization Algorithm to Provide Individual Illuminance in Workplaces,Proc IEEE Int Conf Syst Man Cybern,Vol.2, pp.941-947, 2009

[8] S.Tomoaki,M.Hiroyuki,N.Yoshiki, Research on the Perception of Lighting Fluctuation in a Luminous Offices Environment, Journal of the Illuminationg Engineering Institute of Japan, Vol.85 No.5, pp.346-351, 2001

A New Competitive Ensemble Algorithm (ERAB)

Honghai Feng[1,2] , Junhui Huang[1], Lijuan Liu[1], Kaiwei Zou[1], Zhuo Ran[1], Qiannan Zhu[1], Lijin Ji[1], Sai Liu[1], Xiangyong Lu[1]

[1]School of Computer and Information Engineering

[2]Institute of Data and Knowledge Engineering

Henan University, Kaifeng, Henan, 475001, China

Abstract - *Ensemble algorithms are popular methods for improving the accuracy of a classifier. This paper proposes a rough set based ensemble algorithm that generates 10 rules for every instance and assigns one rule to one base classifier to construct an ensemble learner (Every rules for a base classifier, ERAB). We run an experiment of the new ensemble algorithm on 68 datasets compared with other 20 existing ensemble methods, and results show that the new algorithm is more accurate than the 20 prevailing ensemble algorithms in terms of 8 performances measures.*

Keywords: AdaBoostM1, Bagging, RandomForest, Rough Set, Ensemble

1 Introduction

Many methods have been proposed for the creation of ensemble of classifiers. Mechanisms that are used to build ensemble of learners [1] include: (i) Using different subset of training data with a single learning technique, (ii) Using different learning methods, (iii) Using different training parameters with a single learning method. A handy and informal reasoning, from computational, statistical and representational viewpoints, of why ensembles can get better results is provided in Dietterich (2001) [2]. The key for success of ensembles is whether the classifiers in a system are diverse enough from each other, or in other words, that the individual classifiers have a minimum of errors in common. If one classifier makes an error then the others should not be likely to make the same error.

Four of the most popular ensemble algorithms are bagging (Breiman 1996) [3], boosting (Freund et al. 1996) [4], rotation forest (Rodríguez et al. 2006) [5] and Random subspace method (Ho 1998) [6] and RandomForest [7].

Bagging, a name derived from "bootstrap aggregation", uses multiple versions of a training set by using the bootstrap, i.e. sampling with replacement. Each of these data sets is used to train a different model. The outputs of the models are combined by averaging (in case of regression) or voting (in case of classification) to create a single output. Bagging is only effective when using unstable (i.e. a small change in the training set can cause a significant change in the model) nonlinear models.

Most boosting algorithms [8-9] consist of iteratively learning weak classifiers with respect to a distribution and adding them to a final strong classifier. When they are added, they are typically weighted in some way that is usually related to the weak learners' accuracy. After a weak learner is added, the data is re-weighted: examples that are misclassified gain weight and examples that are classified correctly lose weight (some boosting algorithms actually decrease the weight of repeatedly misclassified examples, e.g., boost by majority and BrownBoost). Thus, future weak learners focus more on the examples that previous weak learners misclassified.

In this paper, We propose a rough set based algorithm that generates 10 rules for every instance and assigns one rule to a base learner to construct an ensemble learner (ERAB). We give experimental evidences that the new algorithm is more accurate than or competitive to prevailing ensemble algorithms such as AdaBoostM1, Bagging and RandomForest.

2 Existing Exnsemle Algorithms

AdaBoost iteratively employ another algorithm known as the base learner to generate a series of models. Any algorithm normally used for classification or prediction can be employed as the base learner, providing it allows weighting of samples. Normally simple base learners such as decision trees, rough sets based rule learner, and associative rules based rule learners are used. Initially all samples have equal weights, then after the first iteration the accuracy of the model produced is measured and the samples weights are adjusted so that the weights of misclassified samples are increased while those of correctly classified samples are reduced. At the next iteration the base learner will concentrate on the misclassified samples. A series of models is then produced with the sample weights being adjusted each time. These models are then combined into an ensemble voting committee. Different boosting algorithms vary in the way that the sample weights are adjusted and the way the votes of the committee members are set.

Bootstrap aggregation, or bagging, is a technique proposed by Breiman (1996) that can be used with many classification methods and regression methods to reduce the variance associated with prediction, and they improve the prediction process. It is a relatively simple idea: many bootstrap samples are drawn from the available data, some prediction method is applied to each bootstrap sample, and then the results are combined, by averaging for regression and simple voting for classification, to obtain the overall

prediction, with the variance being reduced due to the averaging.

Breiman (2001) provides a general framework for tree ensembles called "random forests". Each tree depends on the values of a random vector sampled independently and with the same distribution for all trees. Thus, a random forest is a classifier that consists of many decision trees and outputs the class that is the mode of the classes output by individual trees. Random forests have been shown to give excellent performance on a number of practical problems. They work fast, generally exhibit a substantial performance improvement over single tree classifiers such as CART, and yield generalization error rates that compare favorably to the best statistical and machine learning methods. In fact, random forests are among the most accurate general-purpose classifiers available.

RandomSubSpace method [10]constructs a decision tree based classifier that maintains highest accuracy on training data and improves on generalization accuracy as it grows in complexity. The classifier consists of multiple trees constructed systematically by pseudo randomly selecting subsets of components of the feature vector, that is, trees constructed in randomly chosen subspaces.

RandomCommittee method builds an ensemble of randomizable base classifiers. Each base classifiers is built using a different random number seed (but based one the same data). The final prediction is a straight average of the predictions generated by the individual base classifiers.

NestedDichotomies.ND [11] is a meta classifier for handling multiclass datasets with 2-class classifiers by building a random tree structure.

MultiScheme is for selecting a classifier from among several using cross validation on the training data or the performance on the training data. Performance is measured based on percent correct (classification) or mean-squared error (regression).

MultiBoostAB [12] is an extension to the highly successful AdaBoost technique for forming decision committees. MultiBoosting can be viewed as combining AdaBoost with wagging. It is able to harness both AdaBoost's high bias and variance reduction with wagging's superior variance reduction.

LogitBoost [13] performs classification using a regression scheme as the base learner, and can handle multi-class problems.

Decorate [14] is a meta-learner for building diverse ensembles of classifiers by using specially constructed artificial training examples. Comprehensive experiments have demonstrated that this technique is consistently more accurate than the base classifier, Bagging and Random Forests.Decorate also obtains higher accuracy than Boosting on small training sets, and achieves comparable performance on larger training sets.

3 The New Algorithm ERAB

Below shows the new ensemble algorithm that generates 10 rules from one instance and assigns the 10 rules to 10 different base learners.

Given $(x_1, y_1), \quad ,(x_m, y_m)$ where $x_i \in X, y_i \in Y = \{-1, +1\}$

- For every instance
  - Generate a rule from this instance
  - Fix the first antecedent of the rule and vary the others to generate ten other rules from the same instance.
    - If ten rules can not be generated , copy the generated rules to form ten rules.
  - Assign the ten rules to the 10 different base learners $h_t(x)$.
- Output the final hypothesis

$$H(x) = \text{sign}(\sum_{t=1}^{T} h_t(x)).$$

4 Empirical Evaluation

This section will investigate the performance of our ensemble method in practice. In addition, our new algorithm ERAB will take a new rough set method to induce a rule for every instance. The new rough set method has three features: the first is that the new algorithm can be used in inconsistent systems. The second is its ability to calculate the core value without attributes reduction before. The third is that every example gives a rule and the core values are added first in rule generation process.

The new ensemble algorithm ERAB will be compared with the Bagging, AdaBoostM1, RandomForest, and other 17 ensemble algorithms. The total 21 algorithms are

(1) OrdinalClassClassifier taking J48 as base learner.
(2) NestedDichotomies.ND taking rough set algorithm as base learner.
(3) NestedDichotomies.DataNearBalancedND taking rough set algorithm as base learner.
(4) NestedDichotomies.ClassBalancedND taking rough set algorithm as base learner.
(5) MultiClassClassifier taking rough set method as base learner.
(6) MultiBoostAB taking rough set algorithm as base learner.
(7) ClassificationViaRegression taking trees.M5P as the base learner.
(8) AttributeSelectedClassifier taking trees.J48 as base learner.
(9) Dagging taking rough set algorithm as base learner,
(10) CVParameterSelection taking rough set algorithm as base learner.
(11) Decorate taking rough set algorithm as base learner.
(12) FilteredClassifier taking rough set algorithm as base learner.
(13) NestedDichotomies.ND taking trees.J48 as base learner.
(14) Grading taking rough set algorithm as base learner.

(15) RandomSubSpace taking rough set algorithm as base learner.
(16) Vote taking rough set algorithm as base learner.
(17) MultiScheme taking rough set algorithm as base learner.
(18) ERAB taking rough set algorithm as base learner.
(19) Trees.RandomForest taking J48 as base learner.
(20) Bagging 'taking rough set algorithm as base learner.
(21) AdaBoostM taking rough set algorithm as base learner.

The baseline learner are the 18th algorithm ERAB.

It should be mentioned that the algorithm 1, 8, 13, and 19 take J48 as their base learner. Although J48 performs better than rough set algorithm, the new ensemble algorithm ERAB performs better than J48 based ensemble methods (see Table 2 and Table 3). Additionally, the algorithm 7 takes M5P as its base learner, and M5P is a regression learner that performs better than rough set method, this causes the M5P based ensemble learner ClassificationViaRegression performs better than rough set based ERAB.

4.1 Data Sets

68 data sets were used to run the experiment. The 68 data sets were obtained from the repository of machine learning databases at UCI , see their characteristics in Table 1. Since the new rough set based algorithm does not fit for continuous data, continuous data should be discretized in advance. Some data sets were discretized by supervised discretization methods with WEKA and denoted as like hepatitis_dis, and some data sets can only be discretized by unsupervised discretization methods with WEKA and denoted as like heart_disease_undis. In WEKA, the java class weka.filters.supervised.attribute.AttributeSelection is used for supervised discretization and discretization is by Fayyad & Irani's MDL method (the default). The java class weka.filters.unsupervised.attribute.Discretize is used for unsupervized discretization, and discretization is by simple binning. The default value of bins is 10.

To analyze the performance of our ERAB approach, we conducted several experimental studies under the WEKA 3.6.0 framework.

The new algorithm is programmed with JAVA and embedded into WEKA 3.6.0. The experiment uses a ten-fold cross validation procedure that performs 10 randomized train and test runs on the dataset.

Since the experiment compares pairwise algorithms, t-test is used with 0.05 significance level.

Table 1. Data Sets

Dataset	Features	Classes	Cases
adult-stretch	4	2	20
audiology	69	24	224
australian	14	2	690
autos	24	7	205
balance	4	3	625
transfusion	3	2	748
breast-cancer-wisconsin-w	9	2	286
breast-cancer-wisconsin-d	8	2	699
breast-cancer	32	2	196
bridges	9	4	105
bridges-version2	11	7	107
cleve	11	2	303
cmc	9	3	1473
horse-colic	16	2	368
cpu_performance	8	8	209
crx	15	2	690
cylinder-bands	31	2	540
Dermatology	34	6	366
pima_diabetes	6	2	768
echocardiogram	11	3	132
ecoli	6	8	336
flags	26	8	194
flare_data1	12	6	323
flare_data2	26	8	194
german	19	2	1000
Glass	7	6	214
haberman	3	2	306
hayes-roth	4	3	132
heart-c	11	2	303
heart-h	12	2	294
heart-statlog	9	2	270
hepatitis	19	2	155
ionosphere	33	2	151
iris	4	3	150
labor	16	2	57
led7	7	10	3200
LED_24	24	10	1000
lenses	4	3	24
liver-disorders	6	2	345
lung-cancer	56	3	32
lymphography	18	4	148
mammographic	5	2	961
monks1	6	2	432
monks2	6	2	432
monks3-weka.	6	2	432
new_thyroid	5	3	215
postoperative	8	3	90
primary-tumor	17	21	339
promoter_gene	57	2	106
Robot_Failure_LP4	90	3	118
Robot_Failure_LP5	90	5	165
shuttle-landing	6	2	15
sonar	11	6	323
soybean	35	19	683
space_shuttle	2	3	23
spect	22	2	80
sponge	44	3	76
Teaching Assistant Evalu	4	3	151
tic-tac-toe	9	2	958
trains	32	2	10
urinary	6	4	120
vehicle	18	4	846
vote	16	2	435
vowel	11	11	900
wine	13	3	178
yeast	8	10	1484
yellow-small	4	2	20
zoo	16	7	101

4.2 Results

The classification performances were evaluated in terms of 8 performance measures, see the measures in Table 3. Table 2 shows the results in terms of “percent correct” where the Win/lost counts embodies the counts the other 20 ensemble algorithms against ERAB. For space saving the tie counts are removed.

Table 2 Comparison results in terms of “percent correct”

ata	(1)			(2)			(3)	
balance	75.50	66.86*	led7	68.84	72.94v	led7	68.84	72.94v
cleve	82.55	74.95*	promote	84.73	69.64*	promote	84.73	69.64*
Cylind	79.81	68.15*	monks2	45.14	57.18v	monks2	45.14	57.18v
flags	63.42	54.16*	sonar	75.05	59.55*	sonar	75.05	59.55*
LED_24	69.70	62.80*						
monks2	45.14	67.14v						
tic-t-t	92.69	84.55*						
vowel	84.95	65.56*						
(v/ /*)	(1/60/7)			(2/64/2)			(2/64/2)	

Data	(4)			(5)			(6)	
Derm	93.18	89.35*	balance	75.50	70.38*	flare2	71.49	73.92v
pima	77.09	74.49*	bridge2	63.55	52.27*	liver	65.53	61.45*
Glass	76.13	72.88*	led7	68.84	72.97v	monks2	45.14	68.51v
led7	68.84	72.88v	monks2	45.14	55.55v	promote	84.73	60.09*
monks2	45.14	55.55v	sonar	75.05	60.12*	sonar	75.05	60.98*
sonar	75.05	60.12*	promote	84.73	69.64*	soybean	92.38	88.86*
						vowel	84.95	82.83*
(v/ /*)	(2/62/4)			(2/62/4)			(2/61/5)	

Data	(7)			(8)			(9)	
balance	75.50	69.60*	balance	75.50	69.59*	adult-s	100.00	60.00*
b-c-w-d	74.21	67.16*	cmc	50.64	57.03v	audiolo	78.30	54.80*
cmc	50.64	55.60v	Cylind	79.81	68.15*	autos	85.33	59.36*
Cylind	79.81	67.59*	labor	96.67	78.67*	balance	75.50	72.46*
flare2	71.49	75.24v	led7	68.84	73.34v	b-c-w-w	96.42	87.98*
heart-c	81.47	86.09v	monks1	96.77	75.02*	b-c-w-d	74.21	58.55*
labor	96.67	75.67*	monks2	45.14	67.14v	bridges	71.36	54.36*
led7	68.84	73.28v	tic-t-t	92.69	78.08*	cmc	50.64	54.58v
LED_24	69.70	73.20v	vowel	84.95	73.84*	Cylind	79.81	71.48*
monks2	45.14	88.91v				Derm	93.18	75.41*
Teaching	44.96	52.25v				flare2	71.49	74.86v
vowel	84.95	79.39*				german	74.20	70.90*
						Hayes	76.48	47.64*
						labor	96.67	70.00*
						led7	68.84	73.38v
						lung-c	64.17	30.83*
						monks2	45.14	65.75v
						promote	84.73	42.18*
						LP4	91.67	76.29*
						sonar	75.05	55.74*
						soybean	92.38	74.51*
						tic-t-t	92.69	80.06*
						vowel	84.95	65.45*
						wine	98.30	93.79*
						zoo	92.18	80.18*
	(7/56/5)			(3/59/6)			(4/43/21)	

Data	(10)			(11)			(12)	
bridge2	63.55	49.45*	audiolo	78.30	69.96*	bridge2	63.55	49.45*
flare2	71.49	73.55v	b-c-w-d	74.21	62.61*	flare2	71.49	73.74v
led7	68.84	70.13v	bridge2	63.55	40.18*	led7	68.84	70.13v
monks2	45.14	57.18v	flare2	71.49	73.55v	monks2	45.14	57.18v
promote	84.73	69.64*	led7	68.84	70.13v	promote	84.73	69.64*
LP4	91.67	82.27*	LED_24	69.70	66.70*	LP4	91.67	82.27*
sonar	75.05	59.55*	liver	65.53	62.03*	sonar	75.05	59.55*
soybean	92.38	87.69*	monks2	45.14	57.18v	soybean	92.38	87.69*
vowel	84.95	81.72*	promote	84.73	65.82*	vowel	84.95	81.72*
			sonar	75.05	63.50*			
			soybean	92.38	88.42*			
			vote	96.10	92.42*			

	(3/59/6)			(3/56/9)			(3/59/6)	
Data	(13)			(14)			(15)	
balance	75.50	70.55*	autos	85.33	78.02*	adult-s	100.00	65.00*
cleve	82.55	74.95*	b-c-w-w	96.42	92.42*	balance	75.50	69.27*
cmc	50.64	54.11v	bridge2	63.55	51.27*	heart-c	81.47	58.24*
Cylind	79.81	68.15*	LED_24	69.70	64.80*	led7	68.84	43.34*
flare2	71.49	73.93v	monks2	45.14	59.73v	LED_24	69.70	58.60*
led7	68.84	73.09v	promote	84.73	64.09*	monks1	96.77	86.36*
monks2	45.14	67.14v	sonar	75.05	61.55*	monks2	45.14	67.14v
tic-t-t	92.69	84.55*	soybean	92.38	85.04*	tic-t-t	92.69	79.23*
vowel	84.95	80.40*	vowel	84.95	80.61*	vowel	84.95	74.65*
						yeast	58.15	36.25*
	(4/59/5)			(1/59/8)			(1/58/9)	
Data	(16)			(17)			(19)	
bridge2	63.55	49.45*	bridge2	63.55	49.45*	balance	75.50	70.56*
flare2	71.49	73.74v	flare2	71.49	73.74v	led7	68.84	73.16v
led7	68.84	70.13v	led7	68.84	70.13v	monks2	45.14	37.51*
monks2	45.14	57.18v	monks2	45.14	57.18v			
promote	84.73	69.64*	promote	84.73	69.64*			
LP4	91.67	82.27*	LP4	91.67	82.27*			
sonar	75.05	59.55*	sonar	75.05	59.55*			
soybean	92.38	87.69*	soybean	92.38	87.69*			
vowel	84.95	81.72*	vowel	84.95	81.72*			
	(3/59/6)			(3/59/6)			(1/65/2)	
Data	(20)			(21)				
flare2	71.49	73.55v	flare2	71.49	73.55v			
led7	68.84	71.97v	crx	85.65	81.59*			
monks2	45.14	61.60v	monks2	45.14	61.60v			
promote	84.73	65.91*	promote	84.73	65.91*			
sonar	75.05	61.10*	sonar	75.05	61.10*			
	(3/63/2)			(2/63/3)				

From Table 3, it can be found that only the algorithm 7 gains better performance on 5 measures among the all 8 measures. But the base learner of algorithm 7 is M5P, M5P's classification performance is better than rough set algorithm. The other 19 algorithms have better performances only on 1 or 2 measures among 8 measures. In short, the rough set based new ensemble algorithm shows excellent classification performance on 68 datasets compared with other 20 ensemble algorithms. Additionally, From Table 2 it can be found that the new algorithm has a property that it bias to the dataset the other algorithms does not, and vise verse. For example, ERAB perform worse than almost other algorithms on datasets "monks2" and "LED7", whereas perform better almost on "balance", "promotor-gene", "vowel", "sonar", and "soybean". On the others datasets ERAB shows almost the same performances except the algorithm 9 Dagging.

It should be mentioned that "win" means that an algorithm's measure is bigger than the ERAB's, whereas "lost" means that an algorithm's measure is smaller than ERAB's. And the bigger the "Mean absolute error" is, the worse the performance is, whereas the bigger the other 7 measures are, the better the performance is.

Table 3 Comparison results in terms of 8 performance measures

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)
1.	(40/26/2)	(4/57/7)	(4/58/6)	(6/57/5)	(38/27/3)	(5/46/17)	(35/31/2)	(33/29/6)	(50/15/2)	(4/57/6)
2.	(1/60/7)	(2/64/2)	(2/64/2)	(2/62/4)	(2/62/4)	(2/61/5)	(7/56/5)	(3/59/6)	(4/43/21)	(3/59/6)
3.	(4/45/19)	(0/49/19)	(1/47/20)	(1/46/21)	(4/54/10)	(6/57/5)	(15/51/2)	(5/48/15)	(11/48/9)	(0/42/26)
4.	(1/61/6)	(3/63/2)	(4/61/3)	(4/60/4)	(4/60/4)	(2/61/5)	(5/59/4)	(4/56/8)	(3/46/19)	(3/58/6)
5.	(2/57/9)	(2/63/3)	(3/60/5)	(1/63/4)	(1/62/5)	(2/63/3)	(2/64/2)	(3/58/7)	(1/49/18)	(1/62/5)
6.	(1/60/7)	(2/64/2)	(3/61/4)	(2/62/4)	(2/62/4)	(2/61/5)	(7/56/5)	(3/59/6)	(4/42/21)	(3/58/6)
7.	(4/58/6)	(6/60/2)	(4/61/3)	(4/61/3)	(6/59/3)	(5/59/4)	(8/57/3)	(6/53/9)	(6/44/18)	(6/59/3)
8.	(1/60/7)	(2/64/2)	(3/61/4)	(2/62/4)	(2/62/4)	(2/61/5)	(7/56/5)	(3/59/6)	(4/43/21)	(3/59/6)

Table 3 Comparison results in terms of 8 performance measures (continue)

	(11)	(12)	(13)	(14)	(15)	(16)	(17)	(19)	(20)	(21)
1.	(37/26/4)	(4/58/6)	(38/27/3)	(2/51/14)	(49/18/1)	(4/58/6)	(4/58/6)	(26/39/3)	(12/54/2)	(10/52/6)
2.	(3/56/9)	(3/59/6)	(4/59/5)	(1/59/8)	(1/58/9)	(3/59/6)	(3/59/6)	(1/65/2)	(3/63/2)	(2/63/3)
3.	(1/59/8)	(0/41/27)	(11/49/8)	(1/25/42)	(5/59/4)	(0/41/27)	(0/41/27)	(6/61/1)	(6/60/2)	(2/63/3)
4.	(3/59/6)	(3/59/6)	(3/59/6)	(2/57/10)	(1/50/11)	(3/59/6)	(3/59/6)	(1/65/2)	(3/62/3)	(2/63/3)
5.	(1/62/4)	(1/62/5)	(3/59/6)	(1/59/7)	(0/61/7)	(1/62/5)	(1/62/5)	(0/66/2)	(2/62/4)	(2/62/4)
6.	(3/55/9)	(3/59/6)	(4/59/5)	(1/58/8)	(0/59/9)	(3/59/6)	(3/59/6)	(1/65/2)	(3/63/2)	(1/64/3)
7.	(6/58/4)	(6/59/3)	(6/56/6)	(2/56/10)	(1/54/13)	(6/59/3)	(6/59/3)	(2/64/2)	(3/62/3)	(2/61/5)
8.	(3/55/9)	(3/59/6)	(4/59/5)	(1/58/8)	(1/58/9)	(3/59/6)	(3/59/6)	(1/65/2)	(3/63/2)	(1/64/3)

1. Mean_absolute_error
2. Percent_correct
3. Weighted_avg_area_under_ROC
4. Weighted_avg_F_measure
5. Weighted_avg_IR_precision
6. Weighted_avg_IR_recall
7. Weighted_avg_true_negative_rate
8. Weighted_avg_true_positive_rate

5 References

[1] Sotiris Kotsiantis, Combining bagging, boosting, rotation forest and random subspace methods. Artificial Intelligent Review.12(2010)

[2] Dietterich TG, Ensemble methods in machine learning. In: Kittler J, Roli F (eds) Multiple classifier systems. LNCS, 1857(2001), 1–15.

[3] Breiman L, Bagging predictors. Mach Learn, 24(1996):123–140.

[4] Freund Y, Robert E Schapire, Experiments with a new boosting algorithm. In: Proceedings of ICML'96 (1996) ,pp 148–156.

[5] Rodríguez JJ, Kuncheva LI, Alonso CJ Rotation forest. a new classifier ensemble method. IEEE Trans Pattern Anal Mach Intell 28(10)(2006):1619–1630.

[6] Ho TK. The random subspace method for constructing decision forests. IEEE Trans Pattern Anal Mach Intell 20(8)(1998):832–844.

[7] L. Breiman. Random forests. Machine Learning, 45:5–32, 2001.

[8] Freund, Y., Schapire, R.E.: A decision-theoretic generalization of on- line learning and an application to boosting. Journal of Computer and System Sciences 55 (1997) ,119-139.

[9] Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning: Data Mining, Inference, and Prediction. 2nd edn. Springer (2009).

[10] Tin Kam Ho . The Random Subspace Method for Constructing Decision Forests. IEEE Transactions on Pattern Analysis and Machine Intelligence. 20(8)(1998):832-844.

[11] Lin Dong, Eibe Frank, Stefan Kramer: Ensembles of Balanced Nested Dichotomies for Multi-class Problems. In: PKDD, 84-95, 2005.

[12] Geoffrey I. Webb (2000). MultiBoosting: A Technique for Combining Boosting and Wagging. Machine Learning. Vol.40(No.2).

[13] J. Friedman, T. Hastie, R. Tibshirani (1998). Additive Logistic Regression: a Statistical View of Boosting. Stanford University.

[14] P. Melville, R. J. Mooney: Constructing Diverse Classifier Ensembles Using Artificial Training Examples. In: Eighteenth International Joint Conference on Artificial Intelligence, 505-510, 2003.

Education Environment Comparative Studies: Actors And Resources In Distance

Edgard Thomas Martins Author[1], and Isnard Thomas Martins Co-author[2]
[1]Professor Dr. Universidade Federal de Pernambuco, Recife, PE, Brasil
[2]Professor Dr. Universidade Estácio de Sá, Rio de Janeiro, R.J, Brazil

Abstract - *The reading of information in a book replaced by the reading on a display represents the mere replacement of the boredom of reading books by reading on video. The constructivist philosophy of learning defines the cognitive process as the internal construction of the knowledge in terms of long-term memory, with the personal knowledge built through experience and interaction with the world, which he called active learning. Upon replacing the traditional technology of the "Computer-Teacher" binominal, where human contact is lost, students are isolated and the educational experience is passive, limited and alienated by environments where the WEB represents the means of interaction, enabling research and consultation of information and interaction with tutors and fellow students. The role of the teacher and inflexible rigid teaching methods has been questioned by educators who advocate cooperative learning, as opposed to the dictatorship of tax knowledge, practiced by traditionalists who believe that teaching authority, and assimilation of knowledge for apprentices are necessarily two sides of the same coin.*

Keywords: Ergonomics, Distance Education, Constructivism

1 Introduction

Comparative studies aim to identify, describe and evaluate conceptual and technological innovations focused on the continuous training of students with emphasis on distance education methodologies and tools. The learning process in web-based instructional models becomes a complex commitment to educator, just as the blackboard and the chalk paradigm had been in the past, when the blackboard and chalk were deemed the formal means of transfer of knowledge between the professor and the student in the traditional school environment. Breaking this paradigm is just the initial step to accept the "Invisible School" and the "remote" teacher, who can be "turned on and off" by the students, according to their time constraints and learning conveniences, which are as flexible as the selection of roadmaps on the development of the curriculum frameworks of what should be learned, and where to learn, with whom to learn and when to learn. Some Distance Education environments are being designed by web-designers and html language programmers, and it is expected that the final product, besides presenting a friendly interface and pleasant visual, can also present a pedagogical environment and ergonomic relationships with its users. The nature of the relationship between learners, teachers and mediators with these environments sets, however, the greater or lesser degree of interaction of actors with the pedagogical proposals offered, directly reflecting the level of interest of the user-customer, reflecting itself directly on the overall results achieved in the Distance Education Courses. The restructuring of the knowledge teaching and transfer process from teacher to student through new paradigms that determine a deep change in the manner and means by which this knowledge is constructed by the actors involved in the instructional environment that comprise the new education model [1]. The most promising direction, actually, translates the perspective of collective intelligence in the educational field.

Resources and technologies used must be efficient and ergonomically connected to the information and knowledge sources, which includes exploration of ideas from a research perspective, data acquisition and synthesis, and problem-solving models that are unique and specific to each subject. When working with multiple cultures, the ignorance of the subject can be a major obstacle to the success of the enterprise. A more humanized, intuitive, ergonomic and metaphorical man-machine interface should be progressively more available to the end user and, when added to the teachers and students' intelligences, we will have an educational model, integrating a powerful network endowed with intellectual, cognitive, collective and cooperative energy, where human essence must prevail above all [2].

The relevance of this subject is recognized in countries like Brazil, India and China, among other countries with continental dimensions, where the education-directed resources almost always lie below the real needs of the target audience and where the term "distance" gains, therefore, a certain importance in the range of concerns of the authorities related with the educational system, as well as teachers, researchers or institutions dedicated to education, specialization or training of its universe of users.

The consideration of the geographical and social factors related to the democratic context of Distance Education Systems arouse discussions where the Piaget's theories could significantly influence the usability, construction

and specifications of models focused on user's cultural process [3]. The undergraduate and continuing graduate education represent strong needs for getting qualified professionals in the Brazilian professional scene [4]. However, the huge geographical dispersion, added to reduced investments, greatly restricts the qualification and continuous refreshment activities of university-educated professionals who wish to improve their studies and create specialties in their respective educations. For this reason, short or medium term Distance Education courses can stand as a rational solution to the potential and idle demand and fill the gap represented by the education deficiency.

2 Contextualization

The huge drain of students in Web courses is widely known, especially due to the lack of usability of instructional tools and the lack of courses focused on the real needs of the target audience, which is not prepared for activities involving interaction with automated tools applied in Distance Education courses. The interactive technologies led to the development of the Computer-Aided Distance Education systems, questioning the pedagogical efficiency of the conventional educational system, exclusively based on the use of classroom, in a fully synchronous mode, requiring the physical and simultaneous presence of both teacher and student [5]. Distance Education comprises the education through printed or electronic media provided to people engaged in a learning process at a time and place that are different from the instructor or other learners'. Since early times, when paper and ink were used for mailing in England, the Distance Education incorporated several technologies to meet requirements of the different medias, such as radio, television and computers. Computers and communications have been used in different ways: chat groups, email and, more recently, the World Wide Web. In the evolution of the pedagogical tools made available from the systemic modeling, modern computational resources, combined with high-speed communication networks, have been applied, transforming the previous and restricted scenario of instructional environments in a new educational model. The new system allowed the asynchronous learning offer, delivering teacher's lessons through multimedia contents. The student keeps contact more often with the "invisible teacher", through chats and discussion forums. The learner interacts with the syllabus of distance education classes and participates in constructive discussions with colleagues who are geographically far from him, meeting them virtually through the educational distance education tools. Virtual environments supported by new information and communication technologies combine synchronous and asynchronous resources - these are the "multi synchronous" environments.

3 Preparation of an educational environment

The constructivist model is advocated by many authors as being the teaching method with the best cognitive usability, comprising a high degree of interaction in Distance education cooperative environments. However, it requires ergonomic tools for the actors involved, aiming at better operating and pedagogical results [6].

An efficient Distance education requires an extensive preparation, adaptation of traditional methods to the new learning environment, as well as a deep observation of the social and technical environment of the target audience of the educational material. According to Campos, the systemic environment can be rated according to the degree of interactivity allowed to it and initiative afforded to the student:

- High interactivity: It enables the unexpected discovery and the discovery with free exploration.
- Average interactivity: it enables the guided discovery.
- Low interactivity: it favors the directed learning, the inductive exposure and the deductive exposure.

The degree of interactivity with the instructional environment is also related with the learning theories that distinguish educational environments, with greater or lesser student's participation and control in the knowledge construction process. Masson listed the properties considered essential in the development of the pedagogical content of the distance education environment: ease reading, motivating and interesting, well structured, clear and defined objectives, practical and relevant, activities, accurate instructions, appropriate information density, nice presentation [7].

4 Results and discussion

The consideration of the geographical and social factors related to the democratic context of Distance Education Systems arouse discussions where the Piagetian theories could significantly influence the usability, construction and specifications of models focused on user's cultural process. Ritto and Machado Filho say that Piaget's ideas feed our reflection [8]. The Constructivism has continuously contributed to our formulations, being constantly present in our educational background. Freitag quotes the interference of the structure of classes in the very construction of the consciousness structures, differentiating individuals as to their competence to understand and assimilate the world. Often, individuals are unable to change the external material reality or assimilate changes that occur in the social structure, simply because they do not have cognitive condition to perceive, criticize or try to overcome them [9]. There is a play of forces (social

class, school level, language and biological maturity) that acts either in favorable or unfavorable manner on the psychogenetic development, and that ranges in accordance with the age variations and social constellation in which the individual is.

Thus, we see that there are sharp implications related with the usability factors, in face of the heterogeneity of the prospective target audience, which hardly ever is culturally and socially leveled. The perception of these differences becomes relevant when the system's developer works in the specifications and models an environment that will be applied in a distance education-oriented context [10].

5 Teacher and students as actors of the model - Changing factors

The student is the subject who builds his own knowledge. Shortfalls occur when the exchange of action between the subject and the object of knowledge fails. Considered as the father of constructivism, Jean Piaget, a Swiss psychologist, studied the development of the human being's intelligence, from birth to old age. Piaget examined the evolution of thinking, being it the most widespread pedagogical line among teachers who advocate the "Active School" instead of the "Traditional School". Based on the Piagetian theory, three basic principles regarding the learner can be inferred: Respect to his production, space to test his hypotheses, group work to facilitate his learning..

Lévy put the teacher's role as the epicenter of a problem related to qualitative changes in the learning process to an emphatic discussion, where he sought to establish new paradigms for knowledge and learning acquisition. In response to it, there was the restructuring of the teaching process and the transfer of knowledge from teacher to student through new paradigms that determined a deep change in the manner and means through which this knowledge is constructed by the actors involved in the instructional environment. Teachers should be trained in creating attractive and competitive educational content, presenting quality results backed by support for the construction of educational tours, web designers, literary editors, software engineers, project managers and experts in industrial processes, librarians and art managers [11].

The institution that promotes a distance instructional environment must prepare itself to compete with cinema, newspaper, radio and television. In the technical and pedagogical sphere, it is observed the redirection of methods brought to the participating student, emphasizing the ergonomics of the tools used for distance learning. In models of non-ergonomic automated instructional environments, which feature a significant part of the free courses offered over the Internet, the content provider represents the remote "teacher", using the computer as a unique vehicle for the transfer of knowledge and means of control of participation, assessment of results, tests and application of educational contents. Kanuka cites that the complexity related to the control and motivation of a distance instructional environment provided with interactivity, efficiency and pedagogical quality should not be underestimated. If, on one hand, the innovations and tools offered in the new instructional designs practices surprise the Teacher-Facilitator's practices, on the other hand, this diversified model presents such a complexity degree that requires a reflection on the methodology to be employed, resulting in the reorganization of tasks to be performed. This also occurs with the Teacher-Facilitator interface with the assessment methodology on content generated by student-users [12].

The professional responsible for building this mediating interface arises in this scenario and is defined as the art/science capable of creating an educational environment, using specific technological materials that will allow the student to learn the educational content, enabling him to perform certain specific tasks. The instructional design is based on research in the cognition, educational psychology and problem-solving practice areas. Kanuka alerts to the growing use of instructional designers as pedagogical specialists involved in e-learning activities, emphasizing the importance of these professionals' mastering the design and development of pedagogical contents referring to each subject transported to the virtual environment . When instructional designers are employed as pedagogical experts, but are not specialized in the content addressed in the virtual environment, or online tutors specialized in educational contents, but with no domain of pedagogical practices, the result of the educational model will present an undesirable bifurcation dividing content and pedagogy. Thus, the knowledge of the pedagogical content shall integrate itself to the functional competence of the instructional designers, incorporating this important feature to their professional profiles.

The changes noticed in the role of teachers and students in this scenario of changes. The former fails to act merely as a transmitter of knowledge to become the facilitator and provocateur of the learning process of this relationship. Besides offering a basic content, the teacher must also assume his/her presence as support for the development of ideas and interactions with the disseminated content. The teacher, while an actor in this scenario, must be treated carefully, after decades of rigidity of presence-based classroom, starring in a new role in the forms and means of communication of digital spaces. However, the success or failure of the technical and ergonomic model is conditioned to factors that can decisively contribute for the efficient achievement of their pedagogical goals. This set of factors is subdivided into four groups: context of use (physical, emotional, functional factors); expertise domain (influence on system, influence on user); user's characteristic (computing experience, frequency of computer use, knowledge of content addressed in hypertext, reading ability, cognitive factors); characteristic of browsing tools (forms as they are related to skip zones, browsing

features, audible and visual receptive feedback, level of confidence to go back to the homepage).

6 Conclusions

The reading of information in a book replaced by the reading on a display represents the mere replacement of the boredom of reading books by reading on video. According to Bostock (1996), the constructivist philosophy of learning defines the cognitive process as the internal construction of the knowledge in terms of long-term memory, with the personal knowledge built through experience and interaction with the world, which he called active learning. Upon replacing the traditional technology of the "Computer-Teacher" binominal, where human contact is lost, students are isolated and the educational experience is passive, limited and alienated by environments where the WEB represents the means of interaction, enabling research and consultation of information and interaction with tutors and fellow students.

The role of the teacher and inflexible rigid teaching methods has been questioned by educators who advocate cooperative learning, as opposed to the dictatorship of tax knowledge, practiced by traditionalists who believe that teaching authority, and assimilation of knowledge for apprentices are necessarily two sides of the same coin. Resources and technologies used must be efficient and ergonomically connected to the information and knowledge sources, which includes exploration of ideas from a research perspective, data acquisition and synthesis, and problem-solving models that are unique and specific to each subject. When working with multiple cultures, the ignorance of the subject can be a major obstacle to the success of the enterprise. A more humanized, intuitive, ergonomic and metaphorical man-machine interface should be progressively more available to the end user and, when added to the teachers and students' intelligences, we will have an educational model, integrating a powerful network endowed with intellectual, cognitive, collective and cooperative energy, where human essence must prevail above all .

7 References

[1] Killen, J.D., Fortmann, S. P., Schatzberg, A. F., Hayward, C., Sussman, L., Rothman, M., Strausberg, L., et al.(2000). Nicotine patch and paroxetine for smoking cessation. Journal Consulting and Clinical Psychology, 68, 883-889.

[2] Isnard Martins. Estudo Ergonômico de Ambientes Instrucionais na Internet. Rio de Janeiro, PUC-Rio, Dissertação de Mestrado em Ergonomia, 2001, pp. V-VI.

[3] Niskier, A Ciência e Tecnologia do Estado do Rio de Janeiro. Cedibra, Rio de Janeiro, 1968. Cap 5. P. 38-41.

H. Fuks, C. Lucena. A Educação na Era da Internet, Clube do Futuro, Rio de Janeiro, 2000. C.2-6. P 67-81

[4] Azevedo W. Muito Além do Jardim de Infância: O Desafio do Preparo de Alunos e Professores Online,. London, Routledge,1998.Lastaccessed 5 january, 2012 http://pt.scribd.com/doc/49666884/Muito-para-Alem-do-Jardim-

[5] Campos, G.. Vantagens, Desvantagens e Novidades da Ead. Rio de Janeiro, 2000. Last Acessed 15 may 2010. http://www.timaster.com.br/ext_materia.asp? codigo=205

[6] Ritto, N. Machado Filho. O Caminho Da Escola Virtual - Um Ensaio Carioca, Rio de Janeiro, Edições Consultor, 1995. Cap.5. P. 83

[7] Masson, J. Sur La Satisfaction Des Etudiants Dans Un Contexte De Formation À Distance: La Télé-Université. Paris. 1988, Au Press, Vol 3, No 2

[8] Ritto, N. Machado Filho. O Caminho Da Escola Virtual - Um Ensaio Carioca, Rio de Janeiro, Edições Consultor, 1995. Cap.3. P. 61

[9] Freitag, B.. Sociedade e Consciência - Um Estudo Piagetiano na Favela e na Escola, São Paulo, Cortez Editora, 1984. Cap.6. P. 109

[10] Falkembach, G. A. M.. Adaptive hypermedia: An option for the development os educacional systems in order to getting more effective learning. 2000. Last acessed 12 march 2001 http://www.conex.com.br/user/fabrizio/construtivismo.htm

[11] Lévy, P. . Cybercultura, São Paulo, Editora 34, 1999. Cap.2-4. P. 22-54 . Lévy, P. . Inteligência Coletiva. Edicoes Loyola, 2007, Cap 3. P.39-40.

[12] Kanuca, S. I. (2008). Human psychology (10th ed.). Boston: McGraw-Hill Higher Education

SESSION

XIV TECHNICAL SESSION ON APPLICATIONS OF ADVANCED AI TECHNIQUES TO INFORMATION MANAGEMENT FOR SOLVING COMPANY-RELATED PROBLEMS

Chair(s)

Dr. David de la Fuente
University of Oviedo
Spain
Dr. Jose A. Olivas
University of Castilla
Spain

On improving failure mode and effects analysis (FMEA) from different artificial intelligence approaches.

Javier Puente[1], **Paolo Priore**[1], **Isabel Fernandez**[1],
Nazario García[1], **David de la Fuente**[1], **Raul Pino**[1]
[1] Business Administration Department, University of Oviedo, Gijon, Spain

Abstract - *This study describes the Failure Mode and Effects Analysis (FMEA) from the point of view of different artificial intelligence approaches. After discussing the main drawbacks of the traditional methodology and summarize the main techniques recommended for its improvement in the recent literature, three techniques of Artificial Intelligence are compared: a fuzzy inference system (FIS), a case reasoning based method (CBR) and a vector support machine based method (VSM). From the results of this study we conclude that the best approach to properly classify the causes of risk of a system or service is the fuzzy inference system, method that, in addition, allows to overcome most of the drawbacks associated with the traditional methodology*.

Keywords: Failure Mode and Effects Analysis (FMEA), Fuzzy Inference Systems, Case Based Reasoning, Vector Support Machine

1 Introduction

Failure mode and effects analysis (FMEA) first emerged from the aerospace industry in the 1960s and then spread to the manufacturing scope. FMEA is an efficient tool to identify and remove known and potential failures in a system which allows establishing their causes of appearance and preventing their further occurrence [1]. It has been extensively used in many industrial sectors -aerospace, automotive, nuclear, electronics, etc.- [2].

The paper is organized as follows. In Section 2 the traditional method is explained, pointing out the main criticisms on it and outlining the main methodologies employed to gain more efficiency in the method. In Section 3 the foundations of the three methodologies proposed in this work and their comparative results are described. Finally, the conclusions of the study are exposed.

2 The FMEA method

The FMEA method is based on a systematic brainstorming session aimed to uncover the failures that might occur in a system or process [3]. After that, critical analysis is performed on these failure modes taking into account the valuation (between 1 to 10) of three risk indexes (See Table 1): occurrence (O), detection (D) and severity (S) - associated to the likelihood of occurrence, no detection and severity of each failure respectively- . FMEA searches for prioritising the failure modes of a system in order to assign the available and limited resources to the most serious risk items. Generally, the prioritisation is determined through the risk priority number (RPN), which is obtained multiplying the indexes O, S and D of each failure.

FMEA determines the critical level (or risk score) of these failures and proceeds to put them in order, reviewing each design detail and proposing the relevant modifications. The most critical failures (related to higher RPNs) will head the ranking, and will therefore be considered first during design review or during control actions taken to minimise the likelihood of such failures occurring. Those modes of failure with high rates O or S are also considered critical although their RPN is not high. Finally, RPNs should be recalculated after the corrections to check its efficiency.

2.1 Shortcomings about traditional FMEA

Despite the good results obtained with the traditional FMEA methodology, numerous investigations have shown different problems in its application. Table 2 shows the most relevant and frequently cited according to the last review on the issue [4].

2.2 Techniques used to improve the traditional FMEA

Among the techniques used to overcome the drawbacks described in the previous paragraph those included in the five categories and summarized in Table 3 stand out.

As shown in Table 3, the category of method most frequently applied to FMEA was found to be AI (32 times out of 80; that is 40.0% of all the reviewed papers) and within it, the most popular approach is Fuzzy Rule-based system (29 out of 32, which represent 36,25% of the 80 analyzed papers). This can be caused by the different advantages brought by the fuzzy inference systems [5], [6], [7], [8], [9]:

Table 1. Explanation of the risk indexes (O), (D) and (S).

	Score	Likelihood of occurrence
Remote	1	0
Low	2 3	1/20000 1/10000
Moderate	4 5 6	1/2000 1/1000 1/200
High	7 8	1/100 1/20
Very High	9 10	1/10 1/2
O - Occcurrence Index		

	Score	Likelihood of no detection
Remote	1	0-5
Low	2 3	6-15 16-25
Moderate	4 5 6	26-35 36-45 46-55
High	7 8	56-65 66-75
Very High	9 10	76-85 86-100
D - No-detection Index		

	Score
The customer will not perceive (VL)	1
Small nuisance (L)	2 3
No satisfaction (M)	4 5 6
High level of No satisfaction (H)	7 8
Serious safety consecuences (VH)	9 10
S – Severity Index	

Table 2 .The major shortcomings of FMEA. (Extracted from [4])

	Shortcomings	Frecuency of citation	% citation
1	The relative importance among O, S and D is not taken into consideration	45	24,32%
2	Different combinations of O, S and D may produce exactly the same value of RPN, but their hidden risk implications may be totally different	33	17,84%
3	The three risk factors are difficult to be precisely evaluated	21	11,35%
4	The mathematical formula for calculating RPN is questionable and debatable	14	7,57%
5	The conversion of scores is different for the three risk factors	13	7,03%
6	The RPN cannot be used to measure the effectiveness of corrective actions	12	6,49%
7	RPNs are not continuous with many holes	10	5,41%
8	Interdependencies among various failure modes and effects are not taken into account	10	5,41%
9	The mathematical form adopted for calculating the RPN is strongly sensitive to variations in risk factor evaluations	9	4,86%
10	The RPN elements have many duplicate numbers	9	4,86%
11	The RPN considers only three risk factors mainly in terms of safety	9	4,86%

Table 3. Main risk evaluation methods in FMEA. Frequency of use. (Adapted from [4])

Method	n	%
Multi-Criteria Decision Making	18	22,50%
Mathematical Programming	7	08,75%
Artificial Intelligence	32	40,00%
Hybrid	9	11,25%
Others	14	17,50%

- "Ambiguous, qualitative or imprecise information, as well as quantitative data can be used in criticality/risk assessment and they are handled in a consistent manner".
- "It permits to combine the occurrence, severity and detectability of failure modes in a more flexible and realistic manner".
- "It allows the failure risk evaluation function to be customized based on the nature of a process or a product".
- "The fuzzy knowledge-based system can fully incorporate engineers' knowledge and expertise in the FMEA analysis and substantial cost savings can thus be realized".

As far as the disadvantages of this method is concerned, the difficulty of adapting the model to real-life circumstances is often argued, basically for design reasons (need to define a large number of rules and the membership functions in the model variables). On the other hand, it is suggested that the new methods on FMEA should obviate the subjective weighting of risk factors involved in the model (weights which are the most frequently used in the analyzed literature).

3 Artificial intelligence approaches for the FMEA improvement

In this section the theoretical foundations of three techniques that attempt to improve some of the above- mentioned drawbacks. Nine risk categories associated to the risk priority numbers given by the traditional methodology have been considered (see Table 4).

The classification error from these methodologies for the cases corresponding to all the integer inputs in the range of 1-10 for the "O", "D" and "S" input variable will be calculated for comparison purposes. The classification error will be measured as the percentage of cases misclassified for the used test set (800 cases) -lack of concordance between the output assigned by the method and the correct output of Table 4-.

Table 4: Risk priority categories corresponding to different intervals of risk priority numbers

RPN (Class interval)	Class Score	Category (RPC)
0-50	25	VL
50-100	75	VL-L
100-150	125	L
150-250	200	L-M
250-350	300	M
350-450	400	M-H
450-600	525	H
600-800	700	H-VH
800-1000	900	VH

3.1 FMEA with Fuzzy Inference Systems

Fuzzy inference systems are based on the theory of fuzzy sets [10], and allow an uncertainty component to be incorporated into models, making them more effective in terms of approximating to reality [11]. Linguistic variables can be used to handle qualitative or quantitative information, so that its content can be labelled taking words from common or natural language as values. This contrasts with numeric variables, which can only take numbers as values [12]. All decision problems require a knowledge base provided by an expert who is able to explain how the system works through a set of linguistic rules involving the system's input and output variables; the system's variables, that is, the form and range of the labels for each variable, must therefore be defined in fuzzy form. Mamdani Fuzzy Inference Systems depend on this to model systems in a process which has five stages: the fuzzification of the input variables, the application of fuzzy operators (AND/OR) to each rule's antecedent, the implication process from each rule's antecedent to consequent, the consequent aggregation process, and the defuzzification process [13].

A fuzzy inference system to develop the FMEA has been designed based on qualitative rules using MATLAB 6.5 – Toolbox 'Fuzzy' (v. 2,0). The system assigns a risk priority class (RPC) to each of the causes of failure in an FMEA, depending on the importance given to the three already mentioned indexes "O", "D" and "S" (which will be the input variables in the decision system –with integer scores between 1 and 10-). The output variable of the decision system is the risk priority category (RPC) assigned to the cause of a failure. Here, a division of the traditional domain of the Risk Priority Number (RPN) from 1 to 1000 into nine class intervals was opted for, so that each of the class intervals has a different RPC (very low: "VL", between very low and low: "VL-L", Low: "L", ... very high: "VH"). Figure 1 shows the potential linguistic labels to be assigned to all these variables.

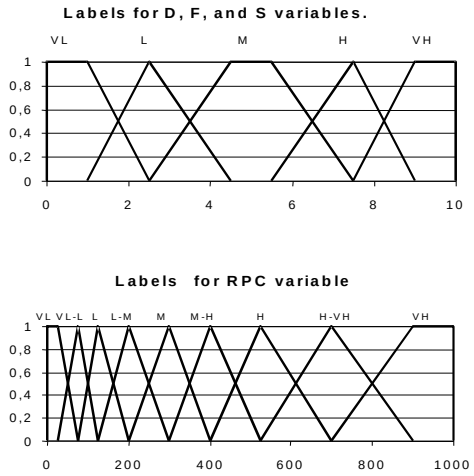


Figure 1. Labels for input and output variables

As each of the three input variables can be given one of five categories or classes, we have as many as 125 rules at our disposal to assign the RPCs to each of the causes of failure analysed in the FMEA. The rules, given by the expert about the RPC evaluation, are shown in Figure 2.

The rule structure in the system is of the type: "if ((O=M) & (D=VL) & (S=H)), then (RPC=H)"; this would mean that if "occurrence" is moderate, "no-detection" is very low, and "severity" is high for a cause of failure, then the risk priority category should be high.

The results obtained for different configurations of parameters in the designed fuzzy inference system (FIS) are shown in Table 5. Three of the five configurations tested, produce 0% classification errors, highlighting the goodness of the FIS method to carry out the evaluation of risk categories in the FMEA.

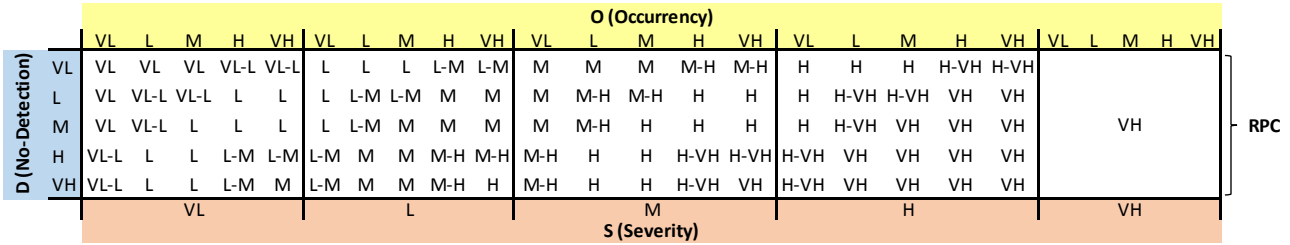


Figure 2. Rule Base of the decision system proposed
VL: Very Low, L: Low, M: Medium, H: High, VH: Very High

Table 3: Classification errors for different configuration parameters of the FIS

AND Method	Implication Method	Aggregation Method	Errors (%) for different defuzzification methods				
			Centroid	Bisector	MOM*	LOM*	SOM*
MIN	MIN	MAX	33%	8.9%	0%	0%	0%

** MOM: middle of maximum, LOM: largest of maximum, SOM: smallest of maximum.*

3.2 FMEA with Case Based Reasoning - C4.5.

C4.5, a learning system based on examples that produce decision trees or sets of decision rules [14], is the second of the systems proposed in this study to assign criticality to error causes in FMEA.

For the case in question, three attributes corresponding to occurrence, non-detection, and severity indexes were used in each training example. All were defined as continuous attributes in the 1-10 domain and each example's class corresponds to the Risk Priority Category (RPC) given to each failure cause (maintaining the structure of nine possible categories: "VL"(1), "VL-L"(2), ... , "VH"(9)). The training examples were 20% of all possible discrete cases of the input variables ("O", "D", and "S") with the corresponding output RPC. Thus, 200 cases were selected from the base total of 1000 discrete traditional possible cases.

Once the decision tree had been formed from the training examples with C4.5, the 1000 possible discrete inputs for the 'O', 'D' and 'S' indexes were processed to determine the criticality assigned by the algorithm to each of them and compare results with the system initially proposed. The obtained results showed a 14,5% classification error, only superior in efficiency to the FIS when this latter method uses the center of gravity as a method of defuzzyfication.

A comparison with the results from the other systems shows that the C4.5 system proposed gives rather poor classification compatibility compared to the initial system (note that certain fuzzy system parameter combinations gave zero classification error).

3.3 FMEA with Support Vector Machines

Support vector machines (SVMs) [15] were originally designed for binary classification. Let (x1, y1), (x2, y2), … , (xn, yn) be a group of data belonging to Class 1 or Class 2, where xi□RN and the associated labels be yi=1 for Class 1and -1 for Class 2 (i=1, … , n). The formulation of SVMs is as follows:

$$Min \quad \frac{1}{2} w^T w + C \sum_{i=1}^{n} \xi_i \qquad \text{(Eq. 1)}$$

subject to the constraints:

$$y_i (w^T \phi(x_i) + b) \geq 1 - \xi_i \quad i = 1,..., n$$

$$\xi_i \geq 0 \quad i = 1,..., n$$

where w is the weight vector; C is the penalty weight; ξ_i are non-negative slack variables; b is a scalar, and xi are mapped into a higher dimensional space by a non-linear mapping function ϕ . Mapping function ϕ needs to satisfy the following equation:

$$k(x_i, x_j) = \phi(x_i)^T \phi(x_j)$$

Where $k(x_i, x_j)$ is called kernel function.

Minimizing $\frac{1}{2} w^T w$ implies that SVMs tries to maximise $\frac{2}{\|w\|}$, which represents the margin of separation between both classes. The data that satisfy the equality in Eq. (1) are called support vectors. Moreover, by adding a set of non-negative Lagrange multipliers α_i and β_i to generate the Lagrangian, the upper- mentioned constrained optimization problem can be worked out with the dual form shown below:

$$Max \quad \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j k(x_i, x_j)$$

subject to the constraints:

$$\sum_{i=1}^{n} \alpha_i y_i = 0$$

$$0 \le \alpha_i \le C \quad i = 1,..., n$$

Having obtained the support vectors (SVs), the decision function for an unseen data (x) is as follows:

$$y = sign \left\{ \sum_{SVs} \alpha_i y_i k(x, x_i) + b \right\}$$

The use of SVMs in FMEA essentially implies a multi-class classification problem. This study uses the one-against-one method to extend the binary SVMs to generate the multi-class scheduler since this method is more suitable for practical use than other methods [16]. It was introduced in [17], and the first use of this strategy on SVMs was in [18] and [19].

One of the steps that precedes the application of the proposed method to the different sets of examples is that of standardising the attributes so that their maximum and minimum values are one and zero respectively. In the same way, in this study, the radial basis function (RBF) and the polynomial function have been used as kernel functions. After several preliminary tests, it has been decided to make use of the RBF Kernel since it is the one that shows a better performance. Furthermore, by employing the grid search technique on the examples, the best performance for the SVMs is obtained when C=8 and σ=0.03. The program that has been used to perform the upper-mentioned study is LIBSVM [20]. The test error obtained with this methodology from the 800 examples is 21,63% lagging behind both the C.4.5 and the FIS methods.

Nevertheless, all the employed methodologies allow overcoming some of the drawbacks mentioned in Table 2. In particular, the best of the three methodologies, FIS, not only gets an excellent classification error but also it permits circumvent the shortcomings 1, 2, 4, 5, 7, and 9 of the above-mentioned table, which corroborates the suitability of the method for the evaluation of risks with the FMEA methodology.

4 Conclusions

This study reviews the traditional FMEA methodology, analyzing their main drawbacks as well as the main categories of tools used in their assessment. These evidences have been observed in the unique and recent review written on the FMEA method in the academic literature.

In view of these drawbacks, a proposal is made to structure expert knowledge to assign risks to the system failure causes in the form of qualitative decision rules whereby a risk priority category (RPC) can be assigned to each cause of failure. The method being proposed here effectively mitigates one of the main criticisms imputed to the traditional model, since the structure of the proposed rule system allows to emphasize one index over the rest (in this case, the severity index "S" associated to a cause of failure). Furthermore, the method being proposed is flexible and easily implemented, making it a useful new tool for most risk classification problems.

In addition, recent techniques of artificial intelligence have been tested to evaluate categories of risk priority of different and potential system or service failure causes. In particular a Fuzzy Inference System (FIS), a Case Based Reasoning system (CBR) developed with the program C4.5 and a Vector Machine Support method of classification (VMS) have been analyzed. The results show that none of the studied methods exceeds in efficiency the behavior of the FIS, which is moreover able to overcome most of the drawbacks targeted in the recent literature regarding the traditional methodology.

5 References

[1] Sankar, N. R., & Prabhu, B. S. (2001). Modified approach for prioritization of failures in a system failure mode and effects analysis. International Journal of Quality & Reliability Management, 18, 324–336.

[2] Chang, K. H., & Cheng, C. H. (2011). Evaluating the risk of failure using the fuzzy OWA and DEMATEL method. Journal of Intelligent Manufacturing, 22, 113–129.

[3] Clifton J.J. Risk Prediction, in Keller, A.Z. & Wilson, H.C. (Eds.), Disaster Prevention, Planning and Limitation Unit, University of Bradford and the British Library. 1990.

[4] Liu H-C, Liu L. & Liu N. (2013). Risk evaluation approaches in failure mode and effects analysis: A literature review. Expert Systems with Applications 40, 828–838

[5] Bowles, J. B., & Peláez, C. E. (1995). Fuzzy logic prioritization of failures in a system failure mode, effects and criticality analysis. Reliability Engineering & System Safety, 50, 203–213.

[6] Braglia, M., Frosolini, M., & Montanari, R. (2003). Fuzzy criticality assessment model for failure modes and effects analysis. International Journal of Quality & Reliability Management, 20, 503–524.

[7] Sharma, R. K., Kumar, D., & Kumar, P. (2005). Systematic failure mode effect analysis (FMEA) using fuzzy linguistic modelling. International Journal of Quality & Reliability Management, 22, 986–1004.

[8] Tay, K. M., & Lim, C. P. (2010). Enhancing the failure mode and effect analysis methodology with fuzzy inference techniques. Journal of Intelligent & Fuzzy Systems, 21, 135–146.

[9] Xu, K., Tang, L. C., Xie, M., Ho, S. L., & Zhu, M. L. (2002). Fuzzy assessment of FMEA for engine systems. Reliability Engineering & System Safety, 75, 17–29.

[10] Zadeh, L.A. "Fuzzy Sets", Information and Control. Vol. 8, pp.338-53. 1965.

[11] Kaufmann, A.; Gupta, M. "Introduction to Fuzzy Arithmetic. Theory and Applications" .Van Nostrand Reinhold. 1991.

[12] Driankov, D.; Hellendoorn, H.; Reinfrank, M. "An Introduction to Fuzzy Control. 2nd Edition". Springer. 1996.

[13] Cox, E. The Fuzzy Systems Handbook. Academic Press, Inc, London. 1994.

[14] Quinlan, J.R. C4.5: Programs for Machine Learning, Morgan Kaufmann Publishers, San Mateo, CA. 1993.

[15] Cortes, C., and V. Vapnik. 1995. Support-vector network. Machine Learning 20:273-297.

[16] Hsu, C.W., and C.J. Lin. 2002. A comparison of methods for multi-class support vector machines. IEEE Transactions on Neural Networks 13:415-425.

[17] Knerr, S., L. Personnaz, and G. Dreyfus. 1990. Single-layer learning revisited: a stepwise procedure for building and training a neural network in Neurocomputing: Algorithms, Architectures and Applications, J. Fogelman, Ed. New York: Springer-Verlag.

[18] Friedman, J. 1996. Another approach to polychotomous Classification. Dept. Statist., Stanford Univ., Stanford Univ., Stanford, CA. [Online]. Available: http://www.stat.standord.edu/report/friedman/poly.ps.Z

[19] Krebel, U. 1999. Pairwise classification and support vector machines in Advances in Kernel Methods-Support Vector Learning, B. Shölkopf, C.J.C. Burges and A.J. Smola, Eds., 255-268. Cambridge, MA: MIT Press.

[20] Chang, C.-C. and C.-J. Lin. 2011. LIBSVM: a library for support vector machines. ACM Transactions on Intelligent Systems and Technology 2(3):27:1--27:27.

ILS with Multiple Variable Neighbourhood Descendent for the Vehicle Routing Problem with Multiple Trips

Gómez A., Argüelles D., Gallego I., López J., Parreño J., De La Fuente D.

Contact Author's Name: Alberto Gómez
Complete mailing address of Contact Author: Alberto Gómez, Escuela Politécnica de Ingeniería de Gijón, Campus de Viesques s/n, 33204 Gijón (Asturias), Spain.
E-mail address of Contact Author: albertogomez@uniovi.es

Abstract - *This paper describes a new approach to solving the vehicle route problem with Multiple Trips. The proposed algorithm (IVNDS) is based on the Iterative Local Search meta-heuristic which uses two consecutive Variable Neighbourhood Descendent (VND) procedures in the improvement phase. The VND procedures are differentiated by the search spaces they explore. The first VND performs moves in routes of the same vehicle and the second one relocates customers to routes of different vehicles. The algorithm has been tested against well-known benchmark instances and the results obtained are competitive with previous approaches found in the literature.*

Keywords: meta-heuristic, vehicle routing, iterative local search, variable neighbourhood descendant

1 Introduction

Due to economic growth and globalization transportation has to become more and more efficient for society. It is not uncommon that 20% of the final cost of a product is directly related to movement costs [1]. Being able design efficient routes will reduce costs and increase the enterprise's competitive advantage in the markets.

The basic Vehicle Route Problem (VRP) aims to design good routes to distribute goods and services to final customers. These routes start from a depot where the goods are loaded into vehicles. Then, the vehicles visit customers and return to the depot of origin following the route designed. Usually the vehicles have load limitations; in this case the problem becomes a Capacitated Vehicle Route Problem (CVRP). Other more general problems studied were Vehicle Route Problem with Multiple Depots (MDVRP), customers with time windows (VRPTW) or heterogeneous vehicles (HVRP). These problems have been intensively studied for the last four decades; however, few studies have considered that vehicles may take more than one route over a working day. This paper focuses on the Vehicle Route Problem with Multiple Trips (VRPMT). The VRPMT is a CVRP generalization which allows vehicles to take more than one route. This approach may reduce the number of vehicles required for the solution and costs.

As the VRP is an NP-Hard problem [2], all of its generalizations are NP-Hard. Due to the computational complexity of the VRPMT, most studies use heuristics to solve instances of realistic size in a reasonable running time.

The first study of the VRPMT was made by Salhi [3]. Restricted to double-trips, the routes were assigned to vehicles using a matching algorithm within a refinement. A modification to the savings algorithm of Clarke and Wright [4] and the Best Fit Decreasing (BFD) heuristic to assign the routes to vehicles was suggested by Fleischmann [5]. Taillard et al. [6] used a two-phase heuristic. The first starts building a set of feasible CVRP routes with Tabu Search and the second selects a subset of routes for the bins packed into feasible working days using the BFD heuristic. The solution obtained is improved by Tabu Search and then its routes are added to the set of routes. Taillard et al. [6] established the benchmark instance which is still used to compare new methods.

Brandão and Mercer [7] solved a real VRPMT problem with time windows, a heterogeneous fleet and vehicle restrictions to visit some customers by using Tabu Search. Later, they proposed a two-phase tabu algorithm [8] to solve a simplification of their previous study [7]. It builds an initial solution with a nearest insertion heuristic to build a VRPMT solution where all routes are load-feasible, but it can be infeasible with regard to distance. Next, an improvement phase to reduce overtime is used before undertaking a Tabu Search to reduce routing costs, with insert and swap moves and long-term memory for diversification. They consider routes with overtime as feasible routes. In this case, a penalty is added to the fitness function. Petch and Salhi [9] suggested a faster multi-phase constructive heuristic. The first phase starts building a set of CVRP feasible solutions and improves each solution with two-optimum and three-optimum procedures. Then, each CVRP solution is transformed to a VRPMT solution, filling the vehicle with the longest route that does not exceed the constraints. Unlike other approaches, the algorithm ends when a feasible solution is found.

Olivera and Viera [10] defined an algorithm based on the Adaptive Memory Programming principle, similar to Taillard et al. [6], obtaining more feasible solutions than other previous approaches. The most significant difference was that

this new approach [10] worked with non-feasible solutions. Salhi and Petch [11] suggested another population-phase meta-heuristic, a Genetic Algorithm. Other broader problems that involve VRPMT were also studied, such as VRPMT with time windows by Battarra et al. [12] or Azi et al. [13], the Profitable VRPMT by Chbichib et al. [14] or the site-dependent multi-trip periodic vehicle routing problem (SDMTPVRP) studied by Alonso et al. [15].

Recently, Cattaruzza et al. [16] proposed a Memetic Algorithm with very good results. Their algorithm is able to find a new feasible solution in the benchmark instances. The population is evaluated with an adapted Split procedure proposed by Prins [17]. A Combined Local Search with non-improved movements and route swapping are also used to reduce infeasibility penalties.

The main contribution of this paper is a new approach to solving the VRPMT with an excellent performance. The new algorithm, IVNDS, is based on the Iterative Local Search (ILS) [18], where the local search has been substituted by two consecutive Variable Neighbourhood Descendent (VND) [19] heuristics and a new intensification/diversification technique is used to improve the performance of the ILS. The first VND reduces the length of the routes through moves between routes performed by the same vehicle, also known as a tour. This reduces the total length of the tour of each vehicle; therefore, it is easier to relocate customers between different vehicles. These inter-tour relocations are performed by the second VND. This is the first time that this approach has been used for this kind of problem.

The rest of this paper is organized as follows: the next section introduces the principal mathematical concepts of the VRPMT. A detailed explanation of each part of the proposed algorithm is available in Section 3. Section 4 explains the benchmark instances used and compares the results with previous approaches and finally, Section 5 draws some conclusions.

2 Problem description

The VRPMT can be formally defined as follows. Let $G = (V,E)$ be a undirected graph where $V = (v_0,v_1,\ldots,v_n)$ is the set of vertices representing cities or customers and $E \subseteq V\times V$ is the set of arcs. If $(i, j) \in E$ then it is possible to travel from i to j with a cost of c_{ij} considering the cost as either time or distance. A homogeneous fleet $K = (k_1,k_2,\ldots,k_n)$ of m available vehicles with a capacity Q is available in v_0 which represents the depot. Nodes in $V\backslash v_0$ represent the n customers, each one having a demand q and a service time δ. There exists a time horizon L that represents the duration of a working day. It is assumed that Q, L, qi and δ are non-negative integers. The VRPMT solution is a sequence of m tours $T = (T_1,T_2,\ldots,T_m)$ where $T = (r_1,r_2,\ldots)$ denotes a set of routes assigned to vehicle k. The objective of the problem is to find a solution S that minimizes function (1) considering that:

(a) Each route starts and ends at the depot v_0.
(b) Each customer is visited exactly once by just one vehicle.
(c) The total demand of the customers in a route does not exceed Q.
(d) The total duration of a tour, including service times, does not exceed L.

$$F_1(S) = \sum_{T_i\in T}\sum_{r\in T_i}\sum_{(v_i,v_j)\in r} c_{ij} \quad (1)$$

For a complete formal definition of the problem, the mathematical model is available in [10]. Due to the nature of the problem, it is hard to find a feasible solution so the algorithm needs flexibility to work with infeasible solutions. It also helps to explore a wider search space and to find a feasible solution. Hence the objective function (1) is improved as follows:

$$\begin{aligned} F_2(S) = \; & F_1(S) \\ & +\alpha \sum_{r_i\in T}[VL(r_i) - Q]^+ \\ & +\beta \sum_{T_i\in T}[DT(T_i) - L]^+ \end{aligned} \quad (2)$$

where $VL(r_i)$ is the vehicle load required to service route i, $DT(T)$ is the total driver time required to service tour i, $[x]+$ is $max(0,x)$ and α and β are two positive penalizing parameters for overcapacity and overtime constraints. These penalizing parameters are adjusted by the algorithm during its evolution. α (β) is multiplied by the parameter ρ if the last φ iterations were all infeasible with regard to capacity (distance). Otherwise, if the last φ iterations were all feasible with regard to capacity (distance) then α (β) is divided by ρ.

3 Algorithm overview

Algorithm 1 – IVNDS algorithm's structure

1. S_{CVRP} ← Build Initial CVRP Solution
2. S ←Assign S_{CVRP}'s routes to vehicles with BFD
3. Set: S* ← S, itrations←0
4. while iterations < η do
5. $VND_{intra\text{-}tours}$
6. VND_{inter_tours}
7. if S improves S* then
8. set: S* ← S, iterations ← 0
9. end if
10. if iterations ≥ Ω then
11. S ← S*
12. end if
13. Adjust(α, β))
14. Perturb(S)
15. end while
16. return S*

The IVNDS heuristic suggested is described in Algorithm 1. It is based on the ILS algorithm by Lourenco et al. [18]. As with the ILS heuristic, the method needs a start point. To build the initial solution, the problem is reduced to a CVRP and solved with the Nearest Neighbour solver. With the CVRP solution, a BFD heuristic (Section 3.1) is used to assign routes to vehicles and achieve the transformation to the VRPMT initial solution. The Local Search phase from the original ILS method has been substituted with two consecutive and different VND searches [19] detailed in 3.2. The acceptance criteria, described in Algorithm 1 Steps 7-12, handle the intensification/diversification of the method which is guided by parameter Ω and it is explained in 3.4. The ILS is used while η iterations, without improving the best solution, are performed, where η is a parameter.

3.1 Initial solution

To provide a good entry point to the algorithm, an initial solution is built by Steps 1 and 2. This solution is formed with a set of feasible CVRP routes built with a fast and non-optimum algorithm, such as Nearest Neighbour. These routes are used to create a VRPMT solution using the BFD heuristic. This heuristic assigns the routes to m vehicles. The routes are sorted by length in decreasing order and the longest m routes are assigned to each vehicle. Then, the longest non-assigned route is selected and assigned to the vehicle with the shortest total distance. This is repeated until all routes are assigned to a vehicle.

3.2 Improvement phase

To our knowledge, this is the first time that the custom improvement phase performed by Steps 5 and 6 from Algorithm 1 has been suggested. It is formed by two different and consecutive VNDs (Mladenović and Hansen [19]). The first uses the neighbourhood search spaces to perform improved movements between customers from the same tour only. This reduces the length of the routes and generates free space to relocate customers from other tours into improved tours. The relocation is made by the second VND which considers only movements between different tours.

Both VNDs use a few neighbourhood search spaces to leave a local minimum. These search spaces are based on λ and $\lambda-optimum$ interchange methods (Osman [20]). The neighbourhoods implemented are:

- One Point Move (1PM) selects a customer from a route and relocates it to another position.
- Two Point Move (2PM) is a swap method which interchanges positions between two customers.
- Three Point Move (3PM) is similar to 2PM, but in this case interchanges the positions in routes between a single customer and two consecutive ones.
- Two Optimum (2OPT) eliminates two route connections in the solution and evaluates the different organizations of the customers involved, performing the best.

Each neighbourhood is explored sequentially allowing only movements that improve the current solution. After a neighbourhood search is undertaken, if no improved movement has been made the next neighbourhood is explored; otherwise, the process restarts at the first neighbourhood. The same search spaces are used in both VNDs, but in the second the 2OPT neighbourhood is also used after each local search considering only intra-route moves. The objective of this extra search is to improve route quality and to reduce the total distance of the modified routes.

In order to reduce the running time of each neighbourhood search, a move is only examined if v_j is one of the P closest customers to v. This reduces the search space and speeds up the algorithm, where P is a parameter.

3.3 Perturbations

The perturbation gives the ILS the ability to escape from the local optimum solution reached by the VND and guarantees the exploration of a wider search space. Three different perturbations, Swap, Insertion and Exchange, are used randomly by mix perturbation during the search.

The Swap perturbation selects two customers and makes a swap move. The first customer is selected randomly from an infeasible route if one exists; otherwise, it is selected from a random route. The second must be in a different route and it is selected from the first customer's neighbourhood. This process is made φ times. The Insertion perturbation selects a random customer from an infeasible route, if one exists, and $\varphi-1$ neighbour customers. These customers are removed from the solution and then inserted, by extraction order, in the less expensive position. The exchange perturbation consists of swapping a string $s_1 = (v_i,\ldots,v_j)$ from one route $r_1 = (\ldots,v_{i-1},s_1,v_{j+1},\ldots)$ with another string $s_2 = (w_1,\ldots,w_m)$ from a different route $r_2 = (\ldots,w_{l-1}, s_2, w_{m+1},\ldots)$. The size of each string is randomly generated between 2 and φ.

The value φ is critical to the method evolution. A small value of φ may generate cycles and no gain is obtained. Otherwise, if the value is too large, the perturbation will have lost the good properties of the local optimum.

3.4 Intensification/Diversification

In the IVNDS algorithm, the control of the intensification and diversification of the search is based on the solution selected to perturb and it is controlled by the parameter Ω. There are two available solutions to perturb: the current solution S and the current best solution S^*.

The algorithm starts perturbing the current solution S, thus ensuring a wider search space is explored. Initially, *S* has the properties of a promising solution but, after a number of iterations without improving the solution, these characteristics are lost. To prevent this effect, after Ω iterations without improving, the algorithms perturb the current best solution S^*. Perturbing S^* the algorithm intensifies the search exploiting the characteristics of the promising solution.

This new approach is able to control the diversification and the intensification of the algorithm with Ω parameter.

4 Results

This section presents the results obtained with the suggested algorithm. It has been tested over the classical benchmark instances in the VRPMT literature. These are a set of 104 instances proposed by Taillard et al. [6] for the CVRP which are based on problems 1–5 and, 11 and 12 proposed by Christofides et al. [21] and problems 11 and 12 proposed by Fisher [22]. For each instance, several values of m were used with two time horizons $L_1 = \left\lceil \frac{1.05z^*}{m} \right\rceil$ and $L_2 = \left\lceil \frac{1.10z^*}{m} \right\rceil$ for each value of *m*.

The instances were solved ten times and the best solution found was taken. The GAP measure suggested by Olivera and Viera [10] has been used to compare the algorithms' behaviour. This measure compares a feasible solution against the corresponding CVRP best known solution value z* and it is defined as follows:

$$GAP(S) = 100\left(\frac{F_1(S)}{z^*} - 1\right) \quad (3)$$

In previous studies, infeasible solutions were compared to determine which solution was better. In this paper, it is not possible. This algorithm works with infeasible solutions with regard to capacity and distance and the other studies only accept solutions infeasible with regard to distance.

The algorithm was executed in an AMD FXTM6100 Six-Core Processor at 3.30GHz, with 16GB RAM and running Windows Server 2008. It was developed in C++ regardless of the parallelism of the computer.

4.1 Parameter tuning

As in most meta-heuristics, there are parameters that must be correctly adjusted to maximize the algorithm's performance. In our case, some of the parameters were pre-fixed: $\eta = n$, $\Omega = \sqrt{n}$ and $\phi = \sqrt{n}$. To prevent odd behaviour of the algorithm, we also fixed $\alpha_{min} = \beta_{min} = 1$.

Neighbourhood size *P* is a critical parameter for computation time and the optimal value is tightly related to customer distribution. For example, if it should be that in the studied problem the customers are grouped in clusters, *NP* is able to have all the customers of the cluster. In this study, the parameter has been fixed at $P = 15$, which gives good results between computational time and solution costs.

For parameter tuning of ρ, α, β and φ, a set of good candidates was selected for testing over a small benchmark. This benchmark was formed by 30 of those instances for which it was harder to achieve feasible solutions and those for which higher errors were obtained in previous tests. The selected values for each parameter were: $\rho \in \{1.5,2\}$, $\alpha \in \{1,10\}$, $\beta \in \{1,10\}$ and $\varphi \in \{3,7\}$. All the 16 possible combinations were tested against the benchmark. Table 1 shows the obtained results of those parameters which found a feasible solution for all the instances in the benchmark.

Only two of the 16-parameter combinations studied found some feasible solution for all instances in the benchmark. Although both combinations need a similar time to resolve all the benchmark instances, with $\rho = 1.5$, $\alpha = 10$, $\beta = 1$ and $\varphi = 7$, the GAP is reduced by 0.25. Therefore this configuration will be used in the following tests.

Table 1 – Parameter tuning resume. ρ, α, β and φ = input parameters; #fs = total feasible solutions found; time = total time needed to perform the complete benchmark;

ρ	α	β	φ	#fs	GAP	time
1.5	10	1	7	223	1.93%	1378
2	10	10	7	205	2.18%	1421

4.2 VND neighbourhood searches and perturbation impact

To analyse the effect of the different neighbourhood searches, the 104 instances of the benchmark were solved with different neighbourhood search configurations. The first one uses the four neighbourhoods in both VND heuristics. The following configurations use only three of the four available neighbourhood searches. These configurations show the effect made by each heuristic on the algorithm.

Table 2 displays the rate of improved movements made by each heuristic. The 3PM heuristic made few improved moves in comparison with the other heuristics. However, according to

Table 3, this heuristic had a very good effect in the results. The 2OPT heuristic has a significant impact over the execution time. When it is not used, the time needed to solve the whole benchmark is reduced but the quality of the solution is also worse. Each heuristic provides some improvement to the search. Only when the four heuristics were used could the search maximize the number of feasible solutions found and minimize the average error of the solutions.

As Section 3.3 explains, three different perturbations were implemented for the ILS algorithm. To compare the performances of each one and determine which would be better to use, the four perturbations, Exchange, Insert, Swap and Mix, were used individually in the search with the four heuristics. As the neighbourhood search analysis, the perturbations were tested against the complete benchmark of 104 instances with ten repetitions using the parameters selected in Section 4.1.

Table 2 – Average rate of improve moves made by each neighbourhood search

	1PM	2PM	3PM	2OPT
%moves	53.20%	31.30%	4.42%	11.08%

Table 3 – Neighbourhood search impact. time = average time in minutes for resolve whole benchmark; #fs reps = total number of feasible repetitions found; #fs sol = total number of feasible solutions found;

	GAP	*time*	*#fs rep*	*#fs sol*
1PM+2PM+3PM+2OPT	1.19%	73.4	887	98
2PM+3PM+2OPT	1.50%	73.3	872	96
1PM+3PM+2OPT	1.40%	63.8	851	96
1PM+2PM+2OPT	1.22%	64.5	862	96
1PM+2PM+3PM	1.53%	46.5	777	95

Table 4 – Perturbations effects

Perturbation type	GAP	*#fs rep*	*#fs sol*
Exchange	1.15%	894	97
Insert	1.32%	868	98
Swap	1.34%	873	97
Mix(Exchange+Insert+Swap)	1.19%	887	98

Table 4 shows the results obtained with each perturbation. When the perturbations are used separately, the lowest error is obtained with the Exchange perturbation. However, the insert perturbation is able to find one more feasible solution for the tested problems than the Exchange perturbation. Although the swap perturbation does not offer good results by itself, it helps to improve the result when the perturbation mix is used.

4.3 Discussion

The algorithm was run to solve the whole benchmark set using the selected parameter configuration described in Section 4.1, the four heuristics and the mix perturbation. For each instance, the IVNDS algorithm was executed ten times to minimize the effect of the random numbers and the results obtained are summarized in Table 5. This table also compares the results against those of Olivera and Viera [10] and Cattaruzza et al. [16]. The comparison is made because Olivera and Viera [10] had the best and most detailed results in the literature until Cattaruzza et al. [16] published their results. It is important to point out that the results given by Olivera and Viera [10] were specified by the GAP measure with only one decimal and this hampered the solution comparison.

The GAP of the IVNDS algorithm is usually between those of Olivera and Viera and Cattaruzza et al.. The algorithm average error is about 15% lower than Olivera and Viera's and is able to improve or match theirs in 70 of the 104 instances of the benchmark. The IVNDS compared to Cattaruzza et al.'s algorithm matches its results 40 times and improves it in 4 instances.

Notably for problem CMT5_L_1_10 a new better solution has been found by the algorithm. The new solution available in Appendix A improves the GAP of the previous best solution, found by Cattaruzza et al., in 0.10. This solution is 1321.75 in length and performs 17 routes.

As Table 5 shows, the most significant feature of the IVNDS algorithm is its speed to resolve the problems. While Olivera and Viera needed 125 seconds to achieve the best running of a 200-client problem in 2007 and Cattaruzza et al. more than 21 minutes, the IVNDS algorithm is able to perform the search in less than 12 seconds on average.

5 Conclusions

In this paper, a new approach based on the ILS and VND search has been suggested to solve the VRPMT. Instead of the Local Search, two consecutive VNDs have been used to improve the solution. A new intensification/diversification method and a mix perturbation were also suggested. Consequently, a competitive fast algorithm was obtained which is also easy to replicate.

A set of problems was solved and the results were compared against the most significant previous studies. The new algorithm found 98 feasible solutions from the 104 problems and it has achieved a good balance between the quality of the solution and the running time of the algorithm. Furthermore, the new algorithm tested has been able to find a better solution for the problem CMT5_L_1_10 that is showed in Table 6.

Table 5 – Algorithms results grouped by instances. time = average time to solve one problem;

		Algorithms								
Instance		OV			Cattaruzza			IVNDS		
Name	#	GAP	Time	#fs	GAP	Time	#fs	GAP	Time	#fs
CMT1	8	2.02	16	30	2.01	14	30	2.13	0.4	53
CMT2	14	0.83	29	61	0.36	81	63	1.56	0.7	120
CMT3	12	0.66	27	60	0.39	119	60	0.73	1.6	107
CMT4	16	1.26	0	73	0.78	493	74	1.29	3.9	135
CMT5	20	2.45	125	95	1.07	1284	100	1.84	11.7	166
CMT11	10	0.93	28	46	0.35	220	48	0.48	1.9	97
CMT12	12	0.37	27	55	0.39	50	55	0.47	1.2	102
F11	6	1.92	13	24	1.73	40	25	1.77	0.5	47
F12	6	0.56	31	30	0	160	30	0.02	4.2	60

Table 6 – New best solution for CMT5_L_1_10

Vehicle	Route	Length	Load	Route
1	1	123.84	198	0 57 15 145 41 75 23 67 170 25 55 165 130 54 195 0
	2	10.06	48	0 112 156 0
2	1	132.36	194	0 31 189 10 126 63 181 64 49 143 36 46 124 82 153 0
3	1	66.37	198	0 18 114 8 174 45 125 199 83 60 118 147 0
	2	68.9	196	0 111 50 102 157 33 81 120 9 51 1 176 0
4	1	76.79	200	0 21 197 56 186 39 187 139 4 155 110 179 149 26 0
	2	57.71	193	0 27 167 127 190 88 148 62 182 7 194 52 146 0
5	1	48.32	196	0 94 95 92 151 98 85 93 104 99 96 183 0
	2	87.39	198	0 106 48 47 168 123 19 107 175 11 159 0
6	1	122.53	200	0 89 84 17 113 86 140 38 44 119 192 14 43 142 42 172 0
7	1	48.67	194	0 28 184 116 68 80 150 177 109 12 154 138 0
	2	86.83	200	0 132 69 101 70 30 20 188 128 160 131 32 90 108 162 0
8	1	72.77	200	0 6 59 37 100 193 91 191 141 16 61 173 5 166 0
	2	58.69	197	0 53 40 73 171 22 133 74 72 198 180 105 0
9	1	125.06	200	0 78 34 164 135 35 136 65 66 71 161 103 122 0
10	1	53.52	179	0 13 117 97 87 144 178 115 2 137 152 58 0
	2	81.93	195	0 76 196 77 3 158 185 79 129 169 121 29 24 163 134 0
Total length:		1321.75		

6. References

[1] Rodrigue, J.-P., et al.: *The geography of transport systems*, Routledge, 2009.

[2] Lenstra, J. K. & Kan, A.: *Complexity of vehicle routing and scheduling problems*. 11: 221-227, 1981.

[3] Salhi,S. : *The integration of routing into the location-allocation and vehicle composition problems*. PhD thesis, University of Lancaster, 1987.

[4] Clarke, G. & Wright, J.W.: *Scheduling of vehicles from a central depot to a number of delivery points*, Operations research, 12(4): 568-581, 1964.

[5] Fleischmann, B.: *The vehicle routing problem with multiple use of the vehicles*. Technical report, Universitdt Hamburg, 1990.

[6] Taillard, E.D. et al.: *Vehicle routeing with multiple use of vehicles.* Journal of the Operational Research Society, 1065-1070, 1996.

[7] Brandão, J. & Mercer, M.: *A tabu search algorithm for the multi-trip vehicle routing and scheduling problem*. European Journal of Operational Research, 100(1): 180-191, 1997.

[8] Brandão, J. and Mercer, A.: *The multi-trip vehicle routing problem*. Journal of the Operational research society, 799-805, 1998.

[9] Petch, R. & Salhi, S.*: A multi-phase constructive heuristic for the vehicle routing problem with multiple trips.* Discrete Applied Mathematics, 133(1-3): 69-92, 2003.

[10] Olivera, A. & Viera, O.: *Adaptive memory programming for the vehicle routing problem with multiple trips.* Computers & Operations Research, 34(1): 28-47, 2007.

[11] Salhi, S. & Petch, R.J.: *A GA based heuristic for the vehicle routing problem with multiple trips.* Journal of Mathematical Modelling and Algorithms, 6(4): 591-613, 2007.

[12] Battarra, M., et al.: *An adaptive guidance approach for the heuristic solution of a minimum multiple trip vehicle routing problem.* Computers & Operations Research, 36(11): 3041-3050, 2009.

[13] Azi, N. et al: *An adaptive large neighborhood search for a vehicle routing problem with multiple trips.* CIRRELT, 2010.

[14] Chbichib, A. et al.: *Profitable vehicle routing problem with multiple trips: Modeling and constructive heuristics.* In Logistics (LOGISTIQUA), 500-507, 2011.

[15] F Alonso, et al.: *A tabu search algorithm for the periodic vehicle routing problem with multiple vehicle trips and accessibility restrictions*. Journal of the Operational Research Society 59(7): 963-976, 2007.

[16] Cattaruzza, D.a, et al.: *A memetic algorithm for the multi trip vehicle routing problem.* European Journal of Operational Research, 2013.

[17] Prins, C.: *A simple and effective evolutionary algorithm for the vehicle routing problem.* Computers & Operations Research, 31(12): 1985-2002, 2004.

[18] Lourenco, H.R., et al.: *Iterated local search*. In "Handbook of Metaheuristics", Ed. F. Glover and G. Kochenberger, ISORMS, 57:321-353, 2002.

[19] Mladenović, N. & Hansen, P.: *Variable neighborhood search.* Computers & Operations Research, 24(11):1097-1100, 1997.

[20] Osman, I.H.: *Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problem.* Annals of Operations Research, 41(4):421-451, 1993.

[21] Christofides, N. et al.: *The vehicle routing problem.* Combinatorial Optimization, Wiley, Chichester, 315-338, 1979.

[22] Fisher, M.L.: *Optimal solution of vehicle routing problems using minimum k-trees.* Operations research, 42(4):626-642, 1994.

Predictive capacity of VIX and VDAX in relation to the first maturity

J. Giner[1], S. Morini[1], R. Rosillo[2], D. De la fuente[3] B. Ponte[3] and J. Lozano[3]
[1]Finance and Economics Department, University of La Laguna, Tenerife, Spain
[2]Business Management Department, University of Leon, Leon, Castilla y León, Spain
[3]Business Management Department, University of Oviedo, Gijón, Asturias, Spain

Abstract – *The aim of this paper is to study how the volatility indexes can provide useful information content for the prediction of realized volatility. In this paper, using data for the USA and the German indexes, VIX and VDAX data in the period 2001-2012, we measure how accurate are the volatility predictions of each volatility index, considering different times to the nearest options maturity. The results show an equal 30 day-ahead forecasting power of the indexes for different times to the nearest options maturity, measured by the determinant coefficient R^2.*

Keywords: Volatility indexes, stock markets, trading system.

1 Introduction

In 1993, the Chicago Board of Options Exchange (CBOE) introduced the VIX a volatility index which has become the *de facto* benchmark for stock market volatility. In 2003, the CBOE changed the definition and calculation of the VIX in order to enhance its significance. Anyway, often VIX is referred to as the *fear index* or the *fear gauge*, and it represents one measure of the market's expectation of stock market volatility over the next 30 day period.

Most studies indicate that implied volatility is the best indicator of future volatility. In the American options market on the S&P 100, [1] find that the truly relevant information is included in the implied volatility. One of the greatest challenges in financial markets is to manage the elevated volatility of the prices of the assets. In this regard, a lot of research is developed to predict the future volatility, looking for strategies to take better advantage of the opportunities of the market.

However, options are traded (calls and puts) with different strike prices and maturities, so the characterization of an index volatility, with a constant 30 days maturity horizon, is a compromise between the available information and the index definition. To capture the desired 30 days time horizon for the realized volatility, the methodology is to use an average of the volatility referred to its nearest maturity, with a minimum of seven days, and the next nearest maturity. The goal of this paper is to measure how can influence this process in the obtained results.

2 The VIX and VDAX indexes

The main features of different indexes methodologies are presented in this section.

2.1 The new VIX

On September 23, 2003 the CBOE adopted a new methodology for calculating the VIX index. This change has two major developments: 1) replacing the S&P 100 for the S&P 500, based on the increased liquidity of the latter, 2) modifying the calculation method replacing the implied volatilities with a weighted sum of OTM (out of the money) option prices.

The generalized formula used is:

$$\sigma^2 = \frac{2}{T}\sum_i \frac{\Delta K_i}{K_i^2} e^{rT} Q(K_i) - \frac{1}{T}\left(\frac{F}{K_o} - 1\right)^2$$

where:

- σ^2 is the daily variance derived from the option prices
- T is the time to expiration of the option prices
- F is the forward index level derived from index option prices
- K_o the first strike below the forward index level, F
- K_i the strike price of i^{th} out of the money option; a call if $K_i > K_o$ and a put if $K_i < K_o$; both put and call if $K_i = K_o$
- ΔK_i is the interval between strike prices, half the difference between the strike on either side of K_i

$$\Delta K_i = \frac{K_{i+1} - K_{i-1}}{2}$$

- r is the risk-free interest rate to expiration

- $Q(K_i)$ is the midpoint of the bid-ask spread for each option with strike K_i

The VIX index is defined as the 30 day-ahead volatility derived from the option prices, so the CBOE methodology say that it must be measured as a weighted average of variances calculated with the option prices across all strikes at two nearby maturities: T_1 the near or first maturity and T_2 the next or second maturity. So, it is calculated as a weighted average of the variances of the first and second maturity, $T_1\sigma_1^2$ y $T_2\sigma_2^2$ which is interpolated to a capture the 30 days period definition:

$$Var_{30} = T_1\sigma_1^2\left(\frac{T_2 - 30}{T_2 - T_1}\right) + T_2\sigma_2^2\left(\frac{30 - T_1}{T_2 - T_1}\right)$$

Finally is adjusted to obtain the VIX index:

$$\text{VIX} = 100\sqrt{\frac{365}{12}Var_{30}}$$

More information about the new VIX can be obtained at www.cboe.com or consulting [2] and [3].

2.2 Other volatility indexes: the VDAX

Over time different countries have incorporated volatility indexes to their markets.

Thus, in the German market the VDAX is calculated from 1994 in a form fairly similar to the old VIX, and from 2007 it is calculated according to the new methodology of the new VIX. The reference index is the DAX 30, the more important German stock index.

From October 1997, MONEP calculated two indexes of volatility on the CAC-40, the VX1 and VX6, based on the definition of [4], and since 2007 uses the new VIX methodology for calculating the VCAC.

More recently, other countries such as Canada (TSX VIX), Japan (VXJ), Russia (RTSVX), Australia (ASX VIX) or India (India VIX) have incorporated volatility indexes on their markets, following the same methodology of the CBOE for VIX in its new version.

About the Nasdaq-100 CBOE calculates the VXN from December of 2003 following the same methodology as for the VIX.

In this paper we will study the German VDAX data with the American VIX.

2.3 VIX on Forecasting Markets

Volatility Indexes (VIX in general) are a key measure of market expectations of near-term volatility based in the informational content of the negotiated option prices.

The calculation method is an average of weighted prices of out-the-money puts and calls options on the S&P500 index for the VIX and DAX 30 index for the VDAX.

Volatility indexes have several characteristics that make it interesting to use in order to forecast stock markets. It grows when uncertainty and risks increase. During falling markets, the VIX rises, reflecting increasing market fear. This stylized facts can be seen in figure 1 and figure 2.

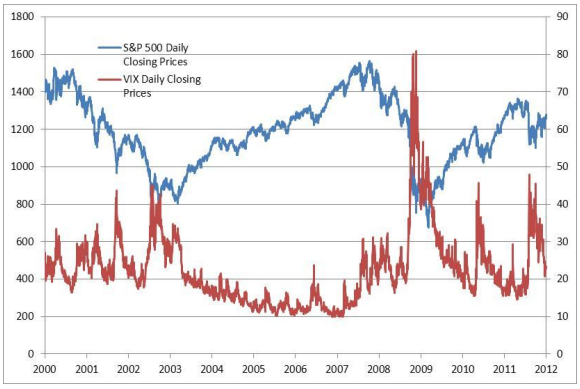


Figure 1. S&P 500 and VIX daily closing prices

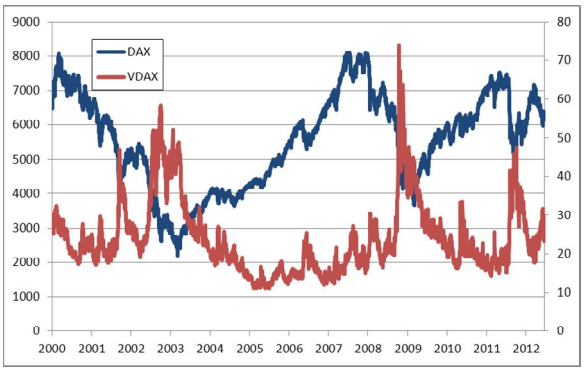


Figure 2. DAX (left axis) and VDAX (right axis)

Volatility index reverts to the mean after high volatility situations and after low volatility situations such us interest rates. Rising markets usually the VIX goes down, reflecting a reduction of fear. So VIX is negatively correlated with stock or index level, and usually stays high after large downward moves in the market.

3 Predictive capacity in relation to the first maturity

As stated in section 2, the volatility indexes are calculated by linearly weighting the calculated variances for the first and second nearest maturity.

In this section we analyse whether volatility indexes can predict better or worse depending on the number of days until the closest first maturity T_1, and also the number of days left until second nearest maturity T_2.

3.1 The nearest maturity interval

By definition, the second maturity date passes to become the first maturity, when there are less than 7 days to the nearest maturity, to avoid price distortions.

Therefore, the shortest time until first maturity is 7 days.

Both in the American as in the German market the options expiration date is the 3rd Friday of each month. This feature determines the range of values that can take the number of days until the closest first maturity, between 7 and 39 days.

As shown in Figure 3, when the month starts on Friday 1st, we have a month with a very early expiration date, the 3rd Friday will be the day the 15th.

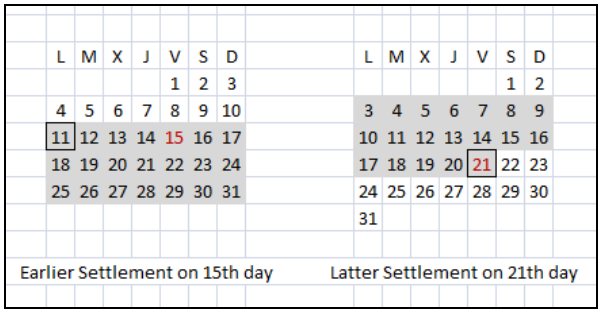


Figure 3. Months with earlier and latter expiration days.

When the month starts on Saturday 1st, we get a very late maturity month, so the 3rd Friday would be the 21th.

These are the limits within which you can move the monthly due date of options, between day 15th and day 21th of the month.

So the longest time until first maturity is 39 days, resulting as the sum of 20 days of an earlier expiration day month, starting on Monday 11th (31 -11 = 20 days), plus 19 days of a latter expiration date month, starting on Monday 3th (21 – 3 = 19 days).

This can be also calculated when we have a month to which, after maturity, still has two calendar weeks. And the next month, has three weeks before the expiration date. So we have to, the first Monday that is the 1st maturity and until the next one, there are six weeks left, resulting in 42-1 days, minus 1 weekend, so we have the 39 days.

3.2 The weighted average of variances

The issue is that the VIX is constructed with options information regarding two different maturities.

Assuming for simplicity that the 2nd maturity occurs 30 days after 1st maturity, we can find a 1st and a 2nd maturity in 7 to 37 days or on the contrary 39 and 69 days, respectively (assuming that between the first and second maturity mediate exactly 30 days).

Both weights are equal when missing maturities 15 and 45 days, respectively, as shown in Table 1.

T1	T2=T1+30	W1=(T2-30)/(T2-T1)	W2=(30-T1)/(T2-T1)
7	37	0.23	0.77
8	38	0.27	0.73
9	39	0.30	0.70
10	40	0.33	0.67
11	41	0.37	0.63
14	44	0.47	0.53
15	45	0.50	0.50
16	46	0.53	0.47
17	47	0.57	0.43
18	48	0.60	0.40
21	51	0.70	0.30
22	52	0.73	0.27
23	53	0.77	0.23
24	54	0.80	0.20
25	55	0.83	0.17
28	58	0.93	0.07
29	59	0.97	0.03
30	60	1.00	0.00
31	61	1.03	-0.03
32	62	1.07	-0.07
35	65	1.17	-0.17
36	66	1.20	-0.20
37	67	1.23	-0.23
38	68	1.27	-0.27
39	69	1.30	-0.30

Table 1. Weighting Values when $T_2 = T_1+30$

The question is whether the VIX predicts better for combinations of times (T_1, T_2) than for others. In view of the above table it seems reasonable to assume that the best quality of information of the options is when T_1 is close to 30, as T_1 coincides with the maturity of the VIX reference, and furthermore it is assumed that there will be much more liquidity and trading in options of T_1 than in options of T_2, contributing to better information.

3.3 Realized volatility

There is no general consensus on how to measure realized volatility. However, the usual trend is calculating realised volatility through the standard deviation of future performance, as evidenced by [5] and [6].

Our goal is to evaluate the usefulness of volatility indexes studied in the previous section as forecasters of realized

volatility within the referred period. For the case of T=30 calendar days we would have:

$$\sigma^{R}_{t+1,t+T} = \sqrt{\frac{1}{m-1}\sum_{k=1}^{m}\left(r_{t+k} - \bar{r}_{t,t+T}\right)^2}$$

where r_{t+k} is the daily return of the market index (S&P 500 or DAX) at the moment $t+k$ and $\bar{r}_{t,t+T}$ is the average return of the T calendar days considered (with T equals 30 calendar days), and m is the effective number of trading days.

3.4 Forecasting with VIX

The question is analyse how accurate are the VIX predictions for different times to nearest maturity T_1. In the period 2001-2012, 2799 trading days, we perform a regression model, but distinguishing 25 different grouping regressions for each of the possible times T_1, where $T_1 \in [7,39]$ days. So 25 different regressions are performed:

$$\sigma^{R,T1}_{t+1,t+30} = \alpha + \beta\ \mathrm{VIX}^{\mathrm{T1}}{}_{t} + u_t$$

In a way that we group the date of $\mathrm{VIX}^{\mathrm{T1}}$ referenced to $T_1 \in [7,39]$ with its corresponding realised volatilities.

The results of the quality of the adjustment, based on the R^2 determination coefficient, are shown in Table 2.

T near	7	8	9	10	11
R^2 VIX	0.66	0.62	0.61	0.62	0.61
R^2 VDAX	0.68	0.65	0.64	0.64	0.62
N obs.	128	132	132	132	130
T near	14	15	16	17	18
R^2 VIX	0.6	0.62	0.56	0.53	0.59
R^2 VDAX	0.61	0.61	0.6	0.61	0.63
N obs.	128	127	128	127	123
T near	21	22	23	24	25
R^2 VIX	0.59	0.54	0.6	0.62	0.62
R^2 VDAX	0.64	0.63	0.65	0.62	0.65
N obs.	125	128	127	128	124
T near	28	29	30	31	32
R^2 VIX	0.61	0.58	0.64	0.58	0.58
R^2 VDAX	0.66	0.65	0.65	0.66	0.68
N obs.	127	132	132	132	128
T near	35	36	37	38	39
R^2 VIX	0.74	0.73	0.72	0.69	0.66
R^2 VDAX	0.72	0.69	0.67	0.67	0.67
N obs.	46	46	46	46	45

Table 2. R^2 determination coefficient related with the number of days to nearest maturity T_1.

The results are always higher than 0.50 and lower of 0.75, in general good results to consider the forecasting capacity of volatilities indexes.

But there are little differences in the predictive ability of the indexes when we classify according to days until the first maturity, T_1, neither in the VIX or in VDAX indexes.

Figure 4 we present this information graphically. Neither in the USA index, nor in Germany index, appears significant differences between the coefficients of determination calculated for different times to the closest maturity. In general, all the values are close to the global values (without distinguishing with T_1), $R^2_{VIX} = 0.610$ and $R^2_{VDAX} = 0.645$.

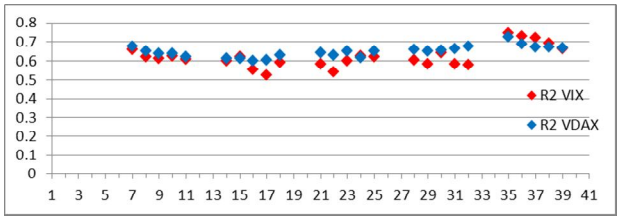


Figure 3. R^2 determination coefficient related with the number days to nearest maturity T_1.

Neither is the patterns observed between the different days of the week remarkable.

It can be also remarkable the effect that the predictive quality of the German index is slightly better than the American.

4 Conclusions

The results show a proved forecasting capacity of index volatilities, and this is equally distributed related with the time to nearest expiration day. One could expect better results on near term periods, or 30 day-ahead expiration dates, but the results similar for different times to the nearest options maturity, measured by the determinant coefficient R^2.

5 Acknowledgment

Financial support given by Ministry of Economy and Competitiveness, Reference ECO2011-23189, and the Government of the Principality of Asturias is gratefully acknowledged.

6 References

[1] Blair, B.J., Poon, S.H. y S.J. Taylor, 2001. Forecasting S&P 100 volatility: the incremental information content of implied volatilities and high-frequency index returns. *Journal of Econometrics* 105, 5-26.

[2] Demeterfi, K., Derman, E., Kamal, M. y J. Zou, 1999. A guide to volatility and variance swaps. *Journal of Derivatives* 6 (4), 9-32.

[3] Carr, P. & Wu, L. (2006). A Tale of Two Indexes. *Journal of Derivatives* 13 (3), 13-29.

[4] Brenner, M., & Galai, D. (1989). New financial instruments for hedging changes in volatility. *Financial Analysts Journal* 45(4), 61-65.

[5] Andersen, T.G. y T. Bollerslev, 1998. Answering the skeptics: yes standard volatility models do provide accurate forecasts. *International Economic Review* 39, 885-905.

[6] Andersen, T.G., Bollerslev, T., Diebold, F.X. y H. Ebens, 2001. The distribution of realized stock return volatility. *Journal of Financial Economics* 61, 43-76.

From the RCPSP to the DRCMPSP: Methodological foundations

F. Villafañez, A. Lopez-Paredes, J. Pajares
INSISOC
University of Valladolid (Spain)
Valladolid, Spain
pajares@insisoc.org

D. de la Fuente
University of Oviedo (Spain)
david@uniovi.es

***Abstract*— In this paper we review different approaches for scheduling projects in multi-project environments. We show the evolution of the methodologies from the Resource Constrained Project Scheduling Problem to the Decentralized Resource Constrained Multi-Project Scheduling Problem.**

We argue that traditional methods used in project scheduling cannot cope with the complexity of current real portfolios performed by project-based organizations. We advocate that a decentralized approach can help project managers to deal with complex restrictions and objectives, including not only operational constraints, but also financial and strategic issues. We model complex portfolios as Multi-Agent System, so that we can include complex behavior and restrictions.

Keywords*— *RCPSP; Resource-Constrained Project Scheduling Problem; RCMPSP;Resource-Constrained Multi-Project Scheduling Problem

I. INTRODUCTION

In practice, firms always run several projects at the same time. Therefore, firms have to schedule a set of projects (project portfolio) where some resources (human, machines, facilities, etc.) are shared by several projects.

For this reason, researchers and academics in the area of Project Management and Project Portfolio Management have focussed their attention on solving the Resource Constraint Multi-Project Scheduling Problem (RCMPSP). The problem is an extension of the classical Resource Constrained Project Scheduling Problem (RCPSP) to multi-project environments.

But in a competitive and global word, the complexity of project portfolios is continually increasing, especially for project-based organizations engaged in multi projects at different geographical locations.

For instance, in real projects, some resources are shared by several projects, while others resources are assigned only to a particular project at a time. Some trade-offs between project objectives and global portfolio objectives must be fulfilled. The hypothesis of "renewable resource" cannot be applied in some real cases. Financial constraints and objectives have to be included within the problem. And decisions about project priority (and project selection) have to be aligned with corporate strategy.

Although the initial RCMPSP approach has been extended to cope with this complexity, in practice, some real restrictions are difficult to model under the classical approach. And, on the other side, when it is possible to model the problem, usually it is hard to find a solution.

In order to cope with the complexity of real portfolios, we advocate for decentralized computational methods, as they allow to model multi-project environments from a "bottom-up" approach, where particular constrains can be easily modelled. In particular, we model multi-project environments as a combinational problem.

In this paper, we review the different approaches to multi-project scheduling from the initial RCPSP to the more recent computational methods used in the Decentralized Resource Constrained Multi-Project Scheduling Problem (DRCMPSP). We show the main characteristics of the decentralized approach that we implement by means of multi-agent technologies.

II. THE BASIC APPROACHES TO PROJECT SCHEDULING.

In figure 1, we show an historical overview of project and multi-project scheduling from the XX Century. Early tools where mainly graph based methodologies (Gantt charts[1], [2], Harmonygraph [3], Flow-line scheduling, Lines of balance methods [4], [5], Milestone charts, etc.). During the fifties, researchers developed the mainstream methods based on Graph Theory (PERT [6], CPM [7], ROY [8], PEP [9], PDM [10]). Currently, those methods are still widely used, and they have been implemented in most common project management software. RCPSP was first formulated during the sixties and both mathematical and heuristics methods where developed. The first methods, mainly based on linear programming techniques (*Simplex Method*, *Explicit* or *Implicit Enumeration Methods*, such as *"Branch&Bound"*, etc.) [21]–[23]) could not deal with more complex real constraints and, therefore, heuristics methods became more appropriate.

During the eighties, the multi-project problem became interesting for researchers and academics. We have to wait until de current century to see decentralized approaches.

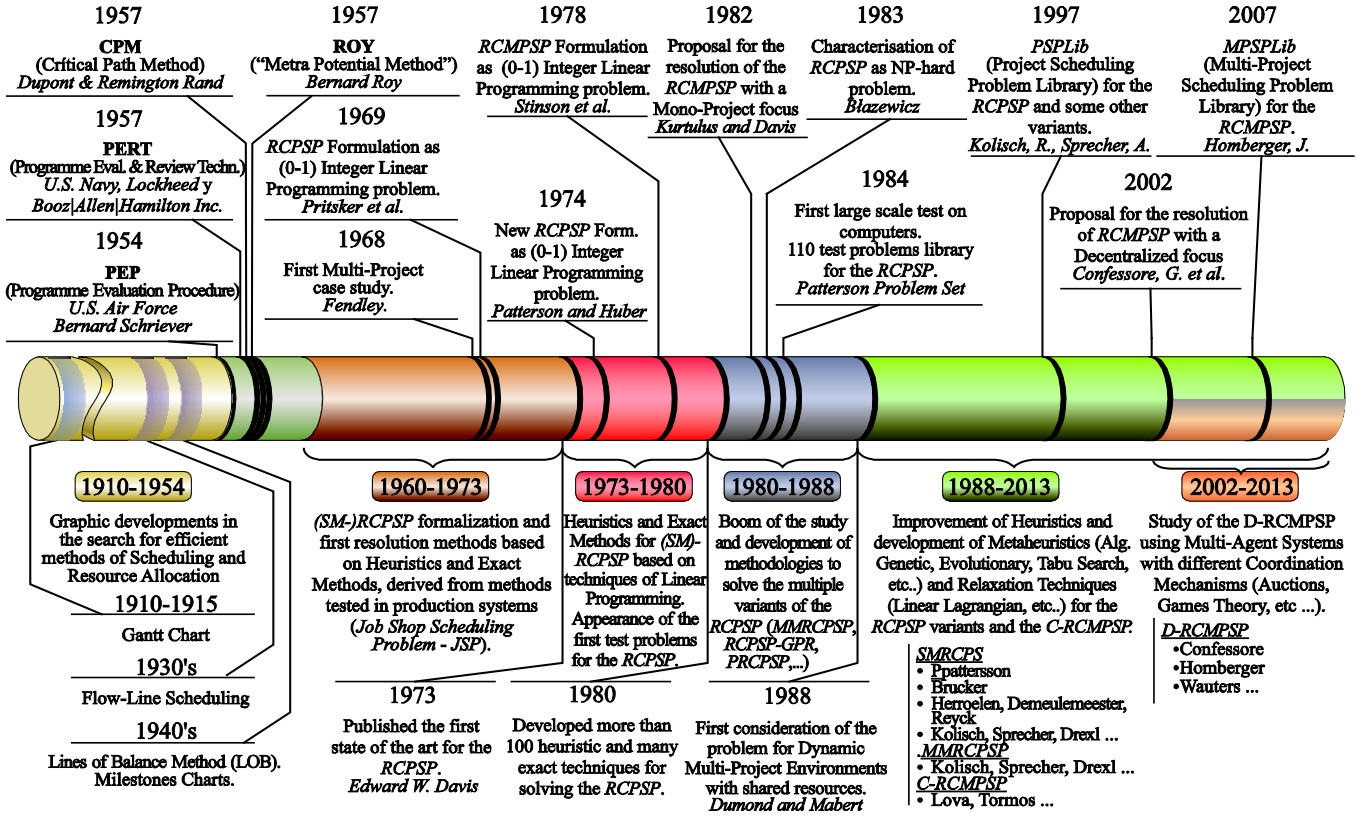


Fig. 1. Timeline with the stages and most important milestones in the study of project scheduling problems with time and resource constrains.

III. MULTI-PROJECT SCHEDULING WITH CONSTRAINTS (RCMPSP)

The *Resource-Constrained Multi-Project Scheduling Problem*", commonly call by the acronym RCMPSP [27], [28] has been widely used to model multi-project environments.

Most of the frameworks distinguish between "local resources" (allocated to a particular project) and "global resources", which can be allocated to all the projects belonging tothe portfolio.

Usually, global resources are human resources with specific high performance competences (i.,e.. Expert engineers) or expensive physical assets with a particular purpose. Firms are not interested in having "excess capacity" of these kind or resources, especially in the short term. On the other side, firms can address more projects by sharing global resources.

Common objective functions are: Minimizing the total duration of the whole portfolio (total makespan); minimizing the average project duration (average makespan); minimizing average project delay or the standard deviation of the project delays. Other firms are more interested in reducing global resources idleness or optimizing the amount of global resources in order to address the highest number of projects.

The solution of the problem shows the schedule of all the projects and the resources needed by each activity.

For simple cases (simple precedence relations, few activities, low number of global resources, etc.), classical Operations Research methods can be used [22] (*Simplex Method*, *Explicit* or *Implicit Enumeration Methods*, such as *Branch&Bound"*, etc.).

More complex problems can be dealt with methodologies like Heuristics, Metaheuristics, Genetic Algorithms, Evolutionary Algorithms, Taboo Search, Ant Colony Algorithms, Simulated Annealing, etc. [30], [31].

A first approach to solve the problem is the Centralized one (CRCMPSP), where all the activities of each project are considered as activities of a global "macro-project" ([30], [32], see fig. 2). The original precedence relations are translated into the global project and two dummy activities are added: One for the beginning and other for the finish of the "macro-project".

The CRCMPSP resolution methods are just extensions of the traditional methods used for solving the simple RCPSP, and they work reasonably well for objective functions like total makespan or resource levelling optimization.

The basic approach can be improved by means of multi-step methodologies. For instance, it is possible to use the feasible schedules obtained in a first step and then apply enhancement algorithms to accommodate particular objectives of individual projects (without losing global feasibility).

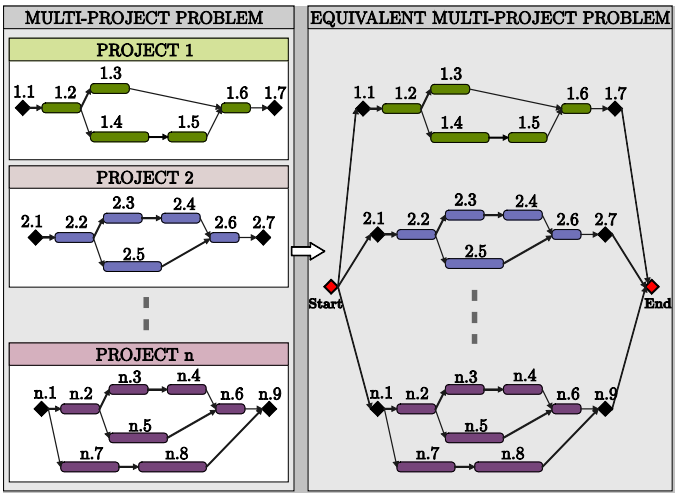


Fig. 2. Conceptual description of the RCMPSP centralized approach.

Anyway, in the centralized approach, there is only a "single programming entity" with all the decision making power. This entity creates a global schedule satisfying global objectives, regardless of possible local particular targets.

IV. The Decentralized Approach to the Resource-Constrained Multi-Project Scheduling Problem (DRCMPSP)

Project based firms are used to manage simultaneously several projects, sometimes in different geographical locations. Each project has its own constraints, customers (internal or external to the firm), stakeholders, etc. In practice, there are overruns and the project manager has to take decisions with local information. At the same time, all the projects should be aligned with corporate strategy. In other words, it makes no sense to develop a project which does not contribute to corporate strategy and financial objectives. In order to include all these issues into the model, real portfolio management demands decentralized methodologies.

The limitations observed by researchers for the centralized approach to the RCMPSP are the origin of the proposal and recent adoption of a new perspective to solve the RCMPSP. It's known as the *Decentralized Resource-Constrained Multi-Project Scheduling Problem* (DRCMPSP) [33].

The term decentralized comes from the fact that these methodologies provide some freedom to individual projects in order to take its own scheduling decisions. Now, the decision-making capacity is distributed among all projects. Therefore, it is required a coordinating entity to organize, arbitrate and check the overall feasibility of the schedules proposed by each project.

The methodology also requires to establish information and communication flows between the coordinating entity (the program or portfolio management) and the project managers (each one managing a particular project). Additionally, it is also possible to include entities acting as *resource managers*, which decide about the process of resources allocation.

In figure 3, we show a picture with the agents of the system. Project manager entities (of individual projects) are responsible for individual project scheduling. One or more Resource Manager entities are engaged in managing the availability of resources.

Above them and in a tactical level, the *Project Portfolio Manager* entity aligns the objectives of each project manager with the objectives of the portoflio. Depending on the context, it can assume roles of coordinating or programming entity.

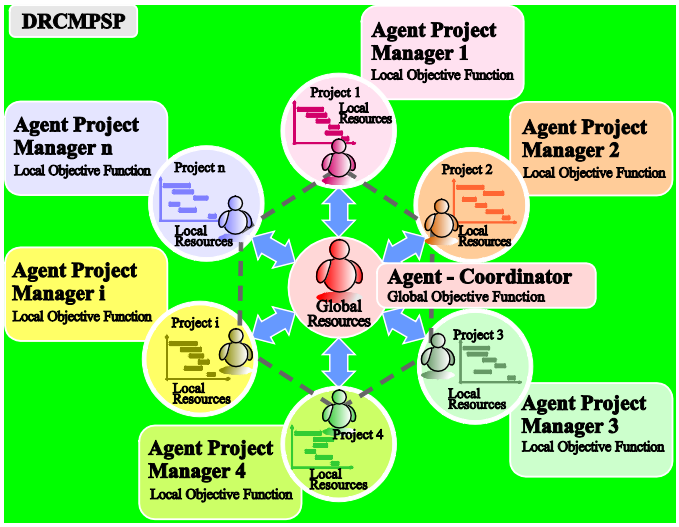


Fig. 3. Conceptual description of the RCMPSP decentralized approach.

The Project Manager entities may contribute to build their initial schedules, but the final decision concerning the best global schedule for all the projects belongs to the Project Portfolio Manager.

A. Coordination Mechanism in DRCMPSP

There are many ways to implement coordination mechanisms among projects, including some methods inspired by other sciences, such as economics, political science and some methodologies used in distributed allocation and decision making systems.

For example, some researchers propose mechanisms that emulate the operation of economic institutions used in the goods and services markets, especially those based on allocation by auctions [33], [34]. Each unit of resource available on a different moment (*slot resource)* is a good, which can be traded in an auction. Project managers bid in the auction in order to take the slot resources they need to fulfil the activities of the project.

Among the different types of auctions (English, Dutch, etc...), Combinational Auctions (CA) seem to be the ones providing better results [35], [36]. It is a kind of multi-item auction that allows the bidders to submit bids not only for each individual auctioned item, but also allows to bid simultaneously for combination of items.

Within the metaphor of combinational auctions, the individual project managers bid for slot-resource combinations that allow them to complete the activities of the project according to the precedence relations.

Other interesting decentralized allocation mechanisms apply Game Theory [37], or Voting Systems [38].

B. Multi-Agent Systems

The computational implementation of most of these decentralized mechanisms use the *Multi-Agent Systems* paradigm [39], [40].

A *Multi-Agent System* (MAS) is a system composed of multiple entities known as agents. These agents are endowed, to a greater or lesser extent, with some kind of individual intelligence and some level of autonomy in their decision-making. The agents also have the possibility of establish some level of communications and interactions between them and with the environment around them.

MAS are *Distributed Artificial Intelligence* (DAI) systems, usually network implemented, where the combined behaviour of their elements can produce a smart result although their individual behaviours can be simple. For this reasons, multi-agent system are particularly suitable for implementing a computational methodology to solve the RCMPSP with a decentralized approach.

In computational terms, each agent would be an independent process of the software system, which exists within a certain context or environment, and with a pattern of more or less self-interested behaviour. This behaviour is usually defined through rules of own performance and responsiveness to external *stimuli*, as simple as possible. In addition, agents can interact following some predefined *agent-agent* and *agent-environment* communication protocols. The environment is usually materialized in form of parameterized scenarios.

Following with the analogy explained in figure 3., we can model the DRCMPSP by means of a *Multi-Agent System* with the following agents:

- The "*Project Manager Agents*", who are responsible for planning the activities of their projects (operational level).
- They may also exist at the operational level, the "*Resource Manager Agents*", who are responsible for managing the availability of resources. They can take decisions in the resource allocation process.
- Above them lie the "*Project Portfolio Manager Agent*", sometimes called simply the Coordinator Agent. It works at a tactical level, trying to align the objectives of the other agents with the objectives of the Organization. It can may assume roles of *coordinator entity* or *programming entity* depending on context.
- Some methods include "*Activity Agents*" or "*Task Agents*", which represent each of the activities of each project, and the "Resource Agents", which represent each of the existing resources.

The Project Manager agents are focussed on developing an optimal (or at least feasible) schedule for their project, based on their goals (objective functions). We need to define how project manager agents interact with other agents and with the environment (in case of modification of scenarios, data requests, coordination possibilities, etc...).

At an operational level, each *Project Manager Agent* is programmed to solve a RCPSP problem, usually using one of the well-proven methodologies to solve this problem (Heuristics, Genetic Algorithms, etc...). The difference now is that the availability of resources to schedule their project is not a fixed data from the beginning, because it can be modified externally through the coordination mechanism.

In some cases, the assigned resource availability should be notified explicitly to each *Agent Project Manager*. In other cases it will be a common information available more or less explicitly or implicitly for all agents.

Moreover, there must be a communication channel that allows the *Project Portfolio Manager Agent* to have feedback and determine the global status of the projects schedules, and thus to know which ones met their local goals. Therefore, communication mechanisms and information flows have to be properly defined during the system modelling phase.

C. Iterative Improvement Processes

Generally, the decentralized resolution methods are performed as an iterative improvement processes. After each iteration with unsatisfactory results, the process is repeated until satisfactory results are achieved (if this is possible). In some methods, the *Coordinator Agent* explores a different allocation of resources between projects for the next iteration, and then the *Project Managers Agents* have to check if they are able to perform a feasible schedule with the new distribution of resources

In other methods, the *Project Management Agents* are "*forced to*" rebuild their schedules. For example, to adjust them to their available budgets due to the change of the resources prices. This creates a new different profile of resource consumption for each project, which must be studied jointly by the *Coordinator Agent*, in order to verify its compatibility with the total availability of resources.

In both cases, the process continues until a satisfactory solution for all parties is reached. Obviously, the process could continue for an indeterminate manner in time, so it is needwe need to include some stop algorithms. For example, after a particular number of interactions without significant improvement or after a maximum computational time.

D. "Acceptably Good" solutions

As mentioned above, these methods cannot guarantee that the best solution obtained is a global optimum of the problem. Usually, the method will provide an "acceptably good" solution (or solutions), close enough to the optimal solution. In practice, the deviations from optimal levels are small, and therefore, the methodology allow multi-project schedules in a reasonable time. This is especially important for large and complex project portfolios, with projects concerning hundreds of activities.

V. CONCLUSIONS

We have reviewed the different approaches to multi-project scheduling from the initial RCPSP framework to the decentralized approaches (DRCMPSP). We advocate for a decentralized approach to deal with the complex constraints of real project portfolios. In particular, we model complex project portfolios by means of a combinational auctions and we use multi-agent methodologies to implement the system.

By means of this framework, we can model not only operational constraints, but also financial and strategic issues.

ACKNOWLEDGMENT

This paper has been partially financed by the Regional Government of Castile and Leon (Spain), project reference GR 251/09.

REFERENCES

[1] H. L. Gantt, *Work, wages, and profits*, vol. 8. Hive Pub. Co., 1913, p. 194.

[2] H. L. Gantt, *Organizing for Work*. New York: Harcourt, Brace and Howe, Inc., 1919.

[3] K. Adamiecki, "Harmonygraph. Przeglad Organizacji.," *Prz. Organ. (Polish J. Organ. Rev.*, 1931.

[4] Office of Naval Material Washington D.C., *LINE OF BALANCE TECHNOLOGY*. Defense Technical Information Center, 1962.

[5] A. C. Gehringer, "Line of Balance," *Armed Forces Comptrol.*, vol. 12, no. 3, pp. 16–21, 1958.

[6] D. G. Malcolm, J. H. Roseboom, C. E. Clark, and W. Fazar, "Application of a technique for research and development program evaluation," *Oper. Res.*, vol. 7, no. 5, pp. 646–669, 1959.

[7] J. E. J. Kelley and M. R. Walker, "Critical-path planning and scheduling," in *Papers presented at the December 13 1959 eastern joint IREAIEEACM computer conference on IREAIEEACM 59 Eastern*, 1959, pp. 160–173.

[8] M. Algan, B. Roy, and M. Simonard, "Principes d'une méthode d'exploration de certains domaines et application à l'ordonnancement de la construction de grands ensembles," *Cah. du Cent. Mathématiques Stat. Appliquées aux Sci. Soc.*, no. 3, p. 41, 1962.

[9] U. S. W. A. D. Division, *Directorate of Systems Management: PEP: Program Evaluation Procedure*. 1960.

[10] J. W. Fondahl, *A non-computer approach to the critical path method for construction industry*. Dept. of Civil Engineering, Stanford University, 1961.

[11] E. W. Davis, "Resource allocation in project network models - A survey," *J. Ind. Eng.*, vol. 17, no. 4, pp. 177–188, 1966.

[12] E. W. Davis, "Project Scheduling under Resource Constraints—Historical Review and Categorization of Procedures," *A I I E Trans.*, vol. 5, no. 4, pp. 297–313, 1973.

[13] A. A. B. Pritsker and L. J. Watters, "A Zero-One Programming Approach to Scheduling with Limited Resources," Santa Monica, California, 1968.

[14] L. A. Kaplan, "RESOURCE-CONSTRAINED PROJECT SCHEDULING WITH PREEMPTION OF JOBS.," University of MICHIGAN, 1988.

[15] R. Klein, *Scheduling of Resource-Constrained Projects*, Volumen 10. Kluwer Academic Publishers, 2000, p. 369.

[16] R. Alvarez-Valdés Olaguíbel and J. M. Tamarit Goerlich, "The project scheduling polyhedron: Dimension, facets and lifting theorems," *Eur. J. Oper. Res.*, vol. 67, no. 2, pp. 204–220, 1993.

[17] A. Mingozzi, V. Maniezzo, S. Ricciardelli, and L. Bianco, "An Exact Algorithm for the Resource-Constrained Project Scheduling Problem Based on a New Mathematical Formulation," *Manage. Sci.*, vol. 44, no. 5, pp. 714–729, 1998.

[18] P. Brucker, "Resource-constrained project scheduling: Notation, classification, models, and methods," *Eur. J. Oper. Res.*, vol. 112, no. 1, pp. 3–41, Jan. 1999.

[19] S. Hartmann and D. Briskorn, "A survey of variants and extensions of the resource-constrained project scheduling problem," *Eur. J. Oper. Res.*, vol. 207, no. 1, pp. 1–14, Nov. 2010.

[20] J. Blazewicz, J. K. Lenstra, and A. H. G. Rinnooy Kan, "Scheduling subject to resource constraints: classification and complexity," *Discret. Appl. Math.*, vol. 5, no. 1, pp. 11–24, 1983.

[21] E. L. Demeulemeester and W. S. Herroelen, *Project Scheduling: A Research Handbook*, vol. 102. Kluwer Academic Pub, 2002, p. 710.

[22] J. H. Patterson, "A Comparison of Exact Approaches for Solving the Multiple Constrained Resource, Project Scheduling Problem," *Manage. Sci.*, vol. 30, no. 7, pp. 854–867, 1984.

[23] J. H. Patterson, "Alternate methods of project scheduling with limited resources," *Nav. Res. Logist. Q.*, vol. 20, no. 4, pp. 767–784, 1973.

[24] R. Kolisch and S. Hartmann, "Heuristic Algorithms for the Resource-Constrained Project Scheduling Problem: Classification and Computational Analysis," in *Project Scheduling SE - 7*, vol. 14, J. Węglarz, Ed. Springer US, 1999, pp. 147–178.

[25] R. Kolisch and S. Hartmann, "Experimental investigation of heuristics for resource-constrained project scheduling: An update," *Eur. J. Oper. Res.*, vol. 174, no. 1, pp. 23–37, Oct. 2006.

[26] S. Hartmann and R. Kolisch, "Experimental evaluation of state-of-the-art heuristics for the resource-constrained project scheduling problem," *Eur. J. Oper. Res.*, vol. 127, no. 2, pp. 394–407, Dec. 2000.

[27] E. W. Davis, "An exact algorithm for the multiple constrained project scheduling problem," Yale University, 1969.

[28] L. G. Fendley, "Towards the development of a complete multi-project scheduling system," *J. Ind. Eng.*, no. 19, pp. 505–515, 1968.

[29] A. A. B. Pritsker, L. J. Watters, and P. M. Wolfe, "Multiproject Scheduling with Limited Resources: A Zero-One Programming Approach," *Manage. Sci.*, vol. 16, no. 1, pp. 93–108, 1969.

[30] I. Kurtulus and E. W. Davis, "Multi-Project Scheduling: Categorization of Heuristic Rules Performance," *Manage. Sci.*, vol. 28, no. 2, pp. 161–172, 1982.

[31] A. Lova and P. Tormos, "Analysis of Scheduling Schemes and Heuristic Rules Performance in Resource-Constrained Multiproject Scheduling," *Ann. Oper. Res.*, vol. 102, no. 1–4, pp. 263–286, 2001.

[32] A. Lova, C. Maroto, and P. Tormos, "A multicriteria heuristic method to improve resource allocation in multiproject scheduling," *Eur. J. Oper. Res.*, vol. 127, no. 2, pp. 408–424, Dec. 2000.

[33] Y. H. Lee, S. R. T. Kumara, and K. Chatterjee, "Multiagent based dynamic resource scheduling for distributed multiple projects using a market mechanism," *J. Intell. Manuf.*, vol. 14, no. 5, pp. 471–484, 2003.

[34] G. Confessore, S. Giordani, and S. Rismondo, "A market-based multi-agent system model for decentralized multi-project scheduling," *Ann. Oper. Res.*, vol. 150, no. 1, pp. 115–135, Dec. 2006.

[35] S. De Vries and R. Vohra, "Combinatorial Auctions - A Survey - Presentation." 2000.

[36] P. Cramton, Y. Shoham, and R. Steinberg, "Combinatorial Auctions," 2004.

[37] T. Wauters, K. Verbeeck, P. De Causmaecker, and P. Decausmaecker, "A Game Theoretic Approach to Decentralized Multi-Project Scheduling (Extended Abstract)," in *9th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2010)*, 2010.

[38] J. Homberger, "A (μ, λ)-coordination mechanism for agent-based multi-project scheduling," *Or Spectrum*. pp. 1–26, 2009.

[39] M. Wooldridge, C. Street, M. Manchester, and N. R. Jennings, "Intelligent Agents : Theory and Practice," *October*, no. January, pp. 1–62, 1995.

[40] M. Wooldridge, *Introduction to Multiagent Systems*, vol. 30. John Wiley and; Sons, Inc., 2002, pp. 152–183.

Improving Business Processes Management Systems by introducing Soft Computing Elements

Francisco P. Romero[1], **Arturo Peralta**[2], **Jose A. Olivas**[1], **Jesus Serrano-Guerrero**[1] **and Sergio Moreno**[2]
[1]Dept. of Information Technologies and Systems, University of Castilla La Mancha, Ciudad Real, Spain
[2]Docpath Document Solutions SL, Madrid, Spain

Abstract— *Business document management systems may provide a lot of useful features in a wide range of document storage and retrieval situations. However, the provision of search systems only based on static indexes will not meet the requirements of the users. Regarding to this background, smarter business management systems are needed with adaptive capabilities for flexible information retrieval, semantic annotation, etc.*

In this work, different soft computing techniques such as fuzzy logic have been applied in order to enhance information retrieval capabilities in business document management systems. The core of the proposed system is based on the application of natural language processing, machine learning algorithms and fuzzy linguistic modeling to semantic annotate business documents and to improve search engine results.

A prototype was implemented and successfully tested in several study cases (invoices, reports, medical records, etc.)

Keywords: document management systems, soft computing, natural language processing, fuzzy logic, document categorization

1. Introduction

Business documents range from brief accounting documents (invoices, bills of entry, transport documents, etc.), to complex legal agreements, circular letters, statistical reports, and so on. These kind of documents are used extensively by professionals in their execution of their own work and to share information with others [1]. Moreover, a company uses documents to communicate, transact business and analyze its productivity, thus, ineffective information retrieval leads to a lack of productivity because it contributes lost work time and duplication of effort [2].

Moreover, in today's business environment, the volume of information is increasing at an unstoppable and constant rate [3]. The storage of an increasing number of business documents and their proper management has become a real challenge for businesses.

A document management system (DMS) can be defined as a computer system that is used to store, track, and retrieve electronic documents [4]. Aware of the above mentioned challenges, the current document management systems tools allows storage of documents and web-based queries.

These document management systems can be used for both individual paper management and business process management [5]. For example, an official document management system is implemented in [6], the proposed system is more oriented to a framework to model business process management than an information retrieval system.

Important causes of the business documents retrieval problem are an incorrect automatic understanding of the information need and of the content of business document texts, and a poor strategy for matching query and document content [7]. In order to solve this problem, a semantic information retrieval mechanism to enable the user to obtain required information with semantics-based query is proposed in [8].

On the other hand, current technological possibilities allow management systems to exploit these texts by using some well known text analysis software frameworks like Unstructured Information Management Architecture (UMIA)[1] and General Architecture for Text Engineering (GATE)[2]. In the work of [9] the goal is to identify and annotate structural parts of regulation documents by using the UMIA framework.

The goal of this work is to enhance information retrieval performance of a business DMS according to the user's view. The proposed approach incorporates in a DMS an array of advanced features that go far beyond mere storage, resulting in an semantic annotation of the documents and a deep analysis of the information contents extracted from the documents.

For this purpose, it required the integration of innovative soft computing techniques and well established natural language processing algorithms to enable an efficient content-based retrieval. These tools allow us a highly advanced document management software for optimal document management.

The rest of the paper is organized as follows: The proposed model based on Natural Language Processing and Soft Computing techniques is explained in section 2. Section 3 contains the description of the preliminary experiments carried out. Finally, section 4 presents some conclusions derived from the experienced.

[1]http://uima.apache.org/
[2]http://gate.ac.uk

2. Proposal

A DMS provide an initial basis for collaboration by organizing your documents and implementing different tools to optimize the flow of documents between employees. This kind of tools has some key functionalities such as the registry and classification of documents with versioning, the forwarding of documents with alarms and the procedural organization of documents [10].

A DMS includes two main components: the Data Storage System, a fundamental element in the infrastructure layer, integrated and used by other company applications[11]; and the Content Management System, that helps to mask the complexity of underlying data management tasks and provides an efficient platform for users to contribute, share and retrieve particular content [12].

2.1 DMS Workflow

One of the main problems that DMS have to deal in an electronic context is the issue of workflow. Our tool users to access organization documents and interact in a simple way. This means that users are able to retrieve and organize documents. The basic workflow can be seen in figure 1.

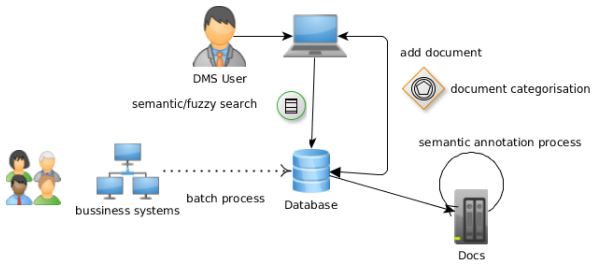


Fig. 1: Basic Workflow.

Different business applications can produce documents according to the business context in a standardized form (invoices, contracts, ...), often in different languages. In this case the users cannot decide whether or not to save a document into the DMS but a batch process loaded these documents into the repository. In this process, the documents are annotated using the most relevant concepts extracted from their content by another batch process (the semantic annotation process 2.2). On the other hand, a DMS user can load individually documents into the repository. The content is annotated on-line in the same procedure and, moreover, the user receive a recommendation (see Section 2.5) about how to classify the document in the DMS System (folder, type of document, etc). This recommendation is very important in order to keep the repository structure consistent. Moreover, this system includes semantic and fuzzy search capabilities to effectively offset existing defects of the traditional keyword-based search.

2.2 Semantic Annotation Process

The proposal presented in this section includes several phases of data processing to automatically extract relevant concepts from the selected document set, as can be seen in figure 2.

The Semantic Annotation Process is a batch process in order to transform and analyze the documents stored in the DMS. This process can process the new documents, the document which previously fail, and the documents under certain conditions established by the system administrator. The aim is to extract a set of relevant concepts that represent a brief summary of the document content.

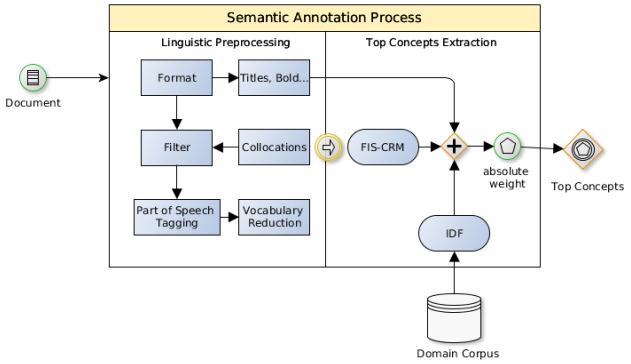


Fig. 2: Semantic Annotation Process.

2.2.1 Linguistic Preprocessing

In this step, the document set is preprocessed in order to characterize texts by their topically significant words. The goal is to extract textual information in the form of individual words from the selected documents. The process begins with a document transformation to deal with the different format types (.doc, .txt, .pdf, xml, etc.). Then, all non-textual information like digits, dates and punctuation marks, is removed from the documents (lexical analysis). Plus, some relevant structural information is extracted (titles, bold and italic words, etc.). Next, collocations were extracted according to the method described in [13]. After that, the system carried out a part-of-speech (POS) tagging process using the Apache Open NLP tool[3]. The POS tagging is the task of labeling each word or token in a sentence with its appropriate syntactic category called part of speech (e.g., adjective, noun, verb, etc.) [14].

Finally, three techniques are used to reduce the vocabulary and make the representation of texts more meaningful: stop words [15], stemming [16] and zipf law, therefore, words that provide low semantic meaning for the domain context are eliminated. Language detection and spelling correction processes are also included in this step.

[3]https://opennlp.apache.org/

2.2.2 Top Concepts Extraction and Storage

The main purpose in this step is to extract a set of chosen concepts which will help to generate a semantic representation of the document content.

In this research work, we used a concept weighting measures to quantify the value or usefulness of a concept in a document and, thus, determine the so-called relevance weight of a concept that was used to select it as "top concept". The concept weighting measure used is an extension of the classic *tf-idf*, which we call *absolute weight* (aw):

$$aw_i = \frac{k * w_{ij}}{|j| * idf^D} \quad (1)$$

The absolute weight of a concept i (aw_i) is calculated mainly using its weight on the document (w_{ij}). Each concept has a weight on each document according to FIS-CRM Model [17]. The fundamental basis of FIS-CRM is to "share" the occurrences of a contained word among the fuzzy synonyms that represent the same concept,and to "give" a fuzzy weight to the words that represent a more general concept that the contained one. In this way, a word may have a fuzzy weight in the new vector even if it is not contained in it, as long as the referenced concept underlies the document. As the total number of concepts in a document ($|j|$) is generally much bigger than a single concept weight, the formula gives a very low-value result. To avoid this problem, a factor k is incorporated to make the result more significant. The value of this factor depends of three parameters: if the concept occurs in a title, if the concept is a collocation, and if the selected word is bold or italic or not. On the other hand, in order to consider the concept discriminative power within a document, it is employed the idf^D factor (Inverse Document Frequency in a Domain), which can be calculated based on the number of times the concept occurs within a domain corpus as a whole. The domain corpus is build from the templates of business documents used in the company or from a standardized set of business documents templates in several languages[4].

The absolute weight value is not used directly as the relevance, but is normalized by dividing it by the average of the values obtained. This operation prevents relevance degrees from appearing as higher in large groups than in small ones.

2.3 Semantic Search

Document Management Repository not support often very large queries, therefore, when users need retrieve documents use short queries in which the number of terms is at most three. Concept-based retrieval arises because it is getting more difficult to satisfy users with only short query-based retrieval.

Query techniques such as query expansion can be very interesting in order to retrieve relevant documents even if they do not contain the words used in the original. This is because the ambiguity of natural language and also the difficulty in using a single term to represent an information concept [18].

The proposed procedure of query expansion in the DMS contains the following steps:

1) *Selection of new terms*: Once the user sends a query to the system, the first step is to build a set of additional queries that complement the original one. In these queries new terms semantically related to those of the original one are included. This process is guided through the set of top concepts previously extracted from the documents (described in the previous section).
2) *Word sense disambiguation*: The main handicap of the search process is managing weak words (words with several meanings) [19]. So, sense disambiguation of this kind of words is fundamental for carrying out this process successfully. The disambiguation of a weak term consists on identifying its right semantic area. Automatic disambiguation methods are suggested by papers such as [20]. In this work, the disambiguation method proposes in [21] is used. This method can be applied when some semantic relation between the terms of the query can be determined. To be able to apply it, it is necessary to have an ontology or thesaurus that allows discovering that relation. An ontology automatically built from the domain templates previously obtained in the semantic annotation process is used for this purpose.
3) *Weighting of new terms*: All terms in the expanded query should have equal weighting or whether the new terms should have a higher/lower weighting, Voorhees [22] found that assigning lower weights to added concepts enhances retrieval accuracy. She used a factor between 0 and 1 for weighting added terms. In this step the approach described in [23] is used with the aim of generating a set of queries/concepts whose meaning is very similar to the original query and which can help to maximize the search process. Each new query can be submitted as an individual request and the final results retrieved by the queries could be merged in order to build a final list of results.
4) *Ranking and filtering*: The relevancy of each object is calculated to determine the order in which they will appear. A ranking of the new items is obtained according the similarity between the expanded query and the original query, the position of the item in the results, and the sum of relevance in all queries (chorus effect).

[4]http://office.microsoft.com/es-es/templates

2.4 Fuzzy Search

Our purpose is to include fuzzy search capabilities in the DMS without modifying its technical architecture. The use of the proposed fuzzy techniques allow the user to write flexible conditions over numeric attributes based on linguistic labels instead of precise (crisp) values.

In our system, an attribute is capable of fuzzy treatment if it is in an ordered underlined fuzzy domain[24], in this case, linguistic labels can be defined on it. Therefore, an attribute can be seen as a linguistic variable. A linguistic variable is a variable whose values are linguistic words or sentences in a natural language[25] and can be divided into various linguistic values (for example: *high*, *low*, *medium*). Triangular and trapezoidal membership functions are used for each linguistic value defined in each quantitative attribute for simplicity. Hence, each linguistic value is a fuzzy number, which is a fuzzy subset in the universe of discourse that is both convex and normal [26].

The number of fuzzy sets refers to the number of partitions that should be done in each ordered continuous domain, i.e., the number of labels that should be used to describe the continuous numeric variable. This number is manually established by the system administrator when the document type is introduced in the DMS. There are an automatic mapping between this number and the labels, for example, if three fuzzy sets are defined, the values are mapped to "high", "medium" and "low" labels.

The system allows to automatically define the fuzzy sets on each continuous domain based on trapezoidal functions and the maximum and the minimum. In a first approach, the attribute A and its membership functions $\mu^{A}_{K,i_m}(x)$ can be represented as follows [24]:

$$\mu^{A}_{K,i_m}(x) = max\{\frac{1-|x-a^{K}_{i_m}|}{b^{K}}, 0\} \quad (2)$$

where:

$$a^{K} = mi + (ma - mi) * \frac{i_m - 1}{K - 1} \quad (3)$$

$$b^{K} = \frac{ma - mi}{K - 1} \quad (4)$$

where K is the predefined number of linguistic values in a linguistic variable, i_m $(1 \leqslant i_m \leqslant K)$ is the i_mth linguistic value of K various linguistic value defined in the attribute , ma and mi are the maximum and minimum values, respectively, of the attribute's domain.

For example, suppose that 3 fuzzy sets should be defined, the fuzzy sets are defined as can be seen in figure 3:

The system also allows to automatically define the fuzzy sets on each ordered continuous domain based on trapezoidal functions and a set of percentiles like the indicate in [27].

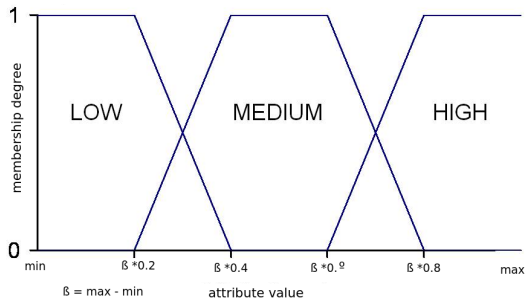


Fig. 3: Fuzzy Partition.

2.5 Document Categorization

Text categorization (also known as text classification) is the task of automatically sorting a set of documents into categories from a predefined set. In automatic text categorization, the decision criterion of the text classifier is usually learned from a set of training documents, labeled for each class.

Each category is established using two different ways, manually or automatically by the application of a clustering algorithm. Once the categories are established, a set of chosen concepts is extracted to semantically defined them. Considering each concept has a weight on each document according to FIS-CRM Model ($fis - crm(c_i, d)$); we distinguished three levels of concept relevance (relevant, subrelevant and other). The weight $kmod^{\zeta}_{i}$ of a concept c_i within a category ζ can be calculated as proposed in the equation 5.

$$kmod^{\zeta}_{i} = \sum_{j \in c} w_{ij} \times \left(1 + \frac{docs\,(c_i, \zeta)}{|D^{\zeta}|}\right) \times Ln\left(\frac{|\zeta|}{\zeta\,(c_i)} + 1\right) \quad (5)$$

where w_{ij} represents the relevance degree of the concept c_i in the document d_j using the FIS-CRM model, $docs(c_i, \zeta)$ is the number of documents in the category ζ in which the concept c_i occurs, $\left|D^{\zeta}\right|$ is the total number of documents considered in the category, $|\zeta|$ is the total number of categories in the context and $\zeta(c_i)$ represents the number of categories in which the concept c_i has a positive membership degree.

According to this formula a concept is regarded as relevant in a category if it occurs more frequently than other concepts in a certain user document set, but occasionally elsewhere. A concept that occurs frequently in several document set could be a relevant domain concept but it is not useful to represent the category.

Once the concepts weights are calculated, it is possible to identify the relevance distribution of all the concepts and use them to construct the groups according to the following statements: *Relevant concepts* (concepts with higher relevance degrees), *Sub-Relevant concepts* (concepts with relevance degrees higher than the average) and *Other concepts* concepts with relevance degree lower than the average.

The category is defined by the most relevant concepts from the Relevant Concepts Set (75% to 100%). These concepts have an associated weight proportional to their importance.

The process to classify new incoming documents into different categories (folders or types) is carried out according to the following steps:

- **Linguistic preprocessing**: The document is automatically analyzed using Natural Language Processing Techniques. (see Section 2.2.1)
- **Conceptual representation**: The document is represented using the FIS-CRM model to consider all the concepts that not occurs in the content (see Section 2.2.2.
- **Comparison with different groups**: A conceptual comparison process will be done between the document and each category (folders or types) obtaining a compatibility degree among these elements (X). This compatibility value depends on two fundamental factors: the number of common concepts and the relevance degree of these common concepts. This way, the result of the conceptual comparison will be obtained from the following formula (Eq. 6):

$$X = Ln\left(\sum_{i \in c}\left(w_i^d * w_i^\zeta\right)\right) * \frac{|(R \cup S) \cap C|}{|R \cup S|} \quad (6)$$

where w_i^d is the weight of the concept i in the document d, w_i^ζ is the weight of the term with in the category, $|R \cup S|$ is the number of relevant and sub-relevant concepts that define the category, and C the set of concepts with weight in the document (originals + obtained from conceptual readjustment).

3. Experiments

This section presents the results obtained from the analysis of the enhanced DMS and its comparison to the results obtained from the original DMS. Accordingly, experiments and evaluations are divided into two sections. User query development, is analyzed in the first section and the second section deals with the document categorization process. The obtained results show us that the use of soft computing and fuzzy logic techniques increases the relevant results and number of the retrieved documents which are consistent with the user's request.

3.1 User Query Development Evaluation

In the case of DMS, it is necessary a test collection to evaluate the effectiveness of the proposed improvements. A test collection is an experimental tool to understand, compare and reproduce the results. In this research work, an Spanish Corpus has been used. This corpus contains more than one thousand files documents collected from State Commission on Access to Public Information of Sinaloa[5] from the years 2008 and 2009. The complete collection includes a lot of business documents like invoices, contracts, balances, letters, etc. Each file is stored in Portable Document Format (pdf) and either represents one page from a document representing a collection of separate sheets. Each page was processed using a standard OCR software[6]. The resulting textual file containing the OCR recognition result is included in the corpus. For every file, a reconstruction of the original text has been prepared using a post-correction process.

Ten queries, established by a panel of experts, has been used to carried out the experiment. Each query is given to the improved DMS and the original DMS and the precision, recall and f-measure values are computed. Table 1 depicts the difference between the enhanced using and no enhanced DMS. It shows that precision per topic after the application of improvements is similar or better than the process without improvements, therefore, the new functionalities implemented in the enhanced has improved retrieval results. This improvement is more evident in the more complex queries, which are less accurate.

Table 1: Results.

	Precision	Recall	F-Measure
DMS+	0.75	0.66	0.70
DMS	0.47	0.32	0.38

3.2 Document Categorization Evaluation

In order the evaluate the document categorization method the TREC 2011 Medical Records track has been used [28]. This track includes a retrieval task aiming to find EMRs that are relevant to a given natural language query. These EMRs are de-identified medical records, provided by the University of Pittsburgh BLULab NLP Repository. There is a total of more than one hundred thousand of medical reports. The categorization algorithm is tested on the types of reports existing in the corpus (radiology reports, and emergency department reports or cardiology reports).

Test collections for experiments are usually split into two parts: one part for training and a second part for testing the effectiveness of the system. To evaluate performance a 2-fold cross-validation is carried out and examined the accuracy as performance measure. The accuracy is defined the ratio of documents correctly classified (Eq. 7).

$$Ac = \frac{N_c}{|D|} \quad (7)$$

Where $|D|$ expresses the total number of documents and N_c is the total number of documents correctly classified.

[5]http://www.ceaipes.org.mx/index.php
[6]https://code.google.com/p/tesseract-ocr/

The results on correctly classified documents are superior to 95%. Comparing this classifier with other similar classifier such as Naive Bayes Classifier, the results are similar in quality or even superior. There exist classifiers which offer in better results than the obtained in this work (SVM) but over experiments more annotated and with a minor volume of documents to deal with.

4. Conclusions

Currently, well tailored and highly personalized document business management systems are needed because there is an overwhelming need for document storage and retrieval. Our approach employs innovative techniques, such natural language processing, soft computing and fuzzy logic that have a large potential for business documents retrieval. The experiments results demonstrate the proposed innovations enhance the information retrieval capabilities of the DMS and improve the user satisfaction. Future work includes working with a larger number of users and a greater volume of documents.

References

[1] M. Hertzum, "Six roles of documents in professionals' work," in *Proceedings of the Sixth Conference on European Conference on Computer Supported Cooperative Work*, ser. ECSCW'99. Norwell, MA, USA: Kluwer Academic Publishers, 1999, pp. 41–60.

[2] M. D. Gordon, "It's 10 a.m. do you know where your documents are? The nature and scope of information retrieval problems in business," *Inf. Processing & Management*, vol. 33, no. 1, pp. 107–122, 1997.

[3] B. Thalheim, "Open problems of information systems research and technology," in *BIR*, ser. Lecture Notes in Business Information Processing, A. Kobylinski and A. Sobczak, Eds., vol. 158. Springer, 2013, pp. 10–18.

[4] H. Zantout and F. Marir, "Document management systems from current capabilities towards intelligent information retrieval: an overview," *International Journal of Information Management*, vol. 19, no. 6, pp. 471–484, 1999.

[5] H. Asili and O. O. Tanriover, "Comparison of document management systems by meta modelling and workforce centric tuning measures," *CoRR*, vol. abs/1403.3131, 2014.

[6] J.-Y. Kuo, "A document-driven agent-based approach for business processes management," *Information and Software Technology*, vol. 46, no. 6, pp. 373–382, 2004.

[7] H. Sahilu and S. Atnafu, "Change-aware legal document retrieval model," in *Proceedings of the International Conference on Management of Emergent Digital EcoSystems*, ser. MEDES '10. New York, NY, USA: ACM, 2010, pp. 174–181.

[8] M.-Y. Chen, H.-C. Chu, and Y.-M. Chen, "Developing a semantic-enable information retrieval mechanism," *Expert Systems with Applications*, vol. 37, no. 1, pp. 322–40, 2010.

[9] P. Rudzajs and I. Buksa, "Business process and regulations: Approach to linkage and change management," in *BIR*, ser. Lecture Notes in Business Information Processing, J. Grabis and M. Kirikova, Eds., vol. 90. Springer, 2011, pp. 96–109.

[10] P. Gomes and M. Antunes, "Mobile edoclink: A Mobile Workflow and Document Management Application for Healthcare Institutions," *Procedia Technology*, vol. 5, no. 0, pp. 932–940, 2012, 4th Conference of ENTERprise Information Systems – aligning technology, organizations and people (CENTERIS 2012).

[11] C. H. Kao and S. T. Liu, "Development of a Document Management System for Private Cloud Environment," in *Proceedings of the 2nd International Conference on Integrated Information (IC-ININFO 2012)*, vol. 73, 2013, pp. 424–429.

[12] M. Nath and A. Arora, "Content management system : Comparative case study," in *Software Engineering and Service Sciences (ICSESS), 2010 IEEE International Conference on*, July 2010, pp. 624–627.

[13] K. Church and P. Hanks, "Word association norms, mutual information, and lexicography," *Computational linguistics*, vol. 16, no. 1, pp. 22–29, 1990.

[14] K. Toutanova and C. D. Manning, "Enriching the knowledge sources used in a maximum entropy part-of-speech tagger," in *Proceedings of the 2000 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora: Held in Conjunction with the 38th Annual Meeting of the Association for Computational Linguistics - Volume 13*, ser. EMNLP '00. Stroudsburg, PA, USA: Association for Computational Linguistics, 2000, pp. 63–70.

[15] R. R. Korfhage, *Information storage and retrieval*. John Wiley & Sons, Inc, 1997.

[16] T. Korenius, J. Laurikkala, K. Järvelin, and M. Juhola, "Stemming and lemmatization in the clustering of finnish text documents," in *CIKM '04: Proceedings of the thirteenth ACM international conference on Information and knowledge management*. New York, NY, USA: ACM, 2004, pp. 625–633.

[17] J. A. Olivas, P. J. Garcés, and F. P. Romero, "An application of the fis-crm model to the fiss metasearcher: Using fuzzy synonymy and fuzzy generality for representing concepts in documents," *Int. J. Approx. Reasoning*, vol. 34, no. 2-3, pp. 201–219, 2003.

[18] J. Bhogal, A. Macfarlane, and P. Smith, "A review of ontology based query expansion," *Inf. Process. Manage.*, vol. 43, no. 4, pp. 866–886, 2007.

[19] A. Soto, J. Olivas, and M. Prieto, "Fuzzy Approach of Synonymy and Polysemy for Information Retrieval," in *Granular Computing: At the Junction of Rough Sets and Fuzzy Sets*, ser. Studies in Fuzziness and Soft Computing, R. Bello, R. Falcon, W. Pedrycz, and J. Kacprzyk, Eds. Springer Berlin / Heidelberg, 2008, vol. 224, pp. 179–198.

[20] R. Navigli, P. Velardi, and A. Gangemi, "Ontology Learning and Its Application to Automated Terminology Translation," *IEEE Intelligent Systems*, vol. 18, no. 1, pp. 22–31, 2003.

[21] P. J. Garcés, J. A. Olivas, and F. P. Romero, "Concept-matching IR systems versus word-matching information retrieval systems: Considering fuzzy interrelations for indexing Web pages," *J. Am. Soc. Inf. Sci. Technol.*, vol. 57, no. 4, pp. 566–576, 2006.

[22] E. M. Voorhees, "Query expansion using lexical-semantic relations," in *SIGIR '94: Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA: Springer-Verlag NY, 1994, pp. 61–69.

[23] J. Serrano-Guerrero, J. Olivas, E. Herrera-Viedma, F. P. Romero, and J. Ruiz-Morilla, "A Model for Generating Related Weighted Boolean Queries," in *Trends in Applied Intelligent Systems, Proceedings of the 23th International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, IEA/AIE 2010, Cordoba, Spain*, ser. Lecture Notes in Artificial Intelligence, vol. LNA 6098. Springer, 20010, pp. 429–438.

[24] Y.-C. Hu, R.-S. Chen, and G.-H. Tzeng, "Discovering fuzzy association rules using fuzzy partition methods," *Knowledge-Based Systems*, vol. 16, no. 3, pp. 137–147, 2003.

[25] L. Zadeh, "The concept of a linguistic variable and its application to approximate reasoning," *Information Sciences*, vol. 8, no. 3, pp. 199–249, 1975.

[26] W. Pedrycz and F. Gomide, *An Introduction to Fuzzy Sets: Analysis and Design*, ser. Complex adaptive systems. NetLibrary, Incorporated, 1998.

[27] F.-J. Lopez, A. Blanco, F. Garcia-Alcalde, and A. Marin, "Extracting biological knowledge by fuzzy association rule mining," in *FUZZ-IEEE*. IEEE, 2007, pp. 1–6.

[28] E. M. Voorhees, "The trec medical records track," in *Proceedings of the International Conference on Bioinformatics, Computational Biology and Biomedical Informatics*, ser. BCB'13. New York, NY, USA: ACM, 2013, pp. 239–246.

Summarizing Information by Means of Causal Sentences

C. Puente[1], J. A. Olivas[2], I. Prado[1],
[1] Advanced Technical Faculty of Engineering – ICAI, Comillas Pontifical University, Madrid, Spain
[2] Information Technologies and Systems Dept, University of Castilla-La Mancha Ciudad Real, Spain

Abstract –*In this paper we explore the role that causal sentences play to summarize information. In particular we analyze causality in texts belonging to medicine and law, to evaluate if it is possible to establish a mechanism able to store the most important ideas of a text without losing relevant information.*

Keywords: Causal sentences, Causality, Causal representation.

1 Introduction

Causality plays and has played an important role in human cognition, in particular in human decision-making, providing a basis for choosing an action which is likely to lead to a desired result. There are many works and theories about this theme, philosophers, scientists, physics, mathematicians, computer scientist and many others have explored the field of causation starting with the ancient Greeks three thousand years ago.

In particular, causality is a fundamental notion in every field of science, in fact there is a great presence of causal sentences in scientific language [1]. Causal links are the basis for the scientific theories as they permit the formulation of laws. In experimental sciences, such as physics or biology, causal relationships only seem to be precise in nature. But in these sciences, causality also includes imperfect or imprecise causal chains.

Regarding causality in law [2], causation is the "causal relationship between conduct and result". Causation of an event by itself is not sufficient to create legal liability, though establishing causation is required to establish legal liability. Usually it involves two stages:

1) The first stage involves establishing 'factual' causation. For example, Did the defendant act in the accuser's loss?.
2) The second stage involves establishing 'legal' causation. For example, Is this the sort of situation in which, despite the outcome of the factual investigation, the defendant might be released from liability, or impose liability?

There have been many works related to causal extraction in text documents, like Khoo and Kornfilt [3], who developed an automatic method for extracting causal information from Wall Street Journal texts, or Khoo and Chan [4] who developed a method to identify and extract cause-effect information explicitly expressed without knowledge-based inference in medical abstracts using the Medline database.

On the other hand, the ability to synthesize information losing as little information as possible from a text has been studied in various disciplines throughout history. In this area the work of linguistics, logic and statistics [5] are the most relevant issues to take into consideration to propose a suitable answer to a given question.

Taking these works as inspiration, this paper deals with two experiments. The first one is focused on the extraction of causal sentences from texts belonging to medicine, using them as a database of knowledge about a given topic. Once the information has been selected, a question is proposed to choose those sentences where this concept is included. These statements are treated automatically in order to achieve a graphical representation of the causal relationships with nodes labeled with linguistic hedges that denote the intensity with which the causes or effects happen, and the arcs marked with fuzzy quantifiers that show the strength of the causal link. In turn, nodes are divided into cells denoting location, sub location or other contextual issues that permit a better understanding of the meaning of the cause. The node 'cause' includes also a special box indicating the intensity with which the cause occurs.

Once the graph has been created, a second algorithm is in charge of the generation of a summary by reading the information represented by the causal graph obtained in the previous step. Redundant information is removed, and the most relevant information is classified using several algorithms such as collocation algorithms like SALSA or classical approaches like keywords depending on the context, TF-IDF algorithm.

The summary is generated in natural language thanks to another algorithm which is able to build phrases using a generative grammar.

With these processes, we have tested the performance with medical texts and with legal ones, to check if it is possible apply the same procedure to different environments.

2 Selection of the input information

In [6], Puente, Sobrino, Olivas & Merlo described a procedure to automatically display a causal graph from medical knowledge included in several medical texts.
A Flex and C program was designed to analyze causal phrases denoted by words like 'cause', 'effect' or their synonyms, highlighting vague words that qualify the causal nodes or the links between them. Another C program receives as input a set of tags from the previous parser and generates a template with a starting node (cause), a causal relation (denoted by lexical words), possibly qualified by fuzzy quantification, and a final node (effect), possibly modified by a linguistic hedge showing its intensity. Finally, a Java program automates this task. A general overview of the extraction of causal sentences procedure is the following (fig.1):

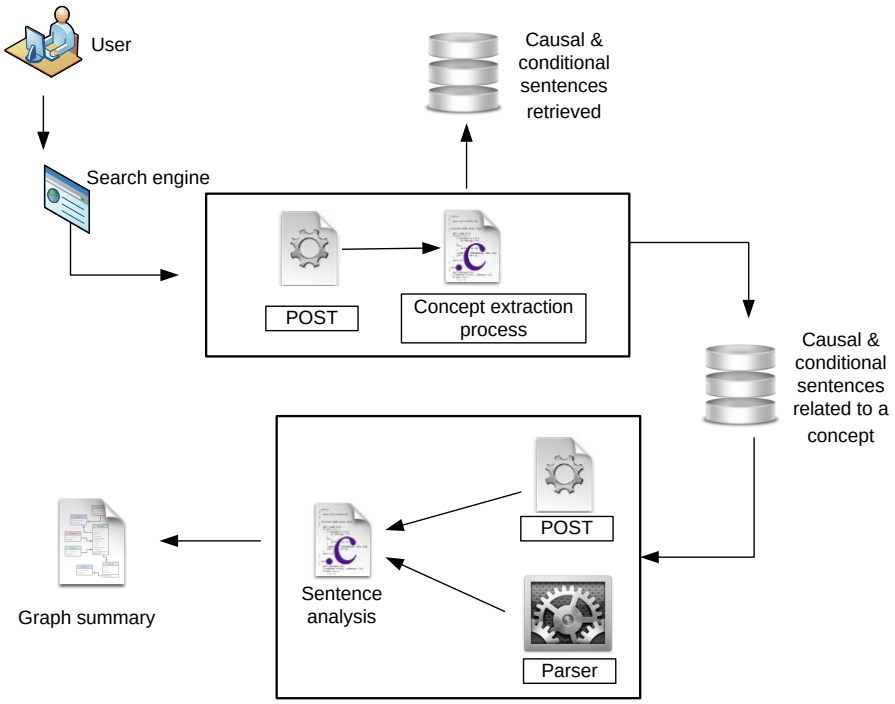


Fig. 1 Procedure to extract causal sentences from text documents.

Once the system was developed, an experiment with medical texts was performed to answer the question *What provokes lung cancer?*, obtaining a set of 15 causal sentences related to this topic which served as input for a causal graph representation.

The whole process was unable to answer the question directly, but was capable of generating a causal graph with the topics involved in the proposed question as shown in figure 2.

3 Summarizing the content of a graph

With this causal graph, in [7] we wanted to go a step further to generate the answer to the proposed question by means of a summary, processing the information contained in the causal nodes and the relationships among them.

The ideal representation of the concepts presented in Fig. 2 would be a natural language text.
The size of the graph could be bigger than the presented one as not all the causal sentences are critical to appear in the final summary and in the graph (they may contain redundant information). It is necessary to create a summary of the information of the graph in order to be readable by a human as if it was a text created by another human.
The causal graph presented in Fig. 2 has the problem that the concepts represented could have a similar meaning in comparison to other concepts. For example, "smoking" and "tobacco use" have the similar meaning in the graph so one of these concepts could be redundant.
Not only is this relationship of synonymy but other semantic relations such as hyperonymy or meronymy are important as well.

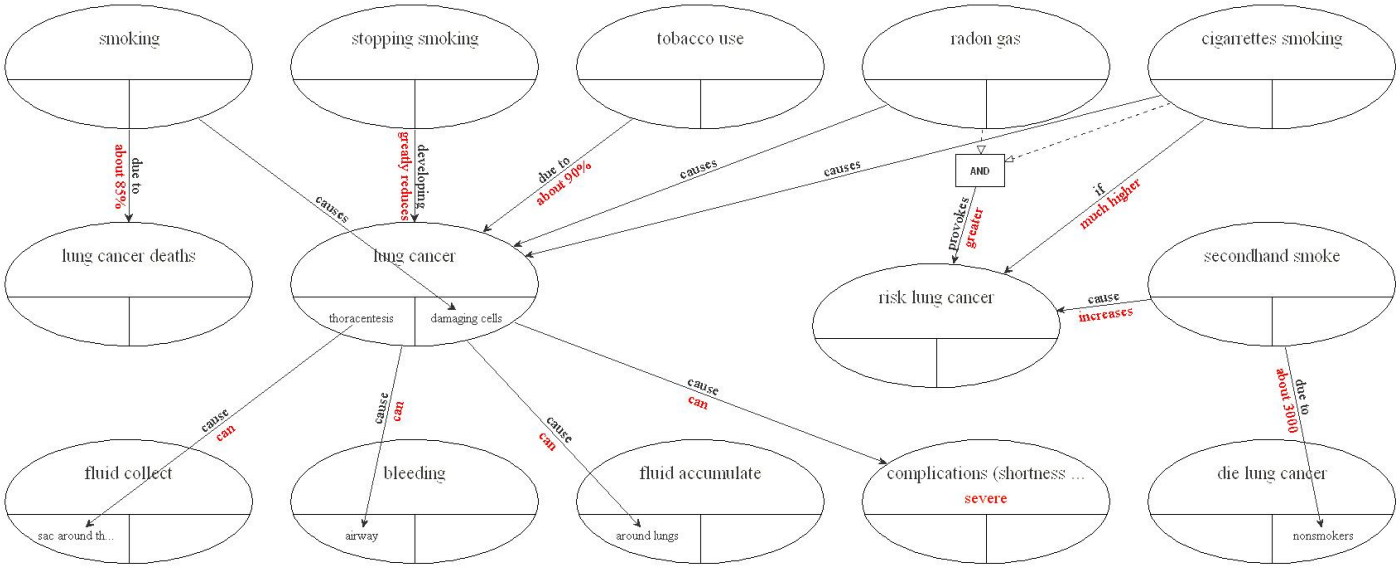


Fig. 2 Medical causal graph.

It is possible to create a process to read the concepts of the graph sending them to an ontology like Wordnet or UMLS and retrieving different similarity degrees according to each relationship [8].

Several analysis and processes need to be executed to obtain a summary by means of an automatic algorithm. The following diagram (Figure 3) shows the design of the summary system created to solve this issue, including the main processes and the main tools needed.

This procedure is not simple, due to the fact that the meaning of the words has to be discovered with help of the context.

A redundancy analysis process is created to solve this problem, taking into account the multiple synsets of every word of analyzed concepts. It is also taken into account the context of the text having keywords of every context and other measures.

To do so, Wordnet synsets are queried from Java thanks to Jwnl and RiWordnet tools to find out the meaning of these terms. The output of the process will consist of the possible relations between all pairs of entities compared, declaring the type and intensity of each relationship.

The degree of their similarity with other concepts is computed as well, being a measure to take into account in the relevance analysis. Different algorithms of similarity between concepts such as Path Length, Leacock & Chodorow [9] or Wu Palmer [10], are executed through platforms like Wordnet::Similarities.

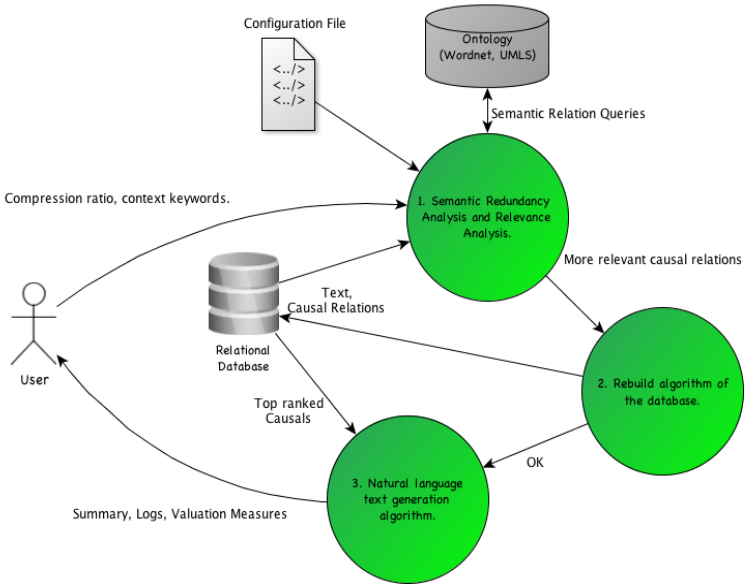


Fig. 3 Basic design of the summarization process.

After these analyses, the information of the graph is summarized (in base to a compression factor introduced by the user), obtaining the following graph:

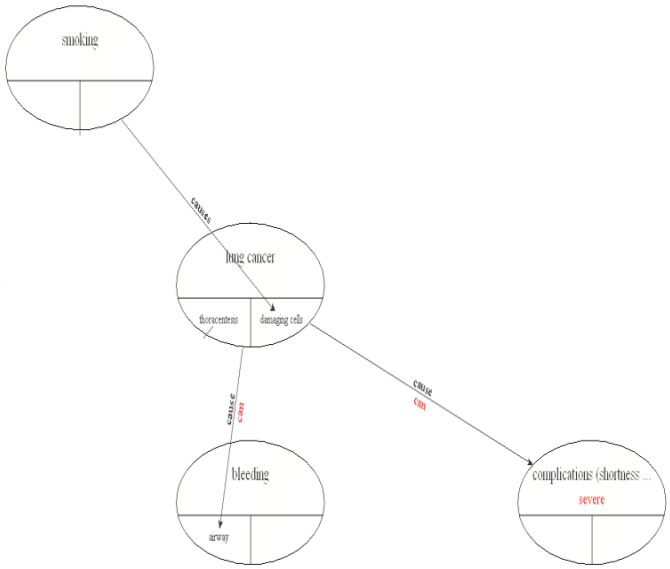


Fig. 4. Causal graph summarized.

The summary process has a configuration module depending on the users preferences and the nature and context of the text to be analyzed. All the modules and measures can be parameterized by means of a weight-value algorithm.
In order to have a better reading of the graph, the information needs to be expressed in natural language. The last process consists of an algorithm that generates natural language given the top ranked causals by the semantic redundancy analysis and relevance analysis.
We have performed two experiments with medical texts, varying the compression rate to evaluate the obtained results and check the configuration of the algorithm.

In the first experiment, we used a compression rate of 0.3, obtaining as a result the following summary:

"*Cigarettes smoking causes die lung cancer occasionally and lung cancer normally.Tobacco use causes lung cancer constantly and die lung cancer infrequently.Lung cancer causes die lung cancer seldom and fluid collect sometimes.*
It is important to end knowing that lung cancer sometimes causes severe complication."

The original text length was 1497 words and the summary length was 311 so the system has been able to achieve the compression rate, being the summary less than the 30% of the original text.

In this case, the main information has been included, removing redundant information. The system has chained sentences with the same causes to compose coordinate sentences and reduce the length of the final summary. As seen, the grammatical and semantic meaning is quite precise and accurate, without losing relevant information.

In the second experiment, the compression rate was the lowest, to remove all the redundant and irrelevant information, so the summary represented a 10% of the original text. The result was the following:

"*Lung cancer is frequently caused by tobacco use. In conclusion severe complication is sometimes caused by lung cancer*."

In this case, the system just takes the information of the three most relevant nodes, one cause, one intermediate node, and an effect node, creating a summary with the most relevant nodes included in the graph.

As it can be seen, the length of the summary is of 118 words, what represents less than a 10% of the length of the original text, the system is able to modify its behaviour according to different configurations of the weights of the redundance and relevance algorithms and the compression rate.

4 Evaluation of legal texts

These experiments have been performed using medical texts. In [11] we explored the role of causality in other fields like legal ones. The conclusion that we obtained was that legal text had a high representation of causal statements, so we decided to see if this amount was enough to summarize legal statements.

To perform these experiments, we run the extraction and analysis algorithms to extract causal sentences included in legal texts (fig. 1). We selected 10 cases related to the prohibition of torture, and the results of causal sentences were the following:

Document title	Nª Pages	Nº of words	Detected	Classified	Not classified	%
Ireland v. United Kindom	5	2.180	6	6	0	100,0
Aksoy v. Turkey	2	880	3	3	0	100,0
Aydin v. Turkey	2	797	0	0	0	0,00
Selmouni v. France	4	1.770	1	0	1	0,00
Chitayev v. Rusia	2	737	2	1	1	50,00
Gäfgen v. Germany	5	2.424	7	3	4	42,86
Ribitsch v. Austria	2	1.046	2	2	0	100,0
Price v. United Kindom	3	1.397	13	13	0	100,0
Selcuk v. Turkey	2	661	2	2	0	100,0
Kurt v. Turkey	2	775	2	2	0	100,0

Fig. 5. Causality in legal texts.

As seen in fig. 5, the amount of causal sentences in this type of texts is very low, being 13 in the best situation. With these results, we analysed if in the case of Price v. United Kindom we had enough information to summarize the facts and establish a causal chain of events. The result was that when we deal with lower amounts of information, vital information is missing, as in the case of the sentences: *"The applicant is four-limb deficient as a result of phocomelia due to thalidomide. She also suffers from problems with her kidneys"* The program was able to detect and classify the first sentence, but *She also suffers from problems with her kidneys* was missing, being relevant information for understanding the case.

5 Conclusions

This work proposes a set of algorithms to summarize information from a set of texts related to the same themes.

As seen in the experiments, results vary in base to the field of study and the amount of information. When it is something general with a wide knowledge and information about it as in the medical case, results are quite useful to provide a summary of information. On the other hand when is something concrete as in the case of the legal sentences, a summary can provide information, but can miss relevant information to understand the development of the causal chain.

As future work, we will explore more fields of study were information may be expressed in a causal way.

6 Acknoledgements

Partially supported by TIN2010-20395 FIDELIO project, MEC-FEDER, Spain.

7 References

[1] M. Bunge, Causality and modern science. Dover, 1979.

[2] A.C. Becht and F.W. Millar, The Test of factual Causation in Negligence and Strict Liability. St. Louis: Committee on Publications, Washington University, 1961.

[3] C. Khoo, J. Kornfilt, R. N. Oddy, S. H. Myaeng, Automatic extraction of cause-effect information from newspaper text without knowledge-based inferencing. Literary and Linguistic Computing. Vol 13(4), pp. 177-186. (1998).

[4] C. S. G. Khoo, S. Chan, Y. Niu, Extracting causal knowledge from a medical database using graphical patterns. In ACL Proceedings of the 38th Annual Meeting on Association for Computational Linguistics, pp. 336-343, Morristown, USA. (2000).

[5] R. Basagic, D. Krupic, B. Suzic, "Automatic Text Summarization", Information Search and Retrieval, WS 2009, Institute for Information Systems and Computer . Media, Graz University of Technology, Graz, (2009).

[6] C. Puente, A. Sobrino, J. A. Olivas, R. Merlo, Extraction, Analysis and Representation of Imperfect Conditional and Causal sentences by means of a Semi-Automatic Process. Proceedings IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2010). Barcelona , Spain , pp. 1423-1430, 2010.

[7] C. Puente, E. Garrido, J. A. Olivas, Answering Questions by Means of Causal Sentences. Proc. Of the 10th Intenational Conference Flexible Question Answering Systems FQAS, Granada, Spain, pp. 91-100, 2013.

[8] G. Varelas, E. Voutsakis, P. Raftopoulou, G. M. E. Petrakis, E. Milios. Semantic Similarity Methods in WordNet and their Application to Information Retrieval on the Web. WIDM'05, Bremen, Germany. November 5, (2005).

[9] C. Leacock, M. Chodorow, Combininglocal context and WordNet similarity for word sense identification. In Fellbaum 1998, pp. 265–283.

[10] Z. Wu, M. Palmer, Verb semantics and lexical selection. In 32nd Annual Meeting of the Association for Computational Linguistics, pp.133-138, Resnik, 1994.

[11] C. Puente, J. A. Olivas, Analysis, detection and classification of certain conditional sentences in text documents. Proceedings of the 12th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, IPMU'08, Torremolinos, Spain, 2008.

SESSION

POSTERS

Chair(s)

TBA

Computational Evaluation of Planarian Regeneration Models

Marianna Budnikova, Jeffrey W. Habig, and Tim Andersen
Computer Science, Boise State University, Boise, ID, USA

Abstract— *We have combined a cell-based modeling platform with a database of formally encoded morphological experiments and their outcomes in order to automate the search for computational models of cellular networks involved in maintenance of morphology and to validate such models against existing experimental outcomes. To achieve automated validation of virtual cellular models, we have developed an algorithm for converting a formal graph representation of planarian morphologies into a polygon representation which can be overlayed with a cell-based model. We show how the overlay algorithm can be used as a basis for the fitness function to guide automated searches for models of cellular development.*

Keywords: planaria, regeneration, tissue modeling

1. Introduction

For the last 200 years scientists have been trying to understand and explain the processes behind the regeneration of planaria worms [1], [2]. A large amount of experimental data has been accumulated on planarians over the years, and a plethora of planarian regeneration models have been proposed [2] [3] [4]. However, the proposed models are able to explain only a few experimental outcomes. The studies performed on planarians could greatly advance the understanding of planarian regeneration if they were united in one central database and combined with a computational search engine. Recently, the Levin lab introduced a graph-based formalism for a database to store planarian experiments. [5]. Flexible graph notation allows organisms to be described in terms of nodes connected by links at specific angles. Due to the flexibility of the graph formalism, the PlanformDB database allows searching the experiments performed on planaria not only by keywords, but also by the worm's shape as the key. PlanformDB can be used to create and validate new regeneration models if combined with an automated generate-and-test tool [6]. A great example of such a tool is the cell-based modeling platform used by the Andersen lab to discover models of planarian regeneration by performing evolutionary search [7].

We are working on combining the PlanformDB and the CellSim evolutionary search engine to develop an automated system for searching and validating computational models of development. The automation calls for mechanisms to compare individuals produced in the simulation platform to morphologies taken from the database. This comparison method will serve as a fitness function to guide the evolutionary search for planarian regeneration models. In this work we present a novel fitness function that compares individuals produced in CellSim to PlanformDB morphologies by creating a graph to cell overlay.

2. Graph Edit and Overlay Fitness Functions

During the evolutionary search, the CellSim platform simulates an experiment taken from PlanformDB and returns the cellular outcome of the experiment. In our experiments, the cellular morphology was approximated starting with a flat sheet of 420 autonomous cells, where each cell is capable of growing, dividing, dying and regulating metabolic and genetic networks in response to changes in their local environment (Fig. 1). The cellular morphology returned by the simulation platform is compared to the target graph from the PlanformDB database, yielding a fitness value that can be used to guide the evolutionary search.

We have been using a graph matching algorithm to calculate the fitness of the regeneration model. With this approach, a cellular morphology generated by the simulation platform is converted into graph representation and compared against the target graph using the graph edit distance technique [8]. The graph edit distance evaluator is a very flexible fitness function, since the graph formalism can be used in simulation platforms other than CellSim. However, the evaluator is not optimal for the purposes of the evolutionary search due to the coarse grained nature of how a cell region type is determined during the conversion of a cell-based morphology into graph. The assignment of a cell to a region is a very complex question, since many factors determine whether a cell belongs to a particular region. The component gathering algorithm used to convert a cellular morphology to graph assigns a cell to a region type based on the maximum concentration of the indicator molecule. This approach may not work well when a cell has several indicator molecules and thus the potential of becoming the sought-for region type. For example, consider two cells, where the head indicator concentration of the first cell is 0.1, the second cell 0.9, and both cells have the trunk indicator concentration of 1.0. Suppose the sought-for region type is head. The graph edit distance algorithm will assign both of the cells to the trunk region resulting in equal fitness, even though the second cell is closer to becoming the head type.

We would like to be able to reward cells that are close to becoming the sought-for type. In the example above, the second cell with 0.9 concentration of the head indicator should be rewarded more than the cell with 0.1 concentration. To approach this problem, we have designed and implemented an overlay evaluator as an alternative to the graph edit distance evaluator. The overlay works by taking the converted graph representation of the target morphology and pasting it onto the cellular outcome from the simulation platform. Once the overlay is created, each cell in the cellular morphology can be examined individually against the closest graph region.

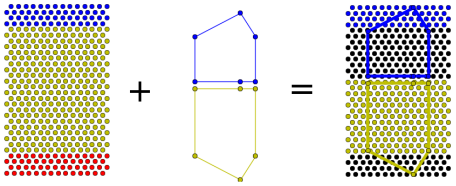


Fig. 1: In this example of overlaying a cellular morphology with a graph morphology converted into polygons, the cellular morphology consists of three regions, head (blue), trunk (yellow) and tail (red), while the overlayed graph only has a head and a trunk. When the graph is overlayed with the cellular morphology, some of the cells do not match the type of the overlayed graph (colored in black).

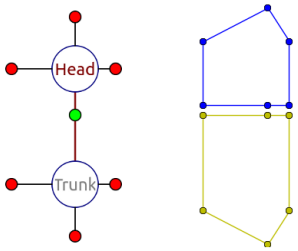


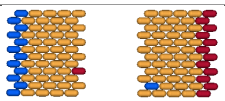
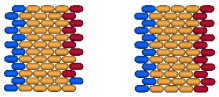
Fig. 2: Sample conversion of a two-node morphology graph into lines. The left image shows a graph for morphology consisting of connected head and trunk regions. The lines with red dots extending from the centers of the regions indicate the general shape of the morphology. The right image shows this graph converted into polygons consisting of connected lines.

In contrast to graph edit distance where a cell is labeled as one region, a cell in the overlay evaluator is examined in respect to many of the cell's features. Therefore, the overlay evaluator can reward a cell that is close to becoming the expected region type. The current graph formalism as presented in the Planform database is not conducive to being used as an overlay. For our overlay implementation we chose to cover every cell in the simulation platform outcome by the overlaying structure. This overlaying structure was obtained by converting the region nodes of the graph morphology into polygons (Fig. 2). The algorithm starts by converting the graph into polygons. Then, the best fit of the graph polygons to a cellular morphology is calculated by scaling the graph so the graph's height and width match those of a cellular morphology 1. For each cell in the cell-based morphology, the closest polygon region of the overlayed polygon graph is found. A subfitness value between 0.0 and 1.0 is calculated for each cell measuring how far the cell is from becoming the expected region. The final fitness of the model in the overlay difference evaluator is calculated by dividing the sum of the subfitness values for every cell by the number of cells.

3. Experiments and Conclusion

We have run several evolutionary searches with various parameters. To get the starting population for the search, different knockouts were performed on a hand-crafted gradient regeneration model, such as removing the promoters of head and tail in the genome and the head regeneration equations in the metabolic description of the model. Table 1 shows the kinds of solutions that the GA found. Morphology with ID 1 was found using the graph edit distance as a fitness function, and individual with ID 2 was found with the overlay evaluator.

Table 1: Graph Edit Distance and Overlay Evaluator Fitness Values for Different Morphologies

ID	Morphology	Graph Edit Distance	Overlay
1		0.983	0.817
2		0.587	0.978

The first morphology had only one cell of head and tail regions regenerated. Graph edit distance fitness function converted those cells into full regions, and assigned a high fitness to this morphology. This fitness assignment is not intuitive, since one would expect a head or a tail region to have more than one cell to be considered a head or a tail. The overlay evaluator matched our intuition and assigned a lower fitness of 0.817 to this individual. The second morphology found by the GA search closely resembled the target model, except that it had a single head generated in the tail. The graph edit distance assigned this morphology a very low fitness value, since the algorithm considered the single head cell as an extra added region. When we examined this cell, it had 32.0 concentration of head indicator, and 14.0 concentration of tail indicator, which means it was slowly evolving into tail region. From these experiments, it can be seen that the overlay evaluator produced more intuitive fitness values from evolutionary search prospective than the graph edit distance evaluator. In future work, we plan to run more complex searches where we knock down the equations responsible for entire region regeneration.

References

[1] P. A. Newmark and A. S. Alvarado, "Not your father's planarian: a classic model enters the era of functional genomics," *Nat Rev Genet*, vol. 3, no. 3, pp. 210–9, March 2002.

[2] D. Lobo, W. S. Beane, and M. Levin, "Modeling planarian regeneration: A primer for reverse-engineering the worm," *PLoS Comput Biol*, vol. 8, no. 4, 2012.

[3] H. Meinhardt, "Models for the generation and interpretation of gradients," *Cold Spring Harbor Perspectives in Biology*, vol. 1, no. 4, 2009.

[4] F. Azuaje, "Computational discrete models of tissue growth and regeneration," *Briefings in Bioinformatics*, vol. 12, no. 1, pp. 64–77, 2010.

[5] D. Lobo and M. Levin, "A new bioinformatics of shape for regenerative science," *Proceedings of the American Association for the Advancement of Science, Pacific Division*, vol. 31, no. 1, pp. 63–65, 2012.

[6] O. Grandstrand, *Agent Based Evolutionary Search*. Berlin, Heidelberg: Springer-Verlag, 2010, ch. Agent Based Evolutionary Approach: An Introduction.

[7] T. Andersen, R. Newman, and T. Otter, "Shape homeostasis in virtual embryos," *Artificial Life*, vol. 15, no. 2, pp. 161–183, 2009.

[8] M. Neuhaus and H. Bunke, *Bridging the Gap Between Graph Edit Distance and Kernel Machines*. Singapore: World Scientific, 2007, ch. Chapter 3. Graph Edit Distance.

Current Events Information Providing System based on Personal Information and General Importance

[1]Hirokazu Watabe, [2]Misako Imono, [3]Eriko Yoshimura, and [4]Seiji Tsuchiya
Dept. of Intelligent Information Engineering and Science,
Doshisha University, Kyotanabe, Kyoto, 610-0394, Japan
[1,2]hwatabe@mail.doshisha.ac.jp, [3] sk109716@mail.doshisha.ac.jp, [4]stsuchiy@mail.doshisha.ac.jp

Abstract *- In this study, we took into account the personal information of users and the general importance of current events information to develop a system that provides current events information that is suited to the user. The current events information that was provided consisted of the headlines and body text of articles found on the websites of newspaper companies.*

Keywords: current events information, degree of association, Concept-Base

1 Concept-Base and Degree of Association

A concept base(CB)[(1)] is a knowledge base comprising the aggregate of words(concepts) that are mechanically constructed from multiple dictionaries and newspapers. As shown in the following equation, concept A is defined by the aggregate of words(attributes) a_i that express the meanings and characteristics of the concept, and their weight w_i.

$$A = \{(a_1, w_1), (a_2, w_2), \cdots, (a_n, w_n)\} \quad (1)$$

However, each attribute a_i must be a concept that is registered to the CB. In this study, we defined concepts that are not registered to the CB as undefined words.

The Degree of Association(DoA)[(1)] is a value ranging from 0.0 to 1.0 that quantifies the strength of the relationship between words and words registered to the CB. Conceptualization attaches the attributes and weight to a given word of the format of Equation (1).

2 Proposed System

The proposed system acquires the current events information and implements conceptualization. The system then determines which current events information is of interest to a user by using the personal information of the user as a preference. The system also takes into account the frequency of current events information and the sex and age group of the user to determine which current events information is high in general importance to the user. By combining these two results, the system can provide current events information that is suited to the user.

2.1 Acquisition of Current Events Information and Conceptualization

We gathered current events information by using the websites of three newspaper companies: asahi.com, Yomiuri Online, and Mainichi jp. First we collected headlines and body text from the websites of each newspaper company. We then conceptualized the current events information using the headlines as the concepts, and independent words in the body text as attributes. We applied the TfIdf value that used the current events information for the last month, for the weights of the attributes. By implementing this processing to conceptualize the current events information, we could calculate the DoA and quantify the relationship of current events information and information that concerns the user.

2.2 Assignment of Points to Current Events Information Based on Personal Information

We obtained personal information by having users directly record words for items that we established beforehand, then determined which current events information would be of interest to the users. The items were divided into two main categories: person information and preference information. There were 13 items for person information including names and places of birth, and 16 items for preference information including animals and sports. Reference information items were divided into likes and dislikes. For example, in the case of animals, there were items for liked animals and disliked animals.

We examined the relationship of the acquired personal information and current events information. A strong relationship to the person information or preference information suggests that the user will be interested in the current events information. We processed the person information as follows. When there is a word that relates to the user in the current events information, it attracts the user's interest. Therefore, we checked to see if there was a matching description in the current events information for words that were contained in the person information, and used the percentage of words as the relationship to the person information. We then used the DoA to the current events information as the relationship to the preference information. We conceptualized words that were contained in the preference information items as having an attribute and weight of "1" and conceptualized the likes and dislikes respectively. We then examined the DoA to the current events information and deducted the DoA to dislikes from the DoA to likes, and used this as the relationship based on preference information. We based this on the reasoning that current events information with a strong DoA to dislikes is not of interest to the user as a topic. Lastly, we added the relationship based on person information and the relationship based on preference information, and used this as the points for the current events information.

2.3 Assignment of Points to Current Events Information Based on General Importance

We assigned points to current events information based on general importance, by taking into account the frequency of current events information and preferences by sex and age group. If the points assigned by frequency and preferences are not high, it suggests that it is not high in general importance, so we multiplied the respective points and used the resulting value as the points based on general importance. We used the following methods to assign the respective points.

2.3.1 Assignment of Points Using Frequency

The more often that news is reported, the greater the general importance of the news. Therefore, we gathered headlines for one day's worth of articles and extracted the nouns in each headline. We checked the frequency of each noun's appearance and used it as the weight of the noun. We then added up the weight of the nouns in the headlines and divided it by the number of nouns, which we used as the points for the current events information.

2.3.2 Assignment of Points Taking into Account Preference Information by Sex and Age Group

We used a trending ranking[(2)] to acquire preference information by sex and age group, and examined the relationship to current events information in order to assign points to the current events information. For the trending ranking, we used the search engine from the BIGLOBE site to tabulate and rank the top 20 trending searched words by sex (man/woman) and age group (from teens to 50s). In this study, we acquired the trending words for the last week, as keywords indicating the preference information by sex and age group.

We conducted auto-feedback[(3)] (AF; a method that uses the Web to capture combinations of attributes that indicate the characteristics of undefined words and their weights) for all keywords acquired by sex and age group, and acquired attributes indicating the connotations of keywords. When there are multiple keywords with the same attributes, it suggests that a topic with those attributes will be of interest to that sex and age group. By examining the number of duplications, we acquired the top 20 word attributes ranked by frequency. We then conceptualized these words using the attributes, as genres that are of interest. We assigned weights based on the ranking by number of duplications. Due to the characteristics of the trending ranking, there were few cases in which the same keyword appeared multiple times. Therefore, when the same keyword was acquired multiple times, we judged that there was interest in the keyword itself and we acquired the keyword. We then used the attributes to conceptualize the words that we acquired, as keywords that were of interest. For the weight, we used the number of duplications of the keywords.

We assigned points by examining the relationship of the preference information for the same sex and age group as the user, and the current events information. We calculated the DoA between current events information and the conceptualized genres and keywords respectively, and summed them as the points for the current events information.

2.4 Provision of Current Events Information

We used the value derived by summing the points based on personal information and points based on general importance, as the points for the current events information. We ranked the current events information starting from the highest points and output the results.

3 Evaluation

We then evaluated the output from the system. The test subjects viewed the headlines and body text for all current events information beforehand, and judged whether the current events information was of interest to them. When the current events information was judged to be of interest to the test subject, it was considered to be correct. For the evaluation index, we used the percentage of number of information that was of interest in the top 20.

For the experiment, we used current events information gathered over a three-day period from December 2–4, 2013, and the trending ranking from November 25 to December 3, 2013. We collected personal information from five test subjects, and conducted the evaluation using a total of 15 types of output results from five test subjects and for the three days of the experiment. For comparison, we also used the results from taking into account only the personal information from the test subjects, and results from taking into account only the general importance.

We found that the accuracy increased by approximately 5% compared with other results, and that the system gave a high ranking to current events information that cannot be provided by personal information or general importance alone.

4 Conclusion

In this study, we proposed a system that provides current events information that is suited to the user by using the two perspectives of personal information of users and general importance of current events information.

Acknowledgements

This research has been partially supported by the Ministry of Education, Science, Sports and Culture, Grant-in-Aid for Scientific Research (Young Scientists (B), 24700215).

References

(1) H. Watabe and T. Kawaoka: "The Degree of Association between Concepts using the Chain of Concepts", Proc. of SMC2001, pp.877-881, 2001.
(2) http://search.biglobe.ne.jp/ranking/
(3) H. Watabe, E. Yoshimura, and S. Tsuchiya: "Development of a System for Providing Current Events Information Based on User-Profile Information", Proc. of ICAI2010, Vol.I, pp.202-208, CSREA Press, 2010.

Judging Emotion from EEGs Using a Clustered EEG Feature Knowledge Base

Seiji Tsuchiya[1], **Misako Imono**[2], **Eriko Yoshimura**[1], **and Hirokazu Watabe**[1]
[1]Dept. of Intelligent Information Engineering and Science, Faculty of Science and Engineering, Doshisha University, Kyo-Tanabe, Kyoto, Japan
[2]Dept. of Information and Computer Science, Graduate School of Engineering, Doshisha University, Kyo-Tanabe, Kyoto, Japan

Abstract - *For a robot to converse naturally with a human, it must be able to accurately gauge the emotional state of the person. Techniques for estimating emotions of a person from facial expressions, intonation and speech content have been proposed. This paper presents a technique for judging the emotion of a person using EEGs, differing from the data used to date. Accuracy of emotion judgment was 33.0%, which is still low, but we believe we obtained sufficient clues that will lead to improved accuracy, such as the validity of knowledge base clustering.*

Keywords: EEG, judging emotion, clustered knowledge base

1 Introduction

For a robot to converse naturally with a human, it must be able to accurately gauge the emotional state of the person. Techniques for estimating emotions of a person from facial expressions, intonation and speech content have been proposed. This paper presents a technique for judging the emotion of a person using EEGs, differing from the data used to date.

2 Overview of Proposed Technique

The objective of this technique was to read the emotions of a conversation partner from EEGs.

EEGs acquired from the subject are used as source EEGs. Emotions of the subject at that time are acquired simultaneously. Spectrum analysis of the source EEGs to which emotion flags have been assigned is performed every 1.28 s, and the EEG features of θ waves, α waves and β waves are determined. A knowledge base of EEG feature is created by clustering the determined EEG features.

To judge emotion from EEGs, the distance between EEG features of the EEG to be judged and the data in the EEG feature knowledge base to which emotion flags have been assigned was determined in this study, and emotions were then judged. Emotions judged in this study were pleasure, anger, sadness, and no emotion.

3 Acquision of Source EEGs and Emotions

EEGs were measured at 14 locations, at positions conforming to the International 10-20 system [1]. Subjects fitted with an electroencephalography [2] cap were asked to watch a Japanese film for approximately 2 h while trying to gauge the emotions of the speakers in the film, and source EEGs were acquired. Images were frozen for each of the 315 speakers in the film, and the subject was asked what emotion the speaker was feeling at that time.

Sixteen subjects were used, and viewing was divided into four sessions to reduce the physical burden on subjects. Before and after the film, EEGs corresponding to open-eye and closed-eye states were measured for approximately 1 min each, and these data were used when normalizing EEG features.

4 Normalization of EEG Features

EEGs show changes in voltage intensity over time within an individual, and base voltage intensity differs among individuals. For this reason, the possibility of misjudgment exists because those values differ greatly even among EEGs with similar waveforms. To solve this problem, linear normalization and non-linear normalization were performed.

4.1 Linear Normalization

This was performed to take into account how EEGs vary over time depending on the subject. Since the eyes were open while viewing the film, linear normalization was performed based on EEG features from the eyes-open state, acquired both before and after the experiment.

EEG feature $Linear\_al_{ij}$, obtained by linear normalization of first EEG feature al_{ij} at a certain point in time during the experiment, is expressed by Formula 4.1:

$$Linear\_al_{ij} = al_{ij} + \left\{ \left(\frac{q_1 - q_2}{p_2 - p_1} \times l + q_2 \right) - \left(\frac{q_2 - q_1}{p_2 - p_1} \times l + q_2 \right) \right\} / 2 \qquad (4.1)$$

4.2 Non-linear Normalization

This was performed to take into account the differences in base voltage intensity among individuals.

Non-linear normalized values were determined using Formula 4.2, where $f(x)$ is the EEG feature after non-linear normalization has been applied, x is the EEG feature applied in non-linear normalization, x_{min} is the minimum EEG feature of the individual, and x_{max} is the maximum of the same data. As a result, EEG features with large values are compressed and EEG features with small values are expanded. The degree of intensity of voltage of an individual's EEGs can thus be accounted for.

$$f(x) = \frac{\log(x - x_{\min})}{\log(x_{\max} - x_{\min})} \qquad (4.2)$$

5 Clustering of EEG Feature Knowledge Base

As described in Section 3, a total of 315 EEG features from each of the 16 subjects were obtained, for a total of 5040 EEG features. If these numerous data points were used as a knowledge base, there would be concern about degradation of emotion judgment performance because of the large number of objects for comparison. Thus, if the EEG feature knowledge base that will serve as an object of comparison can be clustered and grouped based on specific emotions, the number of data points can be limited and performance degradation reduced.

The k-means method, which uses a threshold value, was employed as the clustering technique. The algorithm applied is shown below. In clustering, total error between EEG features was defined as distance, using EEG features determined by compiling data from 14 locations and three frequencies, for a total of 42 elements, into one element. EEG features of compared EEGs A and B were taken as *al* and *bl*, respectively. When EEG features of the i^{th} frequency and j^{th} electrode of each EEG feature were taken as al_{ij} and bl_{ij}, respectively, $Distance(al_{ij}, bl_{ij})$ was expressed using Formula 5.1. The distance of EEGs *A* and *B* overall, *All_Distance(A, B)*, was thus expressed using Formula 5.1.

$$Distance\ (al_{ij}, bl_{ij}) = \left| al_{ij} - bl_{ij} \right| \quad (5.1)$$

$$All\_Distance\ (al_{ij}, bl_{ij}) = \left| al_{ij} - bl_{ij} \right| \quad (5.2)$$

The clustering technique was as follows:

(1) The number of clusters was set, and clusters are randomly assigned to each of EEG features $x_i (i = 1, \ldots, n)$.

(2) The centroid $V_j (j = 1, \ldots, K)$ of each cluster is determined based on the assigned EEG feature. The centroid is the mean of the elements of the EEG feature in the cluster.

(3) The distance between each x_i and each V_j is determined, and when the distance of the nearest centroid is equal to or less than the threshold value, x_i is assigned to that cluster.

(4) The process is complete if the centroids of all x_i clusters have not changed in the above process. Otherwise, Steps 2 and 3 are repeated.

5.1 Determination of the Number Clusters

When the number of clusters was set to a power of 2 between 1 and 2048, accuracy was evaluated by clustering based on the various threshold values. As a result, the cluster number in the range of 32 to 128 at which the correct rate was maximized was used.

5.2 Determination of Threshold Value

In the k-means method, clusters to be partitioned into are set first, but sometimes the number of clusters is less than the set number depending on the integration method. In such cases, the EEG knowledge base after clustering is smaller than the number of clusters initially set. As a result of examination, 100 was used as the threshold value in this study.

6 Evaluation Experiment

6.1 Experimental Method

The evaluation method used was a leave-one-out cross-validation, a technique in which one data point from all test data was extracted and compared with all the remaining data.

This study used a EEG feature knowledge base constructed by clustering all but the principal one of the 3436 EEG features obtained by excluding outliers from the total of 5040 EEG features described in Section 5.

The emotions of the 3436 EEG features used in this study comprised 1159 anger features, 575 sadness features, 1164 no-emotion features, and 538 pleasure features. In cases where the number of data points differed greatly among emotions, a noticeable difference ended up occurring in the number of data points in clusters when clustering was performed under the same conditions. The number of clusters for pleasure, sadness, anger and no emotion was thus set to 64, 64, 32 and 32, respectively, to result in the same ratio as the ratio of the four emotions of 1:1:2:2.

6.2 Evaluation of Accuracy

Accuracy of emotion judgment from EEGs using the clustered EEG feature knowledge base was 33.0%. As a comparison, an emotion judgment system offered by a company that develops EEG measurement equipment [2] showed 27.9% accuracy. The method proposed herein thus appears valid.

However, performance accuracy remains low, and continued development is required through further development of methods for both reducing noise mixed in with EEGs and constructing knowledge bases.

7 Conclusion

We have presented a technique for gauging the emotions felt by a person from EEGs. Accuracy of emotion judgment was 33.0%, which is still low, but we believe we obtained sufficient clues that will lead to improved accuracy, such as the validity of knowledge base clustering. We plan to continue research aimed at improving the accuracy of emotion judgment by EEGs in the hopes of developing robotic systems that can participate in conversation and activities while gauging human emotional states.

Acknowledgements

This research has been partially supported by the Ministry of Education, Science, Sports and Culture, Grant-in-Aid for Scientific Research (Young Scientists (B), 24700215).

References

[1] J. Clin, Guideline therteen : guidelines for standard electrode position nomenclature, American Electroencephalographic Society, Neurophysiol 11, pp.111-113, 1994.

[2] T. Musha, Y. Terasaki, H. A. Haque, G. A. Iranitsky, Feature extraction from EEG associated with emotions, Art Life Robotics, pp.15-19, 1997.

Human-Robot Teaming with Sliding Autonomy

Jacob Longazo, Michael Heid and Fang Tang
Computer Science Department
California State Polytechnic University, Pomona
Pomona, CA, 91768
Email: {jwlongazo|maheid|ftang}@csupomona.edu

Abstract— **This paper describes a sliding autonomy approach for coordinating a team of robots to assist human operators to accomplish tasks while adapting to new or unexpected situations by requesting help from the human operators.** [1]

I. The Problem

Multi-robot systems are widely used in today's robotic applications because of their recognized advantages over single robot systems [2]. However, there are also challenges arising with this type of complex system, such as mechanisms for coordinating team members and the operator control of the team. While researchers strive to build a fully autonomous system to perform various tasks, the robot team inevitably faces many unforeseen circumstances in an open and dynamic world. The team can either adapt to the dynamics through some life-long learning process, or by seeking help from the more competent human operators. Our goal is to develop a human-robot system with robots assisting humans to accomplish tasks while adapting to new or unexpected situations with the help from human operators.

As a motivating example, consider a site clearing application that requires a specific area to be cleared of obstacles, which we simplify to be boxes with different weights or sizes. The objective of the application is to clear the site in as little time as possible. For the purposes of this discussion, we assume that the box locations are known and robots can localize in the environment. This task can be decomposed into series of subtasks with each removing one obstacle. We further assume that there is a general planner that determines the ordering constraints of subtasks. Note that obstacles come with different weights. Whether or not a subtask requires single or multiple robots depends on the capabilities of the robots (e.g., push forces). Our task allocation approach is able to find combinations of robot capabilities that can accomplish the task in both cases.

In the above application scenario, there are many situations where humans and robots can work together to ensure the successful completion of the task. A human operator can oversee the execution of the task through a graphical control interface. The operator can intervene at any point, for example, fixing a sensor failure on a robot, or even work as peers to push the boxes together. Meanwhile, the robots can request help when it encounters a problem that cannot be resolved by itself. The question is how and when to initiate different levels of interactions between humans and robots.

[1] This paper is submitted as a poster.

II. The Approach

Our approach to the above problem is to develop a multi-robot graphical control interface that enables sliding autonomy of controls [3], varying from fully autonomous operations, to human-intervened operations, and to pure teleoperation. This control interface allows human operators to monitor the task execution status and react to unforeseen issues. It also helps establish the interaction between humans and robots by allowing them to influence each other's action selection. In order to coordinate the team of robots to autonomously accomplish the high-level task, we use an auction-based task allocation mechanism to allocate subtasks to appropriate robots.

A. Sliding Autonomy

The motivation of using sliding autonomy is that we recognize that robots work in a dynamic world with various levels of uncertainties, which cannot be easily handled with an autonomous solution. On the other hand, even though pure teleoperation provides precise control and human knowledge, challenges remain in areas such as providing situation awareness, handling operator fatigue, and dealing with communication delay. The core concept behind sliding autonomy is to combine the control methods of autonomy and teleoperation in a way that best highlights each control method's advantages.

Under sliding autonomy, a robot can dynamically switch between autonomy, teleoperation, and various mixes of the two depending on the situation. In our approach, a switch can occur in one of two ways. First, if a robot meets a certain error condition that it cannot recover from while under the control of autonomy, it can request help from a human operator to guide it to a condition the system can recover from. Second, if the human operator observes a robot behaving incorrectly, the human can initiate a switch in control to correct the robots behavior. The ability of both the robot and the operator to issue a control switch helps the system respond to nearly any type of situation. In addition, we expect that the robots can behave autonomously most of the time, thus very few human operators and little operation time are required. This type of system should solve the

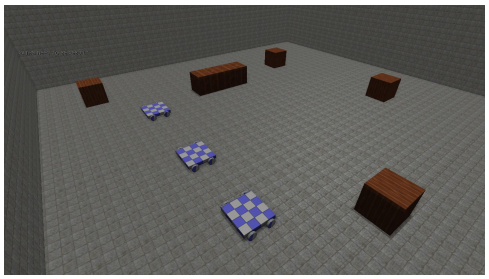


Fig. 1. Our task scenario with three robots and five boxes.

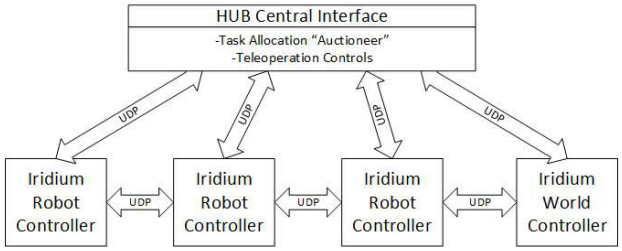


Fig. 2. The overall architecture.

problems of full autonomy and teleoperation without the associated drawbacks of the two.

B. Task Allocation and Monitoring

In order to handle multiple tasks of pushing obstacles, we implement a task allocation approach based on the Contract Net Protocol [4]. The process is described as follows:

1) Task announcement: Initially, the human operator introduces the site clearing task T to an "Auctioneer" agent built in the operator control interface. Assuming there's a planner to decompose the task into subtasks and save them in a task queue. The auctioneer then announces the subtask one at a time to the robots. Each subtask t_i holds task specific information, such as the weight and the position of the obstacle to be removed.
2) Bid submission: Each idle robot submits a bid to the auctioneer including its current distance from the obstacle and its pushing forces.
3) Winner determination: Once bids are collected, the auctioneer then uses a greedy approach to determine the winning robot(s) for the current task. The winning robot(s) should have a summed force (f_{sum}) greater than the weight of the box. There are further two preferences. First, we favor robots that are closer to the obstacle and thus ensures fast completion time. Second, we try to minimize f_{sum} so that we allocate as little resources as possible to execute the current tasks. Unsuccessful allocation will result in the subtask being reinserted back to the task queue.

III. Simulation

To demonstrate our approach, we implemented the site clearing task using USARSim [1], UDK version 1.2, developed to run on Unreals UDK (Unreal Development Kit) platform. USARSim is a high-fidelity, extensible simulation developed to facilitate human-robot interaction in the search and rescue domain. Our task involves a team of three robots working cooperatively to push boxes against two opposite walls in the test room (see Figure 1). The robots used are skid robots with differential drives, equipped with an INS sensor for location and orientation detection, as well as a range scanner for collision avoidance.

To control the robots, an open source controller application developed to work with the UDK version of USARSim known as Iridium was chosen. Iridium was selected because it provided the required features and came with the install package of the UDK version of USARSim. Iridium has been heavily modified to act as each robot's simulated control system with new features for navigation, obstacle avoidance, improved teleoperation, and additional UDP communications that allow an Iridium agent to communicate with other agents. Each robot uses a separate instance of Iridium and is assigned a specific ID number. In this way, the Iridium controller simulates a robot's on-board system. The Iridium controllers are also given the ability to create bids for task allocation as well as detecting erroneous situations in which to request human aid to support sliding autonomy. An additional Iridium client is used as a world controller, meant to act as the interface for outside software to interact with and manipulate the simulation environment. To act as the overarching software, a central controller interface, nicknamed the HUB, contains the task allocator, teleoperation controls, as well as a graphical user interface that allows it to control the simulation and all its various components. The overall architecture is displayed in Figure 2.

We have implemented and tested our approach with a pure autonomous mode, a teleoperation mode and a sliding autonomy mode. Currently, a robot will initiate a request for help when it enters an erroneous state. For example, when a robot has taken too long to complete a task, it will trigger a timeout and a request for help. We are in the process of performing more thorough evaluations of our proposed approach. As a future work, we will extend our current sliding autonomy approach with more sophisticated mechanisms for the robot to decide when and how to initiate help. We would like to include humans in the task allocation process such that humans can be part of the team to perform peer-to-peer collaboration. We would also like to enhance the current graphical control interface to provide more situation awareness.

References

[1] M. Lewis, J. Wang, and S. Hughes. Usarsim : Simulation for the study of human-robot interaction. *Journal of Cognitive Engineering and Decision Making*, 2007:98–120, 2007.
[2] L. E. Parker. Distributed intelligence: Overview of the field and its application in multi-robot systems. *Journal of Physical Agents, Special Issue on Multi-Robot Systems*, 2(2):5–14, 2008.
[3] B. Sellner, F. Heger, et al. Coordinated multi-agent teams and sliding autonomy for large-scale assembly. *Proceedings of the IEEE*, 94(7), 2006.
[4] R. Smith. The contract net protocol: High-level communication and control in a distributed problem solver. *IEEE Transactions on Computers*, C-29, 1980.

SESSION

LATE BREAKING PAPERS AND POSITION PAPERS: ARTIFICIAL INTELLIGENCE AND APPLICATIONS

Chair(s)

Prof. Hamid R. Arabnia

Automatic Selection and Analysis of Synonyms in Japanese Sentences Using Machine Learning

Masaki Murata, Yoshiki Goda, Masato Tokuhisa
Department of Information and Electronics
Tottori University
4-101 Koyama-Minami, Tottori 680-8552, Japan

Abstract— **In this study, we conducted automatic selection of an appropriate synonym among multiple candidate synonyms for a certain sentence using machine learning. We experimentally confirmed that the accuracy rate of the proposed method using machine learning (0.86) was higher than that of the baseline method (0.70), which always outputs the synonym that appears most frequently in the training data set. Based on the performance of machine learning, we classified synonym pairs into pairs that tend to require proper synonym selection for a certain sentence and pairs that do not tend to require proper synonym selection, i.e., either synonym from the pair can be used for a certain sentence. We also examined the features used in machine learning for some synonym pairs and clarified the information that is useful for selection and usage of synonyms.**
Keywords: Synonym, Sentence, Automatic Selection, Machine Learning, Feature

I. Introduction

Synonyms are different words with similar meaning. For example, "research" and "study" are a synonym pair. There have been some studies of synonyms. Many researchers extracted synonyms from corpora and dictionaries [1], [2], [3].

Murata et al. conducted automatic selection of an appropriate notational variant among multiple candidate notational variants for a certain sentence using machine learning [4]. Notational variants are different expressions of a certain word. Examples of notational variants are "3," "III" and "three" in English. Examples of notational variants in Japanese are 桜 (expressed with Chinese characters; cherry in Japanese) and サクラ (expressed with Katakana characters; cherry in Japanese). Although Murata et al. used machine learning, they did not handle synonyms.

In this study, we conduct automatic selection of an appropriate synonym among candidate multiple synonyms for a certain sentence using machine learning. For example, we automatically select of a word "study" among candidate synonyms "research" and "study" for a certain sentence. We refer to automatic selection of an appropriate synonym among candidate multiple synonyms for a certain sentence as automatic selection of synonyms. Our study is useful for the selection of appropriate synonyms in sentence generation [5], [6]. We handle automatic selection of Japanese synonyms.

Although there are many studies on extracting synonyms automatically [1], [2], [3] including using machine learning and there are some studies on analyzing synonyms using similarity (without machine learning) [7], there have been very possibly no studies on mainly handling automatic selection of an appropriate synonym among synonyms with very similar meanings for a certain sentence using supervised machine learning. In this study, our study has an originality that we handled automatic selection of synonyms using machine learning.

For word selection including lexical substitution, there are many studies [8], [9], [10], [11]. They studied word sense disambiguation for word selection in machine translation. They handled selection of an appropriate word among words that have different meaning. In contrast, we handle selection of an appropriate word among words that have very similar meaning. Furthermore, we handle classification of synonym pairs into pairs that tend to require proper synonym selection for a certain sentence and pairs that do not tend to require proper synonym selection, which was not handled by their studies.

We used synonyms in EDR dictionary in this study. Synonyms in the EDR dictionary have very similar meanings. In this study, we handle automatic selection of an appropriate synonym among synonyms with very similar meanings.

Although synonyms have a similar meaning and they seem to have no need for selection and can be used for all cases, some circumstances require the selection of a more appropriate synonym. For example, the two Japanese words, *iryou* (clothing) and *irui* (clothes), have the same meaning as "a thing we wear" has in the EDR dictionary. However, *iryou* (clothing) can appear just before *hin* (item) and *irui* (clothes) cannot appear just before *hin* (item), i.e., we can say *iryou hin* (clothing item) but we cannot say *irui hin* (clothes item). When we use *hin* (item), the pair of *iryou* and *irui* must be appropriately selected for a certain sentence and requires proper synonym selection.

The primary goals of this study are as follows. The first goal is to obtain high performance for automatic selection of synonyms using machine learning. The second goal is to classify synonym pairs into pairs that tend to require proper synonym selection and pairs that do not tend to require proper synonym selection. When the selection of synonyms using machine learning is easy, the pair of synonyms is judged to be a pair that requires proper synonym selection. When the selection of synonyms using machine learning is difficult, the pair of the synonyms is considered not to require proper synonym selection; both synonyms in the pair are likely to be applicable to a sentence. The results of our study will contribute to the selection and usage of synonyms.

II. Task and proposed method

A. Task

Two synonyms A and B that have a similar meaning are given. We refer to words A and B as target words in this paper. We collect sentences containing either target word. We then eliminate target words in the collected sentences. Then, the task is to estimate which word existed in the eliminated part of a sentence. When a system selects a word that was

in the original sentence, the system's selection is considered correct.

B. Proposed method

In this study, we estimate which word among two target words is in the original sentence using machine learning. We use sentences containing either of the two target words as training data. We conduct machine learning by handling a target word included in the original sentence as the category (class) of the sentence. The maximum entropy method [12] is used for supervised machine learning.

The maximum entropy method [12], [13] is used for machine learning because it can produce as good performance as support vector machine, and it can output the importance of features.

One machine learning cycle is used for a pair of target words (synonyms). For n pairs of target words, we use n machine learning cycles.

In our experiments, we use the most frequent two target words. Features are extracted from sentences containing either of the two target words. From the two target words, our machine learning approach selects the target word that is more appropriate in a sentence.

The features used in our method are words in sentences, the three words just before and just after the target word, the words in the bunsetsu (phrase) containing a target word, the content words in the bunsetsu modifying the bunsetsu containing a target word, the content words in the bunsetsu modified by the bunsetsu containing a target word, the POSs (part of speech) of those words, and the first 1, 2, 3, 4, 5, and 7 digits of the category number of those words. We consult Murata et al.'s paper [4] for the features.

The category number in the features is a ten-digit number that is described in Bunrui Goi Hyou, a Japanese thesaurus. Words with similar meanings have similar ten-digit numbers. In this study, we use the first five and three digits of the number as features. Therefore, we use the upper concept of each word as features.

The words adjacent to a target word can be used for synonym selection. The syntactic information of a sentence can also be used in synonym selection.

To classify synonyms into a certain group of synonyms where automatic selection of words is conducted properly by machine learning and other groups, we used "high," "medium," and "low" categories on the basis of recall rates of synonym selection. A recall rate is a concept similar to accuracy. A recall rate is the ratio of the number of correct outputs over the number of correct data items.

When the lowest recall rate of the two notational variants of a target word is greater than or equal to 0.8, the word is classified in the "high" category because when both recall rates are high, the estimation is accurate. When the lowest recall rate is greater than or equal to 0.5 and less than 0.8, the word is classified in the "medium" category. When the lowest recall rate is lower than 0.5, the word is classified in the "low" category because when one of the two recall rates is low, the estimation is not accurate.

III. Data sets used in experiments

In this section, we describe the data sets used in our experiments.

In our experiments on synonym selection, we used synonyms appearing in newspapers. We used Mainichi newspaper articles (all the articles in one year edition, 1991). We extracted pairs of words that satisfy all of the following conditions and used them as pairs of synonyms. Condition 1: The two words in the pair have the same concept identification number in the EDR dictionary. Condition 2: Both words have only one identification number in the EDR dictionary, i.e., the two words have only one meaning. Condition 3: Both words appear 50 times or more in the Mainichi newspaper articles (one year edition, 1991). Condition 4: The representative words of the two words that are given using the morphological analyzer JUMAN are different, i.e.,, the two words differ and are not notational variants.

Condition 1 is used to extract words with very similar meaning. Condition 2 is used to identify and exclude words with multiple meanings because they are complex and not handled by the proposed method. Condition 3 is used to investigate frequently used words in newspaper articles. Condition 4 is used not to handle notational variants because word selection in notational variants has already been addressed in a previous study [4].

In our data set, 90 word pairs satisfy all four conditions. From these, we randomly extracted 45 pairs to use in our experiments. For each word pair, 100 to 10000 data items exist.

IV. Experiments on synonym selection using machine learning

A. Experimental methods

We applied our machine learning approach to the 90 word pairs obtained through the process described in Section III.

In our experiments, we used the most frequent two synonyms. Our machine learning approach selects the word from the two synonyms that is more appropriate for a given sentence.

For each pair of words, sentences containing each word were extracted from the Mainichi newspaper articles (one year edition, 1991) and were used as a data set. Sometimes, the same sentences occurred more than once in Mainichi newspaper articles; thus, we eliminated redundant sentences so that each sentence occurred only once. We conducted a ten-fold cross validation for each data set (each word pair). First, we divided the given data set into ten parts. One part was treated as the test data set, and the remaining nine parts were treated as training data sets. The category (class) of each item in the test data set was estimated by learning the training data sets using the maximum entropy method. Then, the estimated category was evaluated using the correct category in the test data set. This process was repeated for all ten parts. Consequently, all ten parts were evaluated.

We used a baseline method to compare the proposed method. The baseline method always outputs the synonym that appears most frequently in the training data set.

B. Experimental results

The results of classifying 45 words into the three categories by machine learning recall rates are shown in Table I. The three categories are "high," "medium," and "low," as discussed in Section II-B. The average accuracy rates for the proposed method and the baseline method are shown in Table II. Values for each category based on recall rates are also described. The results of a comparison of the proposed and baseline methods relative to the accuracies of each synonym pair are shown in Table III. "Even" shows the number of synonym pairs for which the difference is 0.01 or less. "Proposed method

TABLE I: Ratios of words classified into the three categories on the basis of recall rates

Category	Ratio
High	0.11 (5/45)
Medium	0.51 (23/45)
Low	0.38 (17/45)

TABLE II: Average accuracy rates of the proposed method and baseline method

	High	Medium	Low	All
Proposed method	0.95	0.84	0.80	0.86
Baseline method	0.62	0.69	0.78	0.70

(win)" indicates the number of synonym pairs for which the accuracy rate of the proposed method is higher than that of the baseline method. "Baseline method (win)" indicates the number of synonym pairs for which the accuracy rate of the proposed method is lower than that of the baseline method.

V. Discussion

A. Comparison of the proposed and baseline methods

In Table II, the accuracy rates of the proposed method and the baseline method are 0.86 and 0.70, respectively. The accuracy rate of the proposed method was higher than the baseline method. From Table III, we found that the accuracy rate of the proposed method was higher than that of the baseline method for all "high" and "medium" synonym pairs and was higher for 11 "low" synonym pairs. The synonym pairs where the proposed method won against the baseline method (39 = 5 + 23 + 11) corresponded to 80 percent or more of all 45 synonym pairs. The results indicate that the proposed method and features used in machine learning are useful for synonym selection.

However, in the "low" category, there were four pairs of "even" and two pairs for which the accuracy rate of the baseline method was higher than that of the proposed method. This was likely caused by significant differences in the frequencies of the two words in a synonym pair. This is a case where the frequency of a word in the two words is extremely higher than the frequency of the other word. The baseline method always outputs the synonym that appears most frequently in the training data set among the synonyms. Therefore, when the frequencies of the two words differ significantly, the accuracy rates of the baseline method are likely to be higher. Thus, there would be a small number of cases for which the baseline method was better than the proposed method.

B. Trends based on the degree of recall rates

We examined classified word pairs manually.

Most word pairs that were classified as "high" tend to require proper synonym selection. An Example of "high" is the pairing of *chokin* ("deposit" or "saving") and *chochiku* ("storage" or "saving"). Example sentences are below.

Example sentence 1a (Machine learning correctly selected a synonym):
yuubinkyoku no chokin gaku jougen wa
(post office) (saving deposit) (amount) (maximum)
hikiagerareru
(be raised)
(The maximum amount for a saving deposit at a post office will be raised.)

TABLE III: Comparison of accuracy of the proposed and baseline methods for each synonym pair

	High	Medium	Low
Proposed method (win)	5	23	11
Baseline method (win)	0	0	2
Even	0	0	4

TABLE IV: Results of machine learning for *chokin* and *chochiku* ("high" category)

	Recall rates	Precision rates	Number
chokin	0.89	0.89	403
chochiku	0.86	0.86	332

Example sentence 1b (Machine learning correctly selected a synonym):
toushi ga yowaku chochiku ga chiisai
(investment) (weak) (storage) (small)
(Investment is weak and storage is small.)

A pair of *chokin* ("deposit," "saving," or "saved money") and *chochiku* ("storage," "saving," or "saved thing") was classified as "high." These words have the same concept identification "0fbec6" in the EDR dictionary. In the "high" category, machine learning obtained good recall rates (good performance) and word selection was easy and necessary.

The results of machine learning for the pair are shown in Table IV and the three important features with the highest α values are shown in Table V. A normalized α value represents the degree of importance of the corresponding feature used to estimate a synonym learned using the maximum entropy method. Further details can be found in the literature [13]. From "F1: *teigaku* (fixed-amount)" and "*yubin* (post)" in Table V, we found that *chokin* is used for a sentence related to postal savings. This is also true of Example sentence 1a. In Example sentence 1a, *youbinkyoku* (post office) was used. From "F1: *toushi* (investment)" and "F1: *beikoku no* (USA)," we found that *chochiku* was used for a sentence related to investment or countries. We found that *chokin* and *chochiku* have their own characteristics, and they tend to require proper synonym selection. This result matches the results of machine learning with high recall rates that determine that word selection is necessary.

Word pairs that were classified as "medium" include two cases. The first case occurs when the meaning of one word of the pair includes the meaning of the other word and is less specific than the meaning of the other word. The second case occurs when only one word of the pair can be used when a certain word appears in the sentence.

An example of the first case is *En* (banquet) and *hirouen* (reception banquet). Although *en* and *hirouen* are synonyms in the EDR dictionary, *hirouen* is a certain special banquet (reception banquet), and the meaning of *hirouen* is included and is more specific than the meaning of *en* (banquet).

Examples of the second case are shown below.

Example sentence 2a (Machine learning correctly selected a synonym):
iryou hin no uriage ga okii
(clothing) (item) (sale) (large)
(The sales of clothing items are large.)

Example sentence 2b (Machine learning correctly selected a synonym):
denka seihin to irui to mouhu ga aru
(electric appliances) (clothes) (blankets) (exist)

TABLE V: Important features for machine learning when selecting synonyms *chokin* and *chochiku* ("high" category)

chokin	
Features	Normalized α values
F1: *teigaku* (fixed-amount)	0.74
F1: *yubin* (post)	0.73
F1: *yuuseishou* (MPT)	0.73

chochiku	
Features	Normalized α values
F1: *teki* (like)	0.75
F1: *toushi* (investment)	0.70
F1: *beikoku no* (USA)	0.64

TABLE VI: Results of machine learning for *iryou* and *irui* ("medium" category)

	Recall rates	Precision rates	Number
iryou	0.75	0.75	160
irui	0.63	0.63	106

(There are electric appliances, clothes and blankets.)

A pair of *iryou* ("clothing material" or "clothing") and *irui* ("clothes") was classified as "medium." These words have the same concept identification "0e504a" in the EDR dictionary. The results of machine learning for the pair are shown in Table VI and the three important features with the highest α values are shown in Table VII. From "F1: *hin* (item)" in Table VII and Example sentence 2a, we found that *iryou* (clothing) is used when it appears just before *hin* (item). However, there were no characteristics other than the above characteristic. In many cases, both *iryou* and *irui* can be used. Therefore, we found that *iryou* and *irui* do not have their own characteristics, and the tendency to require proper synonym selection was not significantly high.

Most word pairs that were classified as "low" do not tend to require proper synonym selection. Word pairs that were classified as "low" include pairings of a word and its abbreviated word and pairings of a Japanese word and its imported word (a word derived from a foreign word). An example of a pairing of a word and its abbreviated word is *sho enerugii* (a word) and *sho ene* (it is an abbreviated word) (they have the same meaning, i.e., "energy saving"). An example of a pairing of a Japanese word and its imported word is *utai monku* (a Japanese word) and *kyacchi hureizu* (its imported word); they have the same meaning, i.e., "catch phrase." In these cases, the two words in a synonym pair have very similar meaning and do not tend to require proper synonym selection, and can be used for almost all the sentences.

VI. Conclusion

In this study, we conducted automatic selection of an appropriate synonym among multiple synonyms for a certain sentence using machine learning. In our experiments, we confirmed that the accuracy rate of the proposed method using machine learning (0.86) was higher than that of the baseline method (0.70), which always outputs the synonym that appears most frequently in the training data set. Based on the performance of machine learning, we classified synonym pairs into pairs that tend to require selection for proper usage and pairs that do not tend to require selection. A pairing of *chokin* ("deposit" or "saving") and *chochiku* ("storage" or "saving") was classified as tending to require selection to ensure proper usage. We examined features for some synonym pairs and clarified the information that is useful for synonym selection.

TABLE VII: Important features for machine learning when selecting synonyms *iryou* and *irui* ("medium" category)

iryou	
Features	Normalized α values
F1: *hin* (item)	0.78
F3: aa (abstract entity)	0.66
F2: *hin* (item)	0.62

irui	
Features	Normalized α values
F2: Target word is at the begining of a sentence	0.70
F1: *denki* (electricity)	0.62
F1: *buppin* (thing)	0.61

Acknowledgment

This work was supported by JSPS KAKENHI Grant Number 23500178.

References

[1] D. Lin, "Automatic retrieval and clustering of similar words," in *Proceedings of COLING-ACL*, 1998, pp. 768–774.

[2] R. Barzilay and K. R. McKeown, "Extracting paraphrases from a parallel corpusy," in *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics (ACL 2001)*, 2001, pp. 50–57.

[3] T. Wang and G. Hirst, "Exploring patterns in dictionary definitions for synonym extraction," *Natural Language Engineering*, vol. 18, no. 03, pp. 313–342, 2012.

[4] M. Murata, M. Kojima, T. Minamiguchi, and Y. Watanabe, "Automatic selection and analysis of Japanese notational variants on the basis of machine learning," *International Journal of Innovative Computing, Information and Control*, vol. 9, no. 10, pp. 4231–4246, 2013.

[5] V. H. Kevin Knight, "Two-level, many-paths generation," in *Proceedings of the ACL*, 1995, pp. 252–260.

[6] K. Uchimoto, S. Sekine, and H. Isahara, "Text generation from keyword," in *COLING '2002*, 2002, pp. 1037–1043.

[7] P. Edmonds and G. Hirst, "Near-synonymy and lexical choice," *Computational Linguistics*, vol. 28, no. 2, pp. 105–144, 2002.

[8] K. Y. Kim, S. Y. Park, and S. J. Lee, "Automatic sense disambiguation for target word selection," in *Proceedings of the 10th Pacific Asia Conference on Language, Information and Computation*, 1995, pp. 109–114.

[9] K. Uchimoto, S. Sekine, M. Murata, and H. Isahara, "Word translation based on machine learning models using translation memory and corpora," in *Senseval 2*, 2001, pp. 155–158.

[10] W. Aziz and L. Specia, "Uspwlv and wlvusp: Combining dictionaries and contextual information for cross-lingual lexical substitution," in *Proceedings of the 5th International Workshop on Semantic Evaluation, ACL 2010*, 2010, pp. 117–122.

[11] J. Matsuno and T. Ishida, "Constraint optimization approach to context basedword selection," in *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence*, 2011, pp. 1846–1852.

[12] A. L. Berger, S. A. D. Pietra, and V. J. D. Pietra, "A maximum entropy approach to natural language processing," *Computational Linguistics*, vol. 22, no. 1, pp. 39–71, 1996.

[13] M. Murata, K. Uchimoto, M. Utiyama, Q. Ma, R. Nishimura, Y. Watanabe, K. Doi, and K. Torisawa, "Using the maximum entropy method for natural language processing: Category estimation, feature extraction, and error correction," *Cognitive Computation*, vol. 2, no. 4, pp. 272–279, 2010.

Constraint Based Staff Scheduling Optimization
Using Single Player Monte Carlo Tree Search

CheeChian Cheng, Norman Carver and Shahram Rahimi
Department of Computer Science, Southern Illinois University, Carbondale, IL 62901, USA
cccheng@siu.edu; carver@cs.siu.edu; rahimi@cs.siu.edu

Keywords: Monte Carlo Tree Search, Artificial Intelligence, Optimization, Roster, Staff Scheduling.

Abstract: Single Player Monte Carlo Tree Search approach is used to solve and optimize the constraint-based staff scheduling problem for emergency department in health care industry and the results are compared with the optimum results obtained by brute force search.

1 INTRODUCTION

Optimization is a process of finding the highest achievable performance under a given set of constraints. There are various applications in optimization and one of the applications is to optimize the staff scheduling problem. The schedulers[1] for Critical Care Medicine unit (ER) are constantly facing the challenge of producing rosters that are filled up with providers[2] and at the same time, trying to fulfill every provider's preferences. However, the schedulers' main priority is to ensure that all shifts are filled up with providers. Thus, putting providers' preferences in lower priority, most of the time the rosters hardly meet the providers' preferences. In order to meet the providers' preferences, the schedulers have to spend an average of 25 man-hours to create and optimize the roster manually as staff scheduling is an NP-hard problem (Sotskov and Shakhlevich, 1995).

In this paper, *Monte Carlo Tree Search* (MCTS) is evaluated to solve and optimize the staff scheduling problem in the Health care industry. Several experiments are described and the results of the proposed MCTS approach are compared with the results of *brute force search* (BFS).

1.1 State-Of-The-Art

Throughout the years, various approaches have been proposed to optimize staff scheduling problem including *genetic algorithm* (GA), constraint programming techniques, greedy algorithms and other hybrid models. In recent work, Burke et al. has proposed a hybrid model of *integer programing* (IP) and *variable neighborhood search* (VNS) to optimize nurse rostering with hard and soft constraints (Burke et al., 2010). The article compared the hybrid VNS model with hybrid genetic algorithm model and they claimed that the hybrid VNS model outperformed the hybrid GA model.

Another hybrid model of IP and *Evolutionary Algorithm* (EA) was proposed and the authors claimed that the hybrid model of IP and EA outperformed the earlier model of IP and VNS (Huang et al., 2014).

Genetic algorithm is a popular approach to the scheduling problem and one of the recent work was done by Yang and Wu (Yang and Wu, 2012). They have reported that with the appropriate cross over and mutation operators, the GA method could find optimal or near-optimal work schedules.

Brucker et al. has proposed 2-stage local greedy search to assign as many shifts to the most constrained nurses as possible (Brucker et al., 2010). This may produce a feasible roster but may not produce an (near) optimum roster since the roster might be a local minimum instead of global minimum.

The proposed MCTS approach will shorten the search time significantly while producing an optimum (or near optimum) schedule by taking in all the constraints and preferences. The proposed approach is relatively easy to implement and it does not need any domain knowledge in staff scheduling. However, it does need to know the constraints or rules of the problem so that MCTS will be able to make valid "moves". Moreover, MCTS can be interrupted any time and produces the best solution found so far.

[1]The scheduler is a person who is responsible for creating monthly roster for the providers.

[2]The provider is a person who works at the hospital/facility.

2 BACKGROUND WORK

MCTS is an incremental tree building algorithm using Monte Carlo methods (Browne et al., 2012). A typical 2-player (player A and player B) tree is shown in Fig. 1. The selection of the best (or most promising) node is based on the *upper confidence bound for tree* (UCT) (Kocsis and Szepesvári, 2006) derived from multiarmed bandit algorithm's upper confidence bounds (UCB) (Auer et al., 2002). The UCT as shown in equation 1 acts as a balancer to either explore new node or exploit current best node.

$$UCT = \overline{X} + C\sqrt{\frac{\ln N}{N_i}} \quad (1)$$

In MCTS, the *Selection Phase* is a process of choosing the best (or most promising) node using UCT equantion. It follows by the *Expansion Phase* where the creation of new node is determined by the tree policy and often, the tree policy is a uniform distribution of all possible "moves" (Monte Carlo methods). A new node is randomly chosen from the possible "moves" and later, added to the tree as the descendant of the most promising node mentioned earlier. Play out simulation (*Simulation Phase*) will begin after the *Expansion Phase* by sampling the outcome of the simulation. Often, the outcome of the simulation of a game uses the minimax algorithm with alternate moves and the outcome of the simulation is propagated back to the root node, also known as *Back Propagation Phase*. A 2-player minimax tree (as shown in Fig. 2) shows that player A wins the game if s/he adopts strategy X and loses the game if s/he adopts strategy Y.

Go has a huge search space and by utilizing MCTS, the computer player achieved impressive winning rate against professional players. However, these are 2-player games and there was a need to come up with a MCTS for single player games like puzzle games. *Single Player MCTS* (SP-MCTS) has a UCT as shown in equation 2 (Schadd et al., 2008). It has an additional term to act as a balancer to either explore or exploit current tree.

$$UCT = \overline{X} + C\sqrt{\frac{\ln N}{N_i}} + \sqrt{\frac{\sum X^2 + N_i\overline{X}^2 + D}{N_i}} \quad (2)$$

SP-MCTS was applied to the "re-entrance" scheduling problem and produced impressive results in optimizing the schedule (Matsumoto et al., 2010). However, the "re-entrance" scheduling problem does not have hard and soft constraints like the staff scheduling problem.

3 THE PROBLEM

In Critical Care Medicine unit, creating monthly rosters for providers (doctors and nurses) to work in various shifts are frequent tasks for a scheduler. Most of the time, schedulers have difficulty assigning providers to empty shifts while working out a roster that fulfills every providers' preferences. Each provider may send their working preferences and off-days to the scheduler on monthly basis before the scheduler starts creating the roster. In normal practice, the providers may request five (5) to seven (7) off days per month where they will not be scheduled to work on the off-days. Then, the scheduler would produce a feasible solution before trying to meet all the providers preferences. This approach may produce globally optimized solution (measurement of optimality is described in sub-section 4.6) but often produces locally optimized solution. Hence, there are two types of constraints, hard and soft constraints[3].

[3]Acknowledgment: This is to acknowledge that this work was supported by R&B SOFT LLC, providing information on rules and practices in the emergency departments and the required data for evaluation and testing purposes.

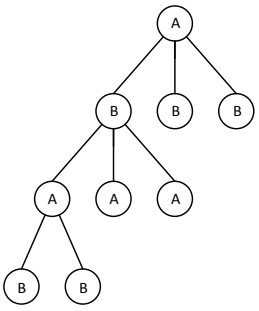


Figure 1: A 2-Player Tree.

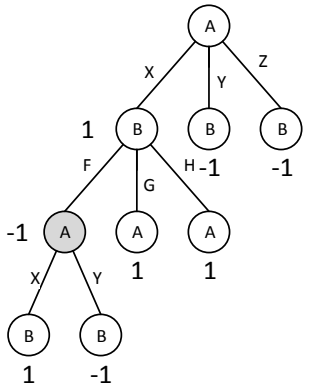


Figure 2: A 2-Player MiniMax Tree.

3.1 Hard Constraints

Hard constraints are the constraints that if the roster violates them, it is considered not a feasible solution/roster. The hard constraints listed below were used in the experiments.

- Hard Constraint #1 - Each provider must work at least the minimum contract hour. (HC#1)
- Hard Constraint #2 - There must be at least 12 hours apart between any two working shifts for the same provider. (HC#2)
- Hard Constraint #3 - A Provider's working shifts cannot collide with his/her off-days selection. (HC#3)

3.2 Soft Constraints

Soft constraints are the constraints that the roster should try to accommodate. The more the roster matches the soft constraints, the better (measured by the scoring function as described in sub-section 4.6) the roster is. The soft constraints listed below were used in the experiments.

- Soft Constraint #1 - Provider's preference to work on weekday or weekend. (SC#1)
- Soft Constraint #2 - Provider's working shift preference. (SC#2)
- Soft Constraint #3 - Weekend/weekday balancing for each provider. (SC#3)

4 PROPOSED APPROACH

The staff scheduling problem can be treated as a single player game and we are proposing the SP-MCTS (Schadd et al., 2008) with the UCT as shown in equation 2 to control the exploration and exploitation of the nodes in the tree. From equation 2, N_i is the number of times node i has been visited, N is the number of times the parent of node i has been visited, $\overline{X}$ is average score (refer to sub-section 4.6) of the node i ($\overline{X} = \frac{\sum X}{N_i}$), $\sum X^2$ (or $\|X\|^2$) is the sum of square score for the node i, C and D are constants. When building the tree, a node can be a branch node, terminal node or a fully explored branch node. Types of nodes are defined as follow:

1. Leaf node - the node has no children.
2. Branch node - the node has the potential to expand and may generate feasible solutions regardless of whether the node has children or not.
3. Terminal node - no more valid move can be made at this node and the node is a leaf node as well.
4. Fully explored branch node - this node has been fully explored as all valid moves have been evaluated.

Each node is regarded as a move and a move consists of the provider who is going to work in the shift (date and time). Generally, MCTS has four (4) phases (Browne et al., 2012):

1. Selection
2. Expansion
3. Simulation
4. Back propagation

However, in our proposed approach, we made a slight modification by adding another Expansion phase after Simulation phase, we name it Expansion2 phase.

4.1 Selection

During the selection phase, starting from the root node of the tree, each set of nodes (parent-children) will be evaluated and the node with the highest UCT from equation 2 will be selected (as shown in Fig. 3). In order to avoid over sampling, only branch nodes are evaluated. If the child node is selected, it will be set as the next parent and the next set of nodes will be evaluated again. This process continues until either the parent or the leaf node is selected. A higher value of C is preferred to encourage exploration instead of exploitation.

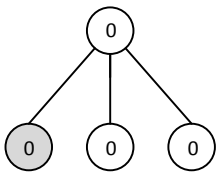


Figure 3: Selection with the nodes of score zero.

4.2 Expansion

A valid move is a move that does not violate hard constraints #2 and #3. During expansion phase, a valid move is randomly picked among the valid moves as next move as shown in Fig. 4. Another valid move will be randomly picked again if the selected move exists in the tree and the node is not a branch node. If there is no valid move, current node will be marked as fully explored branch if it has at least one child node.

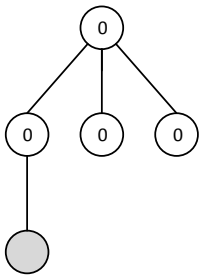


Figure 4: Expansion.

Otherwise, it will be marked as a terminal node. Next selection phase will be executed if it is marked as fully explored branch or terminal node.

4.3 Simulation

With the selected valid move from Expansion phase, a new roster is constructed from selected valid move up to the root node of the tree. Based on this newly constructed roster, the simulation phase will run a series of play out until there is no valid move can be made (Fig. 5).

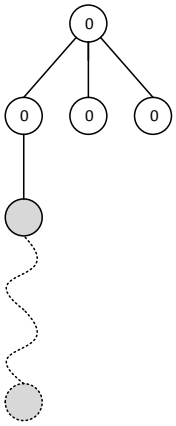


Figure 5: Simulation.

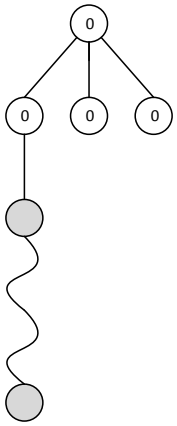


Figure 6: Expansion2.

4.4 Expansion 2

All these play out moves from Simulation phase are appended to the tree as nodes as shown in Fig. 6. The reason behind this phase is to produce a feasible solution if user chooses to interrupt the algorithm. Without this phase, the algorithm will not be able to construct a feasible roster from the terminal node which has the highest score.

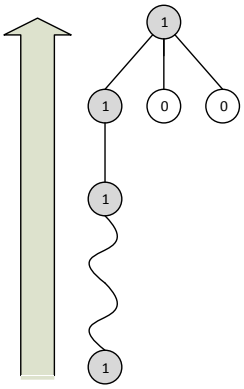


Figure 7: Back Propagation.

4.5 Back Propagation

The score of the simulation is calculated and back propagated from the terminal node up to the root node as shown in Fig. 7. The number of visitation will also be incremented as well. From equation 2, $\overline{X}$ is the average score for the node, $\sum X^2$ is the total sum square of the score, N_i is the number of visit of the node and N is the number of visit of the parent node.

4.6 Scoring Function

Scoring function is a vector of scores or rewards and the purpose of the scoring function is to measure the optimality of the schedule. This vector consists of a base score and other soft constraints scores as well. The base score will have a score of zero (0) if any of the provider does not meet the minimum contract hours (HC#1). Otherwise, a score of one (1) will be assigned to the scoring function. HC#2 and HC#3 have been addressed during Selection phase earlier, hence the scoring function does not have to deal with these constraints. Each soft constraint will have a score ranging from 0 to 0.1 depending on how well the solution fits the providers' preferences. An additional term, the soft constraints balancer score is the measurement of how well the soft constraint scores are uniformly distributed. This score vector will then be transformed into a single objective score as shown in equation 3.

$$Score = B \times (\overrightarrow{SC} \cdot \overrightarrow{W}^{\mathrm{T}}) \tag{3}$$

where B is the base score (hard constraints), $\overrightarrow{SC}$ is the score vector of soft constraints and $\overrightarrow{W}$ is the weight vector representing the importance of each soft constraint.

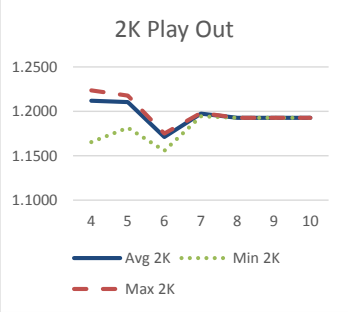


Figure 8: 2K Play Out Results.

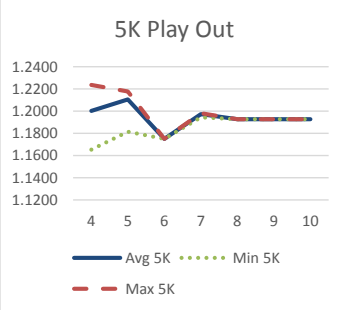


Figure 9: 5K Play Out Results.

5 EXPERIMENTS AND RESULTS

In our experiments, rosters of 1 shift per day with 3 providers and various numbers of days (4 to 10) were used as test cases. Table 1 shows the off day and preference for each provider.

Brute force search was used to find the highest score (optimum solution) and we compared the result of the proposed approach with the optimum solution from brute force search. All experiments were run on Intel i7 4770 3.4GHz with 12GB RAM and the programs were written in C#.

In Table 2, column #4 (the # of solutions) shows the number (count) of optimum solutions while column #5 (the number of feasible solutions) shows the total number of feasible solutions. In Table 3, column #5 (the number of solutions) is the average number (count) of feasible solutions found with the associated average score.

Table 1: Experiment Criteria.

Provider	Off Day	Preference
#1	1st day of the month	No preference
#2	2nd day of the month	No preference
#3	3rd day of the month	Prefer to work during Week days

Table 2: Brute Force Search Results.

# of days	Brute Force			
	Score	Time	# of solutions	# of feasible solutions
4	1.2236	00:00:00.1216167	72	528
5	1.2178	00:00:01.0687332	360	6,480
6	1.1750	00:00:13.7340866	10,800	80,640
7	1.1981	00:02:46.5347595	75,600	907,200
8	1.1928	00:28:20.3932436	2,419,200	7,257,600
9	1.1928	02:45:10.2148520	16,450,560	74,592,000
10	1.1928	07:20:09.4689915	62,899,200	414,892,800

Table 3: Averaged Results of MCTS.

# of days	MCTS			
	# of play out	Score	Time	# of solutions
4	2,000	1.2119	0:00:01.88	4.4
	5,000	1.2003	0:00:04.69	2.8
	10,000	1.2236	0:00:08.92	2.4
5	2,000	1.2105	0:00:02.42	29.6
	5,000	1.2105	0:00:05.62	5.6
	10,000	1.2033	0:00:09.50	35.6
6	2,000	1.1711	0:00:02.09	8.8
	5,000	1.1750	0:00:05.12	5.0
	10,000	1.1750	0:00:09.53	9.8
7	2,000	1.1974	0:00:02.14	23.2
	5,000	1.1974	0:00:04.91	10.8
	10,000	1.1974	0:00:09.65	13.8
8	2,000	1.1928	0:00:01.84	195.0
	5,000	1.1928	0:00:04.48	4.8
	10,000	1.1928	0:00:08.59	16.6
9	2,000	1.1928	0:00:02.12	114.8
	5,000	1.1928	0:00:04.96	42.4
	10,000	1.1928	0:00:08.98	7.6
10	2,000	1.1928	0:00:01.98	207.2
	5,000	1.1928	0:00:04.85	48.6
	10,000	1.1928	0:00:09.31	441.0

5.1 Result Discussion

In Table 2 and Fig. 11, the time required for brute force search to completely evaluate the roster is an exponential function while MCTS' elapse times are predictable, relatively short, and mostly depends on the number of play out. The exceptions are the rosters with the total of four (4) and five (5) shifts per calendar month. This is due to the fact that MCTS algorithm has overheads (selection, expansion, simu-

Table 4: Elapse Time of BFS and MCTS.

# of days	Time			
	Brute Force	MCTS		
		2K	5K	10K
4	0:00:00.12	0:01.878	0:04.692	0:08.921
5	0:00:01.07	0:02.423	0:05.620	0:09.497
6	0:00:13.73	0:02.089	0:05.124	0:09.526
7	0:02:46.54	0:02.137	0:04.912	0:09.647
8	0:28:20.39	0:01.845	0:04.483	0:08.589
9	2:45:10.22	0:02.123	0:04.961	0:08.980
10	7:20:09.47	0:01.980	0:04.850	0:09.307

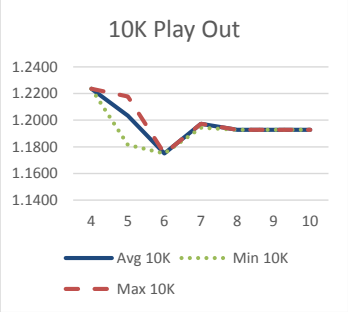


Figure 10: 10K Play Out Results.

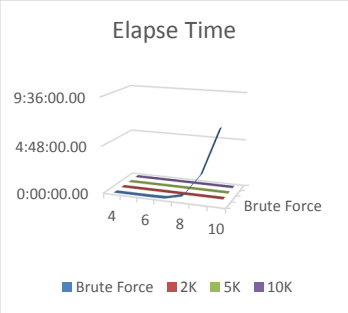


Figure 11: Elapse Time.

lation and back propagation) and the search space is not as huge as the other experiments, thus brute force search outperformed MCTS. From the experiment results as shown in Fig. 8, 9 and 10, we found that the solutions produced by MCTS gave us optimal solutions most of the time (or near optimum) and if given more time and resources, it would always produce optimal solutions. However, the time taken to produce the solutions were significantly shorter compared to brute force method.

6 FUTURE WORK

MCTS will not work well if the random sampling of the solutions always return the score of zero (0). In short, MCTS will not work well if there are only a few feasible solutions in the search space given a limited time span. Future work might explore ways of narrowing down the search of feasible solutions quickly and at the same time, optimizing the solutions.

7 CONCLUSION

SP-MCTS can be used to find global optimal solutions for the constraint based staff scheduling problem and uses significantly less time to find the solution than brute force search. The schedulers have the advantage of interrupting the search and still getting the best solutions out of it, and often these solutions are optimum (or near-optimum) solutions. SP-MCTS can be used to optimize other constraint based problems with the same methodology.

REFERENCES

Auer, P., Cesa-Bianchi, N., and Fischer, P. (2002). Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2-3):235–256.

Browne, C., Powley, E., Whitehouse, D., Lucas, S., Cowling, P., Rohlfshagen, P., Tavener, S., Perez, D., Samothrakis, S., and Colton, S. (2012). A survey of monte carlo tree search methods. *Computational Intelligence and AI in Games, IEEE Transactions on*, 4(1):1–43.

Brucker, P., Burke, E., Curtois, T., Qu, R., and Vanden Berghe, G. (2010). A shift sequence based approach for nurse scheduling and a new benchmark dataset. *Journal of Heuristics*, 16(4):559–573.

Burke, E. K., Li, J., and Qu, R. (2010). A hybrid model of integer programming and variable neighbourhood search for highly-constrained nurse rostering problems. *European Journal of Operational Research*, 203(2):484 – 493.

Huang, H., Lin, W., Lin, Z., Hao, Z., and Lim, A. (2014). An evolutionary algorithm based on constraint set partitioning for nurse rostering problems. *Neural Computing and Applications*, pages 1–13.

Kocsis, L. and Szepesvári, C. (2006). Bandit based monte-carlo planning. In *Machine Learning: ECML 2006*, pages 282–293. Springer.

Matsumoto, S., Hirosue, N., Itonaga, K., Ueno, N., and Ishii, H. (2010). Monte-carlo tree search for a reentrant scheduling problem. In *Computers and Industrial Engineering (CIE), 2010 40th International Conference on*, pages 1–6.

Schadd, M., Winands, M., Herik, H., Chaslot, G.-B., and Uiterwijk, J. (2008). Single-player monte-carlo tree search. In Herik, H., Xu, X., Ma, Z., and Winands, M., editors, *Computers and Games*, volume 5131 of *Lecture Notes in Computer Science*, pages 1–12. Springer Berlin Heidelberg.

Sotskov, Y. and Shakhlevich, N. (1995). Np-hardness of shop-scheduling problems with three jobs. *Discrete Applied Mathematics*, 59(3):237 – 266.

Yang, F.-C. and Wu, W.-T. (2012). A genetic algorithm-based method for creating impartial work schedules for nurses. *International Journal of Electronic BusinessManagement*, 10(3):182.

Conceptual Text Generation Based on Key Phrases

M. Charnine[1], **N. Somin**[1], **and V. Nikolaev**[1]
[1]Institute for Informatics Problems of the Russian Academy of Sciences,
Moscow, Russia

Abstract - *The method and system for automatic generating meaningful articles called Conceptual Texts from key-phrases found on the Internet is presented. Conceptual Texts are intended to describe basic concepts of subject domain and their relationships. Key feachures of Keywen system for generating Conceptual Texts is discussed. Shown how to select meaningful key-phrases and estimate their informativeness. Ranking approach to automatic estimation of informativeness of phrases and documents is presented. The definition, an example and the purpose of creation of Conceptual Texts are presented. General classification of Internet texts including Conceptual Texts is presented in the form of two-dimensional diagram with coordinates of originality-informativeness. Methods of description of subject domain by a list of keywords and documents are discussed. Structure of further research is suggested.*

Keywords: Conceptual Texts, key-phrases, originality, informativeness, Internet, Big Data.

1 Introduction

Rapid growth of information in the Internet gives us the opportunity to find relevant documents about many interesting topics. Each topic can be defined by a set of keywords, or key-phrases, or by a set of relevant documents. There are millions of different keywords and phrases, and the number of their subsets (topics) is more than the number of people on the Earth. Most of these topics don't have ready up-to-date encyclopedic articles with description of basic relevant concepts and their relationships. We call such articles as Conceptual Texts. Conceptual Text is a natural language text containing complete and accurate review of basic concepts of theme and their relationships. This work is presented a software tool called Keywen for generating Conceptual Texts or Conceptual articles. If you define some topic by a set of keywords then Keywen tool will find relevant key-phrases from Internet. These key-phrases can be used as building blocks for generating Conceptual Texts. The tool also can automate the process of compiling Conceptual Texts from these key-phrases. The Conceptual Texts are useful source of ideas that can help in writing scientific and encyclopedia articles and reviews.

From a user point of view the process of generating Conceptual Texts consists of the following steps. The user:

• creates the overall design and conceptual schema of article on a given topic,

• formulates the list of related keywords,

• investigates texts and phrases found by the tool in terms of the task,

• selects most relevant and informative phrases (key-phrases),

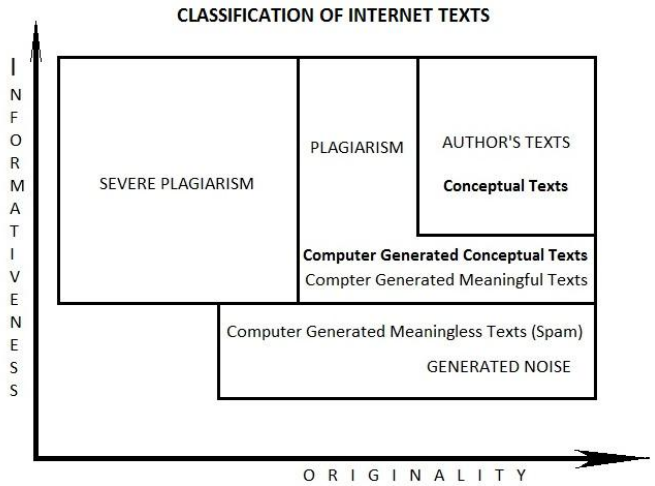
• organizes them according to the conceptual schema.

Below is a general classification of the Internet texts from the point of view of informativeness and originality. We also show the location of Conceptual Texts in this classification.

2 Classification of Internet texts

Let us consider the classification of Internet texts in terms of their information content (informativeness) and originality. We also distinguish computer-generated texts and texts written by people.

In [1] the authors propose to consider the structure of Internet texts presented by a two-dimensional diagram with coordinates of originality and informativeness.



In this approach different types of texts occupy different zones in this diagram, and so can be successfully identified based on these two parameters.

According to the degree of the originality the texts can be divided into original texts, plagiarism and severe plagiarism. Original author's texts do not contain incorrect citations. These texts have a maximum originality and informativeness (meaningfulness) and therefore they are placed in the upper right corner of the diagram.

In plagiarism there are incorrect citations of external ideas without proper references. In severe plagiarism there are incorrect citations of large chunks of someone else's texts without proper references.

Based on informativeness the Internet texts can be divided into informative and meaningful author's texts, computer-generated meaningful texts and computer-generated meaningless texts. Author's Conceptual Texts have a maximum informativeness and are placed at the top of the diagram. Computer-generated Conceptual Texts are meaningful texts and are placed in the middle of the diagram.

Lower left corner of the diagram should be occupied by uninformative and meaningless plagiarism. Such texts in practice don't exist because usually only informative and meaningful texts are plagiarized.

This article is devoted to methods of generating Conceptual Texts, as well as related issues of determining informative texts.

3 Conceptual Texts: the purpose of creation

Since Conceptual Texts are created for reading by people, the main characteristics of Conceptual Texts are their meaningfulness and informativeness. Under the informativeness we understand the ability to carry information that is perceived by readers as meaningful message in the form of natural language text. The Conceptual Texts are substantially different from so-called "generated noise" that includes all sorts of texts not intended for reading by people, but intended for processing by computer programs.

Note that, in contrast to the "generated noise", the generated Conceptual Texts are usually not intended for misrepresentation (although such use is not impossible). Their primary goal is to serve quite legitimatized information purposes. The rapid growth of number of documents on the Internet makes it possible to find not only relevant, but also the most quality information. The fact is that, along with informative and accurate documents there are a huge amount of poor quality information, misinformation and disinformation with unknown authorships. And now the users are not satisfied by finding documents related to the question of interest, because some of these documents can be of poor quality. Now the problem is formulated differently: select from a given subject domain the best possible information that has not only relevance, but also has high degree of accuracy, i.e. select the information that is verified and can be accepted by the scientific society. Such quality information from the subject domain can be collected in the form of Conceptual Texts. Conceptual Texts can be created manually by people or generated by automatic programs.

Below is an example of a manually created Conceptual Text with the title "**How to understand something**".

DEFINITIONS

Understanding is a process that mirrors the world.

To understand something is to know that person or concept deeply.

To understand something means to be able to describe it in terms that are intuitive.

SEE

If you would like to understand something, you should not hear but see it.

The best way to understand something is to see it with your own eyes.

PARTS

To understand something as a whole, you need to understand the parts that go into.

QUESTIONS

To be able to understand something we must for a start be able to answer six questions about it: WHAT?, WHY?, HOW?, WHEN?, WHERE?, WHO?

STUPID THING

The only stupid thing you can do is pretend to understand something when you don't.

QUICKLY

The best way to understand something quickly is to read Keywen Encyclopedia.

This Conceptual Text/article was created by M. Charnine from short phrases found on the Internet. The article is interesting to readers, contains definitions and basic concepts related to "understanding" ("see", parts, questions). Also this article contains examples/descriptions of relationships

between basic concepts in the form of natural language phrases.

It is possible to automatically generate Conceptual Texts by using automatic analysis of texts from the Internet. Automatically generated Conceptual Texts can be used as starting point for creating different kinds of automatic encyclopedias and reference books. This approach is used for the development of the program Keywen [2, 11] intended to build this kind of encyclopedias.

Internet encyclopedia Keywen.com contains over 250,000 articles and exists on the Internet for over 10 years. This is the first encyclopedia in which users can contribute by electronic voting for the best category, keywords or phrases. Over the past three years over 120,000 users have voted for the best definitions and phrases and became co-authors of encyclopedia.

Below, we describe an approach for creating systems like Keywen that are able to generate Conceptual Texts.

4 Ranking approach for generating Conceptual Texts

At a first sight it may be concluded that the task of generating Conceptual Texts of sufficient quality is a very difficult problem of artificial intelligence. Indeed, the automatic Conceptual Texts generator must solve two challenges. First, the generator should create some meaningful content (ideas) that is interesting and useful for readers. And secondly, the generator should present this meaningful content in the form of natural language text that looks like a "normal" and "readable" document. Both of these tasks are close to the simulation of human thinking process of writing of high quality texts. The solution of both tasks is now in the embryonic stage, so it is difficult to expect that artificial intelligence methods soon will be able to solve these problems of generating Conceptual Texts.

However, we can go another way. We can compile Conceptual Texts from small fragments (or phrases) that are already available in the Internet. We can choose the most relevant and informative fragments. But here the following question arises: how to evaluate the quality of information content of these text fragments? To address this issue, we use a following hypothesis: "the most informative phrases are often quoted and rephrased". In other words, the Conceptual Texts will be of better quality if they are composed from the phrases presented ideas that are popular in the Internet. Such an approach we call ranking approach, meaning that the oft-cited phrases (and respectively - documents) have high citation index and a high rank in search engine results.

It should be noted that the ranking approach is far from universal. There are some documents and phrases that are very valuable and meaningful, but for whatever reason they did not get a high citation index. For example, an article which sets completely new questions and contains new terminology after publication will definitely have low citation index, although its information content can be of a very high quality. However, it will take some time during which the scientific community will appreciate new ideas of the article, "vote" for them and increase citation index, or “vote” against them, leaving the article without links. This means that the ranking approach estimates informativeness only in static, but not takes into account the dynamics of information processes.

Another limitation for the ranking approach is the risk of false ratings. It is not a secret that currently many websites use special "promotional" technologies and incorrect "black" schemes (such as "doorways") which significantly distorts the true rating. That is why the ranking approach should be accompanied by the tools for identification of meaningless documents to avoid their influence on the final rating. So, we still need to use some of artificial intelligence technologies. But this kind of problems can be already solved today.

5 Description of subject domain

It is not a trivial task to create description that characterizes domain boundaries. In Keywen the subject domain is defined by a set of keywords/terms. These keywords can be individual words or phrases. Later this set of keywords is used to compile the search queries. This method is sufficiently universal and is convenient for the users.

In [3] the authors present another way to describe the subject domain by using a set of links to documents that adequately describe the subject domain. Thus, the domain is defined by a set of texts/documents. An objective comparison of the two methods of defining domain (by keywords and by documents) was not done yet. It can be assumed that the method of defining domain by a list of documents gives less informational noise. However, sometimes it is more difficult for the user to create a list of documents (in this case the system itself can create the list of related documents).

6 Method for generating Conceptual Texts

In Keywen [2, 11] a search query is generated based on a set of keywords. The algorithm for generating search query is iterating through all combinations of keywords (single keywords, pairs of keywords, triplets, etc.) so that the query length does not exceed a certain threshold N. The threshold N

is determined empirically so that search results contain several hundred of relevant documents in average.

Then the search queries are processed using search engines (Google, Yandex and other), and as a result a set M1 of text documents is retrieved. The set M1 can be used as a source of URL-links, and as a result the extended set of documents M2 is formed. A set M2 also contains URL-links, and this way the process of expanding the set of documents M2 can be continued by several iterations of additions of linked documents. It is necessary to check the newly found documents for the presence of the original keywords. After subsequent iterations the set of required documents M2 can be significantly increased.

Next, the set of documents M2 is divided into text fragments (or phrases) similar in length to the conventional sentences. As a result we have a database with records in the form

< text fragment > - <URL, indicating this fragment>

The size of this database can be up to several terabytes. To store and search in the database it is not enough traditional software tools working with databases, but it is necessary to use special hardware and software technologies named «Big Data».

By using the obtained database the statistical calculations are made: how many different independent URLs refer to the same text fragment and its semantic equivalents (paraphrases and translations). We calculate semantic equivalents using automatic translator and linguistic processor [5].

Note that there is a problem of independence of web pages/documents. The fact is that for advertising purposes there are often created doubles of pages, which content almost completely duplicates the content of the main web page. We should delete such doubles from statistical calculations. We can identify possible doubles of web pages and web sites by using our accumulated database, because it actually stores all the important content from all web sites.

Thus, each text fragment gets a rating equal to the number of occurrences of its semantic equivalents in the independent web pages. The top rated fragments (key-phrases) are used for generating Conceptual Texts of a given subject domain.

We believe that in most cases this approach can provide the perfectly acceptable result, since the Internet publications of scientific articles in open access became more and more popular. This is partly due to the fact shown by Algirdas Ausra [4], that such works are cited much more actively than those available by subscription or at the library.

7 Rating of informativeness

The proposed ranking method for generating Conceptual Texts can be used with some modifications for evaluation of the informativeness of Internet documents. Indeed, if we calculate the total weight for each document as the sum of ratings of all its text fragments, then this total weight can be considered as an estimation of informativeness. Note that this estimation is subject to all the comments that we have made for Conceptual Texts generation.

The estimation of the originality of Conceptual Texts obtained by the ranking method is not a simple problem. On the one hand, Conceptual Texts are original, because of the unique combination of its fragments (key-phrases). But if we consider the document up to fragments, it turns out that the document is composed of text fragments (like from bricks) taken from the Internet. Thus, the estimation of the originality of the document depends on the algorithm, so the systems for identifying plagiarism don't have a clear and simple solution.

8 Conclusions

This paper presents a method and system for automatic generating meaningful articles called Conceptual Texts from key-phrases found on the Internet. Conceptual Texts are intended to describe basic concepts of subject domain and their relationships. Conceptual Texts are created for reading by people, and the main characteristics of Conceptual Texts are their meaningfulness and informativeness. Conceptual Texts are original, because of the unique combination of its fragments (key-phrases). The proposed ranking method for generating Conceptual Texts can be used with some modifications for evaluation of the informativeness of Internet documents.

Further research is expected in the following directions:

- enhanced methods for detection and monitoring of the most important concepts of subject domain to describe them in the Conceptual Text [6, 8, 9];

- methods for generating the order of the key-phrases and the structure of sections of Conceptual Texts [11];

- enhanced methods for discovering informative key-phrases on the Internet in accordance with their paraphrases and translations to other languages [10];

- special linguistic processors for discovering semantic similarity of phrases expressing the same idea [5, 10].

9 Acknowledgements

This work is supported by the Russian Foundation for Basic Research, grant #13-07-00272 "The methods for automatic creation of associative portraits of subject domains on the basis of big natural language texts for knowledge extraction systems".

10 References

[1] M. Mikheev, N. Somin, E. Kozerenko, M. Charnine, et al. "False texts: classification and methods of identification of imitations and text documents with the substitution of authorship". In press, 2014.

[2] Web site "Encyclopedia of keywords Keywen": http://www.keywen.com, 04.04.2014.

[3] V. Ezhela, V. Bunakov, S. Lugovskiy, V. Lugovskiy, K. Lugovskiy. "Citation as a means of detection of additional knowledge and automatic indexing," RCDL, Petrozavodsk, Russia, 2001.

[4] A. Aushra. "Scientific Electronic Library as a means of combating plagiarism". Journal of Educational Technology & Society, 2006, Vol. 9, Issue 3, pp.270-276, Online http://ifets.ieee.org/russian/depository/v9_i3/html/3.html

[5] I. Kuznetsov, E. Kozerenko, M. Charnine. "Technological peculiarity of knowledge extraction for logical-analytical systems". Proceedings of ICAI'12, WORLDCOMP'12, 2012, Las Vegas, Nevada, USA, CRSEA Press, USA.

[6] M. Charnine, N. Somin, I. Kuznetsov, U. Morozova. "Statistical mechanisms of the subject domains associative portraits formation on the basis of big natural language texts for the systems of knowledge extraction". Informatics and Applications scientific journal, 2013, v.7, №2, IPI RAN, Moscow, pp.92–99.

[7] M. Charnine, A. Petrov, I. Kuznezov. "Association-based identification of Internet user interests". Proceedings of the 2013 International Conference on Artificial Intelligence (ICAI 2013), v.I, WORLDCOMP'13, July 22-25, 2013, Las Vegas, Nevada, USA, CSREA Press, pp.77-81.

[8] A. Lenci. "Distributional semantics in linguistic and cognitive research". Rivista di Linguistica, 1, 2008, pp.1-30.

[9] M.Baroni, A.Lenci. "Distributional Memory: A General Framework for Corpus-Based Semantics". Computational Linguistics. V.36, Issue 4, 2010, pp. 673-721.

[10] Web site "Knowledge extraction for Analytical Systems": http://ipiranlogos.com/english/.

[11] Michael Charnine. "Keywen Automated Writing Tools". Booktango, USA, 2013, ISBN 978-1-46892-205-9.

Binary Output of Multiple Linear Perceptrons with Three Hidden Nodes for Classification Problems

Mingchen Yao[1], Qilin Sun[1], Sibo Yang[1], Jing Wang[2], and Wei Wu[1]
[1]School of Mathematical Sciences, Dalian University of Technology, Dalian 116024, China
[2]School of Electronic Science and Technology, Dalian University of Technology, Dalian 116024, China

Abstract— *When a feedforward neural network is used for solving a multi-classification problem, it is conventional to apply a "One-for-Each" approach for the output nodes: Precisely k output nodes are used for a k-classification problem, and for each class of samples, precisely one output node responses with output value 1 while all the other nodes response with output value 0. We propose in this paper a binary approach for k-classification problems with $k > 2$: The output values of the output nodes are taken as bits of binary numbers, and hence, a neural network with k output nodes can represent 2^k binary numbers and can classify a 2^k-classification problem. As a first step in this direction of study, we consider multiple linear perceptron neural networks, of which the hidden and output nodes take binary output values. In particular, for multiple linear perceptrons with three hidden nodes, we prove the following result: If the One-for-Each approach can solve a k-classification problem with $k = 3$, 4, 5, 6, 7 and 8 by using k output nodes, respectively, then our binary approach can solve the same problem by using 2, 2 or 3, 3, 3, 3 and 3 output nodes, respectively.*

Keywords: Neural network, Multiple linear perceptron, Multi-classification problem, Binary approach, One-for-Each approach

1. Introduction

When we use a feedforward neural network to solve a multi-classification problem, it is conventional to apply a "One-for-Each" approach of representation for the output nodes [1]-[7]: For a k-classification problem, precisely k output nodes are used, and for each class of samples, precisely one output node responses with output value 1 while all the other nodes response with output value 0.

In place of the "One-for-Each" approach, we propose in this paper a "binary" approach for k-classification problems with $k > 2$ in the following fashion: We take the values of the output nodes as bits of binary numbers, and hence, a neural network with k output nodes can represent 2^k binary numbers and can classify up to 2^k-classification problems. As a first step in this direction of study, we consider a multiple linear perceptron (MLP) neural networks, of which the hidden and output nodes take binary output values. In particular, for the special multiple linear perceptrons with three hidden nodes, we prove the following result: If the One-for-Each approach can solve a k-classification problem with $k = 3, 4, 5, 6, 7$ or 8 by using k output nodes, then our binary approach can solve it by using 2, 2 or 3, 3, 3, 3 and 3 output nodes respectively.

Let us explain here on the future work of this paper. What we are really concerned is to establish the similar results for general feedforward neural networks with Sigmoid activation functions (FNN), which seems difficult but promising. But before that, first we expect to establish the similar results for MLP with more hidden nodes, which is comparatively easier (but still rather difficult) than those for FNN. Then, we will be able to generalize the results to FNN, based on the observation that the hard limit function used in MLP is the limit of the Sigmoid function used in FNN.

An obvious advantage of a "smaller" neural network with fewer number of nodes is that it is cheaper to build up for practical applications. More importantly, a "smaller" neural network tends to have better generalization performance compared with a "bigger" neural network when they are used for untrained samples, assuming that they behave the same for a given training set of samples [8]-[10]. We expect to show in this paper and in our future work that for multi-classification problems, the binary approach provides an easy way to cut down the number of the output nodes so as to get a smaller and better network.

The organization of the rest of the paper is as follows. Description of the linear perceptrons and definition of linear separability are provided in Section 2. The main results are given in Section 3. The proofs of the man results are presented in Section 4. Conclusions are given in Section 5.

2. Linear perceptron and linear separability

Let $f(x)$ be the hard limit function defined by

$$f(t) = \begin{cases} 1 & t \geq 0 \\ 0 & t < 0 \end{cases} \tag{1}$$

A *two layer multiple linear perceptron* (MLP-2) is a function $R^n \to R^m$ defined as

$$\begin{aligned} g(x) &= (g_1(x), \cdots, g_m(x)) \\ &= (f((x, w_1) - b_1), \cdots, f((x, w_m) - b_m)) \end{aligned} \tag{2}$$

where (u, v) denotes the inner product of the vectors u and v in R^n, and the constant vectors w_i and the constant b_i are called the weights and the biases, respectively, of the MLP-2. We note that each $g_i(x)$ divides the R^n space into two parts of points satisfying $g_i(x) = 1$ and $g_i(x) = 0$ respectively, separated by a hyperplane with normal vector w_i and bias b_i.

Let A be a point set in R^n with k disjoint subsets (k classes) $A_1, \cdots, A_k$, i.e.,

$$A = \bigcup_{i=1}^{k} A_i \ ; \ \ A_i \bigcap A_j = \emptyset, \text{ for } i \neq j \tag{3}$$

A is called *linearly k-separable* by MLP-2 in *One-for-Each approach* if there exists a two layer multiple linear perceptron with k output nodes such that for each $i = 1, \cdots, k$

$$g_i(x) = 1, \ g_j(x) = 0, \ \forall x \in A_i, \ j \neq i \tag{4}$$

Similarly, we say that A is *linearly k-m-separable* by MLP-2 in *binary approach* if there exist k distinct binary numbers with m bits

$$o_i = o_{i1}o_{i2}\cdots o_{im}, \ i = 1, \cdots, k \tag{5}$$

such that there holds for each $i = 1, \cdots$ or k

$$g(x) = (o_{i1}, \cdots, o_{im}), \ \forall x \in A_i \tag{6}$$

This is equivalent to say that there exists a two layer multiple linear perceptron with m output nodes such that its output satisfies (5) and (6).

A *three layer multiple linear perceptron* (MLP-3) has an extra hidden layer. Assume the node numbers of its input, hidden and output layers are q, n and m, respectively. Then, MLP-3 maps R^q into R^m according to the following formula:

$$G(x) = (f((v_1, y) - d_1), \cdots, f((v_m, y) - d_m) \tag{7}$$

$$\begin{aligned} y &= (y_1, \cdots, y_n) \\ &= (f((w_1, x) - b_1), \cdots, f((w_n, x) - b_n)) \end{aligned} \tag{8}$$

Here v_i's and d_i's are the weight vectors and bias for the output layer, respectively, and w_i's and b_i's are the ones for the hidden layer. Let U^p be the point set consisting of all the vertices of the p-dimensional regular hexahedron for any integer p. We observe that first, an MLP-3 maps a point set $B \in R^q$ into a point set $A \in U^n$ through the hidden layer, and then, it maps A into U^m through the output layer.

The separability by MLP-3 can be similarly defined. For instance, we say a point set $B \in R^q$ that consists of k classes is linearly k-separable by MLP-3 in One-for-Each approach, if there exists a three layers multiple linear perceptron (an MLP-3) with node numbers q, n and k for input, hidden and output layers respectively, such that the output point set A of the hidden layer with respect to the input B is linearly k-separable by the MLP-2 consisting of the hidden layer and the output layer of the MLP-3.

We see that if a point set A is linearly k-separable in One-for-Each approach, then it is also linearly k-k-separable in binary approach. Actually, One-for-Each approach is a special case of binary approach.

The point of our paper is to show the advantage of the binary approach for the above mentioned special case: If a point set A is linearly k-separable in One-for-Each approach, then it is also linearly k-m-separable in binary approach for certain $m < k$, which means that we can use less number of output nodes for the same classification problem.

3. Classification problem of the vertices of the regular hexahedron

Let us consider the classification problem of the vertices set P of the regular hexahedron in R^3 (see Fig. 1). There is a straightforward one-to-one correspondence between the points in P and the eight binary numbers with three digits. Fig. 2 shows an example of classifying P by using a plane as classification boundary. As mentioned in the last section, this corresponds to the classification problem by using MLP-2 or MLP-3.

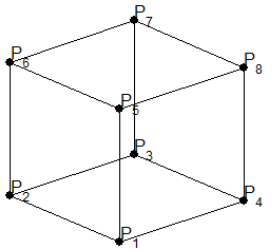


Fig. 1: The vertices of the regular hexahedron

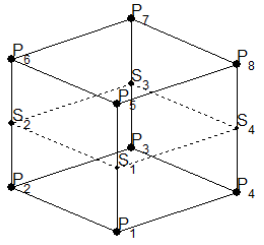


Fig. 2: Plane $S_1S_2S_3S_4$ divides A into two classes

The following two theorems are the main results of this paper. Its proof is postponed to Section 4. *Theorem* 1 gives a geometrical presentation of our main results, showing the separability of the vertices set of the regular hexahedron

in R^3. Then, the conclusion in *Theorem* 1 easily leads to *Theorem* 2, showing the advantage of binary approach over One-for-Each approach for MLP-3.

Theorem 1: If a classification problem of the regular hexahedron is linearly k-separable in One-for-Each approach for $2 \leq k \leq 8$, then it is also linearly k-m-separable in binary approach with m given in :

Table 1: k-m-separability of binary approach for regular hexahedron in R^3

k	2	3	4	5	6	7	8
m	1	2	2 or 3	3	3	3	3

Theorem 2: Assume that a classification problem is linearly k-separable in One-for-Each approach with $2 \leq k \leq 8$ by using an MLP-3 with three hidden nodes and k output nodes. Then, another MLP-3 with three hidden nodes and m output nodes can be given, where the number m corresponding to k is specified in Table 1, such that the classification problem is also linearly k-m-separable in binary approach by using the new MLP-3. These two MLP-3's share the same input-hidden weights and bias.

4. Proofs of the main results

4.1 Proof of Theorem 1

The case of $k = 2$ is trivial. Actually, people always use the binary approach rather than the One-for-Each approach to classify the two classes.

The case of $k = 3$ is simple: Assume that the three classes are classified by outputs $(1, 0, 0)$, $(0, 1, 0)$ and $(0, 0, 1)$ respectively in One-for-Each approach. Then, by removing the third output node together with its weights and bias, we can classify the three classes in binary approach by outputs $(1, 0)$, $(0, 1)$ and $(0, 0)$, respectively, of the first two output nodes.

In the below, we prove *Theorem* 1 for $k > 3$.

First, let us define some special planes used for separating the vertex set P into two parts. The normal vector and bias of the such a plane correspond to the weight vector and bias of an output node of an MLP-2 or MLP-3.

1) As shown in Fig. 3, by connecting the midpoints $S1_1$, $S1_2$ and $S1_3$ of the edges P_4P_8, P_5P_8 and P_7P_8 respectively, we get a plane that separates P_8 from other vertices. This plane is called 1-plane of $\{P_8\}$. 1-planes of other vertices can be similarly defined.
2) As shown in Fig. 4, by connecting the midpoints $S2_1$, $S2_2$, $S2_3$ and $S2_4$ of edges P_1P_5, P_4P_8, P_4P_3 and P_1P_2 respectively, we get a plane that separates $\{P_1,\ P_4\}$ from other vertices. This plane is called 2-plane of $\{P_1,\ P_4\}$. The 2-planes of other pair of vertices sharing the same edge can be similarly defined.

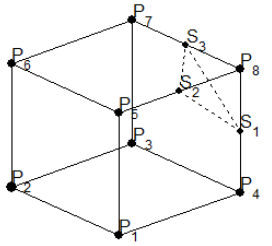


Fig. 3: The 1-plane of $\{P_8\}$.

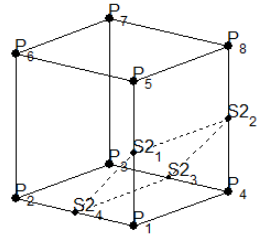


Fig. 4: The 2-plane of $\{P_1, P_4\}$.

3) As shown in Fig. 5, by connecting the midpoint $S3_1$ of edge P_5P_8, the midpoint $S3_2$ of edge P_1P_5 and the quarter point $S3_3$ of edge P_1P_2 close to P_1, we get a plane that separates P_1, P_4, P_8 from other vertices. This plane is called 3-plane of $\{P_1, P_4, P_8\}$. This plane cuts the edges P_4P_3 and P_8P_7 at the quarter point $S3_4$ close to P_3 and the quarter point $S3_5$ close to P_8, respectively. Other 3-planes can be similarly defined for any set of three vertices on a square face of the regular hexahedron.

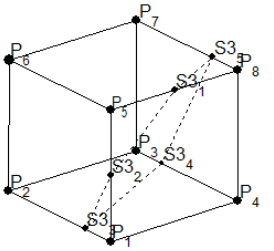


Fig. 5: The 3-plane of $\{P_1, P_4, P_8\}$.

4) As shown in Fig. 6, by connecting the midpoints $S4_1$, $S4_2$, $S4_3$ and $S4_4$ of edge P_1P_5, P_2P_6, P_3P_7 and P_4P_8, respectively, we get a plane that separates the two point sets $\{P_1, P_2, P_3, P_4\}$ and $\{P_5, P_6, P_7, P_8\}$. This plane is called 4(1)-plane of $\{P_1, P_2, P_3, P_4\}$ or $\{P_5, P_6, P_7, P_8\}$. Other 4(1)-planes can be similarly

defined for any set of four vertices on a square face of the regular hexahedron.

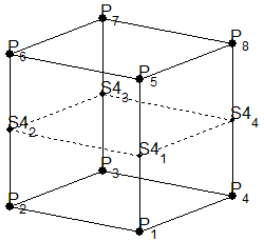


Fig. 6: The 4(1)-plane of $\{P_1, P_2, P_3, P_4\}$.

5) As shown in Fig. 7, by connecting the midpoints $S5_1$, $S5_2$, $S5_3$, $S5_4$, $S5_5$ and $S5_6$ of edges P_1P_2, P_2P_3, P_3P_7, P_7P_8, P_5P_8 and P_1P_5 respectively, we get a plane that separates the two vertex sets $\{P_1, P_3, P_4, P_8\}$ and $\{P_2, P_5, P_6, P_7\}$. This plane is called 4(2)-plane of $\{P_1, P_3, P_4, P_8\}$ or $\{P_2, P_5, P_6, P_7\}$. Other 4(2)-planes can be similarly defined for any set of four vertices that are on three edges sharing a common vertex.

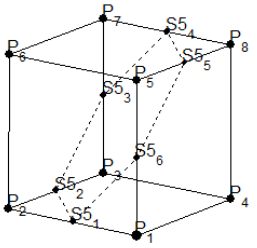


Fig. 7: The 4(2)-plane of $\{P_1, P_3, P_4, P_8\}$.

Next, let us list all the typical classification problems of the regular hexahedron that are linearly k-separable in One-for-Each approach with $4 \leq k \leq 8$. Here by "typical" we mean that any given classification problem of the regular hexahedron that are linearly k-separable in One-for-Each approach with $4 \leq k \leq 8$ is equivalent to one of the cases listed below in the following sense: By a suitable rotation round the center $(\frac{1}{2}, \frac{1}{2}, \frac{1}{2})$ of the regular hexahedron, the given classification problem is identical to one of the classification problem listed below. For instance, the classification problem to separate $B_1 = \{P_1, P_2\}$ from the other vertices is equivalent to the classification problem to separate $B_2 = \{P_3, P_4\}$ from the other vertices, by rotating the regular hexahedron in a 180^o angle around the axis that passes through the point $(\frac{1}{2}, \frac{1}{2}, \frac{1}{2})$ and is perpendicular to the plane $\{P_1, P_2, P_3, P_4\}$.

1) $k = 4$.
   a) $A_1 = \{P_1, P_2, P_3, P_4, P_5\}$, $A_2 = \{P_6\}$, $A_3 = \{P_7\}$, $A_4 = \{P_8\}$.
   b) $A_1 = \{P_1, P_2, P_3\}$, $A_2 = \{P_4, P_5, P_8\}$, $A_3 = \{P_6\}$, $A_4 = \{P_7\}$.
   c) $A_1 = \{P_1, P_2, P_3\}$, $A_2 = \{P_5, P_6, P_7\}$, $A_3 = \{P_4\}$, $A_4 = \{P_8\}$.
   d) $A_1 = \{P_1, P_2, P_3\}$, $A_2 = \{P_5, P_7, P_8\}$, $A_3 = \{P_4\}$, $A_4 = \{P_6\}$.
   e) $A_1 = \{P_1, P_2, P_3\}$, $A_2 = \{P_6, P_7, P_8\}$, $A_3 = \{P_4\}$, $A_4 = \{P_5\}$.
   f) $A_1 = \{P_1, P_2, P_3, P_4\}$, $A_2 = \{P_5, P_6\}$, $A_3 = \{P_7\}$, $A_4 = \{P_8\}$.
   g) $A_1 = \{P_1, P_2, P_3\}$, $A_2 = \{P_5, P_6\}$, $A_3 = \{P_7, P_8\}$, $A_4 = \{P_4\}$.
   h) $A_1 = \{P_1, P_2, P_3\}$, $A_2 = \{P_4, P_8\}$, $A_3 = \{P_6, P_7\}$, $A_4 = \{P_5\}$.
   i) $A_1 = \{P_1, P_2\}$, $A_2 = \{P_3, P_4\}$, $A_3 = \{P_5, P_6\}$, $A_4 = \{P_7, P_8\}$.
   j) $A_1 = \{P_1, P_2\}$, $A_2 = \{P_3, P_4\}$, $A_3 = \{P_5, P_8\}$, $A_4 = \{P_6, P_7\}$.
2) $k = 5$.
   a) $A_1 = \{P_1, P_2, P_3, P_4\}$, $A_2 = \{P_5\}$, $A_3 = \{P_6\}$, $A_4 = \{P_7\}$, $A_5 = \{P_8\}$.
   b) $A_1 = \{P_1, P_2, P_3\}$, $A_2 = \{P_4, P_8\}$, $A_3 = \{P_5\}$, $A_4 = \{P_6\}$, $A_5 = \{P_7\}$.
   c) $A_1 = \{P_1, P_2, P_3\}$, $A_2 = \{P_5, P_6\}$, $A_3 = \{P_4\}$, $A_4 = \{P_7\}$, $A_5 = \{P_8\}$.
   d) $A_1 = \{P_1, P_2, P_3\}$, $A_2 = \{P_5, P_8\}$, $A_3 = \{P_4\}$, $A_4 = \{P_6\}$, $A_5 = \{P_7\}$.
   e) $A_1 = \{P_1, P_2\}$, $A_2 = \{P_3, P_4\}$, $A_3 = \{P_5, P_6\}$, $A_4 = \{P_7\}$, $A_5 = \{P_8\}$.
   f) $A_1 = \{P_1, P_2\}$, $A_2 = \{P_3, P_4\}$, $A_3 = \{P_5, P_8\}$, $A_4 = \{P_6\}$, $A_5 = \{P_7\}$.
3) $k = 6$.
   a) $A_1 = \{P_1, P_2, P_3\}$, $A_2 = \{P_4\}$, $A_3 = \{P_5\}$, $A_4 = \{P_6\}$, $A_5 = \{P_7\}$, $A_6 = \{P_8\}$.
   b) $A_1 = \{P_1, P_2\}$, $A_2 = \{P_3, P_4\}$, $A_3 = \{P_5\}$, $A_4 = \{P_6\}$, $A_5 = \{P_7\}$, $A_6 = \{P_8\}$.
   c) $A_1 = \{P_1, P_2\}$, $A_2 = \{P_7, P_8\}$, $A_3 = \{P_3\}$, $A_4 = \{P_4\}$, $A_5 = \{P_5\}$, $A_6 = \{P_6\}$.
   d) $A_1 = \{P_1, P_2\}$, $A_2 = \{P_5, P_8\}$, $A_3 = \{P_3\}$, $A_4 = \{P_4\}$, $A_5 = \{P_6\}$, $A_6 = \{P_7\}$.
4) $k = 7$.
   a) $A_1 = \{P_1, P_2\}$, $A_2 = \{P_3\}$, $A_3 = \{P_4\}$, $A_4 = \{P_5\}$, $A_5 = \{P_6\}$, $A_6 = \{P_7\}$, $A_7 = \{P_8\}$.
5) $k = 8$.
   a) $A_1 = \{P_1\}$, $A_2 = \{P_2\}$, $A_3 = \{P_3\}$, $A_4 = \{P_4\}$, $A_5 = \{P_5\}$, $A_6 = \{P_6\}$, $A_7 = \{P_7\}$, $A_8 = \{P_8\}$.

In Table 2, it is shown how to solve these classification problems in binary approach by using the separating planes defined in the first part of the proof. For instance, in the case b of $k = 4$, we have a 4-separable classification problem in

Table 2: Separating planes in binary approach for classification problems

k	case	separating planes
4	a	2-plane of $\{P_6, P_7\}$, 2-plane of $\{P_7, P_8\}$
4	b	2-plane of $\{P_6, P_7\}$, 4(2)-plane of $\{P_1, P_2, P_3, P_6\}$
4	c	2-plane of $\{P_4, P_8\}$, 4(1)-plane of $\{P_1, P_2, P_3, P_4\}$
4	d	4(1)-plane of $\{P_1, P_2, P_3, P_4\}$, 4(2)-plane of $\{P_1, P_2, P_3, P_6\}$
4	e	4(1)-plane of $\{P_1, P_2, P_3, P_4\}$, 4(2)-plane of $\{P_1, P_2, P_3, P_5\}$
4	f	3-plane of $\{P_5, P_6, P_7\}$, 3-plane of $\{P_5, P_6, P_8\}$
4	g	3-plane of $\{P_4, P_7, P_8\}$, 4(1)-plane of $\{P_1, P_2, P_3, P_4\}$
4	h	3-plane of $\{P_5, P_6, P_7\}$, 3-plane of $\{P_4, P_5, P_8\}$
4	i	4(1)-plane of $\{P_1, P_2, P_3, P_4\}$, 4(1)-plane of $\{P_1, P_2, P_5, P_6\}$
4	j	2-plane of $\{P_1, P_2\}$, 2-plane of $\{P_3, P_4\}$, 2-plane of $\{P_5, P_8\}$
5	a	2-plane of $\{P_5, P_6\}$, 2-plane of $\{P_5, P_8\}$, 2-plane of $\{P_6, P_7\}$
5	b	3-plane of $\{P_5, P_6, P_7\}$, 3-plane of $\{P_4, P_5, P_8\}$, 3-plane of $\{P_4, P_7, P_8\}$
5	c	3-plane of $\{P_4, P_7, P_8\}$, 3-plane of $\{P_5, P_6, P_7\}$, 3-plane of $\{P_5, P_6, P_8\}$
5	d	3-plane of $\{P_4, P_5, P_8\}$, 3-plane of $\{P_5, P_6, P_8\}$, 3-plane of $\{P_5, P_7, P_8\}$
5	e	4(1)-plane of $\{P_1, P_2, P_3, P_4\}$, 4(1)-plane of $\{P_1, P_2, P_5, P_6\}$, 1-plane of $\{P_7\}$
5	f	3-plane of $\{P_3, P_4, P_7\}$, 3-plane of $\{P_5, P_6, P_8\}$, 3-plane of $\{P_5, P_7, P_8\}$
6	a	2-plane of $\{P_7, P_8\}$, 3-plane of $\{P_4, P_5, P_8\}$, 4(1)-plane of $\{P_1, P_2, P_3, P_4\}$
6	b	2-plane of $\{P_6, P_7\}$, 4(1)-plane of $\{P_1, P_2, P_3, P_4\}$, 4(1)-plane of $\{P_1, P_2, P_5, P_6\}$
6	c	4(1)-plane of $\{P_1, P_2, P_3, P_4\}$, 4(1)-plane of $\{P_1, P_2, P_5, P_6\}$, 4(2)-plane of $\{P_4, P_5, P_7, P_8\}$
6	d	2-plane of $\{P_3, P_7\}$, 3-plane of $\{P_1, P_2, P_6\}$, 4(1)-plane of $\{P_1, P_2, P_3, P_4\}$
7	a	4(1)-plane of $\{P_1, P_2, P_3, P_4\}$, 4(1)-plane of $\{P_1, P_2, P_5, P_6\}$, 3-plane of $\{P_3, P_6, P_7\}$
8	a	4(1)-plane of $\{P_1, P_2, P_3, P_4\}$, 4(1)-plane of $\{P_1, P_2, P_5, P_6\}$, 4(1)-plane of $\{P_1, P_4, P_5, P_8\}$

One-for-Each approach with the four classes $A_1 = \{P_1, P_2, P_3\}$, $A_2 = \{P_4, P_5, P_8\}$, $A_3 = \{P_6\}$ and $A_4 = \{P_7\}$. As shown in Table 2, these four classes can also be separated in binary approach by a 2-plane of $\{P_6, P_7\}$ and a 4(2)-plane of $\{P_1, P_2, P_3, P_6\}$ (cf. Fig. 8). Table 1 in *Theorem* 1 is a direct consequence of Table 2.

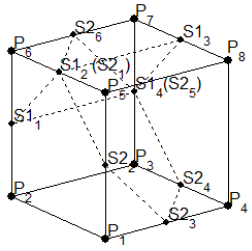


Fig. 8: Case b of k=4 and it's classification planes in binary approach.

4.2 Proof of Theorem 2

Suppose that a linearly k-separable classification problem is solved in One-for-Each approach by an MLP-3 with q input nodes, three hidden nodes and k output nodes. Then, we can construct a new MLP-3 to solve the same problem in binary approach such that the number of output nodes is reduced to m given in Table 1 of *Theorem* 1. To this end, the weights and biases connecting the input and hidden nodes remain unchanged. Denote the set of input samples of the classification problem by B. The hidden layer of the MLP-3 maps B into a point set $A \subset U^3$. By assumption, A is linearly k-separable in One-for-Each approach. Thanks to *Theorem* 1, we know that A is also linearly k-m-separable in binary approach with the m separating planes specified in Table 2. Let the m planes be expressed in a standard form as:

$$v_{i1}y_1 + v_{i2}y_2 + v_{i3}y_3 + v_{i0} = 0,\ i = 1, 2, \cdots, m \qquad (9)$$

where $y = (y_1, y_2, y_3)$ denotes arbitrary points on the plane, and v_{i1}, v_{i2}, v_{i3} and v_{i0} are fixed real numbers. Then, we can finish the construction of the new MLP-3 by leaving the input-hidden weights w_i's and biases b_i's (see (8)) unchanged and setting the hidden-output weights and biases (see (7)) as

$$v_i = (v_{i1}, v_{i2}, v_{i3}),\ d_i = -v_{i0},\ i = 1, 2, \cdots, m \qquad (10)$$

This completes the proof.

5. Conclusions

In this paper, we propose a binary output method instead of conventional "One-for-Each" approach, which can reduce the number of the output nodes for multiple linear perceptron, resulting in a compact neural network. For ordinary

k-separable classification problem ($2 \leq k \leq 8$) in MLP, we have shown the equivalence property by using binary method with k-m-separable conception, where $m \leq \lceil \log_2 k \rceil$. The further study for general feedforward neural networks is still in process.

Acknowledgements

This work is supported by Innovation Training Program for Chinese Undergraduate Students, grant 201210141098, the Fundamental Research Funds for the Central Universities, grant 11171367, and the National Natural Science Foundation of China, grant 11201051. The authors also thank the anonymous reviewers for their valuable comments on the paper.

References

[1] Simon Haykin, *Neural Networks and Learning Machines*, 3rd ed.,New Jersey: Prentice Hall, 2009.

[2] G. Sateesh Babu, S. Suresh, "Meta-cognitive NeuralNetwork for classification problems in a sequential learning framework," *Neurocomputing*, vol. 81, pp. 86–96, Apr. 2012.

[3] Anas Qteishat, Chee Peng Lim, Jeffrey Tweedale and Lakhmi C. Jain, "A neural network-based multi-agent classifier system," *Neurocomputing*, vol. 72, pp. 1639–1647, Mar. 2009.

[4] Bogdan Gabrys, Andrzej Bargiela, "General Fuzzy Min-Max Neural Network for Clustering and Classification," *IEEE Trans Neural Netw*, vol. 11(3), pp. 769–783, May. 2000.

[5] Chris M. Bishop, "Neural networks and their applications," *Review of scientific instruments*, vol. 65(6), pp. 1803–1832, Jun. 1994.

[6] Guobin Ou, Yi Lu Murphey, "Multi-class pattern classification using neural networks," *Pattern Recognition*, vol. 40, pp. 4–18, 2006.

[7] Shigetoshi Shiotani, Toshio Fukuda and Takanori Shibata, "A neural network architecture for incremental learning," *Neurocomputing*, vol. 9, pp. 111–130, 1995.

[8] R. Reed, "Pruning algorithms–a survey," *IEEE Trans Neural Netw*, vol. 8, pp. 185-204, 1997.

[9] R. Setiono, "A penalty-function approach for pruning feedforward neural networks," *Neural Networks*, vol. 9, pp. 185–204, 1997.

[10] Hongmei Shao, Wei Wu and Lijun Liu, "Convergnece of online gradient method with a penalty for BP neural networks," *Communications in Mathematical Research*, vol. 26(1), pp. 67–75, 2010.

Underwater Acoustic Monitoring Using MFCC for Fuzzy C-Means Clustering, Naive-Bayes and Hidden Markov Model-Based Classifiers

Cristian Alzate[1], Samira Ortiz[1] and Vidya Manian[1]
[1]Department of Electrical & Computer Engineering, University of Puerto Rico at Mayaguez
Mayaguez, Puerto Rico

Abstract - *The whale sounds help researchers in population assessments and to follow the migratory path of whales. Acoustics is the best way to study and observe cetaceans since it is automatic and non-invasive. A technique capable of differentiating between whale songs, other marine sounds and man-made sounds would be very useful for the scientific community. This paper presents a system that can classify signals collected from humpback whales vocalization data through acoustic monitoring from Puerto Rico coastal marine habitats. The fuzzy c-means algorithm is used to cluster, in an unsupervised manner, sounds from the same marine source. The Naive-Bayes classifier and a Hidden Markov Model are used to classify marine sounds such as whale songs, man-made sounds and natural background noise from hydrophone recordings under water. The MFCC (Mel-Frequency Cepstral Coefficients) approach is used to extract the most important characteristic of the signals and a vector quantization and clustering approach is used to obtain the states for training the HMM. Satisfactory results are obtained through HMM classifier which outperforms the Naive-Bayes classifier for whale songs and marine sounds.*

Keywords: Classifier, Clustering, Hidden Markov Model, Humpback whales, Naïve-Bayes

1 Introduction

A humpback whale song consists of units of sound which combine to form what is known as a phrase. An unbroken sequence of similar phrases is a theme. Themes are sung in a specific order; and several distinct themes combine to form a song. Once completed, the whale will return to the beginning and start again. Whales will sing for hours or days at a time. Whales in different areas sing in different dialects. And, the songs change gradually over time. [1][2]

Although there is not a sure explanation about the purpose of the songs, they allow researchers to follow the migratory path of whales throughout this season and also to estimate population density within an area. The whale watching season in Puerto Rico is between December and March and reaches its peak in February. The rare and impressive Humpback whales visit Puerto Rico every year on their migratory route to the North Atlantic where they mate during the summer months. Hence, in an effort to monitor marine ecological habitats in the coasts of the island of Puerto Rico and gain new knowledge and further the understanding of harbor acoustic monitoring, we set out to implement different techniques to extract songs features, cluster whales individuals based on their singing patterns and classify underwater sounds to differentiate whale songs from other marine sounds, background and man-made sounds.

With growing concern about environmental problems, cetaceans and their habitats are receiving significant attention and studies have been developed for location and cetacean behavior analysis [3] [4]. However there is not much work in classification of acoustic signals from marine habitats especially those around Puerto Rico. There has been some analysis of West Indian mammals' vocalization [5] [6] and marine mammal behavioral response in southern California [7]. This paper presents the results of clustering and classification of signals recorded from acoustic monitoring of humpback whales in Rincon and Boqueron bay area in Puerto Rico. Due to highly song nature and underwater channel distortions, it has been a challenge the task of classification.

The acoustic signals considered in this work are whale songs, background noise, swimming noise, kayak, boat and jestki sounds. We test different techniques to achieve the best results in classification and make a remarkable contribution in the study of cetaceans. A clustering method and two different classification methods are implemented; for clustering the fuzzy c-means algorithm is used and for classification, the Naïve-Bayes classifier and Hidden Markov Model. The feature extraction method is the Mel-Frequency Cepstral Coefficients (MFCC). The paper is organized as follows. In section two, we first explain data acquisition and preprocessing methods, later we present the feature extraction, data segmentation to later explain the clustering and classifying methods. Section three presents the experimental results and conclusions are shown in section 4.

2 Methodology

2.1 Data Acquisition and preprocessing

Sound under water has important characteristics; it travels rapidly over long distances due to the fact that seawater is approximately 800 times denser than air [8]. The attenuation in sounds is related to frequency. High frequencies are attenuated faster than low frequencies and, given the same

environmental conditions; the louder the sound the further it can travel.

The active vocalization range of humpback whales has been estimated to be 0.02-8.2 kHz at 15 to 160km. Hydrophone recordings of whale songs have been used in this work. Since the highest frequency component of humpback whales is 8.2 kHz, the sampling rate of the hydrophone was set to be at least 22 kHz.

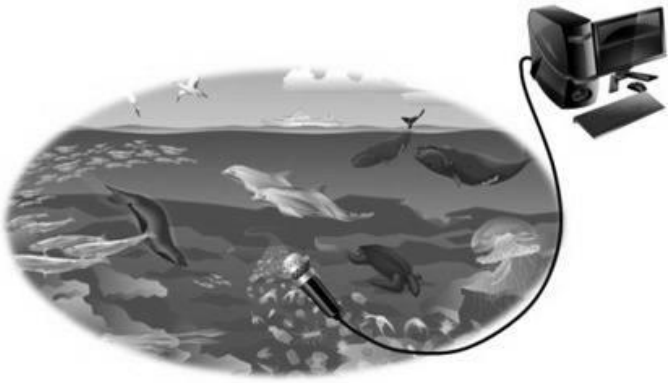
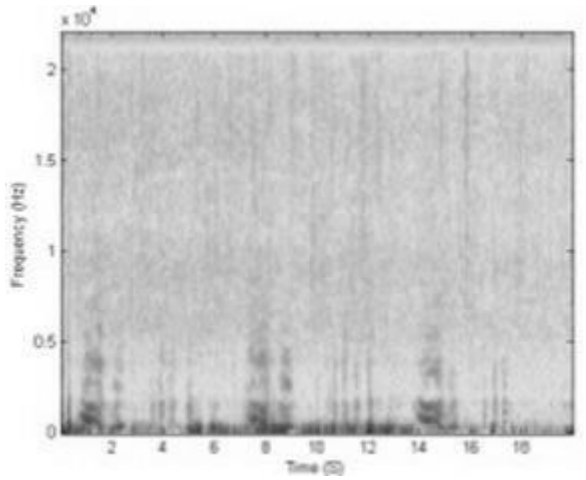


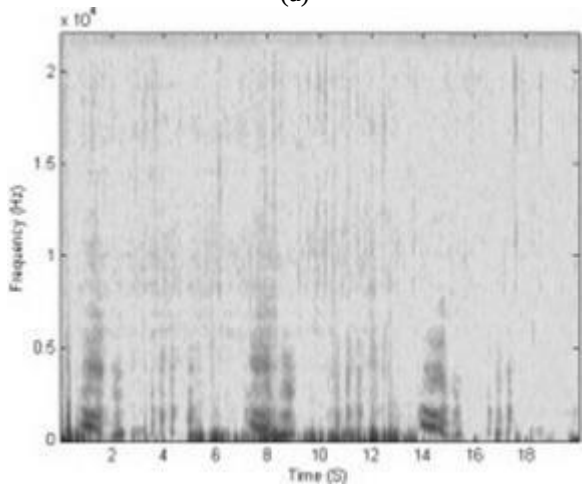
Fig. 1. Marine acoustic data acquisition scheme

Once the sounds are recorded, the next step is the signal preprocessing, at this stage, it is extremely important to reduce the background noise so that the sounds can be processed optimally enhancing features and improving classification. In the case of recording marine species like whales, multiple sources of noise can be identified since the underwater sounds is generated by a variety of natural sources such as rain, marine life and breaking waves. It is also generated by a variety of man-made sources, such as military sonars, ships, boats, fishing and so on. The frequency range of ambient noise underwater is 50- 100 kHz. This noise is due to spray and bubbles associated with breaking waves [9] [10]. It increases with increasing wind speed.

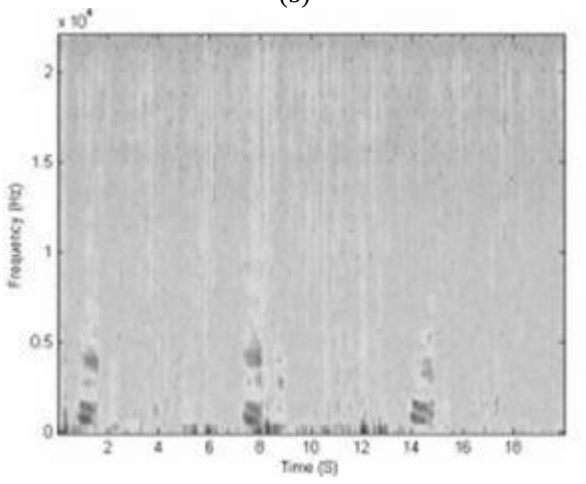
The vocalizations emitted by humpback whales can be defined as broad-band transient signals. Transient signals can be defined as having a short duration and finite energy, whereas their power is zero. For this reason, the correlation function in the analysis of these kinds of sounds is determined using their energy [11] Fig. 2(a) shows the spectrogram (Short Time Fourier Transform) of an original whale sound, the spectrogram after removal of noise due to ocean waves (Fig. 2(b)) and after complete noise removal (Fig. 2 (c)). From the spectrogram, we can see that the principal frequency components are between 0 and 5 KHz. approximately. The noise from the breaking waves was reduced by creating a sound profile of the source and designing a filter to remove it.



(a)



(b)



(c)

Fig. 2. (a) Spectrogram of whale sound, (b) Spectrogram without noise from waves, (c) Spectrogram with full noise duction.

2.2 Feature Extraction using Mel-Frequency Cepstral Coefficients (MFCC)

The Mel-Frequency Cepstral Coefficients is a good parametric representation of acoustic signals. The coefficients are a result of the cosine transform of the real logarithm of the short-term energy spectrum expressed on a Mel-frequency scale. The MFCC calculated for three samples from a whale audio source is shown in Fig. 3. It can be seen that the coefficients are similar as it is for the same individual whale.

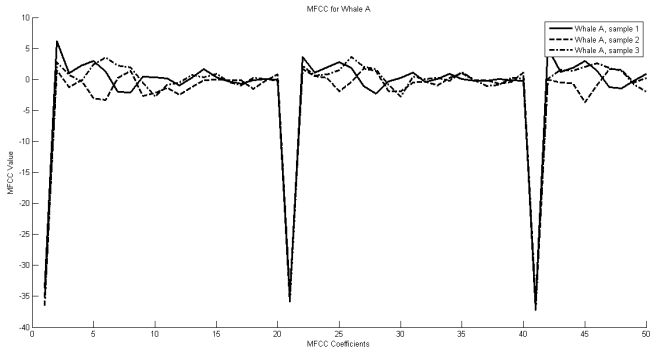


Fig. 3. MFCC applied to different samples of the same whale.

2.3 Fuzzy C-means (FCM)

FCM is a data clustering technique where each data point belongs to a cluster to some degree that is specified by a membership grade the cluster centers, which are updated iteratively. It was proposed by Dunn [12] in 1973 and eventually modified by Bezdek [13] in 1981.

Let x_i represent the N data points of M-dimension to be clustered, where$i = 1,2,\dots,N$. And C the number of clusters to be made. The level of cluster fuzziness f higher than one is chosen and the membership matrix U of size $N \times C \times M$ is initialized at random, such $U_{ijm} \in [0,1]$ and $\sum_{j=1}^{C} U_{ijm} = 1.0$, for each $i$and a fixed value of m. The cluster centers CC_{jm} forj^{th} cluster and its m^{th}dimension is determined by using the expression [14]

$$CC_{jm} = \frac{\sum_{i=1}^{N} U_{ijm}^{f} x_{im}}{\sum_{i=1}^{N} U_{ijm}^{f}} \quad (1)$$

Then the Euclidean distance between i^{th}data point and j^{th} cluster center with respect to m^{th} dimension is defined by:

$$D_{ijm} = \left\| \left(x_{im} - CC_{jm} \right) \right\| \quad (2)$$

Then, the fuzzy membership matrix U is updated according to D_{ijm}. While $D_{ijm} > 0$

$$U_{ijm} = \frac{1}{\sum_{j=1}^{C} \left(\frac{D_{ijm}}{D_{icm}} \right)^{\frac{2}{f-1}}} \quad (3)$$

When$\mathrm{D_{ijm}} = 0$, $\mathrm{CC_{jm}}$ will have the full membership $\mathrm{U_{ijm}} = 0$. This procedure is done until$\mathrm{U} < \in$, where $\in$ is a pre-specified termination criterion [14].

2.4 Classification

2.4.1 Naive-Bayes Classifier

The Naive Bayes classifier is a classification method based on Bayes Theorem [15]. C_j denotes the class of vector X as belonging to the j-th class, $j = 1,2 \dots$ J out of J possible classes. Let $P\left(C_j \middle| X_1, X_2, \dots, X_P \right)$ denote the (posterior) probability of the sample vector belonging in the j-th class given the individual characteristics $X_1, X_2, \dots, X_P$. Furthermore, let $P\left(X_1, X_2, \dots, X_P \middle| C_j\right)$ denote the probability of a sample with individual characteristics $X_1, X_2, \dots, X_P$ belonging to the j-th class and $P(C_j)$ denote the unconditional (i.e. without regard to individual characteristics) prior probability of belonging to the j-th class. For a total of J classes, Bayes theorem gives us the following probability rule for calculating the case-specific probability of a sample vector falling in the j-th class:

$$\mathrm{P}\left(\mathrm{C_j} \middle| \mathrm{X_1}, \mathrm{X_2}, \dots, \mathrm{X_P} \right) = \frac{\mathrm{P(X_1, X_2, \dots, X_P | C_j) \cdot P(C_j)}}{\mathrm{Denom}} \quad (4)$$

Where:

$$\mathrm{Denom} = \mathrm{P(X_1, X_2, \dots, X_P | C_1) P(C_1)} + \cdots + \mathrm{P\left(X_1, X_2, \dots, X_P \middle| C_J\right) P\left(C_J\right)}. \quad (5)$$

All the MFCC of each individual is saved in a data structure named "Codebook". Each MFCC has a dimension of 20x16. The Naive-Bayes algorithm is trained with the entire codebook.

2.4.2 Hidden Markov model

Hidden Markov Model is used for modeling stochastic processes and sequences in various applications, like natural language modeling, handwriting recognition and voice signal processing [16]. Also it is used to solve problem that has states at time t that are influenced directly by a state at time (t-1). Hidden Markov Model has a number of parameters, whose values are set so as to characterize well the training patterns for the known category. Later, a test pattern is classified by the model that has the highest posterior probability. For n events A1 ,… An, the chain rule of probability [17] is given by:

$$P(A_1 \cap A_2 \ldots \cap A_n) = P(A_1) \prod_{i=2}^{n} P(A_i|A^{i-1}) \quad (6)$$

Where,

$$A^{i-1} \triangleq A_1 \cap A_2 \ldots \cap A_{i-1} \quad (7)$$

If the event A_i pertain to the event $\{X_i = a_i\}$ then,

$$P(X^n = a^n) = P(X_1 = a_1) \prod_{i=2}^{n} P(X_i = a_i|X^{i-1} = a^{i-1}) \quad (8)$$

The formula above show the causal decomposition of the joint distribution of x^n into a product of conditional distributions of the present value of X given the past values. X^n is a Markov Process if the future value X^{i+1} is independent of the past value Xi-1 given the present X_i :

$$P(X_i = a_i|X^{i-1} = a^{i-1}) = P(X_i = a_i|X_{i-1} = a_{i-1}) \quad (9)$$

3 Results

The algorithms were trained using two samples of each of the acoustic sources and were tested using 15 to 20 other audio samples that were not used for training. Audio files are 5 seconds length. They belong to different acoustic sources of whales, jet skis, kayaks, human voices and boats. Features were extracted from the MFCC coefficients of the data. One set of coefficients were calculated for each audio sample.

For clustering, FCM is determining the cluster centers, calculating the Euclidean distances and updating the fuzzy membership matrix until the centroid distances are less than a threshold, which is the minimum distance between centroids and our termination criterion. Once this threshold is computed with just the two samples of each of the acoustic sources, the algorithm is implemented in the testing data set. Fig 6 shows the clustering results after the FCM is implemented to group these sounds, the algorithm precisely detects the number of sources.

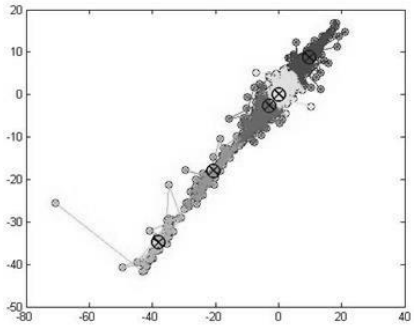
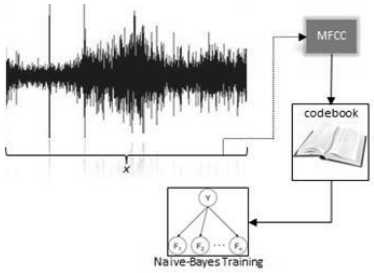


Fig. 4. Clustering using FCM algorithm

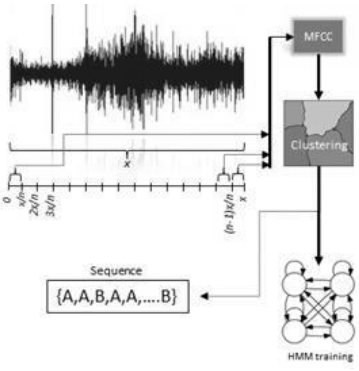
For the naive-Bayes classifier, after one set of features extracted from the MFCC coefficients of the data, two samples of each audio file were stored in a codebook to train the network and the others coefficients were used to test the classifier. The algorithm for naive Bayes classifier is described in Fig. 7 (a).

Tables 1 and 2 show the results for this classifier. In the case of whale songs the accuracy of the algorithm was 72.7%, which corresponds to 30 errors in 110 trials. For the marine sounds, the accuracy was 87.6% which corresponds to 13 errors from 105 trials.

For the classifier based on Markov Model, a segmentation process is used to create a sequence for each sound sample. Each file was segmented into 64 parts, and then the MFCC is calculated for each part and stored into a data structure called codebook. Then, a clustering algorithm based on the Euclidean distance between the codebooks and the coefficients is applied to find similar coefficients and create groups which will represent the hidden states of the HMM. After running the clustering algorithm, a hidden state is assigned to each segment, which results in a sequence for each audio file. Finally, to create the training data set, all the sequences for the same sound source are grouped. Fig. 7 (b) shows the general algorithm. The classification results are shown in Tables 1 and 2 along with the results of Bayes classification for comparison. This classifier results in 95.2% accuracy for classification of whale songs and 95.7 for the other unknown audio files



(a)



(b)

Fig. 5. Algorithm for Naive Bayes (a) and HMM classifier (b).

The HMM is more accurate for whale songs and marine sounds classification. The disadvantage of this classifier is the training time; the segmentation and the creation of sequences are time consuming steps. Table 3 shows the computational run time for both classifier algorithms using Matlab in a computer whose technical specifications are: 24 GB of RAM memory and 2 Intel Xeon processors with a 2.67 GHz clock each. The naive-Bayes classifier is less expensive in computational time compared to HMM classifier, and

depends on the number of sound sources. The computational time for classification of marine sounds (6 sources) is greater than that of whale songs (5 sources). The HMM classification timing cost depends only on the number of files, hence the computational time for classifying whale songs (110 files) is greater than that of marine sounds (105 files).

Table 1. Overall accuracy (OA) classification for Whale songs

Whales	Naive-Bayes (%)	HMM (%)
A	66.6	89
B	50	95
C	50	97.5
D	100	99.5
E	97	95
OA	72.7	95.2

Table. 2. Overall accuracy (OA) classifications for Marine Sounds.

Marine Sound	Naive-Bayes (%)	HMM (%)
Boat	1	88
Jestki	66.7	92.3
Kayak	100	100
Noise	100	99
Swimming	100	100
Voice	67	95
OA	87.5	95.7

Table. 3. Timing results for the Classifiers.

Sound	Time (in seconds)	
	Naive-Bayes	HMM
Whale songs	2.42	79.42
Marine	4.61	41.4

4 Conclusions

In this paper, we implemented a series of algorithms to study and analyze the humpback whales behavior through their songs. The series of algorithms implemented for noise removal, feature extraction, clustering and classification allowed successfully monitoring marine life in the island of Puerto Rico and other coastal areas. Classifiers allowed identifying songs from the same whale, showing an important fact about whales; each one of them has an individual signature. Besides, different sound sources could be successfully clustered and classified.

The feature extraction using Mel-Frequency Cepstral Coefficients proved to be effective for whale songs and marine sounds. By using the MFCC for fuzzy c-means clustering, the number of acoustic sources could be satisfactory obtained. HMM classifier performed better than the naive Bayes classifier in spite of the complexity involved in finding the sequences and states. The HMM classifier performs well once a suitable segmentation of the sequences is obtained and state transitions are set up correctly. Training the naive-Bayes Classifier is comparatively easier and is less complex compared to the HMM in time and memory, it works well with marine sounds and gives an accuracy of 87.5% for the whale songs, hence the techniques implemented can be used for whales and other marine mammal surveys and census.

Acknowledgment

We would like to thank Ms. Mithriel M. MacKay, Director of Research and Education Marine and Coastal Ecology Research Center San German, Puerto Rico, and Mr. Jose Rivera, National Marine Fisheries Service, NOAA for providing us the humpback whale songs We would also like to thank Mr. Samuel Rosario for his technical support in the data collection and the Laboratory for Applied Remote Sensing and Image Processing, University of Puerto Rico, Mayaguez for providing the software platforms for data processing. Finally, we would like to say that the views presented in this paper are the opinions of the authors and do not represent any official position of any of the US federal agencies that partially supported this work.
This work was supported by DHS grant CIMES 2008-ST-061ML0002.

5 References

[1] Songs of Humpback Whales, roger S. Payne and Scott McAvay 13 August 1971, Volume 173, pp 587-597

[2] Clapham, P. J. (2000). The humpback whale. Cetacean Societies, field studies of dolphins and whales. Chicago: The University of Chicago, 173-196.

[3] Mazhar, S.; Ura, T.; Bahl, R., "Effect of Temporal Evolution of Songs on Cepstrum-based Voice Signature in Humpback Whales," OCEANS 2008 - MTS/IEEE Kobe Techno-Ocean , vol.,no., pp.1,8, 8-11 April 2008

[4] Mazhar, S.; Ura, T.; Bahl, R., "Vocalization based Individual Classification of Humpback Whales using Support Vector Machine," OCEANS 2007 , vol., no., pp.1,9, Sept. 29 2007-Oct. 4 2007

[5] Miksis-Olds, P. L. tyack, "Manatee vocalization usage in relation to environmental noise levels", J. Acoust. Soc. Am., 125(3), pp 1806-1815, March 2009.

[6] J.E Reynolds, S.A. Rommel, "Biology of Marine Mammals". Ch 7. 1999, Washington, DC: Smithsonians Institution Press

[7] L. S Brandon, "Marine mammal behavioral response studies in southern california: Advances in

technology and experimental methods", Marine technology society journal, 46(4), pp 48-56, July/August 2012.

[8] M. Berke Gur, C. Niezrecki, "A wavelet packet adaptive filtering algorithm for enhancing manatee vocalizations" J. Acoust. Soc. Am., 129(4), pp 2059-67, April 2011.

[9] G. M Wenz, 1962, "Acoustic ambient noise in the ocean: Spectra and sources". Jornal of the Acoustical Society of America 34: 1936-1956.

[10] T. G Leighton, "Fundamentals of underwater acoustics, in Fundamentals of noise and vibration", F. Fahy and J. Walker, Editors. 1998, Spon Press: London. 9. 373-444.

[11] W.W.L. Au, "Acoustic properties of humpback whale songs". The Journal of the Acoustical Society of America, 2006. 120(2): p. 1103-1110.

[12] Dunn, J. C.: A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters. J. Cybernet, Vol. 3, 1973, pp. 32–57.

[13] Bezdek, J. C.: Pattern Recognition with Fuzzy Objective Function Algorithms. Kluwer Academic Publishers, Norwell, MA, USA, 1981.

[14] Chattopadhyay, S., Pratihar, D. K., & De Sarkar, S. C. (2011). A Comparative Study of Fuzzy C-Means Algorithm and Entropy-Based Fuzzy Clustering Algorithms. Computing & Informatics, 30(4).

[15] Fomby Thom, "Naïve-Bayes Classifier", Southern Methodist University, Texas, pp 1-3, April 2008.

[16] R. Meital, "Information Bottleneck and HMM for speech recognition", School of Computer Science and Engineering, The Hebrew University of Jerusalem. Ch 3, 2006.

[17] R. Duda, P. Hart, D. Stork. "Pattern Classification". John Wiley & sons, inc. 2001, pp 128-133

The Crowdsourcing Linguistic Technology for Experts Assessment

V. Protasov[1], **M. Charnine**[2], **and E. Melnikov**[1]
[1]The Russian Center of Computing for Physics and Technology, Moscow, Russia
[2]Institute for Informatics Problems, Russian Academy of Sciences, Moscow, Russia

Abstract - *Aspects of test materials formation, experts' competency assessment and test questions complexity determination are introduced, using a crowdsourcing technology, Rasch model and adaptive sampling methodology for selection of test questions. Shown how iterative method for simultaneous determination of experts' competency level and complexity of test questions can solve the problem of metrology testing - a clear and objective measurement of these quantities. Because the technology is simple and low-cost compared to existing methods, the community of experts by using crowdsourcing and proposed methods can independently develop test materials and carry out self-certification of their community network. After that certified experts can participate in various individual and collective projects with predictable results. The usage of crowdsourcing technology for solving linguistic problems such as selecting definitions, evaluating of related keywords, and writing of encyclopedia articles are discussed*.

Keywords: crowdsourcing, Rasch model, competence measurement, question complexity, task complexity, tests certification.

1 Introduction

At present, there is an exponential growth in the number of scientific publications about collective intelligence. The invention of social WEB-2 and social computing has led to the creation of new networking tools, such as crowdsourcing. Application of crowdsourcing involves delegating business objectives to the company organizing the remote network community. This way of organizing labor has significant advantages over traditional recruitment, because it allows you to quickly create a global product using cheap remote workforce. At the same time, business risks are shared with the performers, which in most cases are paid after the sale of their work product.

Initially the term «crowdsourcing» was used in 2006 by journalist Jeff Howe [1]. In Russia, this technology has been actively developing by company Witology that has completed several significant projects for the Russian economy [2].

It is very promising to use crowdsourcing for solving linguistic problems, such as choice of the best definition and main category of some concept, selection of related keywords, writing of encyclopedia articles [8]. There are millions different concepts in natural language, so only very large groups of people using crowdsourcing can solve such complex linguistic tasks.

One of the main challenges of crowdsourcing is the unpredictability of the results related to the fact that Condorcet's theorem [3, 7], which is the basis of this method requires that the probability of a correct conclusion of the expert exceeded 0.5 and reliability of the method requires a large number of pre-tested experts. This problem is exacerbated by the fact that in actual practice different tasks included in the project may have different complexity, and it should be taken into account in the preparation, implementation and forecasting results of the project.

2 The Rasch Model

The correlation between test questions complexity and experts' competence in evaluation of correct answer probability was set up in generic theory of test construction – Item Response Theory (IRT) [4]. For our objectives, the Rasch's single-parametric model [5] is the most appropriate, because it is the most simple formula that relates the correct answer obtainment probability G of the testee, his competence θ and task complexity β:

$$G = \frac{1}{1 + e^{\alpha(\theta - \beta)}} \qquad (1)$$

where α is a scale factor. This expression defines the logistic curve. Tasks complexity and experts competence is measured in special units – the logits.

Fig. 1 illustrates the Rasch curves for α = 1, with task complexity β from -3 (the easiest) to 3 logits (the hardest).

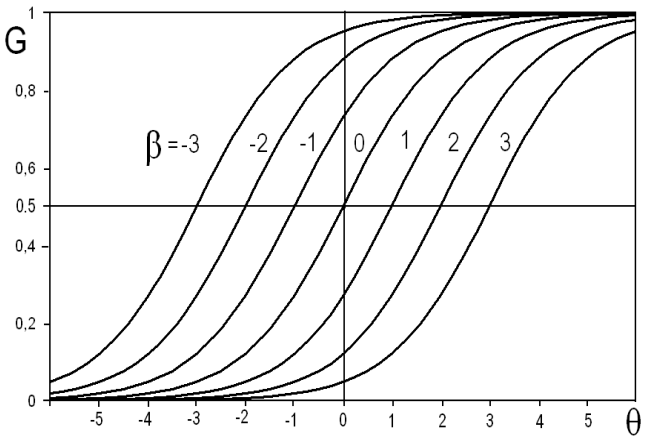


Fig. 1. The dependence of correct answer probability of the testee's on the task complexity and competence.

Usage of the Rasch model guarantees independence of tasks estimation on the testees and also testees estimation on the tasks. The provided dependencies illustrate that the greater competency θ of the testee, the higher probability of success in a task. When θ = β, the probability of correct answer equals 0.5.

3 Experts' competence measurement using small set of test questions

The availability of test base of huge number of various level test questions is the necessary condition for the competence measurement, independent on task complexity. As evaluations show, according to the Law of Large Numbers and results of computer modeling, presented below, for covering the interval of competence from -3 to 3 logits with accuracy ±0.1, it's necessary to prepare a test base over 1000 questions of difficulty from -4 to 4 logits. A methodological difficulty appears here. Commonly, existing sets of questions don't contain such number of questions and, that is more important, they are designed for rather homogeneous group of experts. In another hand, it's hard to imagine an expert, solving a test of hundreds of questions. If we have a large set of questions with known values of β, sorted in descending order of difficulty, then for measurement of expert competence it's able to develop further adaptive procedure, reducing number of questions to 50 – 100.

Let us consider a set of 1000 questions, sorted by difficulty in descending order. Search interval K is set up to 500. After each answer the value halves until it reaches 1. The expert is asked the question number N = K. If the answer is correct, then the next N equals N – K, else N + K (this means that the procedure automatically finds the direction where the experts' competence area is located). The selected question is removed from the list and is not asked again. If the number of the question appears again then the next question in the same direction is picked. When the list acquires a solid sublist of asked questions of more than 50 questions with correct answers portion value 0.5±0.01, the measurement process is finished. The weighted average is taken as the value of competence:

$$\theta = \frac{1}{i_2 - i_1 + 1} \sum_{i=i_1}^{i_2} \delta_i \beta_i \qquad (2)$$

where $\delta_i = 0$, if the answer i is incorrect, and $\delta_i = 1$, if the answer is correct, i_1 and i_2 are the borders of the solid sublist.

4 Application of crowdsourcing in test question base compilation

It's well known that creation of huge test question list is a very laborious procedure. A crowdsourcing, performed among experts of the knowledge area, where the net intelligence of tested experts is supposed to be used further, can be very useful here.

Experts are divided into groups, each group consists of approximately 7 participants, and each group prepares test questions with answers using the technology of Evolutional Decisions Reconciliation method [6]. It's supposed that each participant generates more than 2 such questions with answers, and they are checked, appended or rejected by collective mind of the group. At the end of iterative process 10 best questions remain. Meanwhile the expert group defines approximate questions complexity in logits. If such crowdsourcing organization includes 100 groups then we can obtain 1000 questions of different level with correct answers in the output.

As long as questions complexities were estimated approximately, it's unable to use them in experts' competence measurement. These estimations can be used only in adaptation procedure of experts testing for reducing the number of questions proposed to each expert. In the process of correct answers statistics accumulation it's also necessary to reorder the question list by difficulty to make the ranking more accurate.

5 Experts and tests certification with the help of crowdsourcing

Having a wide base of test questions of various levels it's able to produce the following procedure of the absolute experts' competences estimation and accurate tasks level values obtaining. According the adaptation methodic, described above, each expert is proposed with 50 questions

from the test base. The expert is asked the questions that he has not participated in and, hence he doesn't know the correct answers. His correct and incorrect answers are recorded. During the accumulation of correct and incorrect answers complex of the entire group, the difficulty order of the questions is specified. For this purpose, the question list is periodically sorted by number of correct answers. After testing process is finished, when each of 700 experts has answered his 50 questions, the table $T_{j,i}$ is obtained, where *j* is the expert index, and *i* is the index of test question. A cell of the $T_{j,i}$ table equals one if the expert *j* has answered the question *i* correctly, and equals zero, otherwise. If the question was not offered to the expert then the cell value is dash.

Let us consider a method that allows to define exactly complexities of questions and competences of experts during the table processing. Let us note that we already have test questions complexities in the first approximation, they were defined by the collective intelligence of the experts at the stage of test base assembling.

According to Rasch [4], we evaluate the sums of correct values of all experts S_i in each column *i* and write it in the table bottom, in each *j*-th row we evaluate a sum of correct answers Q_j and write it in the right side of the table:

Table 1

$T_{j,i}$	1	2	3	4	…	*m*	
1	0	1	-	0	…	-	Q_1
2	1	-	0	1	…	0	Q_2
3	0	-	0	0	…	0	Q_3
4	-	0	1	-	…	1	Q_4
…	…	…	…	…	…	…	…
n	1	0	0	1	…	0	Q_n
	S_1	S_2	S_3	S_4	…	S_m	

So, the Table 1 contains the results of testing of *n* experts with *m* test questions.

Then we perform double sorting of the table $T_{j,i}$: by Q_j for rows and by S_i for columns, to place the hardest test questions in left and the weakest experts topmost. In the result, the Table 2 is obtained containing generally zeroes in top left and ones in bottom right corner.

Table 2

$T_{j,i}$	β_1	β_2	β_3	β_4	…	β_m
θ_1	0	0	-	0	…	-
θ_2	0	0	0	-	…	0
θ_3	0	-	0	0	…	1
θ_4	0	0	-	1	…	1
…	…	…	…	…	…	…
θ_n	-	0	1	1	…	1

After the double sorting the experts and the test questions are numerated according to the obtained order, in the top row of the Table 2 the first approximated values of difficulty are placed. Experts' competence values are not yet defined.

Based on the Table 2 for most cells located on *j*-th rows and *i*-th columns intersections it's possible to calculate the value of correct answer probability for *j*-th expert and *i*-th question as the ratio of the correct answers number in some neighborhood of the cell to the number of cells in the neighborhood:

$$G_{j,i} = \frac{1}{(2l+1)^2} \sum_{p=j-l}^{j+l} \sum_{r=i-l}^{i+l} T_{p,r} \delta_{p,r} \ , \quad (3)$$

where l is size of the neighborhood, $\delta_{p,r} = 1$, if the answer is correct, $\delta_{p,r} = 0$, otherwise. There is a need to notice that according to the formula (3) it's possible to calculate the values of probability of correct answers only in the areas from $j=l+1$ to $j=N-1$ and from $i=l+1$ to $i=m-1$. In areas out of this part of the table, values of $G_{i,j}$ can be defined less accurately by reducing the size of the neighborhood. It's obvious that top left values of the table *G* are mostly equal 0 and bottom right are mostly equal 1. Top right and bottom left parts of the table can contain dashes. So, we have obtained the final table of probabilities of correct answers with unknown experts' competences and first approximation of questions complexity in following presentation:

Table 3

$G_{j,i}$	β_1	β_2	β_3	β_4	…	β_{m-2}	β_{m-1}	β_m
θ_1	0	0	0	$G_{1,3}$	…	-	-	-
θ_2	0	0	$G_{2,3}$	-	…	$G_{2,m-2}$	-	-
θ_3	0	-	$G_{3,3}$	$G_{3,4}$	…	$G_{3,m-2}$	$G_{2,m-1}$	-
θ_4	$G_{4,1}$	$G_{4,2}$	$G_{4,3}$	$G_{4,4}$	…	$G_{4,m-2}$	$G_{4,m-1}$	$G_{4,m}$
…	…	…	…	…	…	…	…	…
θ_{n-2}	-	$G_{n-2,2}$	$G_{n-2,3}$	$G_{n-2,4}$	…	$G_{n-2,m-2}$	$G_{n-2,m-1}$	1
θ_{n-1}	-	-	$G_{n-1,3}$	$G_{n-1,4}$	…	$G_{n-1,m-2}$	1	1
θ_n	-	-	-	$G_{n,4}$	…	1	1	1

According to (1) the value $G_{i,j}$ with α = 1 can be expressed by the formula

$$G_{j,i} = \frac{1}{1+e^{\theta_j-\beta_i}} . \qquad (4)$$

In logarithm we obtain the calculation expression:

$$\theta_j - \beta_i = \ln\frac{1-G_{j,i}}{G_{j,i}} = C_{j,i} . \qquad (5)$$

According to (5) the table *C* can be calculated from the Table 3 replacing zeroes and ones with dashes, because it is impossible to divide by zero and logarithm of zero does not exist.

Then we build an iterative procedure of finding values of θ_j and β_i taking into account that the first approximation of β_i is known:

1. Find an approximation for each θ_j, picking all filled values for index *i* from the table *C*.

$$\theta_j = \frac{1}{m_j}\sum_i \beta_i + C_{j,i} ,$$

m_i is the number of filled values in the row *j*.

2. Calculate average value $\theta_s = \frac{1}{n}\sum_{j=1}^{n}\theta_j$.

3. Subtract it from each θ_j:

$$\theta_j = \theta_j - \theta_s$$

4. Calculate the next approximation for questions complexity:

$$\beta_i = \frac{1}{n_i}\sum_j \theta_j - C_{j,i} - \theta_s ,$$

n_i – number of filled values in the column *i*.

5. Repeat from the step 1 until the iterative process of θ_j and β_i calculation converges.

The data base of test questions obtained such way can be used for new experts' certification. The adaptive procedure of experts' competence measurement was discussed above. A group of certified experts can define complexity of new test questions analogically.

6 Testing of the proposed experts' certification methodology on computer model

For the proposed experts certification methodology testing and test questions complexities estimation the computer model experiments were carried out. The computer modeling was implemented the following way.

Using a random numbers generator the vectors of experts' competences θ_j and test questions complexities β_i are generated, and these distributions are considerably nonlinear. One of these distributions realizations is provided on the fig. 2.

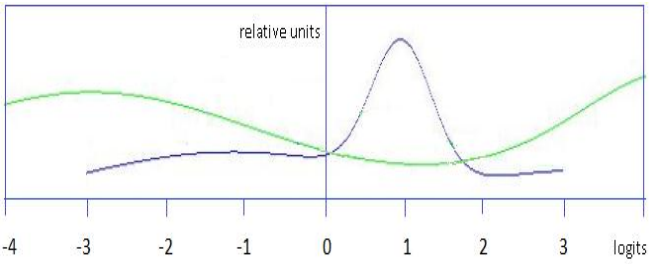


Fig. 2. *Relative distribution of experts by competence and test questions by complexity.*

Questions complexity in this sample varies from -4 to 4 logits, and experts competences from -3 to 3 logits.

According to the distribution, the tables of experts competences θ_j for *j* = 1..500 and test questions complexities β_i for *i* = 1..800 were generated randomly. Virtual experts were put under "testing" – they filled the table *T* the following way. The cell *j, i* of the table is filled with 1 if random value from 0 to 1 is less than the result of the Rasch formula (4), otherwise, it is filled with 0.

Further calculations were carried out the way described above. Randomly generated arithmetic progressions were taken as the first approximation for questions complexity β_i. The fig. 3 illustrates one of realizations of the table T after double sorting. Correct answers are presented by black pixels, incorrect – by white pixels. The region where $G_{i,j}$ values are in the interval from 0.499 to 0.501, is highlighted with solid line. According to the curve analysis, it is can be concluded that nonlinearity of the expert competences and questions complexity can be identified with instrumental means. The described iterative method allowed us to detect unambiguously the required dependencies.

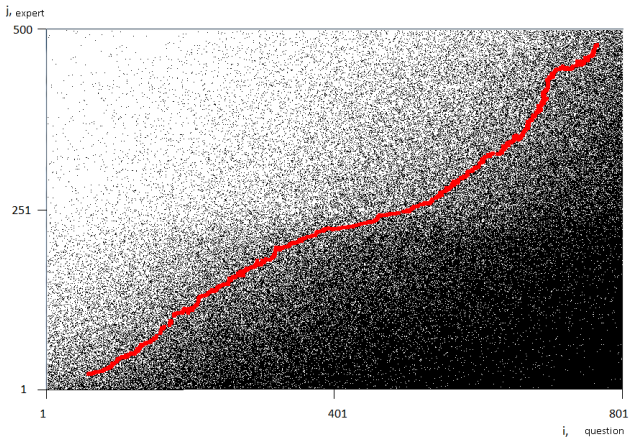


Fig. 3. *Table after T double sorting*

The results of the source tables restoration are presented on the fig. 4.

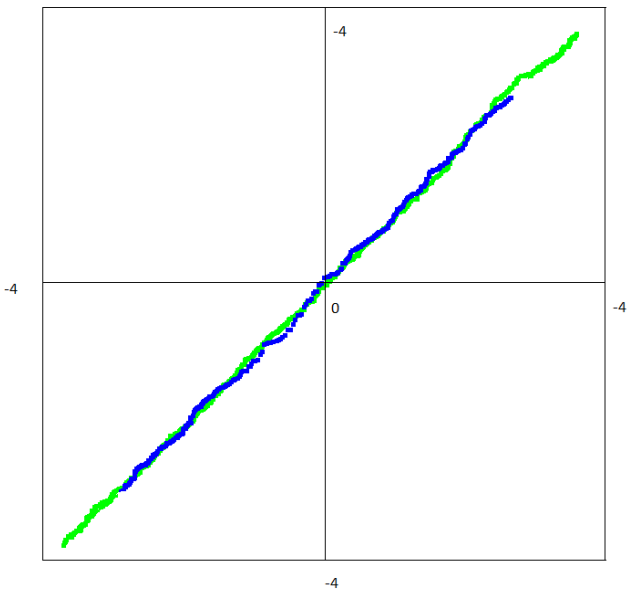


Fig. 4. *Results of sample variant calculation*

On this figure the abscissa represents the source tables values, the ordinate is the restored values – both are in logits. The long curve corresponds to β_i values, the short one – to θ_j values. The errors of restoration in absolute units are not greater than 0.12 logits. The results of computer modelling show that restoration of the dependences doesn't depend on the first approximation.

7 Conclusions

As a result of this work we can conclude that the use of crowdsourcing, Rasch model, adaptive sampling methodology for selection of test questions, as well as the iterative method for simultaneous determination of experts' competency level and complexity of test questions will solve the problem of metrology testing - a clear and objective measurement of these quantities. Because the technology is simple and low-cost compared to existing methods, the community of experts in a particular field of human activity by using crowdsourcing technology and proposed methods can independently carry out the development of test materials and conduct self-certification of their community network. Later these certified experts can participate in various individual and collective projects with predictable results.

It is very promising to use crowdsourcing for solving linguistic problems such as choice of definitions, selection of related keywords, and writing of encyclopedia articles. These linguistic tasks are very complex and only very large groups of people using crowdsourcing can successfully solve them.

8 Acknowledgements

This work was supported by the Russian Foundation for Basic Research, grant #13-07-00958 "Development of the theory and experimental research of a new information technology of self-managed crowdsourcing" and grant #13-07-00272 "The methods for automatic creation of associative portraits of subject domains on the basis of big natural language texts for knowledge extraction systems".

9 References

[1] Jeff Howe. "The rise of crowdsourcing". Wired. 2006, pp.1-4.

[2] Web site "Witology. Collective intelligence": http://witology.com/blog/library/detail/163/

[3] Condorcet, marquis de (Marie-Jean-Antoine-Nicolas de Caritat) (1785), Essai sur l'application de l'analyse à la

probabilité des décisions rendues à la pluralité des voix. Imprimerie Royale, Paris.

[4] V. Druzhinin. "Experimental Psychology: Textbook for Universities". 2nd ed., St. Petersburg, 2003, pp.1-319

[5] Rasch G. Probabilistic Models for Some Intelligence and Attainment Tests /Expanded Edition, with Foreword and Afterword by B.D. Wright. Chicago: University of Chicago Press, 1980.

[6] V. Protasov. "Construction of metasystem transitions". Institute of Physics and Technology, Moscow, 2009, pp.1-197.

[7] E.Baharad, J.Goldberger, M.Koppel, S.Nitzan. "Beyond Condorcet: Optimal Aggregation Rules Using Voting Records", CESifo München, 2011.

[8] Michael Charnine. "Keywen Automated Writing Tools". Booktango, USA, 2013, ISBN 978-1-46892-205-9.

Global Temperature Vs. Greenhouse Emissions Model: Genetic Algorithm Driven Parametric Multiobjective Fitting

Javier J. Sanchez-Medina, Jeffrey Miller and Enrique Rubio-Royo

***Abstract*—There are many elements of proof indicating that the forecasted Earth's starting glaciation is being severely counteracted by the well known anthropogenic greenhouse effect. This seems to be mainly because of the human development related increment of the emission of a number of gases, in particular CO_2.**

In this work we present a model fitting, by the use of a Genetic Algorithms. We have designed a polynomial model to infer the increment in global Average Temperature. We use as independent variables just 5 variables: the known emission volumes of 5 different greenhouse enabling substances, by using the European EDGAR database. And we do a curve fitting using the observed Temperature Increment recurring to the NASA/GISS GLOBAL Land-Ocean Temperature Index. By this experiment we try to shed some light on how every substance relates to each other, and how they are impacting the Earth's average warming.

***Index Terms*—enetic AlgorithmsMultiobjective Optimization-Climate ChangeGreenhouse GasesCurve FittingGreenhouse Climate Modelsenetic AlgorithmsMultiobjective OptimizationClimate ChangeGreenhouse GasesCurve FittingGreenhouse Climate ModelsG**

I. Introduction

summary of the results.

There is a well-established consensus among the scientific community regarding how the human race is and will be suffering from abrupt changes in climate, due to the side effects of the development of modern societies all over the Earth. It seems that, in a few words, the Earth is absorbing more energy from the Sun than it reflects to the Space ([1]), which may be producing a global heating in our planet. Some paleoclimatologists are tagging this as the the Anthropocene Era. The intensive emission of Greenhouse Gases subsequent to the Industrial Revolution, especially, the CO_2 emissions seem to be producing a dramatic tendency change regarding the global average temperature on Earth.

Not just the atmosphere, but the oceans are experimenting sudden changes as well. The well known acidification process, which seems also to be happening at a very alarming rate ([2] and [3]), is expected to produce a deep impact in biodiversity in the oceans, which will again impact our lives in the end, for instance, regarding the global production of food needed for the growing world population ([4]).

A majority of climate researchers believe that the current deviations are provoked, at least partially, due to the human greenhouse gases (GHGs) emissions in the last 150 to 200 years. Some (like [5]) extend that period to thousands of years due to massive ranching.

There is a some controversy on whether this global temperature phase is just an outlier or if it is a real curve trend changing. For example, in [6] they argue that the emission of GHGs may result in the cooling of the Average Temperature of Earth. They think small increments of Carbon Dioxide and Methane will not have an important effect on the climate.

However, it seems that is not small the increment of the CO_2 concentration in the atmosphere. There are recent estimations reckoning we may have surpassed the level of 400 ppm, "for the first time since humans walked the Earth" ([7]).

In this situation there is a very big amount of researchers trying to understand what is happening, and how we can revert or at least stop this process.

Modelling the relationship between the global average temperature and the Anthropogenic GHGs emissions is extremely difficult, as climate by itself is a very complex system. This is mainly because the Earth's climate includes several feedback processes, most importantly, the hydrological cycle ([8]).

Apart from the Anthropogenic causes, it seems to be mainly affected by Earth-orbital variations. [9] explained how accurate these astronomical models represent past climate fluctuations. According to these GHGs ignoring models, we should be facing the beginning of a new global cooling phase, a new glaciation that started 6000 years ago and that will last for the next 23000 years. However, measurements in the last two centuries seem to contradict that forecast.

By the present work we do not pretend to establish a simplistic model of how Greenhouse Emissions and the Earth's temperature are related. We just want to provide another view of the problem of Climate Change, using a new approach. Instead of developing complex multivariate holistic models for the climate evolution, we have just picked a deliberately simple mathematical model. In that model we assume that the short term of the global average temperature deviation is just a function of the amount of 4 different substances. Then we have performed a parametric curve fitting by using a Genetic Algorithm. From this non-deterministic optimisation, as we analyse the resulting adjusted coefficients, we hopefully may provide three benefits:

- The adjusted model may give the climate researchers some new clues on how every substance relates to each other, and how each impacts the Earth's average warming.
- It may also give hints on how quick any of these

J.J. Sanchez-Medina and E. Rubio-Royo are researchers at CICEI (www.cicei.com), and professors at DIS (Computer Science Department) ULPGC, SPAIN.

Jeffrey Miller is professor at University of Southern California, USA

considered substances impact the Earth's climate. This may be used as a tool for the establishment of an order of priorities and emergency among the pollutants.

- Finally, it may also provide another model to estimate future global temperature scenarios, just another argument to the cause of the majority of the science community currently embraced into the moral obligation of persuading decision and policy makers to actively fight anthropogenic climate change.

II. State of the Art

In this section we will describe a list of works that are somehow related to our research. Of course, there may be works left behind, since this topic has a huge number of researchers and groups working on it. There are many parametric models fine tuned through natural computing techniques like Genetic Algorithms/Programming, or Neural Networks.

One of the most cited works, related to the work we are presenting in this paper is in [10]. The authors propose a coupled ocean−atmosphere model for foreseeing what the climate changes will be when the CO_2 concentration doubles. This work uses only CO_2 atmospheric concentration changes as model input. According to their results, the doubling of the CO_2 concentration would produce an increment globally higher than 35% of the equilibrium response.

In [11] they propose a Neural Network based model, using a Perceptron Multi-layer Network to predict the greenhouse micro-climate behaviour using as input divers environmental variables like wind, solar radiation, etc.

In [12] a Genetic Programming approach is used to model the impact of Climate Change into the population of a specific species: the Formosan Landlocked Salmon. This work approaches the complex systems modelling very similarly as we do at this work. Starting from already available databases, they perform a Genetic Programming driven data mining to draw some conclusions and prospects for future scenarios.

III. Proposed Model Description

In this section we represent the mathematical model (Eq. 1, eq. 3 and eq. 2) we have used to adjust a relationship between global emissions and global delta temperature or the average increment of the Earth's surface temperature.

In equation 1 we represent the main equation where one can see what is the relationship between the temperature deviation for a specific year ($\Delta T(y)$), being y the year, and five polynomials, also dependent of y. Each one of these Polynomials ($f_x(y)$) is related to each one of the five considered substances (table I).

In equation 2 we have described $f_x(y)$. It is a fifth degree polynomial. Consequently, it is defined through 6 constant coefficients ($P_{1,x}$ to $P_{6,x}$).

The independent variable of these polynomial is the result of a third equation ($E_x(y)$), defined in equation 3. This function is conceptually represented on figure 1. For each value of the variable y (year) we simply accumulate all the emissions for the x substance, for the period defined by the two additional variable Z_x and A_x.

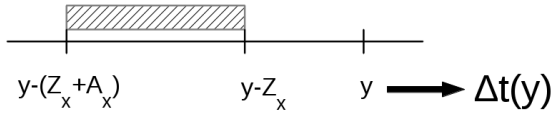


Fig. 1. Time Window Considered for Each Substance

Code	Substance
α	CO_2 (Carbon Dioxide except from Short Burning Carbon Cycle)
β	$SCBBCO_2$ (Carbon Dioxide just from Short Burning Carbon Cycle)
γ	CH_4 (Methane)
δ	N_2O (Nitrous Oxide)
ϵ	$HFCs$ (Hydrofluorocarbons: HFC-23, 32, 125, 134a, 143a, 152a, 227ea, 236fa, 245fa, 365mfc, 43-10-mee)

TABLE I
Codes and Substances

In summary, our model consists of 5 polynomials, each one characterised by 8 variables, 6 polynomial coefficients and 2 range variables. In total we have 40 integer variables to be optimised through the curve fitting process.

We use a genetic algorithm to optimise that variables.

After the optimisation, we may find if there is some information not just regarding the influence of each substance to the global temperature deviation, but also regarding the time lapse between the emissions and its impact into global delta temperature, and hopefully, regarding the persistence of that influence. In other words, we are trying to include in our model two assumptions: the effect on $\Delta t(y)$ of each substance may not be immediate, and we also assume that some substances may disappear from the Earth's atmosphere after a number of years, as it happens with CO_2 when it is absorbed by the oceans (producing other problems than Greenhouse Warming).

We realise the scale of the studied period may be too short to observe meaningful information regarding the influencing period of the studied substances, but we use the Genetic Algorithm to optimise also these two variables, A_x and Z_x, just in case any new information can be obtained from that. Perhaps this approach can be used again with wider periods in future research.

$$\Delta T(y) = f_\alpha(y) + f_\beta(y) + f_\gamma(y) + f_\delta(y) + f_\epsilon(y) \quad (1)$$

$$f_x(y) = P_{1,x} \times (E_x(y))^5 + P_{2,x} \times (E_x(y))^4 + \\ + P_{3,x} \times (E_x(y))^3 + P_{4,x} \times (E_x(y))^2 + P_{5,x} \times E_x(y) + P_{6,x} \quad (2)$$

$$E_x(y) = \sum_{i=y-(A_x+Z_x)}^{y-Z_x} Emi_x(i) \quad (3)$$

The units used are the following: Celsius for temperatures and Eg[1] for total emissions.

[1] $1 exagramme = 10^9 gigagramme = 10^{18} gramme$

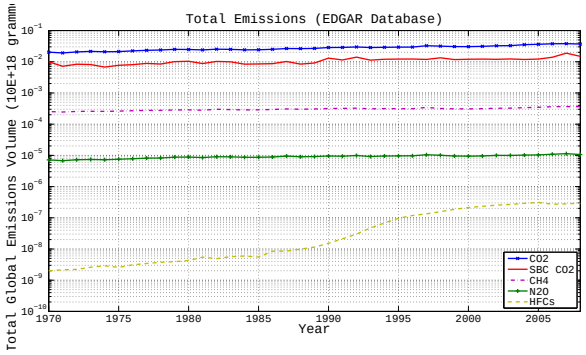


Fig. 2. Total Emissions

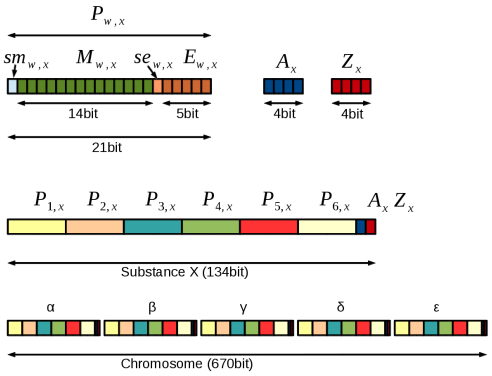


Fig. 3. Chromosome Encoding

IV. Databases used

numbers, language,

To develop the current research we have used two different databases. The first one, coming from the European Commission supported program EDGAR (http://edgar.jrc.ec.europa.eu) in order to have their estimations for global total GHGs emissions in the tackled period (1970 to 2008). We have simply summed all the emissions year by year for each one of the five substances considered. The resulting total emissions curves can be seen in figure 2.

On the other hand, we have used the NASA/GISS GLOBAL Land-Ocean Temperature Index (http://data.giss.nasa.gov/gistemp), during the period from 1970 to 2008 to match the available EDGAR data.

A group of researchers at the NASA Goddard Institute led by James Hansen developed an analysis method to estimate the global average temperature month by month from 1880 to the present ([13] [14] and [15]). To be precise, they produce a set of delta temperatures or *temperature anomalies* with respect to the base period 1951-1980. As they say in the NASA/GISS website,

> The reason to work with anomalies, rather than absolute temperature is that absolute temperature varies markedly in short distances, while monthly or annual temperature anomalies are representative of a much larger region.

We will be using the Delta temperatures in this work as well, to be consistent with the data source used. We have used the average yearly temperature anomaly at our studied period: 1970 to 2008.

V. Material and methods

A. Genetic Algorithm

1) Chromosome Encoding: In figure 3 we represent the Chromosome used for the non deterministic optimisation developed. It is a binary encoded chromosome where the same structure is cloned five times, one per substance considered. For each substance we have defined the following fields:

- $P_{w,x}$: Each one of the 6 polynomial coefficients for the substance x (eq. 3). These coefficients are encoded with 21 bits (see fig. 3). Each coefficient is defined by a signed mantissa and a signed exponent. Range: -2097152 to +2097152.
- $sm_{w,x}$: Mantissa sign for the coefficient number w and the substance x. One bit.
- $M_{w,x}$: Mantissa module for the coefficient number w and the substance x. Encoded using 14 bits. Range: [0, 16384]
- $se_{w,x}$: Exponent sign for the coefficient number w and the substance x. One bit.
- $E_{w,x}$: Exponent module for the coefficient number w and the substance x. Encoded using 5 bits. Range: [0, 31].
- A_x: With 4 bits we encode the span of years of the considered emissions window for each substance x. Range: [0,15].
- Z_x: Years from the modelled year delta temperature to the last considered emissions year, for substance x. Encoded with 4 bits. Range: 0 to 15.

Therefore, each substance is assigned a 134 bits long portion of the chromosome. The chromosome total size is 670 bits.

2) Selection, Crossover and Mutation Operators: In this subsection we will describe the basic Genetic Algorithm operators used. First of all, we have used a truncation and elitism selection operator. The population size was 1000 individuals. The truncation picked the most fitted two thirds of that population, and with them we produce 996 siblings for the new generation. The missing 4 are the result of cloning the best 4 in the former generation.

The Genetic Algorithm ending criterion was just stopping after a fixed number of generations, 700 for the presented experiments.

Regarding Crossover, we have used a standard two point crossover. Since all the binary combinations of our chromosome encoding are feasible combinations, we can randomly select two points at the chromosome parents, and interchange the central chunk to produce two siblings.

Finally, with respect to Mutation, we have used a variable Mutation probability, starting with a hyper-mutation value (0.99) that decreases gradually by a factor of 0.99, generation by generation, to reach at the end of the Genetic Algorithm evolution values in the order of the inverse of the population size. In our experience, this is a quite successful method regarding mutation.

When an individual is picked to be mutated, we simply randomly select one bit in its chromosome and we roll a binary dice for it. Again, since all the possible binary combinations in our chromosome encoding are legal, we can do this very simply and fast.

3) Fitness Function: The more important element when designing a Genetic Algorithm may be its Fitness Function, where we, in a normalised fashion, define what is the optimality criterion (or criteria for multiobjective optimisation). In our case, we have used a very simple and convenient one: we used the Coefficient of Determination, or R^2, and it is basically the square of the Pearson's correlation coefficient. This coefficient is calculated as shown in equations 4, 5 and 6, and is very appropriate for polynomial curve fitting.

In equation 4 we have SS_{total}, the so called total sum of squares. In that equation it is a way to calculate it using the variance of all the values of the actual reference function (ΔT) and the number of elements (years) in the considered series (N).

$$SS_{total} = (N-1) * \sigma(\Delta T) \tag{4}$$

In equation 5 we have SS_{resid}, known as residual sum of squares. To calculate this parameter we used both the actual reference function ($\Delta T(y)$) and the function to be fitted, which represented in that equation as $\Delta T_f(y)$.

$$SS_{resid} = \sum_{y} (\Delta T(y) - \Delta T_f(y))^2 \tag{5}$$

$$R^2 = 1 - \frac{SS_{resid}}{SS_{total}} \tag{6}$$

Values of R^2 over a level +0.8 (80%) are usually said to mean a good fit.

Hence, for each chromosome to be evaluated, we build the corresponding polynomial and compare, year by year, its value with the desired curve, which is the actual evolution of the average global delta temperature (provided by the NASA GISS Database, see section IV). To build our adjusted curve we need each substance total emission curve, obtained from the European Commission EDGAR database (again, see section IV).

Only 2 of 104 optimisations did not get higher than 0.8 for its final fitness (see figure 4)

VI. Results and Discussion

In this section we present the main results obtained. For every table and figure presented we have removed the outliers using 2 times the Standard Deviation as cutting criterion (outliers limit).

We have run the Genetic Algorithm for 104 times. In figure 4 we represent the evolution of the fitness value of the best individual for every test.

In figure 4 we can observe how all but two runs produced a best fitness value over 0.8, which is usually taken as the threshold limit to assume that the polynomial curve fitting is good enough.

In figure 5 we are representing a lot of information. First we have the dot cloud corresponding to the average global Delta

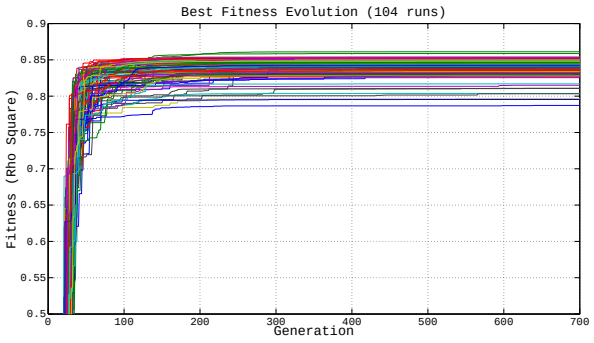


Fig. 4. Best Fitness Evolution for All the 104 Experiments Run

Temperatures obtained from the NASA GISS office database (see IV).

Then, for the studied range (1970-2008) we have the adjusted curve. This curve is the resulting sum of a polynomial curve for each one of the studied substances (see table I), as we explained in section III.

Therefore, we have also represented each one of the five summand curves to see how important each one of them is to the resulting adjusted curve.

It is quite obvious the huge impact of CO_2 to that adjusted curve. But this influence is not unexpected, since CO_2 coming from no Short Burning Cycle processes is, by far, the most abundant GHG substance emitted to the atmosphere, as can be observed in figure 2.

$$CI(x) = \sum_{y=1970}^{2008} \frac{f_x(y)}{Emi_x(y)} \left(\frac{year \times Celsius \times 10^{-2}}{g \times 10^{18}}\right) \tag{7}$$

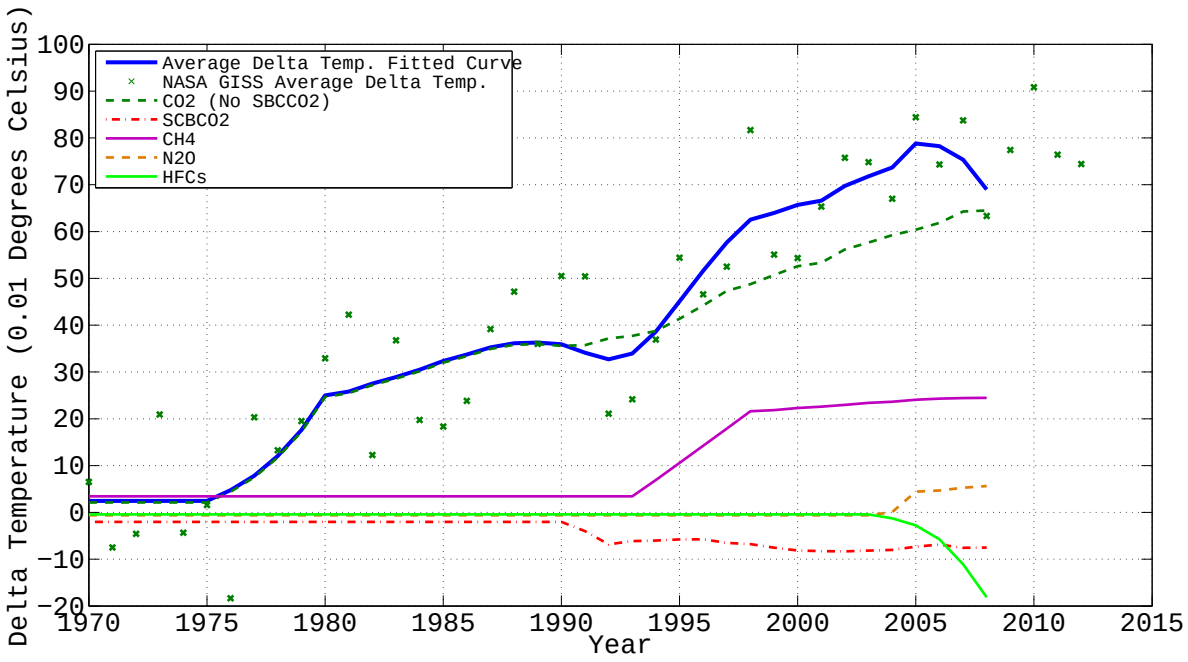
Finally, in figures 6 and 7 we represent the statistical distribution for two variables regarding the emissions period considered for each substance/polynomial (see eq. 1, 3 and 2). It seems that the information obtained from those parameters is not very trustworthy, statistically speaking, except perhaps for CO_2. For that substance, results indicate that the latest relevant year for calculating a year Average Delta Temperature is 6 years before the considered year. And it seems that the time window is 6 years wide. So, the polynomial related to that specific substance takes, on average, the years $(x-12)$ to $(x-6)$ for inputting their emissions into that equation (eq. 3).

Of course, it seems that we should take a much wider period to get some useful news regarding that time versus substance relationship.

VII. Conclusions

In this paper we have tried to solve three questions:

1) What kind of polynomial relationships can be established between five different Greenhouse Gas Substances emitted (EDGAR Database) and the Global Average Temperature variation across time (inferred by the NASA GISS Database).



670

Fig. 5. NASA GISS Average Global Land-Ocean Delta Temperature (Base period 1951-1980) and Model Fitted Curves

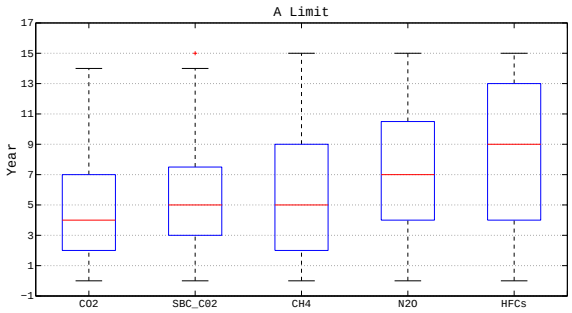


Fig. 6. A_x Box and Whiskers Diagram for Each Substance Removing Outliers

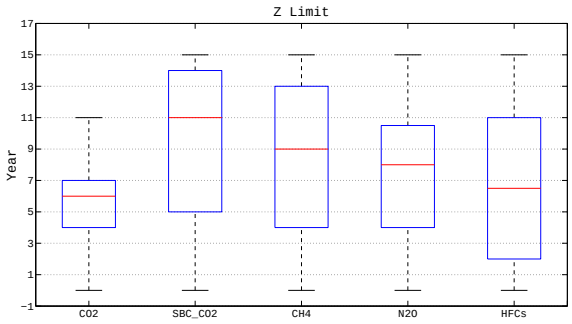


Fig. 7. Z_x Box and Whiskers Diagram for Each Substance Removing Outliers

2) How important is each substance in the fitted curve obtained.
3) Is there any important information regarding the time period that affects the current temperature more for each substance. In other words, how long does it take to affect the global temperature, and for how long does it affect the global temperature.

Some hints may be present within the results obtained with respect to the first two questions. But it also seems that the period studied is too short for obtaining relevant conclusions regarding the third question.

A quite strong convergence is observed regarding the polynomial coefficients. So it seems that the GA is reaching a very good maximum, very likely to be the global maximum.

Getting into the resulting optimised parameters, a strong influence is observed from the CO_2 emissions (not the Short Burning Cycle CO_2, which seems to not be affected very much), the Methane (CH_4) and perhaps the Nitrous Oxide as well (N_2O).

There is no clear information regarding the time period's influence, affecting each substance. The statistical dispersion of that parameter seems to prevent us from inferring any idea other than perhaps we should search for data including a wider period to see any real correlation regarding how long it takes a specific substance to affect the Global Temperature, and for how long does it affect it.

There may be some hints only regarding CO_2 suggesting a $t-12$ to $t-6$ years time window as the most influential to the corresponding polynomial.

The predominant influence of CO_2 in the Average Delta Temperature is very clear. But of course this may be highly

influenced by the abundance of this particular substance in the Earth's atmosphere. If we try to observe the relative influence of each one of the 5 different considered gases, it looks like the most scarce are the most harmful, and CO_2 seems to be the less influencing.

As for future work we must search for a wider time period, optimally from a different set of databases to validate A) the used simplistic model and B) the conclusions obtained with this research.

APPENDIX

ACKNOWLEDGEMENTS

We sincerely want to thank the Joint Research Centre Ispra at the Institute for Environment and Sustainability, Air and Climate Unit, a research centre belonging to the European Commission to provide us with data from their EDGAR Database, with their estimations for global total GHGs emissions. In particular, our kindest thanks to Greet Janssens-Maenhout.

We must also thank the NASA GISS office for their generosity, sharing their Global Delta Temperature Database for its public use.

REFERENCES

[1] J. Hansen, M. Sato, P. Kharecha, and K. v. Schuckmann, "Earth's energy imbalance and implications," *Atmospheric Chemistry and Physics*, vol. 11, no. 24, pp. 13 421–13 449, 2011.

[2] Anonymous, "Ocean acidification," *Journal of College Science Teaching*, vol. 41, no. 4, pp. 12–13, Mar 2012, copyright - Copyright National Science Teachers Association Mar/Apr 2012; Caractersticas del documento - Photographs; ltima actualizacin - 2012-07-02; CODEN - JSCTBN; SubjectsTermNotLitGenreText - California. [Online]. Available: http://search.proquest.com/docview/1014005380?accountid=14705

[3] T. Fenchel, "Ocean acidification," *Marine Biology Research*, vol. 7, no. 4, pp. 418–419, 2011. [Online]. Available: http://www.tandfonline.com/doi/abs/10.1080/17451000.2010.550051

[4] F. Tao, Z. Zhang, J. Liu, and M. Yokozawa, "Modelling the impacts of weather and climate variability on crop productivity over a large area: A new super-ensemble-based probabilistic projection," *Agricultural and Forest Meteorology*, vol. 149, no. 8, pp. 1266 – 1278, 2009. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0168192309000501

[5] W. F. Ruddiman, "The anthropogenic greenhouse era began thousands of years ago," *Climatic Change*, pp. 261–293, 2003.

[6] G. Chilingar, O. Sorokhtinand, L. Khilyuk, and M. Gorfunkel, "Greenhouse gases and greenhouse effect," *Environmental Geology*, vol. 58, no. 6, pp. 1207–1213, 2009.

[7] N. Jones, "Troubling milestone for co2," *Nature Geosci*, vol. 6, no. 8, 8 2013.

[8] J. E. Harries, H. E. Brindley, P. J. Sagoo, and R. J. Bantges, "Increases in greenhouse forcing inferred from the outgoing longwave radiation spectra of the earth in 1970 and 1997," *Nature*, pp. 355–357, 2001.

[9] J. Imbrie and J. Z. Imbrie, "Modeling the climatic response to orbital variations," *Science*, vol. 207, no. 4434, pp. pp. 943–953, 1980. [Online]. Available: http://www.jstor.org/stable/1683550

[10] R. J. Stouffer and et al., "Response of a coupled ocean−atmosphere model to increasing atmospheric carbon dioxide: Sensitivity to the rate of increase," 1999.

[11] M. A. Bussab, J. I. Bernardo, and A. R. Hirakawa, "Greenhouse modeling using neural networks," 2007.

[12] C.-P. Tung, T.-Y. Lee, Y.-C. E. Yang, and Y.-J. Chen, "Application of genetic programming to project climate change impacts on the population of formosan landlocked salmon," *Environmental Modelling and Software*, vol. 24, no. 9, pp. 1062 – 1072, 2009. [Online]. Available: http://goo.gl/aNqzp

[13] J. Hansen, D. Johnson, A. Lacis, S. Lebedeff, P. Lee, D. Rind, and G. Russell, "Climate impact of increasing atmospheric carbon dioxide," *Science*, vol. 213, no. 4511, pp. 957–966, 1981. [Online]. Available: http://www.sciencemag.org/content/213/4511/957.abstract

[14] J. Hansen and S. Lebedeff, "Global trends of measured surface air temperature," *Journal of Geophysical Research: Atmospheres*, vol. 92, no. D11, pp. 13 345–13 372, 1987. [Online]. Available: http://dx.doi.org/10.1029/JD092iD11p13345

[15] J. Hansen, R. Ruedy, M. Sato, and K. Lo, "Global surface temperature change," *Reviews of Geophysics*, vol. 48, no. 4, pp. n/a–n/a, 2010. [Online]. Available: http://dx.doi.org/10.1029/2010RG000345

Dr. Javier J. Sánchez Medina obtained his Bachelor Degree in 1998 and Masters in 2001 in Telecommunications Engineering. He earned his Computer Science PhD in 2008, at the ULPGC. He is assistant professor at the Department of Computer Science (DIS), and researcher at the Innovation Centre for Information Society (CICEI), both in the University of Las Palmas de Gran Canaria (ULPGC). He is glad to volunteer serving as reviewer at some of the most prestigious journals in the area and as organizing committee member in some IEEE ITSS conferences.

Dr. Jeffrey Miller is currently an Associate Professor in the Computer Engineering department at the University of Alaska Anchorage. Prior to joining UAA in 2007, he was an Adjunct Professor in the Computer Science department at California State University, Los Angeles for 5 years. Dr. Miller earned the Bachelor of Science, Master of Science, and Ph.D. degrees in Computer Engineering and Computer Science in 2002, 2002, and 2007, respectively, all from the University of Southern California. His research interests include vehicle-to-vehicle and vehicle-to-infrastructure communication, vehicular ad hoc networks (VANETs), and distributed algorithms and software architectures for Intelligent Transportation System (ITS) applications.

Dr. Enrique Rubio Royo earned his undergraduate degree in Physics in 1971. He read his Sciences PhD in 1980 and has been a Professor since 1990. He is the Director of the Innovation Center for Information Society, a research centre belonging to the University of Las Palmas de Gran Canaria. He has extensive experience in Information Society and Technology.

The SWAM Arabic Morphological Tagger: Multilevel Tagging and Diacritization Using Lexicon Driven Morphotactics and Viterbi

Mohammed A ELAffendi
Department of Computer Science
College of Computer & Information Sciences
P O Box 66833
Prince Sultan University
Riyadh 11586

Majdy ALTayeb
School of Mathematical Sciences
University of Khartoum
Khartoum
Sudan

Abstract-*SWAM is an exhaustive systematic algorithm that employs a sliding window strategy to identify all candidate morphological decompositions (segmentations) of a given Arabic word. At the upper level, the algorithm applies lexicon driven morphotactics and utilizes morpheme level unigram and bigram statistics (inner word context) to rank the resulting morphological segmentations and pick the topmost candidate. As a side effect, the algorithm identifies the surface non-vocalized morphological pattern and the relevant features of the input word. The lexicon-driven morphotactics are used to handle deletions, transformations or transpositions resulting from morpheme interaction during the derivation process. The main tactics used include injecting artificial surface roots and patterns in the respective lexicons. This greatly improves the coverage of the algorithm, but leads to some redundancy. Inverse mappings are then used to identify the real deep morphemes. At the lower level, Viterbi algorithm is used to refine the search and identify the vocalized deep version of the surface pattern. Combining the deep pattern with the lexeme leads to a composite tag that provides a wealth of features associated with the input word. The accuracy of the algorithm is very impressive, ranging from 97.3 to 98.2. SWAM does not impose a restricted tag set and support multilevel fine-grain tagging.*

1 Introduction

The last few years witnessed the appearance of a number of Arabic Part-of-Speech and morphological taggers as part of the general efforts made by international centers to build statistical NLP systems. These include the work done by the NLP group of Stanford University [1], the work done by Nizar Habash and the Columbia NLP Group, [8], [10], [11] - the work done by the Leeds Arabic NLP group (Sawalha & Atwell 2010) [2], plus scattered efforts at different places and centers [5] ,[6] ,[7] and [12], [13].
The emerging Arabic tagging systems may be divided into two main categories:

1- Part of Speech taggers that apply approaches similar to those used for English and some Latin languages. Taggers in this category use a small restricted tag set that are very similar to their English counterparts (e.g. the Penn Arabic Tag Set used by the Stanford Tagger), or an extended version (like the one used by Khoja 2000).
2- Morphological taggers that identify more refined fine-grained tags based on morphological analysis. Examples of these include the work of Habash and Rambow at Columbia [8], [9], [10], [11] and the work of Sawalha and Atwell in Leeds [2].

The second approach appears to be more promising and appropriate for Arabic (Sawalha and Atwell 2010). However, the main problem with this approach is that there is no consensus yet regarding the level of detail and the appropriate size of the tag set.

In building SWAM, we adopted the following points of view:

1- A multilevel tagging approach is more appropriate for Arabic. We differentiate between surface non-vocalized patterns and deep vocalized patterns. The surface patterns represent upper level broad tags that may be associated with more than one morphological class.
2- Deep morphological patterns represent complete classes overloaded with appropriate features and attributes. Deep patterns are represented as Tag Charts in SWAM.
3- A Tag Chart is a data structure providing detailed features based on certain conditions and combinations.
4- For templatic words the deep pattern is the composite tag. For other words (e.g. particles), the morphological class is the tag.

The power of this approach is that users may customize their tag sets based on the nature of the application. Most of

the known tag sets can easily be generated as side effects of the composite tagging process.

SWAM is a two-phase pattern oriented tagger that tries to identify the pattern of a given templatic word, or the class if the word is non-templatic. To achieve this, the algorithm enumerates and ranks all possible morphological decompositions (solutions) for the input word. The ranking process is based on morpheme level unigram and bigram frequencies collected from a training corpus.

Patterns identified in the first phase of the algorithm are surface non-vocalized patterns, and accordingly require further disambiguation. The second phase of the algorithm uses the Viterbi algorithm and sentence level context to deduce the diacritization and identify the corresponding deep pattern.

SWAM applies a sliding window strategy and a number of lexicons to derive the appropriate morphological decompositions of the input word. To maintain the simplicity of the logic and avoid complex morphotactics, the algorithm applies a redundancy principle and inverse mappings to handle irregular cases resulting from transformations in the derivation process.

This paper describes in some detail the SWAM algorithm and provide some preliminary results

2 Related Work:

Our work is similar in philosophy and spirit to the work done by the Columbia NLP group presented in (Habash & Rambow [10] and later implemented in the MADA+TOKAN toolkit (Habash, Rambow and Roth 2009) [8].

In the MADA+TOKAN kit, the morphological analysis is done in the MADA component using an SVM engine. The tokenization and morphological decomposition are performed in the TOKAN component.

One interesting point about MADA is that it does not impose a restricted tag set and leave it for the user to explore a wide range of possible tags and annotations. This is very similar to our approach where no restricted tag set is imposed, but a large number of possible tags embedded in coarse morphological tag classes.

In SWAM, both morphological analysis and tokenization are performed during the first phase in the SWAM component. Diacritization is performed during the second phase through the Viterbi component.

Our system has also some similarities with the fine-grained system built by the Leeds Arabic NLP group (Sawalha & Atwell 2010). In addition to the fact that no restricted tag set is implied, the tag feature vectors used in the Leeds system are similar in spirit to our overloaded Tag Charts.

3 The SWAM Tagging Algorithm:

The SWAM tagger is a two phase Arabic morphological analysis system that identifies and assigns tag classes to input words. The tag classes are based on Arabic morphological patterns or morphological classes in case of non-templatic words. In the context of SWAM, each pattern represents a potential tag class. During the first phase, SWAM applies a sliding window strategy to identify possible tags for a given word. In the process, SWAM produces all possible decompositions (segmentations) of the input word. In the second phase, the SWAM applies the Viterbi algorithm to resolve ambiguities and perform diacritization.

Phase 1: Morphological Analysis (for any input test or production Text):

1- Apply the SWAM analyzer to compute all possible morphological decompositions for all input Arabic words.
2- Apply morpheme level unigram and bigram probabilities to rank the resulting decompositions. This is the first disambiguation step which may eliminate some of the resulting decompositions.

Phase 2: Further Disambiguation and Diacritization:

The morphological decompositions obtained in phase 1 above are associated with non-vocalized morphological patterns. To identify the corresponding vocalized deep patterns, the Viterbi algorithm is applied. At this stage, sentence level bigram probabilities are utilized to capture the context.

3.1 The Training Corpus:

The morpheme level and sentence level unigram and bigram probabilities have been computed using a small training corpus. A portion of the corpus has been hand tagged to bootstrap the process. After that SWAM has been guided by experts to pick the correct solutions. Table 1 below provides some details about the corpus.

Table 1: Components of the Training Corpus

CATTEGORY	*Number of files*	*Total No of words T*	*Distinct words D*	*Ratio D/T*	*Ratio T/Total*
Religion	9	63548	18733	29.47 %	12.97 %
Social	53	39809	9202	23.1%	8.12%
Economics	77	34728	5608	16.1%	7.09%
Cultural	110	125287	18341	14.6%	24.48 %
Political	53	29097	3346	11.49 %	5.94%
Health	17	9732	1372	14%	1.98%
Technology	72	45327	4083	9%	9.25%
Sport	92	44223	4625	10.4%	9.03%
Art	40	24724	2487	10%	5%
Psychology	5	8702	594	6%	1.7%
Language	2	4950	595	12%	1%
Education	19	35915	2307	6%	7.3%
Philosophy	2	2970	351	11.8	0.6%
Sciences	15	20713	1860	8.9%	4.2%
	566	**489725**	**73504**	100%	100%

3.2 The SWAM Morphological Analysis Algorithm:

SWAM is table driven morphological analysis algorithm that has been around for some time. The first version of the algorithm appeared in [3]. Another version has been published in the context of a full framework for Arabic morphological analysis [4].
The algorithm requires the availability of the following lexicons:

- A list of all roots in Arabic.
- A list of all morphological forms (patterns).
- A list of all particles.
- Lists of all invariable (non-templatic) nouns (pronouns, demonstrative nouns, conjunction nouns …etc.), a list for each type.
- A list of all possible prefixes (including proclitics)
- A list of all possible suffixes (including enclitics).

Given these lexicons, the algorithm proceeds as follows:

```
posn:=1;  //The window begins at position 1
For each stored pattern p do
begin
   while ((posn + length(p)-1)  ≤ length(word))  do
    begin
       if (p is a morphological pattern) then
       begin
           Stem:=ExtractStem(posn, p, word);
           Root:=intersect(stem, p);
           Prefix:=ExtractPrefix(posn, p, word);
           Suffix:=ExtractSuffix(posn, p, word);
           If ValidRoot(Root) and ValidPrefix(Prefix)
and ValidSuffix(Suffix) then
           AcceptSolution(Root, p, Prefix, Suffix);
       if (p is a particle or a invariable word) then  //
handle invariable words
      begin
           Prefix:=ExtractPrefix(posn, p, word);
           Suffix:=ExtractSuffix(posn, p, word);
          if ValidPrefix(Prefix) and
ValidSuffix(Suffix) then
          AcceptInVarSolution(p, word, prefix,
suffix);
      end;
       posn:=posn+1;  //move the window one
position to the left
   end;
end;
```

3.3 Lexicon Driven Morphotactics: Injected Artificial Surface Morphemes and Inverse Mappings:

To handle the transformations caused by morpheme interactions, SWAM applies the following solution:

1- Inject all transformed surface roots (defective and partial roots) in the ROOTS lexicon. Based on this strategy, partial and weak roots such as those shown in the left column of table 2 below will be inserted directly in the ROOTS lexicon.
2- Inject transformed surface morphological patterns (defective, transposed and partial morphological patterns) in the PATTERNS lexicon. For example, patterns such as those shown in the left part of table 2 below will now be inserted in the PATTERNS lexicon.
3- Use a binary flag (injected flag) in each of the above lexicons which is set if the entry is injected and left reset if the entry is original (0 if the entry is original and 1 if the entry is injected). For example the value of this flag will be 1 for the injected root (هد/hd/) and 0 for the original root (قول/qwl/).
4- Build a mapping table that maps transformed roots to the corresponding original roots:

Table 2: SWAM Inverse Mappings Examples

Surface Form		Original Form	
Root	Pattern	Root	Pattern
هد /hd/	فع /fç/	هدى /hdý/	فعل /fçl/
خذ /xð/	çل/عل	أخذ /Âxð/	فعل/fçl/
قال /qAl/	فال/fAl/	قول /qwl/	فعل/fçl/
صام/Sam/	فال/fAl/	صوم/Swm/	فعل/fçl/

Note: Transliteration of Arabic terms is based on the HSB transliteration scheme (Nizar Habash 2010), page 25.

The above measures greatly improve the "coverage" of SWAM and makes sure that it does not miss proper words.

Based on the above measures and strategies, SWAM handles each of the above problems as follows:

1- *Deletions:* According to the SWAM "injection strategy", partial roots and corresponding partial morphological patterns are injected in the respective lexicons. A partial pattern will naturally generate a partial root. We know that the resulting partial root is not original because the corresponding "injected" flag is set to 1. Accordingly, the resulting partial root will be used as a key in surface/original mapping table to retrieve the original root. If no match is found, the analysis is dropped.
2- *Transformations:* If the analysis leads to an acceptable defective root, the SWAM "injection strategy" again enables the algorithm to capture the resulting defective root. We know that this root is not original because the corresponding "injected" flag has been set to 1. Accordingly, the resulting defective root will be used as a key to retrieve the original root form the surface/original mapping table.

3- *Transpositions:* According to SWAM "injection strategy", transposed morphological patterns are injected in the PATTERNS lexicon. This implies that the algorithm can easily capture transposed roots, which has also been injected in the ROOTS table. The resulting transposed root will be used as a key to retrieve the original root from the surface/original mapping table.

3.4 Morpheme Level Bigram Probabilities:

SWAM is an exhaustive algorithm that usually generates a wide range of morphological solutions for a given input word. Only one of these solutions is valid in the given context. SWAM considers two levels of context:

The morpheme level context: the training corpus is used to compute unigram and bigram probabilities at the morpheme level including prefix-root bigrams, root-suffix bigrams, prefix-pattern bigrams, pattern-suffix bigrams …etc.
The following formula is used to compute the score for a resulting segmentation:

Score =(Pr(Root | Prefix))/Pr(Prefix)) * (Pr(Suffix | Root)/Pr(Root)) * (Pr(Pattern | Prefix))/Pr(Prefix)) * (Pr(Suffix | Pattern)/Pr(Pattern))

A similar formula is used to compute the score for segmentations that involve non-templatic words or particles.

The sentence level context: the training corpus has been used to compute unigram and bigram probabilities for all words in the corpus in a manner that makes possible to use a variation of Bayes formula to capture the sentence level context.

Table 3: A Sample SWAM Analysis: the word /nšArħ/ نشارة

Word	Resulting Segmentation				Surface Pattern	Probability
	Prefix	Root	Suffix	Stem		
/nšArħ/ نشارة		/nšr/ نشر		/nšArħ/ نشارة	/fçعالةAlħ/	0.23871
/nšArħ/ نشارة	ن /n/	/šwr/ شور		/šArħ/ شارة	فعلة /fçlħ/	9.2431 E-8
/nšArħ/ نشارة	ن /n/	/šrý/ شرى		/šArħ/ شارة	فاعة /fAçh/	2.976 E-9
/nšArħ/ نشارة	ن /n/	/šr/ ش رر		/šArħ/ شارة	/فاعةfAçh/	1.7813 -9

3.5 The Viterbi Component:

The SWAM algorithm proved to be very effective in identifying morphological tags in the first phase. However, the first phase does not perform diacritization. It is also expected that some ambiguities still remain to be resolved. To solve these problems, SWAM applies classical Viterbi on a limited scale.

Using the bigram probabilities for deep patterns, Viterbi is used to identify the corresponding deep patterns as follows:

- The algorithm handles the text sentence by sentence. Assume that the sentence consists of N words.
- Assume that each word in the sentence has already been assigned a surface tag by SWAM (either a pattern or nontemplatic tag).
- The algorithm retrieves all possible deep patterns corresponding to the assigned surface tag and store them in an array (*TagsForWord array*) with columns corresponding to words and rows corresponding to possible deep patterns.
- Create another two arrays with same dimensions: The *Score array* in which we store the maximum scores for tags calculated while the algorithm transitions from one state to another. The other array is the *BackPointer array* in which we store the index of the tag that achieved the highest score.

The algorithm may be summarized as follows:

Prob(Tag j | φ) = Bigram(φ, TAG j) / UNIGRAM(φ) = count(φ,TAG j) /count(φ)
Prob(word1| Tagi) =bigram(tagj , word1) / unigram(tagj) = count(tagj , word1) /count(tagj)

WHILE J <= M-1
*Score(j, 0) = Bigram (φ ,T ag $_j$)/Unigram(φ) * Bigram(T ag $_j$, W ord$_0$)/Unigram(T ag $_j$)*
BackPointer(i, 0) = 0

For i = 1 to N-1
WHILE j <= M-1
*Score(j, i) = M AX$_{k=1,j}$ (Score(j, i-1) * Bigram(T agk, Tagj)/ Unigram(T agk)) * Bigram(T ag j , W ordi)/Unigram(T ag j)*
BackPointer(j, i) = index of tag with max score
sequence (N) = j that maximizes Score(j,N)
for i = N -2 to 0
sequence (i) = BackPointer(sequence (i+1),i+1)

3.6 Accuracy:

This section provides some information regarding the performance of the algorithm. The following tables provide some indicators regarding the accuracy of the algorithm in

identifying the correct morphological segmentation and picking the correct pattern tag. Five test samples (S1 to S5) have been randomly picked from the web.

In the tables below the accuracy at level N is simply computed as:

$$Accuracy - N = \frac{\text{Number of Valid Segmentations in Top N Solutions}}{\text{Number of Words Yeilded to Analysis}}$$

Where N is the number of top solutions in the ranked list of solutions.

Table 4: The Valid Segmentation is the Top Solution

Valid Segmentation is the Top Solution						
Test Samples	S1	S2	S3	S4	S5	Total
Number of distinct words	992	357	315	217	135	2016
Number of words yielded to analysis	983	351	310	210	132	1986
Valid segmentation is the top	963	340	299	202	128	1932
Excluded words	9	6	5	7	3	30
Accuracy	97.8	96.9	96.5	96.1	97	Av=97.3

Table 5: The Correct Solution is in one of the Top Two Solutions

Valid Segmentation is in the Top 2 Solutions						
Test Samples	S1	S2	S3	S4	S5	Total
Number of distinct words	992	357	315	217	135	2016
Number of words yielded to analysis	983	351	310	210	132	1986
Valid Segmentations in Top 2	974	343	305	210	132	1964
Excluded words	9	6	5	7	3	30
Accuracy	99	97.7	98.4	100	100	Av=98.9

Table 6: The Valid segmentation is in the Top Five Solutions

Valid Segmentation is the Top Solution						
Test Samples	S1	S2	S3	S4	S5	Total
Number of distinct words	992	357	315	217	135	2016
Number of words yielded to analysis	983	351	310	210	132	1986
Valid Segmentations in Top 5	979	348	309	210	132	1978
Excluded words	9	6	5	7	3	30
Accuracy	99.5	99.1	99.7	100	100	Av=99.2

Table 7: A Summary for all Samples

All words in Top1, Top2 and Top5	Top1	Top2	Top5
Number of distinct words	2016		
Number of words yielded to analysis	1986		
Valid solutions	1932	1964	1978
Bad solutions	54	22	8
number of word has no solutions	30	30	30
Accuracy	97.3	98.9	99.2

4 Conclusion

SWAM is a promising Arabic Morphological tagger with accuracy ranging from 97.3 to 99.2. It is interesting to note that the levels of accuracy reported in tables (6) and (7) above has been achieved using only morpheme level statistics. SWAM supports flexible multilevel tagging without imposing a restricted tag set. The algorithm can be improved in a variety of ways, including the addition of machine learning engine that captures the generated solutions. The composite Tag Charts require careful revision in terms of design and content. These charts may be equipped with built in logic that resolves some expected residual ambiguities. Another advantage of these Tag Charts is the inherent derivational power that clearly adds a "generative" dimension to SWAM.

5 References:

[1] Kristina Toutanova, Dan Klein, Christopher Manning, and Yoram Singer. 2003. Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network. In *Proceedings of HLT-NAACL 2003*, pp. 252-25

[2] Majdi Sawalha & Eric Atwell 2010 Fine-Grain Morphological Analyzer & Part-of-Speech Tagger for Arabic Text, Language Resources and Evaluation Conference, LREC

[3] Mohammed A. ELAffendi, "An LVQ connectionist solution to the non-determinacy problem in Arabic morphological analysis: a learning hybrid algorithm"**,** *Natural Language Engineering, Volume 8 Issue 1, March 2002, Pages 3 - 23*

[4] Mohammed A ELAffendi "A Suggested computational framework for Arabic Morphological Analysis' A Sliding Window Asymmetric Matching (SWAM) Algorithm and its Implications", *Egyptian Informatics Journal, Vol 9, No 1, 2008*

[5] Mona Diab 2007. Towards an Optimal POS tag set for Modern Standard Arabic Processing. In *Proceedings of Recent Advances in Natural Language Processing (RANLP)*, Borovets, Bulgaria

[6] Mona Diab, Kadri Hacioglu, and Daniel Jurafsky 2004 . Automatic Tagging of Arabic Text: From Raw Text to Base Phrase Chunks. In *Proceedings of the 5th Meeting of the North American Chapter of the Association for Computational Linguistics/Human Language Technologies Conference (HLT-NAACL04)*, pages 149–152, Boston, MA

[7] Mona Diab, Mahmoud Ghoneim, and Nizar Habash 2007. Arabic Diacritization in the Context of Statistical Machine Translation. In *Proceedings of Machine Translation Summit (MT-Summit)*, Copenhagen, Denmark

[8] N. Habash, O. Rambow and R. Roth. 2009, MADA+TOKAN: A Toolkit for Arabic Tokenization, Diacritization, Morphological Disambiguation, POS Tagging, Stemming and Lemmatization. In Proceedings of the 2nd International Conference on Arabic Language Resources and Tools(MEDAR), Cairo, Egypt.

[9] Nizar Habash 2010 *Introduction to Arabic Natural Language Processing*, Morgan & Claypool, San Rafael, California, USA

[10] Nizar Habash and Owen Rambow. Arabic Tokenization, Part-of-Speech Tagging and Morphological Disambiguation in One Fell Swoop. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 573–580, Ann Arbor, Michigan, June 2005. Association for Computational Linguistics.

[11] Roth, Ryan, Rambow Owen, Habash Nizar, Diab Mona, and Rudin Cynthia 2008 Arabic Morphological Tagging, Diacritization, and Lemmatization Using Lexeme Models and Feature Ranking, ACL-08:HLT, 06/2008, Columbus, Ohio, p.117-120, (2008)

[12] Shereen Khoja, Roger Garside, and Gerry Knowles. A tag set for the morphosyntactic tagging of Arabic. In *Proceedings of Corpus Linguistics 2001*, pages 341–353, Lancaster, UK, 2001

[13] Shereen Khoja. APT: Arabic Part-of-Speech Tagger. In *Proceedings of Student Research Workshop at NAACL 2001*, pages 20–26, Pittsburgh, 2001. Association for Computational Linguistics.

Malware Detection with Computational Intelligence

Brittany Byrd, Rakesh Malik, Venkata Kandalam and Qingzhong Liu
Department of Computer Science
Sam Houston State University, Huntsville, TX 77341, USA
Emails: { bas050, rxm058, vnk001, liu} @shsu.edu

Abstract - *This work seeks to detect malware with learning classifiers. In this study, three learning classifiers: fisher linear classifier, automatic neural network classifier, and support vector machine are utilized for the analysis and detection of malicious code within a given data set. To achieve this goal we collected and investigated the sequential component of API calls native to the Windows 32-bit host operating system using API Monitor v2 and general sandboxing techniques. The findings indicate that implementation of a support vector machine algorithm (sustaining a 94.9% detection rate) to conclude the malicious nature of an executable is ideal in behavior-based malware detection.*

Keywords: malware detection, API system calls, learning classifiers

1 Introduction

The anonymity of the digital world provides an invisibility cloak for cybercriminals to lurk behind undetected by Internet users worldwide. In the past decade, the incentive for malware development and distribution by malware writers has shifted from mere bragging rights or proof of concept, to economical profit (e.g., trade of bank account information, theft of credit card numbers, the collection and sale of sensitive user information, etc.). Motivated by monetary gain, cybercriminals are increasingly encouraged to employ more advanced and persistent modes of evasion to undermine malware detection strategies and tools.

The prevalence of novel malware and its variants have inseminated the cyber-world. Created at an alarming rate of millions per day, the exploitation of automation and obfuscation techniques enable cybercriminals to generate new malware more rapidly than analysts can develop detection signatures [1]. Malware authors seek to exploit the vulnerabilities of the cyber-world via copious avenues (e.g., malware can be propagated through web, e-mail, and human-transported media) through the implementation of varying tactics (e.g., executable binary file, script, and active content). Researchers unanimously agree that new detection methods are crucial in the battle against malware identification, detection and eradication – preferably fast and accurate methods that also use automation techniques and output a low false positive rate.

Created from merging the terms 'malicious' and 'software', malware is generally defined as software that performs varying nefarious actions on host systems (e.g., computer and/or mobile devices) without the authorized consent of the host's owner [2]. Implementation of the malicious code continuously poses an imminent threat to our society's capabilities to safeguard both the security and privacy of cyber users.

The purpose of this work seeks to determine via verifiable methods the detection capabilities of three learning classifiers: 1. fisher linear classifier, 2. automatic neural network classifier, and 3. support vector machine – to detect malware in executable files via segment analysis using inputted Microsoft Windows (32-bit operating system) API system calls derived from both the malicious and benign executables. Testing implemented in this work, aims to demonstrate the detection ability of the constructed testing algorithm, based on the analysis of the collected data set outputted by the executable monitoring software, API Monitor v2 created by Rohitab.com [3].

To achieve this task, we collected hundreds of malicious and benign program samples from VX Heaven (a malware collection library website) and the local Windows 32-bit host directory. Analysis of the samples were conducted using the API Monitor v2 tool to extract an output of API system calls, as input into the malware detection algorithm, as a distributed framework for automatic detection.

We explored the malware detection capabilities of varying learning classifiers, as proof of concepts for determining whether a program is malicious or benign through the analysis of API system calls, specific to the Windows 32-bit operating systems.

The remainder of this paper is organized as follows: Section 2 presents prior and related work. Section 3 discusses implementation of the methodology, including an explanation of the data set, the intricate testing procedures, and the results. Lastly, Section 4 we briefly conclude and present future implications.

2 Related Works

This section discusses the contributions of prior and related work, regarding the developmental effort towards malware-detection methodologies that demonstrate advancement against sophisticated malware evasion strategies.

Current research explicably demonstrates that signature-based detection techniques are an ineffective detection tools against advanced stealth and obfuscation techniques. Signature-based detection use a pattern matching schema (signatures or scan strings) to identify malware [8 – how antivirus]. Prominent among commercial malware detectors (e.g., virus scanners), this exhaustive human-error prone reactive approach only proves effective against known malware – and is dependent on providing detection results after infection, due to its emphasis on syntactic(specific characteristics of individual malware instances) as a complete representation of all malware development and complexities. Hence, the practical shift towards behavior-based detection techniques that offer analysts the opportunity to create detection methods that instead target the semantics (general behavior specifications exhibited by an entire family of malicious code) to detect malware [6,8,9].

There is a large body of work in the area of behavior-based malware detection. We focus on the areas most related to our work.

Zhou & Inge [8] presented a malware detection technique that categorizes a program as malware, according a learning engine created by the authors which uses an adaptive data compression model. Our work seeks to analyze malicious code, in contrast to benign executables (native to the Windows 32-bit host operating system). We focused on how each program subset interacted with the Windows API system calls, specifically which DLLs were called and in what order. Similarly, Christodorescu et al. [9] proposed a malware detection algorithm that undermines the weaknesses associated with the use of signature-based detection by incorporating instruction semantics to detect malicious program traits. Their contribution demonstrated that the algorithm can detect all variants of certain malware, has no false positives, and is resilient to obfuscation. Fredrikson et al. [1] also implemented a novel algorithm called HOLMES that demonstrated a detection accuracy of 86% for new unknown malware, with 0 false positives. Their work further elaborated on the effectiveness of behavior-based detection methods overall, in contrast to commercial signature-based anti-virus products.

In contrast, Kolbitsch et al. [10] only focused on the dissection of one malware sample to create a novel malware detection approach. They analyzed the intimate structure and characteristics of the malicious program – later using the analysis results to contrast and test against the runtime behavior of unknown programs to demonstrate the efficiency of their malware detection approach to detect running malicious code on a host system with small overhead. Our data set consisted of the collection of hundreds of benign and malicious executables, wherein our data detection learning strategies were tested via static analysis (e.g., without preprocessing, such as unpacking or disassembling) according to their ability to accurately determine whether a program was malicious based on data analyzed from our sample collection.

3 Detection Method and Experiments

To demonstrate the efficiency of an automated malware detection algorithm, we collected and analyzed the Windows API system calls of a 32-bit host operating system called by both malicious and benign programs.

3.1. Data Set

We collected real malware executable files, specifically known to target the Windows 32-bit operating system, from VX Heaven. VX Heaven is a library collection website that provides a vast selection of known malware source codes that are widely used as an experimental data set [8]. We downloaded and analyzed 200 malware scripts of multiple categories (e.g., backdoor, worm, Trojan, etc.) for this study. These files converted into executables within the Windows platform, form the malware data set.

160 benign executable programs (native to the Windows 32-bit local host operating systems) were retrieved via the directory path C:\Windows\System32. These collected samples were used to perform contrast analysis, in comparison to the malicious files to create detection dependency graphs as input for our detection algorithm. Following, the collection of the sample data set – the execution process of each program was monitored and collected via the API Monitor tool. Our malware samples are derived from varying malicious codes, collectively grouped under the common term *malware*. Hence, the malware used in our data set can be categorized into six distinct subsets, according to their actions [4,5,6,7]: virus, worms, Trojan, rootkit, distributed denial of service, and backdoor.

3.2. API System Call Extraction

We utilized the API Monitor v2 monitoring tool [3], to construct API system call detection tables as input for the learning classifiers. API system calls used by the both the malicious and benign executables, were individually inputted into the detection tables and numbered in order of call sequence, according to each executable collected.

Collection of the API system calls (specifically the malicious samples) were executed under delicate conditions via dynamic analysis, to appropriately analyze each malicious sample. To achieve this task, the malware must be executed in a realistic environment, while ensuring that any potential for leaks into the live network are removed. For this, we used *Oracle VirutalBox* virtualization tool as a sandbox to download and run the malware for monitoring and API system call capture. After collecting the API system calls, we constructed detection tables based on the extracted segmented features. Figure 1 shows the output via API Monitor v2 of the API system calls monitored and detached by the malware sample, named Backdoor.Win32.711.). Figure 2 displays a detection table of the API system calls outputted by *API Monitor v2*, in correspondence with the specified monitored malware samples.

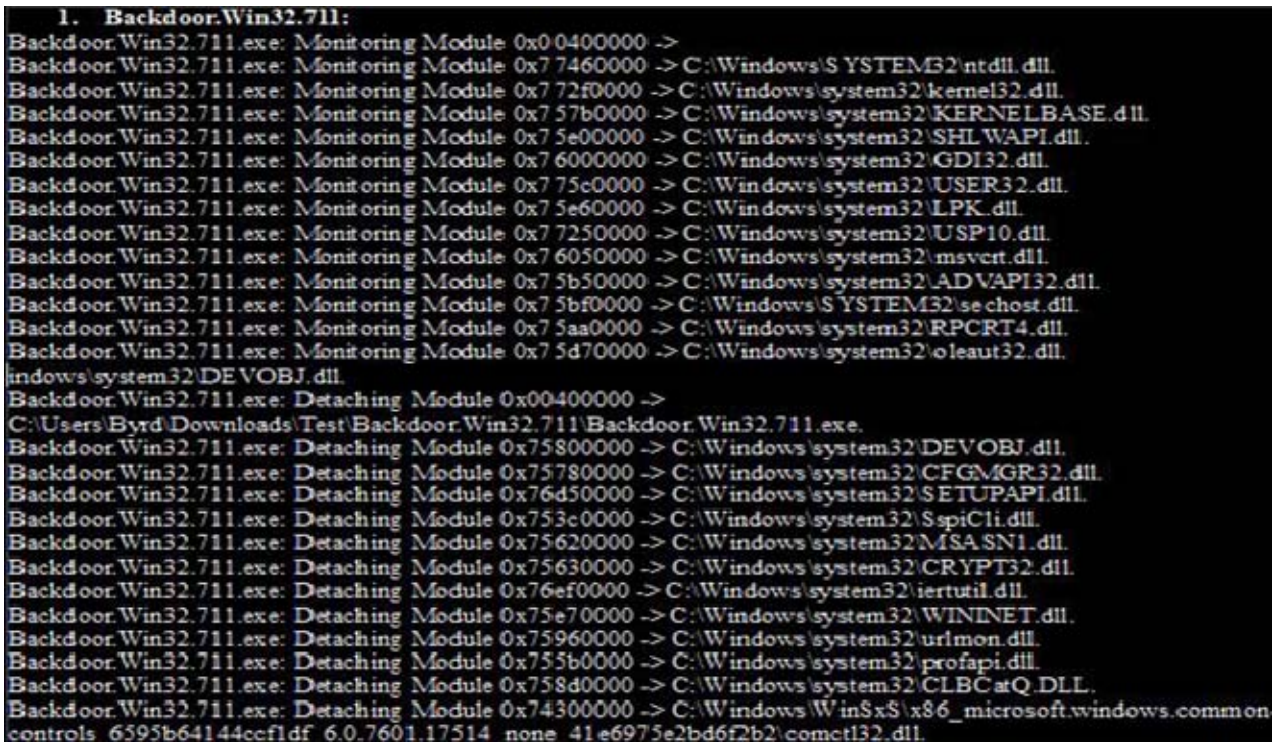


Fig. 1. *API Monitor v2 monitoring output of the malicious executable* ***Backdoor.Win32.711***

	A	B	C	D	E	F	G	H	I	J	K
1	Executable Name	ntdll.dll	kernel32.dll	KERNELBASE.dll	SHLWAPI.dll	GDI32.dll	USER32.dll	LPK.dll	USP10.dll	msvcrt.dll	ADVAPI32.dll
2	Backdoor.Win32.711	1	2	3	4	5	6	7	8	9	10
3	Backdoor.Win32.Acidoor.11	1	2	3	4	5	6	7	8	9	10
4	Backdoor.Win32.Acidsena	1	2	3	4	5	6	7	8	9	10
5	Backdoor.Win32.Aeon.10	1	2	3	4	5	6	7	8	9	10
6	Backdoor.Win32.Aibolit.40	1	2	3	4	5	6	7	8	9	10
7	Backdoor.Win32.Akcom.11	1	2	3	4	5	6	7	8	9	10
8	Backdoor.Win32.Alerter	1	2	3	4	5	6	7	8	9	10
9	Backdoor.Win32.Alfdoor	1	2	3	4	5	6	7	8	9	10
10	Backdoor.Win32.BrainWiper.03	1	2	3	4	5	6	7	8	9	10
11	Backdoor.Win32.Broser	1	2	3	4	5	6	7	8	9	10
12	Backdoor.Win32.Buxtehude	1	2	3	4	5	6	7	8	9	10
13	Backdoor.Win32.Canvas.10	1	2	3	4	5	6	7	8	9	10
14	Backdoor.Win32.CFour	1	2	3	4	5	6	7	8	9	10
15	Backdoor.Win32.Chat	1	2	3	4	5	6	7	8	9	10
16	Backdoor.Win32.Cheeser	1	2	3	4	5	6	7	8	9	10
17	Backdoor.Win32.Buterat.q	1	2	3	4	5	6	7	8	9	10
18	Backdoor.Win32.Chubo	1	2	3	4	5	6	7	8	9	10
19	Backdoor.Win32.Click	1	2	3	4	5	6	7	8	9	10
20	Backdoor.Win32.ColdDeath	1	2	3	4	5	6	7	8	9	10
21	Backdoor.Win32.CRS-Gate	1	2	3	4	5	6	7	8	9	10
22	Backdoor.Win32.DarkPortal	1	2	3	4	5	6	7	8	9	10
23	Backdoor.Win32.DeathCorner.12	1	2	3	4	5	6	7	8	9	10
24	Backdoor.Win32.Dude	1	2	3	4	5	6	7	8	9	10
25	Backdoor.Win32.EzKilla	1	2	3	4	5	6	7	8	9	10
26	Executable Name	ntdll.dll	kernel32.dll	KERNELBASE.dll	SHLWAPI.dll	GDI32.dll	USER32.dll	LPK.dll	USP10.dll	msvcrt.dll	ADVAPI32.dll
27	Backdoor.Win32.Fakemanga.a	1	2	3	4	5	6	7	8	9	10
28	Backdoor.Win32.Gacho	1	2	3	4	5	6	7	8	9	10
29	Backdoor.Win32.Grab	1	2	3	4	5	6	7	8	9	10
30	Backdoor.Win32.Hackboy.10	1	2	3	4	5	6	7	8	9	10
31	Backdoor.Win32.Ident	1	2	3	4	5	6	7	8	9	10
32	Backdoor.Win32.Kneel	1	2	3	4	5	6	7	8	9	10
33	Backdoor.Win32.NetTerrorist	1	2	3	4	5	6	7	8	9	10
34	Backdoor.Win32.Parasite	1	2	3	4	5	6	7	8	9	10
35	Backdoor.Win32.Quimera	1	2	3	4	5	6	7	8	9	10
36	Constructor.Win32.DateChanger.a	1	2	3	4	5	6	7	8	9	10
37	Worm.Win32.AimVen	1	2	3	4	5	6	7	8	9	10
38	Trojan-Spy.Win32.MicroLog	1	2	3	4	5	6	7	8	9	10
39	Trojan-Spy.Win32.KeyHunter	1	2	3	4	5	6	7	8	9	10
40	Trojan-Spy.Win32.DakrOmen.13	1	2	3	4	5	6	7	8	9	10
41	Trojan-Spy.Win32.Cracker	1	2	3	4	5	6	7	8	9	10
42	Trojan-Spy.Win32.AGM	1	2	3	4	9	5	6	7	10	11
43	Trojan-Spy.Win32.AdvKeyLogger	1	2	8	5	9	10	11	12	13	14
44	Trojan-Ransom.Win32.Vb.d	1	2	3	4	5	6	7	8	9	10

Fig. 2. Detection table of collected malware sample API system calls from API Monitor v2 tool output.

Applications that originated from the same development platform tend to share the same libraries and resources. As a result, malware and benign programs generated using the same development platform (e.g., third-party software companies) can resemble each other through the observation of their execution process regarding the use of API system calls [11].

3.3 Learning Classifiers

To determine whether a program is classified as a malicious executable, the extracted API system calls were inputted into learning classifiers, as segmented classification features.

Classifiers enable *statistical classification* (the use of an object's characteristics to identify which class the object belongs to) by evoking a *classification decision* premised on the object's characteristics combination value.

We used three learning classifiers: fisher linear classifier [12]; automatic neural network classifier [13, 15], and support vector machine [14], to perform our malware detection approach by testing and contrasting the detection rate capabilities of each.

3.4 Experimental Results and Analysis

The results indicate that by extracting meaningful features, such as the API system calls from within each different subset (malicious and benign), malware detection is possible via learning classifiers.

In each experiment, 50% benign and 50% malicious of the executables were randomly selected from the data set were used to create the training data set. The other 50% benign and 50% malicious from the data set were used for the testing (prediction) data set – with a mean accuracy of over ***100*** experiments. The results are ***83.2%*** by using fisher linear classifier, ***85.7%*** by using automatic neural network classifier and ***94.9%*** by using support vector machine with a linear kernel.

Implementation of the three learning classifiers to our data set indicates that each strategy is capable of distinguishing between malicious and benign applications via the extraction and input of API system calls as determinants. However, it is also evident that use of the support vector machine is the more efficient approach.

The results further support the notion that measureable and identifiable differences exist between malicious and benign executables. Hence, deeper investigation into these instances can provide analysts in the development of better detection approaches in the future.

4 Conclusion

Our study shows that it is effective for detecting malicious code by analyzing the Windows 32-bit API system calls with the aid of learning maching techniques.

Our future implications aspire to generate work that will further explore malware detection efforts, from the perspective of algorithm modification (specifically the HOLMES algorithm). This current paper will serve as a platform to our future endeavor that seeks to incorporate the concept of infection markers (detectors used by malware to avoid infecting the same host system more than one time) into undermining the effects of malicious code on Windows Registry keys.

Additionally, other areas of malware analysis using raw features extracted from Windows executables (such as, PE header information, security labels, and instruction code sequences, assembly language mnemonics) are practical options for data set derivation for behavior-based malware detection efforts).

5 References

[1] M. Fredrikson, et al., "Synthesizing Near-Optimal Malware Specifications from Suspicious Behaviors," in *2010 IEEE Symposium on Security and Privacy*. doi: [10.1109/SP.2010.11], pp. 45-60.

[2] A. Wichmann and E. Gerhards-Padilla, "Using Infection Markers as a Vaccine against Malware Attacks," in *2012 IEEE International Conference on Green Computing & Communications (GreenCom).* doi: [10.1109/GreenCom.2012], 2012, pp. 737-742.

[3] API Monitor. (2012). *Rohitab Batra.* [Online]. Available: http://www.rohitab.com/apimonitor

[4] M. Kassner. (2009, July). *The 10 faces of computer malware.* [Online]. Available. http://www.techrepublic.com/blog/10-things/the-10-faces-of-computer-malware/

[5] J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.

[6] M.Christodorescu et al., "Mining Specifications of Malicious Behavior," in *Proceedings of the 6th Joint Meeting of the European Software Engineering Conference and the ACM SIGSOFT Symposium on the Foundations of Software Engineering (ESEC/FSE 2007).* Dubrovnik, Croatia, 2007.

[7] D. Maslennikov. (2011, March 22). *Mobile Malware Evolution: An Overview, Part 4.* [Online]. Available: http://www.securelist.com/en/analysis/204792168/Mobile_Malware_Evolution_An_Overview_Part_4

[8] Y. Zhou and M. Inge, "Malware Detection Using Adaptive Data Compression," in *AlSec '08.* ACM 978-1-60558-291/08/10, Alexandria, VA, 2008, pp. 53-59.

[9] M.Christodorescu, et al., "Semantics-Aware Malware Detection," Department of Electrical & Computer Engineering, Carnegie Institute of Technology, 2005.

[10] C. Kolbitsch, et al., "Effective and Efficient Malware Detection at the End Host," in *USENIX Security Symposium*, 2009, pp. 351-366.

[11] G. Tahan, et al. "Mal-ID: Automatic Malware Detection Using Common Segment Analysis and Meta-Features," *Journal of Machine Learning Research 13*, pp. 949-979, April 2012.

[12] J. Stock., "A Note of Linear Classifiers," Department of Economics, Harvard University, 2003.

[13] G. Dahl, et al. "Large-Scale Malware Classification Using Random Projections and Neural Networks," in *ICASSP*, 2013.

[14] D. Gavrilut, et al. "Malware Detection Using Perceptronas and Support Vector Machines," in *ComputationWorld '09* doi [10.1109/ComputationWorld.2009.85], 2009, pp. 283-288.

[15] "Method of setting optimum-partitioned classified neural network and method and apparatus for automatic labeling using optimum-partitioned classified neural network," U.S. Patent EP 1453037 A2, Sep 1, 2004.

Hierarchical Artificial Bee Colony Optimizer for Multilevel Threshold Image Segmentation

Maowe He[1,2], **Hanning Chen**[2,*], **and Kunyuan Hu**[2]
[1] Department of Information Service & Intelligent Control, Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China
[2] University of Chinese Academy of Sciences, Beijing 100039, China

Abstract - *This paper presents a novel optimization algorithm, namely hierarchical artificial bee colony optimization (HABC) for multilevel threshold image segmentation, which employs a pool of optimal foraging strategies to extends the classical artificial bee colony framework to a cooperative and hierarchical fashion. In the proposed hierarchical model, the higher-level species incorporates the enhanced information change mechanism based on crossover operator to enhance the global search ability between species. In the bottom level, with the divide-and-conquer approach, each subpopulation runs the original ABC method in parallel for part-dimensional optimum, which can be aggregated into a complete solution for the upper level. The experimental results on a set of benchmarks demonstrated the effectiveness of the proposed algorithm. Furthermore, we applied the HABC algorithm to the multilevel image segmentation problem. Experimental results of the new algorithm on a variety of images demonstrated the performance superiority of the proposed algorithm*.

Keywords: Hierarchical cooperative optimization; Divide-and-conquer; Crossover; Image Segmentation

1 Introduction

Image segmentation is considered as the process of partitioning a digital image into multiple regions or objects. Among the existing segmentation methods, multilevel threshold technique is a simple but effective tool that can extract several distinct objects from the background [1]. This means that searching the optimal multiple thresholds to ensure the desired thresholded classes is a significantly essential issue. Several multiple thresholds selection approaches have been proposed in literatures: ref [2] proposed some methods derived from optimizing an objective function, which were originally developed for bi-level threshold and later extended to multilevel threshold [1,2]. However, these methods based on exhaustive search suffer from a common drawback, that is, the computational complexity will rise exponentially when extended to multilevel threshold from bi-level.

Recently, to address this concerned issue, swarm intelligence (SI) algorithms as the powerful optimization tools have been introduced to the field of image segmentation owing to their predominant abilities of coping with complex nonlinear optimizations. Akay [3] employed two successful population-based algorithms-particle swarm optimization (PSO) and artificial bee colony algorithm (ABC) to speed up the multilevel thresholds optimization. Ozturk et al. [4] proposed an image clustering approach based on ABC to find the well-distinguished clusters. Sarkar and Das [5] presented a 2D histogram based multilevel thresholding approach to improve the separation between objects, which demonstrated its high effectiveness and efficiency.

Comparing with the huge in-depth studies of other evolutionary and swarm intelligence algorithms, how to improve the diversity of swarm or overcome the local convergence of ABC is still a challenging issue in the optimization domain. Thus, this paper presents a novel hierarchical optimization scheme based on multi-pattern and cooperative evolutionary strategies, namely HABC, to extend ABC framework from flat (one level) to hierarchical (multiple levels). We should note that similar hierarchical schemes have been incorporated into some intelligent algorithms. The essential difference between our proposed scheme and others can be abstracted as follows:(1) The divide-and-conquer strategy with random grouping is incorporated in this hierarchical framework, which can decompose the complex high-dimensional vectors into several smaller part-dimensional components that assigned to the lower level. This can enhance the local search ability (exploitation). (2) The enhanced information exchange strategy, namely crossover, is applied to interaction of different population to maintain the diversity of population. In this case, the neighbor bees with higher fitness can be chosen to crossover, which effectively enhances the global search ability and convergence speed to the global best solution (exploration).Through these significantly effective strategies, HABC can accommodate considerable potential for solving more complex problems.

The rest of the paper is organized as follows. In Section 2 the proposed hierarchical artificial bee colony (HABC) model is given. Section 3 presents the experimental studies of the proposed HABC and the other algorithms with descriptions of the involved benchmark functions, experimental settings, and experimental results. And its application to image segmentation has been presented in Section 4. Finally, Section 5 outlines the conclusion.

2 Hierarchical Artificial Bee Colony Algorithm

The HABC integrates a two-level hierarchical co-evolution scheme inspired by the concept and main ideas of multi-population co-evolution strategy and crossover operation. The flowchart of the HABC is shown in Fig. 1. It includes four key strategy approaches: variables decomposing approach, random grouping of variables, background vector calculating approach, and crossover operation, which is presented as follows.

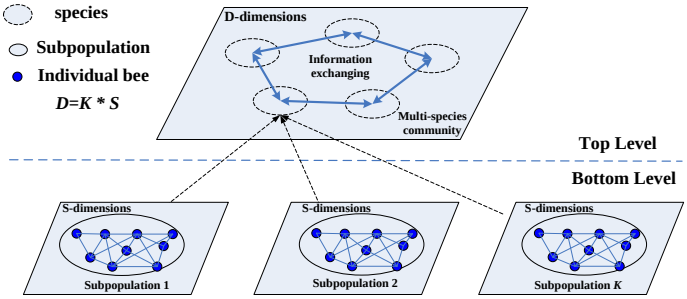


Fig.1 Hierarchical optimization model

2.1 Hierarchical multi-population optimization model

According to the canonical ABC algorithm [6], the new food source is produced by a perturbation coming from a random single dimension in a randomly chosen bee. This causes that an individual may have discovered some good dimensions, while the other individuals that follow this bee likely to choose worse vectors in *D* dimensions and abandon the good ones. On the other hand, when solving complex problems, single population based artificial bee algorithms suffer from the following drawback of premature convergence at the early generations.

Hence, the HABC contains two levels, namely the bottom level and top level, to balance exploring and exploiting ability. In Fig.1, in the bottom level, with the variables decomposing strategy, each subpopulation employs the canonical ABC method to search the part-dimensional optimum in parallel. In each iteration, *K* subpopulations in the bottom level generate K best solutions, which are constructed into a complete solution species that update to the top level. In the top level, the multi-species community adopts the information exchange mechanism based on crossover operator, by which each species can learn from its neighborhoods in a specific topology. The vectors decomposing strategy and information exchange crossover operator can be described in detail as follows.

2.2 Variables decomposing approach

The purpose of this approach is to obtain finer local search in single dimensions inspired by the divide-and-conquer approach. Notice that two aspects must be analyzed: (1) how to decompose the whole solution vector, and (2) how to calculate the fitness of each individual of each subpopulation. The detailed procedure is presented as follows:

Step1. The simplest grouping method is permitting a *D*-dimensional vector to be split into *K* subcomponents, each corresponding to a subpopulation of *s*-dimensions, with *M* individuals (where $D = K$*s). The *j*-th subpopulation is denoted as P_j, $j \in [1...K]$.

Step2. Construct complete evolving solution *Gbest*, which is the concatenation of the best subcomponents' solutions P_j by fowling:

$$Gbest = (P_1.g, P_2.g, P_j.g...P_K.g) \quad (1)$$

where $P_j.g$ represents the personal best solution of the *j*-th subpopulation.

Step3. For each component P_j, $j \in [1...K]$, do the following:

(a) *At employed bees' phase,* for each individual X_i, $i \in [1...M]$, Replace the *i*-th component of the *Gbest* by using the *i*-th component of individual X_i. Caculate the new solution fitness: $f(newGbest(P_1.g, P_2.g, X_i, ...P_k.g))$. If $f(newgbest) < f(Gbest)$, then G*best* is replaced by *newGbest*.

(b) Update X_i positions by using (1);

(c) At onlooker Bees' Phase, repeat (a)-(b)

Step 4. Memorize the best solution achieved so far, Compare the best solution with *Gbest* and memorize the better one.

● *Random grouping of variables*

To increase the probability of two interacting variables allocated to the same subcomponent, without assuming any prior knowledge of the problem, according to the random grouping of variables proposed by [7], we adopt the same random grouping scheme by dynamically changing group size. For example, for a problem of 50 dimensions, we can define that:

$$G = \{2,5,10,25,50\} \quad K \subset G \quad (2)$$

Here, if we randomly decompose the *D*-dimensional object vector into *K* subcomponents at each iteration (i.e., we construct each of the *K* subcomponents by randomly selecting S-dimensions from the *D*-dimensional object vector). The probability of placing two interacting variables into the same subcomponent becomes higher, over an increasing number of iterations.

2.3 The information exchange mechanism based on crossover operator between multi-species

In the top level, we adopt crossover operator with a specific topology to enhance the information exchange between species, in which each species P_j can learn from its symbiotic partner in the neighborhood. The key operations of this crossover procedure are described in Fig.2.

Step 1. Select elites for the best-performing list (BPL)

Individuals from current species P_j's neighborhood (i.e. ring topology) with higher fitness have larger probability to be selected into the best-performing list (BPL) as elites, whose size is equal with current population size.

Step 2. Crossover operation

Step2.1. Parents are selected from the BPL's elites using the tournament selection scheme: two enhanced elites are selected randomly, and their fitness values are compared to select the

elites. The one with better fitness is viewed as parent. Another parent is selected in the same way.

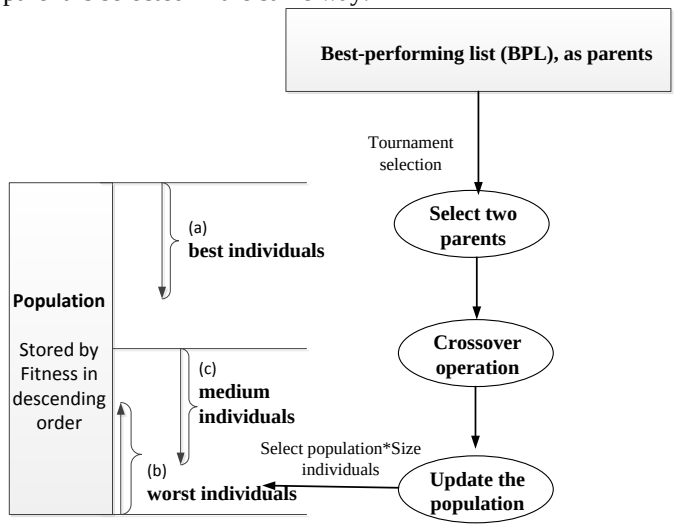


Fig.2. The information exchange mechanism based on crossover operation

Step2.2. Two offspring are created by arithmetic crossover on these selected parents as the following equation:

$$S_{new} = rand(0,1) \times parent1 + rand(0,1) \times parent2 \quad (3)$$

where S_{new} is newly produced offspring, *parent1* and *parent2* are randomly selected from BPL.

Step 3. Update with different selection schemes

If population size is *S*, then the replaced individuals number is *S*CR* (*CR* is a selecting rate).The greedy selection mechanism is adopted to replace the selected individual. There are three replacing approaches of selecting the best individuals (i.e. *S*CR* individuals), a medium level of individuals and the worst individuals. To maintain the population diversity, we randomly select one replacing approach at every iteration.

In summary, in order to facilitate the below presentation and test formulation, we define a unified parameters for HABC model in Table 1. According to the process description as mentioned above, the flowchart of HABC algorithm is summarized in Fig.3.

Table 1. Parameters of the HABC

$HABC=(N,M,P_{ij},C,CR,T,O,S)$	
N	The number of species in top level
K	Subpopulationr dividing *D*-dimensions into *S*-dimensions
M	Subpopulation size
S	Corresponding to a population of *S*-dimensions, where $S = D / K$
D	Dimensions of optimization problem
i	Top level population(species)'s ID counter from 1 to *N*
j	Button level population's ID counter from 1 to *K*
P_{ij}	The i^{th} population (of the j^{th} species)
CR	Selection rate for replacing the offspring to the selected individuals.
T	The hierarchical interaction topology of HABC
O	The objective optimization goals

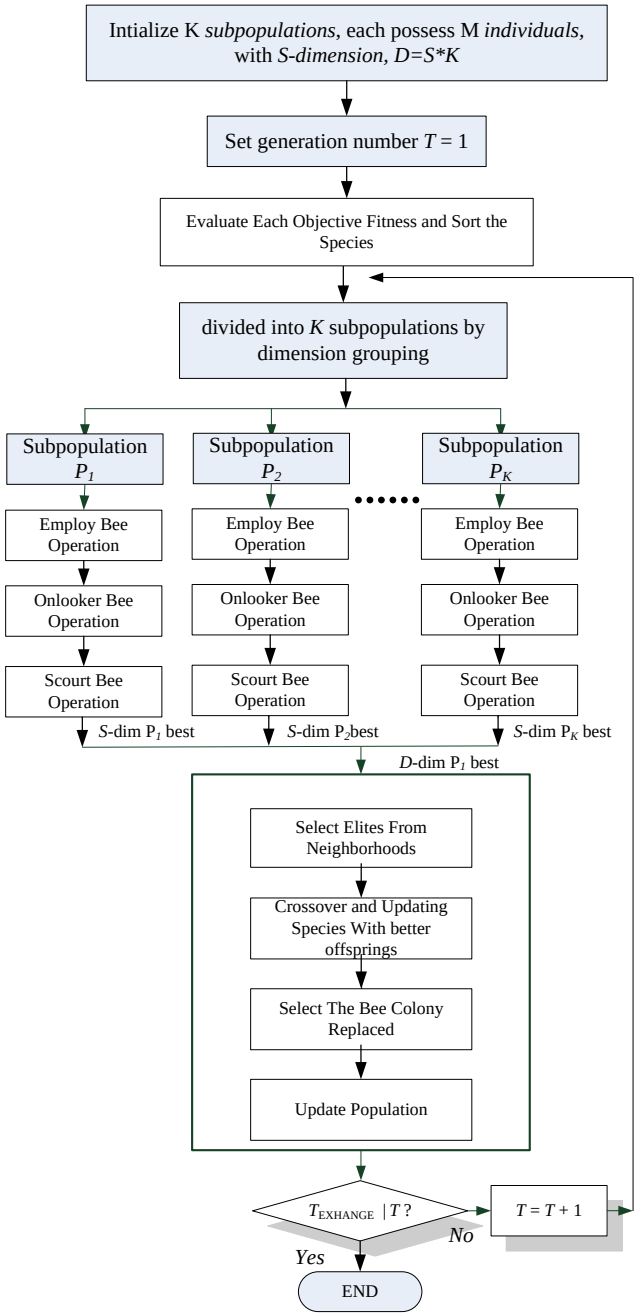


Fig.3. Flowchart of the HABC algorithm.

3 Experimental study

3.1 Experimental settings

A set of six basic benchmark functions (f_1-Sphere, f_2-Rosenbrock, f_3-Quadric, f_4-Sin, f_5-Rastrigrin and f_6-Schwefel) and four CEC2005 benchmark functions (f_7-Shifted Sphere, f_8-Shifted Schwefel's Problem 1.2, f_9-Shifted Rotated High Conditioned Elliptic Function and f_{10}-Shifted Schwefel's Problem 1.2 with Noise in Fitness) are employed to fully evaluate the performance of the HABC algorithm fairly. The number of function evaluations (FEs) is adopted as the time measure criterion substitute the number of iterations.

Eight variants of HABC based on different crossover methods and CR values were executed with six state-of-the-art EA and SI algorithms for comparisons: Artificial bee colony algorithm (ABC), Cooperative Artificial bee colony algorithm (CABC), Canonical PSO with constriction factor (PSO), Cooperative PSO (CPSO), Standard genetic algorithm (GA), and Covariance matrix adaptation evolution strategy (CMA-ES).

In all experiments, population size is set as 50 and the maximum evaluation number is set as 100000. For the ten continuous testing functions used in this paper, the dimensions are all set as 50. All the control parameters for the EA and SI algorithms are set to be default of their original literatures: initialization conditions of CMA-ES are the same as in [8], and the number of offspring candidate solutions generated per time step is $\lambda = 4\mu$; for ABC and CABC, the limit parameter is set to be $SN \times D$, where D is the dimension of the problem and SN is the number of employed bees. The split factor for CABC and CPSO is equal to the dimensions. For canonical PSO and CPSO, the learning rates c_1 and c_2 are both set as 2.05 and the constriction factor χ=0.729. For EGA, intermediate crossover rate of 0.8, gaussian mutation rate of 0.01, and the global elite operation with a rate of 0.06 are adopted.

3.2 Comparing HABC with other state-of-the-art algorithms on benchmark problems

The means and stand deviations obtained all involved algorithms on the 50-dimensional classical test suite for 30 runs were reported in Table 2. On the uni-modal basic benchmark functions ($f_1 - f_4$), from Table 2, HABC converged faster than all other algorithms. HABC was able to consistently find the minimum to functions f_1, f_2 and f_3 within 100000 FEs. On the multi-modal functions (f_5 - f_6), it is visible that HABC algorithms markedly outperforms other algorithms on most of these cases. This can be explained that the multi-population cooperative co-evolution strategy integrated by HABC and CABC enhance the local search ability, contributing to their better performances in the multimodal problems.

From the experimental results in terms of means and stand deviations of CEC2005 benchmarks ($f_7 - f_{10}$) tests, HABC outperformed CMA-ES on eight out of the twelve functions. CMA-ES also outperformed CABC on most functions. HABC can find the global optimum for f_7 and f_9 within 10000 FEs; this is due to the fact that HABC can balance the exploration and exploitation by decomposing high-dimensional problems, and using crossover operations to maintain its population diversity, which is a key contributing factor. On the other hand, CMA-ES converged extremely fast. However, CMA-ES converged very fast or tended to be trapped into local minima very quickly, especially on multimodal shifted and rotated functions. According to the rank values, the performance order of the algorithms involved is HABC > CMA-ES > CABC > ABC > CPSO > PSO>GA.

Table 2. Performance of all algorithms.

Func.		*HABC*	*CABC*	*CPSO*	*CMA-ES*	*ABC*	*PSO*	*GA*
f_1	Mean	0	1.01e-137	3.01e-014	0	9.94e-016	1.85e-009	4.03e-001
	Std	0	1.741e-3	2.45e-137	0	7.85e-016	1.01e-009	1.05e-001
	Rank	1	3	5	1	4	6	7
f_2	Mean	0	4.91	3.41	1.04e-007	1.83e-001	2.98e+001	3.47e+002
	Std	0	5.65	1.96	4.31e-002	1.84e-001	2.48e+001	8.01e+002
	Rank	1	5	4	2	3	6	7
f_3	Mean	0	3.89e+001	2.66e+001	0	3.33e+001	1.99e-003	7.45e+001
	Std	0	7.31e+001	4.71e+001	0	3.97	3.15e-003	3.44e+001
	Rank	1	6	4	1	5	3	7
f_4	Mean	3.02e-032	2.15e-032	4.97e-009	2.34e-032	8.07e-014	8.27e-001	9.10e-001
	Std	4.95e-032	4.47e-032	2.08e-008	8.75e-034	5.03e-014	5.01e-001	5.11e-001
	Rank	3	2	5	1	4	7	6
f_5	Mean	0	0	1.35	1.93e+001	6.11e-009	1.49e+001	3.92e+001
	Std	0	0	0.97	5.98	5.37e-009	1.19e+001	5.41
	Rank	1	1	4	6	3	5	7
f_6	Mean	2.19	3.10e+002	2.32e+003	5.03e+003	1.32e+001	5.14e+003	2.79e+002
	Std	4.99	1.38e+003	1.78e+002	6.08e+002	3.92e+001	6.04e+002	9.01e+001
	Rank	1	4	5	6	2	7	3
f_7	Mean	0	2.85e-013	3.57e+002	4.76e-014	3.65e-012	1.88e+001	7.05e+001
	Std	0	4.62e-014	3.78e+002	7.14e-014	2.35e-012	1.98e+001	4.07e+001
	Rank	1	3	7	2	4	5	6
f_8	Mean	2.34e+001	1.67e+003	4.86e+003	0	4.64e+003	2.14e+002	5.45e+003
	Std	3.86e+001	3.91e+002	2.24e+004	7.31e-014	4.16e+003	1.01e+002	2.50e+003
	Rank	2	4	6	1	5	3	7
f_9	Mean	0	4.04e-013	4.25e+002	3.24e-014	1.69e-011	3.55e+001	3.91e+001
	Std	0	6.00e-014	4.84e+002	2.83e-012	2.96e-012	6.10e+001	3.90e+001
	Rank	1	3	7	2	4	5	6
f_{10}	Mean	1.30e+003	4.07e+003	2.43e+004	7.65e-022	1.46e+004	5.18e+002	1.42e+004
	Std	6.77e+001	7.13e+002	4.42e+005	4.25e-012	4.58e+004	3.64e+002	8.19e+003
	Rank	2	4	6	1	5	3	7
Total Rank		14	35	53	23	39	50	63

4 Multilevel threshold for image segmentation by HABC

4.1 Entropy criterion based fitness measure

The Otsu multi-threshold entropy measure proposed by Otsu [1] has been popularly employed in determining whether the optimal threshold method can provide image segmentation with satisfactory results. Here, it is used as the objective function for the involved algorithms and its process can be described as follows:

Let the gray levels of a given image range over [*0, L-1*] and *h(i)* denote the occurrence of gray-level *i*.
Let

$$N=\sum_{i=0}^{L-1}h(i),\ P(i)=h(i)/N \ \text{ for } 0\le i\le L-1 \tag{4}$$

$$\text{Maximize } f(t)=w_0w_1(u_0-u_1)^2 \tag{5}$$

where

$$w_0=\sum_{i=0}^{t-1}P_i;u_0=\sum_{i=0}^{t=1}i\times P_i/w_0 \tag{6}$$

$$w_1=\sum_{i=t}^{L-1}P_i;u_1=\sum_{i=t}^{L=1}i\times P_i/w_1 \tag{7}$$

and the optimal threshold is the gray level that maximizes Eq. (5). Then Eq. (5) can also be written as:

$$f(t)=\delta^2-w_0\times\delta_0^{\ 2}-w_1\times\delta_1^{\ 2} \tag{8}$$

where w_0,w_1,u_0 and u_1 are the same as given in Eqs.(7) and (8), and

$$\delta_0=\sum_{i=0}^{t-1}(i-u_0)^2\times P_i/w_0,\ \delta_1=\sum_{i=t}^{L-1}(i-u_1)^2\times P_i/w_1$$

$$\delta_0=\sum_{i=0}^{L-1}(i-u)^2\times P_i/w,\ w=\sum_{i=0}^{L-1}P_i,\ u=\sum_{i=0}^{L=1}i\times P_i/w$$

Expanding this logic to multi-level threshold,

$$f_9(t)=w_0\times\delta_0^{\ 2}+w_1\times\delta_1^{\ 2}+w_2\times\delta_2^{\ 2}+.....+w_1\times\delta_N^{\ 2} \tag{9}$$

where *N* is the number of thresholds.
Eq.(9) is used as the objective function for the proposed HABC based procedure which is to be optimized (minimized). A close look into this equation will show it is very similar to the expression for uniformity measure. Here, the Probabilistic Rand Index (PRI) [9] is introduced as a qualitative measure. The value of PRI ranges from 0 to 1, where 0 means no similarities and 1 means all segmentations are identical.

4.2 Experiment setup

The experimental evaluations for segmentation performance by HABC are carried out on the Berkeley Segmentation Database (BSDS). The BSDS consists of 300 natural images, manually segmented by a number of different subjects. The ground-truth data for this large collection shows the diversity, yet high consistency, of human segmentation. These datasets involve a suit of popular standard images, namely 38092.jpg and 14037.jpg , both sizes of which are 321*481, with a unique grey-level histogram. A comparison between the proposed algorithm and other methods is evaluated based on Otsu, which means Eq. (9) is regarded as fitness function to evaluate all involved algorithms.

The numbers of thresholds M-1 investigated in the experiments were 2, 3, 4, 5, 7, and 9 while all the experiments were repeated 30 times for each image for each M-1 value. The population size is 20 and the maximum number of FEs is 2000. In all experiments, to ensure that the initial values of each random algorithm are equal, we use the MATLAB command rand(*X, Y*)*(*maxL-minL*)+*minL*, where *maxL* and *minL* are the maximum and minimum grays of the tested images, respectively. Fig.4 (a)&(b) and Fig. 5 (a)&(b) donate the original images and their histograms.

The basic parameter settings of these algorithms, namely HABC, ABC, PSO, EGA and CMA_ES, are set as default in Section 3.1. It is noteworthy that the split factor *K* of HABC should be adjusted according to the dimension of the image segmentation problems. In this experiment, for 2 dimensional problem, K ⊂ {1,2}; for 3 dimensional problem, K ⊂ {1, 3}; for 4 dimensional case, K ⊂ {1,2, 4}; for 5 dimensional 5, K ⊂ {1,5}; for 7dimensional case, K ⊂ {1,7};for 9 dimensional case, K ⊂ {1,3,9}. In order to test the effect of segmentation, the Probabilistic Rand Index (PRI) is chosen as a qualitative measure.

4.3 Experimental results of multilevel threshold

Table 3 presents the fitness function values, mean computation time, and corresponding optimal thresholds (with M-1 = 2, 3, 4) obtained by Otsu. It is noteworthy that the term CPU time is also an important issue in the real-time applications. From Table 4, there are not obvious differences about CPU times between the involved population-based methods, which are suitably significantly superior in terms of time complexity for high-dimensional image segmentation problems.

As can be seen form Table 5, the proposed HABC algorithm generally performs close to the Otsu method in term of fitness value when M-1=2,3,4, whereas the performance of HABC on time complexity is significantly superior to its counterpart Otsu. Furthermore, the HABC-based algorithm achieves the best performance among the population-based methods in most cases. This means that HABC can obtain an appropriate balance between exploration and exploitation. However, compared with the other population-based algorithms, the PRI results for low dimensional segmentation obtained by HABC have no obvious enhancement. This can be explained that HABC endowed with crossover operation performed better global search in the higher dimensional search space as well as the hierarchical cooperation strategy will be used to emphasize the fine exploitation around the promising area. Moreover, the differences between the HABC and the other algorithms are more evident as the segmentation level increases.

Regarding the high-dimensional segmentation problems with M-1=5, 7, 9, Table 5 demonstrates the average fitness

value, the standard deviation and the PRI obtained by each population-based algorithm, where the correlative results with the larger values, the smaller standard deviations or the higher PRIs, indicate the better achievement.

From Table 5, depending on the crossover method and the fast convergence rate, HABC demonstrates the best performance in terms of efficiency and stability on the high dimensional cases. Furthermore, as the level of segmentation increases, the fitness of HABC increases faster than that of other methods, especially ABC almost didn't achieve any improvement for 5, 7, and 9 levels of segmentation. From the experimental results of Tables 4 and 5, the PRI results for high dimensional segmentation are significantly better than that for the low dimensional cases. Meanwhile, HABC can also achieve better statistical results regarding PRI than its counterparts in most tested datasets. This can be explained that, based on Otsu method, the segmentation with more thresholds can achieve greater consistency to human segmentation and HABC algorithm can demonstrate powerful performance in searching high dimensional space. Figs.4 and 5 show the original images, multilevel threshold segmentations of those image and the ground truth human segmentations of those images. From these results as shown in Figs.4 and 5, it is clearly visible that the HABC-based method is more suitable to deal with such multilevel segmentation problem.

Table 3. Objective values and thresholds by the Otsu method and their PRI

Image	M-1=2			M-1=3			M-1=4		
	Objective values	Optimal thresholds	PRI	Objective values	Optimal thresholds	PRI	Objective values	Optimal thresholds	PRI
38092	2.5064E4	101,180	0.7425	2.5258E4	77,132,190	0.7860	25345.4	64,111,152,197	0.7994
14037	7.5361E3	61,128	0.7361	7.6593E3	33,77,131	0.8019	7.7151E4	33,73,109,147	0.7997
24077	2.1133E4	81,175,	0.6821	2.1411E4	66,126,199	0.7398	2.1499E4	63,112,163,217	0.7637
CPU time		1.4427			61.421			2644.543	

Table 4. The mean CPU time of the compared population-based methods on Otsu algorithm.

Dim	HABC	ABC	PSO	CMA-ES	EGA
2 Dim	0.8575	0.7954	0.2126	0.2438	2.4755
3 Dim	0.8956	0.8422	0.2398	0.2876	3.1928
4 Dim	0.8982	0.8607	0.2395	0.3106	4.3953
5 Dim	0.6322	0.8861	0.2677	0.3061	1.1138
7 Dim	1.0136	0.9830	0.2944	0.4403	1.6599
9 Dim	1.0461	0.9201	0.4356	0.4445	2.3235

Table 5. Objective value, standard deviation and PRI by the compared population-based methods on Otsu algorithm.

Image	M-1	Objective values(standard deviation)/PRI				
		HABC	ABC	PSO	CMA-ES	EGA
38092	2	2.5064E4(3.83477E-12)/ 0.74248	2.5064E4(3.8348E-12)/ 0.74248	2.5064E4(3.8348E-12)/ 0.74248	2.5047E4(18.3953)/ 0.73305	2.5063E4(0.2663)/ 0.74268
	3	2.5259E4(0.0257346)/ 0.78598	2.5259E4(0.0259)/ 0.78611	2.5259E4(0.0462)/ 0.78546	2.5235E4(39.6597)/ 0.77094	2.5256E4(1.4455)/ 0.78657
	4	2.5345E4(0)/ 0.7994	2.5345E4(0.0192)/ 0.79932	2.5344E4(0.8140)/ 0.80059	2.5276E4(65.1685)/ 0.77711	2.5340E4(2.0187)/ 0.79761
	5	2.5387E4(0.130652)/ 0.80876	2.5387E4(0.0367)/ 0.8083	2.5383E4(2.8507)/ 0.80108	2.5347E4(25.8857)/ 0.7798	2.5353(1.3310)/ 0.8031
	7	2.5437E4(0.169583)/ 0.79081	2.5436E4(1.5625)/ 0.79184	2.5425E4(5.1891)/ 0.78483	2.5389E4(40.7006)/ 0.79591	2.5403E4(1.3991)/ 0.79123
	9	2.5458E4(0.565715)/ 0.79661	2.5458E4(0.4417)/ 0.79642	2.5446E4(2.5883)/ 0.79571	2.5430E4(11.0258)/ 0.79659	2.5413(1.0165)/ 0.79513
14037	2	7.5360E3(0.00682035)/ 0.73621	7.5361E3(0.0068)/ 0.73621	7.5361E3(0)/ 0.73614	7.4730E3(99.7498)/ 0.72324	7.5359E3(0.2176)/ 0.73693
	3	7.6593E3(9.58692E-13)/ 0.8019	7.6593E3(0)/ 0.8019	7.6592E3(0.2069)/ 0.80121	7.6006E3(47.7619)/ 0.76189	7.6567E3(1.4649)/ 0.79826
	4	7.7151E3(9.58692E-13)/ 0.79968	7.7151E3(1.9174E-12)/ 0.79968	7.7120E3(2.7500)/ 0.79803	7.6729E3(57.4010)/ 0.78318	7.7110E3(2.3737)/ 0.79919
	5	7.7435E3(0.0188984)/ 0.79629	7.7435E3(0) 0.79639	7.7347E3(3.5134)/ 0.79788	7.6961E3(35.7684)/ 0.78357	7.7341E3(4.3209)/ 0.79849
	7	7.7718E3(0.261583)/ 0.80195	7.7710E3(2.6775)/ 0.80073	7.7614E3(4.4311)/ 0.7995	7.7389E3(16.3182)/ 0.80102	7.7620E3(2.9283)/ 0.79861
	9	7.7852E3(0.111484)/ 0.80293	7.7844E3(0.4046)/ 0.80003	7.7778E3(1.9638)/ 0.80216	7.7674E3(11.5504)/ 0.79963	7.7774E3(2.5778)/ 0.79894

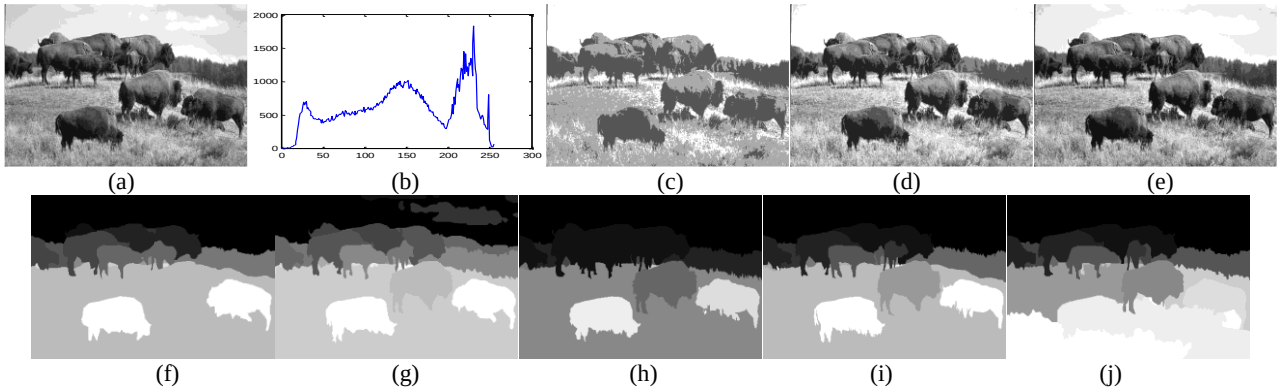


Fig.4. (a) Original Image 38092 (b) histogram (c) M-1=2 PRI=0.74248 (d) M-1=5 PRI=0.80876 (e) M-1=9 PRI=0.79661 (f-j) the ground-truth hand segmentations offered by Berkeley Segmentation Dataset

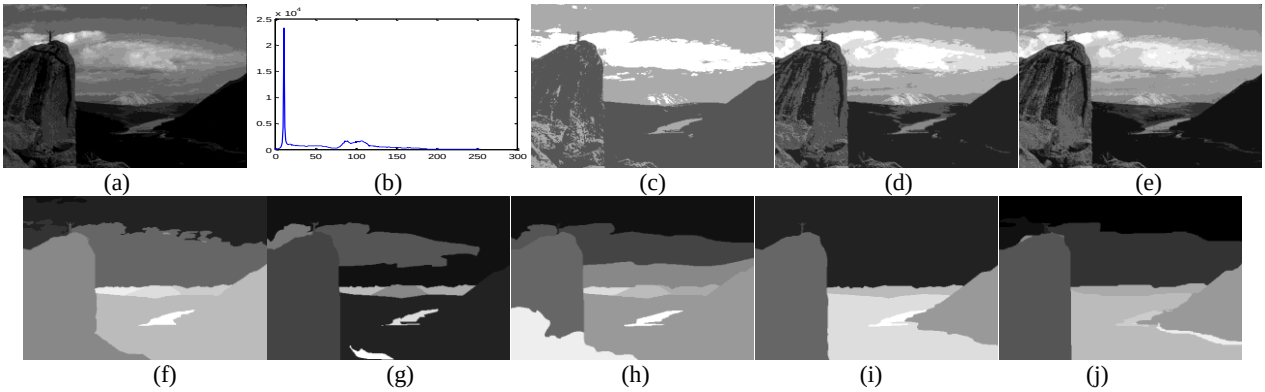


Fig.5. (a) Original Image 14037 (b) histogram (c) M-1=2 PRI=0.73621 (d) M-1=5 PRI=0.79629 (e) M-1=9 PRI=0.80293 (f-j) the ground-truth hand segmentations offered by Berkeley Segmentation Dataset

5 Conclusion

In this paper, we propose a novel hierarchical artificial bee colony algorithm, called HABC, to improve the performance of solving complex problem. The main idea of HABC is to extend single artificial bee colony (ABC) algorithm to a hierarchical and cooperative mode by combining the multi-population vector decomposing strategy and the comprehensive learning method. The experimental results demonstrate the proposed HABC achieved superior performance to other classical powerful algorithms. Then, the HABC algorithm is applied for resolving the real-world image segmentation problems. The correlative results obtained by HABC-based method on each image indicate a significant improvement compared to several other popular population-based methods.

6 References

[1] N. Otsu, "A threshold selection method from gray-level histograms". IEEE Transactions on System, Man, and Cybernetics, vol.9, pp.62-66, 1979.

[2] J. N.Kapur,. "A new method for gray-level picture thresholding using the entropy of the histogram". Computer Vision Graphics Image Processing, vol.2, pp.273–285,1985.

[3] B. Akay, "A Study on Particle Swarm Optimization and Artificial Bee Colony Algorithms for Multilevel Thresholding". Applied Soft Computing, vol.13, pp.3066–3091, 2013.

[4] Ozturk, DervisKarabog, "Improved Clustering Criterion for Image Clustering with Artificial Bee Colony Algorithm". Pattern Analysis and Applications, pp.1-13, 2004.

[5] S. Sarkar, S. Das, "Multi-level image thresholding based on two-dimensional histogram and maximum tsallis entropy - a differential evolution approach". IEEE Transactions on Image Processing, 2013.

[6] D. Karaboga, B. Basturk, "A powerful and efficient algorithm for numerical function optimization:Artificial bee colony (abc) algorithm". Journal of Global Optimization, vol.3, pp. 459-471, 2005.

[7] X. Li, X. Yao, "Cooperatively coevolving particle swarms for large scale optimization". IEEE Transaction on evolutionary computation, vol.16, pp. 210-212, 2012.

[8] N. Hansen and A. Ostermeier, "Completely derandomized self-adaptation in evolution strategies", Evolutionary Computation, vol.9, pp. 159–195, 2001.

[9] R. Unnikrishnan, M. Hebert, "Measures of Similarity", IEEE Workshop on Applications of Computer Vision, pp. 394–400, 2001.

Combination and Analysis of Features From Forensic Letters

Aline M. Malachini Miotto Amaral[1], Cinthia O. de Almendra Freitas[2], and Flávio Bortolozzi[1]
[1]Department of Informatics, UNICESUMAR, Maringá, Paraná, Brazil
[2]Politechinc School, Pontifícia Universidade Católica do Paraná, Curitiba, Paraná, Brazil

Abstract - *Current paper describes a study of a feature set extracted from forensic letters and used to writer identification. Forensic letters are documents devoted to collect the handwriting style, individuality, and other characteristics. In this context, extract automatically handwriting features from such documents could aid forensic examiners in their analyses. Experiments were conducted, and the individual accuracy of the features extracted, in a writer identification approach, was computed, resulting in poor results, except for the axial slant feature. However, when these features are analyzed in combination mode, the accuracy comes to 72.7% with 180 different writers and 84% with 100 different writers.*

Keywords: *Handwriting recognition, Feature Extraction; Forensic Letter, Document Processing***.**

1 Introduction

According to Koppenhaver [1], forensic letters are devoted to collect the handwriting style, individuality, and other characteristics. They have been collected for several reasons. In their practice, document examiners frequently have to collect specimen of handwriting to make a professional examination.

It is important remark that there are many different types of forensic letter in different languages, and no one form is better or worst than any other specimen writing forms. Each has its specific purpose and is generally designed by an expert in forensic analysis who believes that it will provide him with those features of the writer's writing that are most useful in conducting a handwriting examination [2].

Currently, the forensic handwriting examination is performed by experts using optical (optical device) and/or chemicals methods. Based on Sheikholeslami et al. [3] the manual process of feature extraction and observation is tedious and subjective. In addition, different examiners may extract the same features from a particular document in a different way. Then, the use of automatic feature extraction process can be useful and helpful to experts.

Features extracted from a forensic letters can be classified mainly according to their granularity. Taking account the granularity they can be classified as global and local. Global features use information from the entire document, while local features use information from a piece of the document [4].

Besides, according to Amaral et al. [5] features can be differentiated in graphometry and non-graphometry. Graphometry features are those applied by the examiner during its analysis. These experts analyze the forensic letters observing aspects of the handwriting, such as, minimal graphics, beginning/ending strokes, calibre, spacing between characters and words, proportion, slant, alignment to baseline among others. Non-graphometry features use primitives extracted from the document image, such as texture approaches or codebook approaches. These features are not considered graphometry, since they consist of complex computational transformations and procedures on the document image and do not consider the same principles used by the forensic letter examiners.

In this context, current paper presents an analysis of a graphometry feature set extracted from a forensic letters. Analysis of the individual and combining accuracy of these features are performed through a set of experiments using the Brazilian Forensic Letter Database [6].

The paper is divided into the following sections. Section 2 discusses the Forensic Letters. Section 3 presents the features, which can be extracted from a forensic letter. Section 4 presents a graphometry feature set and its extraction process. Section 5 discusses the experimental results and presents a brief discussion based on results obtained. Finally, Section 6 provides some considerations and indicates future investigations.

2 Forensic Letter

According to Freitas et al. [6], the main goal of a forensic letter is to reproduce the association among different letters, words, numerals, and symbols. The letters must be adapted to the local language and it is important to reproduce the upper and lower case, diacritics (á, à, ã, ê, ü), special symbols (ç), numerals ("0" to "9"), and punctuation symbols.

Most work-related forensic letters found in the literature are devoted to English language, and thus they do not include characteristics from Brazilian Portuguese Language and Brazilian writers. In this context, Freitas et al. [6] presented the Brazilian forensic letter database, which was created to address the several particularities of the Portuguese Language.

The Brazilian Forensic Letter (shown in Figures 1 and 2) is composed of all Latin alphabetic characters (upper and lower case) and others Latin symbols such as, special symbol (ç), punctuation symbols, and diacritics (á, à, ã, ê, ü). The special symbol (ç) and diacritics constitute the

feature called minimal graphics categorized as genetic features in graphometrical features [7].


De
Fernando Quintas Zanon
Rua Luiz Kirt Walterez, 87 - Ap. 300
Xenápolis, NovaYolanda 14506-159
Para Dr. Onório Bob Grant

Soube, através de publicação pela imprensa local, que V. Sas. necessitam de um funcionário na Seção de Correspondência do Departamento Pessoal.

Venho, portanto, candidatar-me a esta vaga. Sou brasileiro, solteiro, com 18 anos, curso a 3ª série do Curso Técnico de Contabilidade do Colégio Horácio Alves - Escola Municipal de 2° Grau - e possuo alguma prática de datilografia e arquivos.

Trabalhei durante dois anos nas Lojas Universais Rayon S.A. onde exerci as funções de Auxiliar de Escritório Júnior.
Inicialmente, coloco-me à disposição de V. Sas. para um período de experiência, quando, então, poderão tranquilamente avaliar minhas aptidões.

Na expectativa de uma resposta apresento-lhes cordiais saudações,

Fernando Zanon

Figure 1. Brazilian Forensic Letter

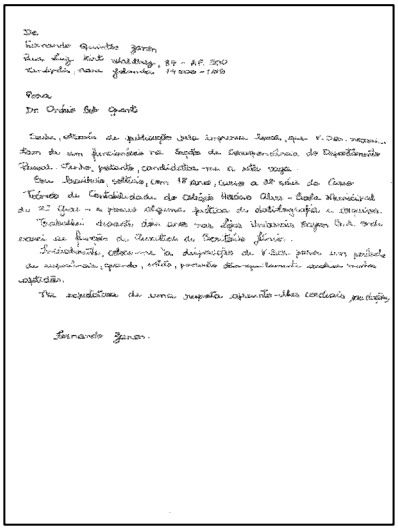


Figure 2. CF017_01 – Sample of Brazilian Forensic Letter

Nowadays, the Brazilian Forensic Letter Database contains 1800 letters being 3 letters from each one of 600 different writers. It is important to emphasize that the letters used in the experiments presented in this paper were extracted from this database.

3 Features From Forensic Letters

There are different forms to classify features in a handwriting document (such as a forensic letter). In this work we use a classification based on the research present by Sreeraj and Idicula [8] (as shown in Figure 3).

Features which consider information from the whole document are classified as global (those involving the extraction of information in the document level, paragraph and line), and features which consider information from a specific part of the document are classified as local (those involving the extraction of information in word level and character). Based on the input method of writing, automatic writer identification system has been classified as online and offline.

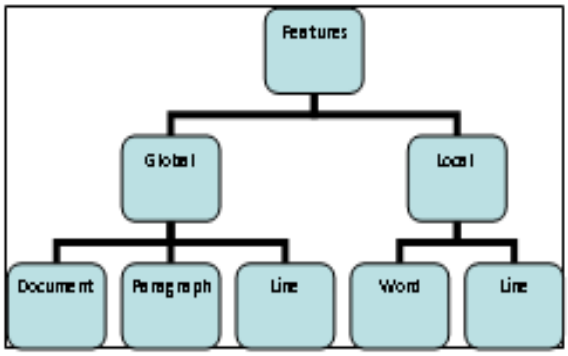


Figure 3. Classification of features (adapted from [8])

3.1 Global Features

Many research use global features extracted from the document image, such as the texture approaches [9-11] or codebook approaches [12, 13]. These approaches consist in applying filters (such as Gabor Filter) in the document image. In these approaches, it is not necessary use forensic letters to realize the handwriting analysis, since the text format is not important to the extraction process. Consequently, these approaches are non-graphometry, since, as stated before, they do not consider the same principles used by the forensic letter examiners.

Researches such as [5, 14, 15, 16] discuss global features, such as, continuity of the stroke, closed regions, upper and lower edges, axial slant and height and slant of the text lines and margin position, inclination and height of the text lines. These features are considered graphometry since they are some of the features used by the experts during their analysis.

Figure 4 present an example of global graphometry feature [5] - the margins positions. This feature demonstrates the relative placement habits of a writer. In Figure 5, another global graphometry feature - the axial slant [5] is presented. This feature represents the general angle of the handwriting.

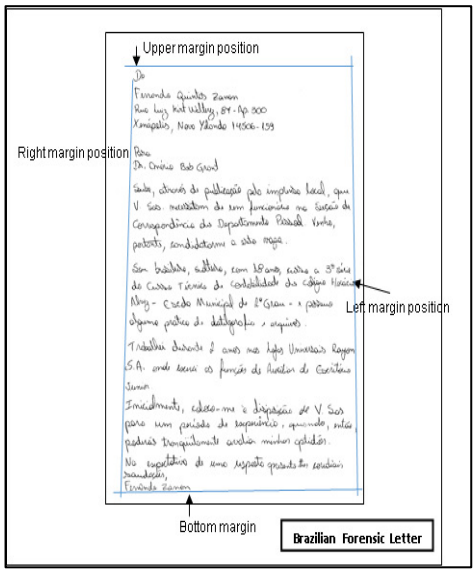


Figure 4. Example of global features – margins position

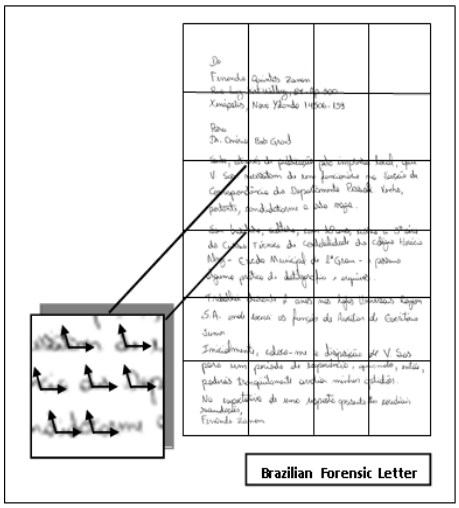


Figure 5. Example of global feature – axial slant

3.2 Local Features

Local features, as stated before, use information from a piece of the document. As occurs in global approaches, there are graphometry and non-graphometry local features. Examples of research which extract non-graphometry features are [17,18]. In these works, allograph and grapheme features are extracted.

In works such as [5, 16, 19] graphometry features are extracted from: specifics characters ("d", "y" and "f") , proportion of words from the middle zone compared with the zones up and down of the words, words slant, height and weight of the words; among others.

Figure 6 present an example of local graphometry feature [5] - the height of the first word of each handwriting line. When this feature had to be computed, the first word of each line was bounded by a box and its height and proportion of black pixels were computed.

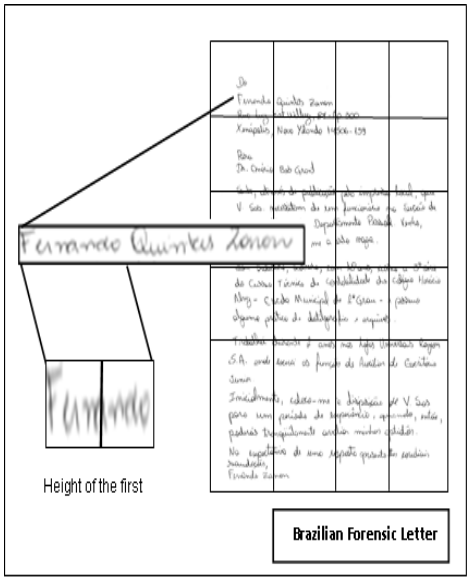


Figure 6. Example of local feature – first word of each handwriting line

4 Feature Extraction

Foregrounded on the study of graphometry, a set of feature used for forensic handwriting analysis was proposed, based on [5]. This set of features is composed of: relative placement habits (f_1,f_3,f_4,f_5,f_6), relative relationship between individual words height (f_2 and f_7), axial slant (f_8) and relative loop habits (f_9,f_{10},f_{11},f_{12}) as presented in Table 1. It is important detach that the last group of features (f_9,f_{10},f_{11},f_{12}) was not explored in [5], and that analysis focused on the individual performance of each feature were not realized in this work.

An important feature related to handwriting individuality is relative placement habits [2]. Writers may make a better use of the paper sheet and write to its physical limit. They may also leave a blank space, usually regular in all lines. Different writers start and stop their writing at different locations.

Another important feature is related to the size of the first word of each handwriting line. When this feature had to be computed, the first word of each line was bounded by a box and its height and proportion of black pixels were computed.

The axial slant is a graphometry feature used by the forensic document examiners and has been extensively used. In fact, it represents the general angle of the handwriting and has the best individual performance in our analyses.

The relative loop habits are a set of graphometry features extracted from words and characters. These features present information about the upward and downward loops of the words (height, width, number of pixels and axial slant).

Figures 7 and 8 present an overview of the extraction process from an image in the Brazilian Forensic Letter Database [6]. The result of the extraction process is a vector containing 85 primitives. This vector is applied to SVM classifier in the training and test stages.

It should be highlighted that, in order to compute each feature, a number of primitives had to be extracted from the image of the document (totaling 85 primitives).

Table 1. Feature description

Group	Description
f_1 (primitive: 1)	Number of lines in each forensic letter.
f_2 (primitives: 2 to 21)	Proportion of black pixels
f_3 (primitives: 22 to 41	Right margin position
f_4 (primitive: 42)	The lower left margin position.
f_5 (primitive: 43)	Upper margin position
f_6 (primitive: 44)	Bottom margin position
f_7 (primitives: 45 to 64)	Height of the first word
$f8$ (primitives: 65 to 81)	Axial slant: general angle of the handwriting.
f_9 (primitive: 82)	Medium height of the upward and downward loops
f_{10} (primitive:83)	Medium width of the upward and downward loops
f_{11} (primitive: 84)	Medium size (in number of pixels) of the upward and downward loops
f_{12} (primitive: 85)	General angle of the upward and downward loops

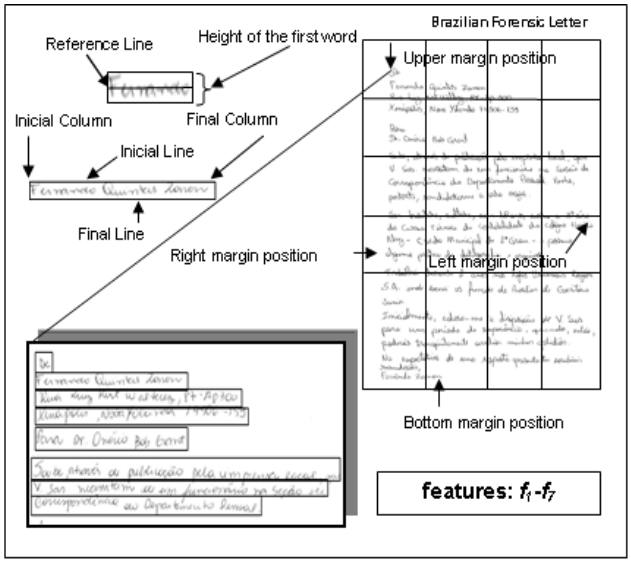


Figure 7. Overview of feature extraction $f_1 – f_7$ (adapted from [5])

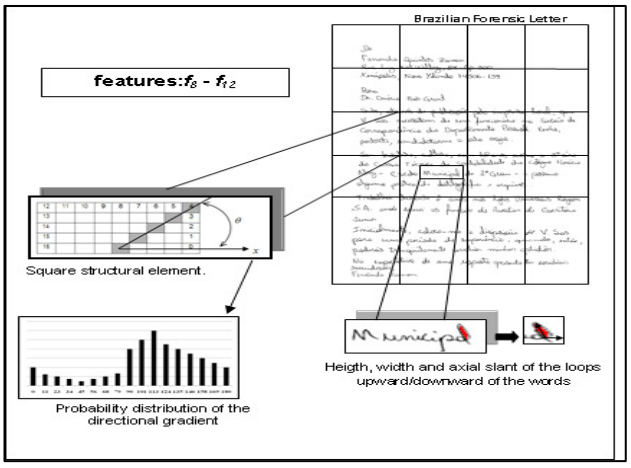


Figure 8. Overview of feature extraction $f_8 – f_{12}$ (adapted from [5])

5 Experimental Results

In order to validate the features extracted from a forensic letter experiments were conducted, in this work the features extracted are used in an automatic writer identification approach (defined in [20]). It is important to observe that this feature set can be used in other kinds of forensic analysis, such as, authenticity verification.

Firstly in the experiments realized, all features are analyzed individually, using forensic letters of different writers randomly selected from Brazilian Forensic Letter Database. The number of writers was gradually increment from 20 to 180, in order to evaluate the feature stability. The results of this experiment can be observed in Table 2.

In this table are also presented the results of experiments conducted with the entire set.

Table 2. Individual accuracy of features in a writer identification approach.

Group of features	Number of Writers/Accuracy (%)				
	20	**60**	**100**	**140**	**180**
f_1	15	3.33	4	4.3	3.89
f_2	25	18.33	17	15	13.89
f_3	20	16.67	15	11.42	8.33
f_4	10	8.33	5	2.86	1.667
f_5	10	6.67	5	4.3	2.22
f_6	30	5	8	3.57	2.22
f_7	35	30	25	21.43	18.33
f_8	**85**	**76.67**	**68**	**67.15**	**62.78**
f_9	5	5	4	2.142	1.667
f_{10}	5	5	3	2.142	1.667
f_{11}	6	5	3	1.43	1
f_{12}	10	5	4	2.142	1.667
f_1 & f_2 & f_3 & f_4 & f_5 & f_6 & f_7&f_8 & f_9& f_{10} & f_{11}& f_{12}	65	55	55	51.43	49.44

It can be clearly seen in Table 2 that individually the accuracy of the most features are not significant, since low rates were obtained. This scenario is not only observed in the axial slant feature (f_8). This feature has the best performance of the feature set, and alone already brings good results for forensic analyses.

The second group of experiments was conducted combining our bets feature - f_8 with all other features, and the result of these experiments can be observed in Table 3. It is important to note, that individually the features f_1 (number of letter lines), f_6 (bottom margin position) and f_{12} (general angle of the upward and downward loops) do not present good rates. However, when these features are combined with feature f_8 (axial slant) the results are significantly enhanced. It occurs since these features aid the classifier to decide, among the possible authors candidates of the forensic letter.

Table 3. Accuracy of second group of experiments

Group of features	Number of Writers/Accuracy (%)				
	20	**60**	**100**	**140**	**180**
f_1 & f_2 & f_3 & f_4 & f_5 & f_6 & f_7&f_8 & f_9& f_{10} & f_{11}& f_{12}	65	55	55	51.43	49.44
f_8 & f_1	**85**	**76.67**	**76**	**71.7**	**71.4**
f_8 & f_2	60	51.67	49	42.5	42.14
f_8 & f_3	60	50	47	40.83	40
f_8 & f_4	80	73.33	69	65	63.57
f_8 & f_5	80	71.67	69	65.83	62.86
f_8 & f_6	**90**	**83.33**	**78**	**75**	**71.43**
f_8 & f_7	75	58.33	59	60	56.43
f_8 & f_9	60	55	33	31.7	32.15
f_8 & f_{10}	60	65	33	22.5	21.43
f_8 & f_{11}	75	35%	33	22.5	11.42
f_8 & f_{12}	**85**	**80**	**76.27**	**73.33**	**68.57**

After the second group of experiments, we could perceive that some features, mainly those with complex extraction process such as f_2 and f_7, introduce noise in the information provided by the feature set, since the results fall when these features are added in the feature set (see the last line of the Table 2 when the entire feature set was tested). Another group of experiments were conducted combining features f_1, f_6 and f_{12}, those that present the best results when combined with f_8. The results of this experiment are compiled in Table 4.

Table 4. Accuracy of third group of experiments

Group of features	Number of Writers/Accuracy (%)				
	20	**60**	**100**	**140**	**180**
f_1 & f_6 & f_8	85	85	80	77.14	71.11
f_1 & f_{12} & f_8	85	83.33	78	71.43	67.78
f_6 & f_{12} & f_8	80	73.33	69	68.57	66.66
f_1 & f_6 & f_8 & f_{12}	**85**	**88.3**	**84**	**80**	**72.77**

In order to prove the results obtained in the experiments described above, a formalized feature selection approach as presented in [5] was applied. This feature selection process determines precisely which feature set present the best accuracy to the writer identification task. The goodness combination set (CS) obtained in this formal process was composed of features f_1, f_6, f_8 and f_{12} (exactly the same set presented in Table 4).

6 Conclusions

Current paper presents an analysis, individual and in group, of a graphometry feature set extracted from a forensic letter. These features could be used in different kinds of forensic analysis, in this work the feature set was validated in a writer identification approach.

The individual accuracy of the features, as demonstrated in Table II, was not interesting, except to the axial slant feature (f_8), which is our best feature. However, when these features are combined, some ensemble of combinations present goods writer identification rates, as can be observed in Table III and IV.

Experiments in authorship verification will be conducted in order to validate the feature set to other type of forensic analyses.

7 References

[1] Koppenhaver, K.M.,"Forensic Document Examination, Principles and Practices". Humana Press, 2007.

[2] Morris, R. N, "Forensic Handwriting Identification – fundamental concepts and principles". Academic Press, 2000.

[3] Sheikholeslami, G.; Srihari, S. N.; Govindaraju, V, Computer aided graphology. Proceedings of the 5th International Workshop on Frontiers in Handwriting Recognition. Essex, England. 1996, pp. 457-460.

[4] Sreejaj, M., Idicula, S. M, "A survey on writer identification schemas". International Journal of Computer Applications, 26, 2, 2011, pp. 23-33.

[5] Amaral, A. M. M. M., Freitas, O. A., Bortolozzi, F, "Feature Selection for Forensic Handwriting Identification". Proceedings of 12th International Conference on Document Analysis and Recognition, 2013, pp. 922-926.

[6] Freitas,C. O. A., Oliveira, L. S; Bortolozzi, F. Sabourin, R, "Brazilian Forensic Letter Database". Proceedings of the 11th International Conference on Frontiers on Handwriting Recognition, 2008, pp. 64-69.

[7] Oliveira, L S., Justino, E., Freitas, C.O.A., Sabourin, R, "The Graphology Applied to Signature Verification". Proceedings of International Conference on Graphonomics Society, 2005, pp. 286-290.

[8] Sreejaj, M. Idicula, S. M, "A survey on writer identification schemas. International Journal of Computer Applications". 2011, pp. 23-33.

[9] Bulacu, M. and Schomaker, L. and Brink, "A. Text-independent writer identification and verification on offline Arabic handwriting". Proceedings of the 9th Conference on Document Analysis and Recognition, 2007, pp.769-773.

[10]He, Z. You, X. Tang,Y, "Writer identification of Chinese handrwriting documents using hidden Markov tree model". Pattern Recognition, 2008, pp.1295-1307.

[11] Helli, B. Moghaddam, E, "A text-independent Persian writer identification based on feature relation graph (FRG)". Pattern Recognition, 2010, pp.2199-2209.

[12]Siddiqi, I., Vincent, N, "Combining global and local features for writer identification". Proceedings of the 11th International Conference on Frontiers in Handwriting Recognition, 2008, pp. 48-53.

[13]Schomaker, L. Franke, K. Bulacu, M, "Using codebooks of fragmented connected-component contours in forensic and historic writer identification". Pattern Recognition Letters, 2007, pp. 719–727.

[14]Hertel, C. Bunke, H, "A set of novel features for writer identification". Proceedings of the 4th international conference on Audio- and video-based biometric person authentication. 2003, pp. 679-687.

[15]Schlapbach, A. Bunke, H, "Off-line Handwriting Identification Using HMM Based Recognizers". Proceedings of the Pattern Recognition, 17th International Conference on ICPR'04, 2004, pp.654-658.

[16]Luna, E. C. H. and Riveron, E. M. F. and Calderon, S. G., "A supervisoned algorihm with a new differentiated-weighting scheme for identifying the author of a handwritten text". Pattern Recognition Letters, 2011, pp.1139-1144.

[17]Bui, Q. A., Visani, M.; Prum S.; Ogier, J. M, "Writer identification using TF-IDF for cursive handwritten word recognition". Proceedings of International Conference on Document Analysis and Recognition, 2011, pp.844-848.

[18]Blankers, V., Niels, R., Vuurpijl, L, "Writer identification by means of explainable features: shapes of

loops and lead-in strokes". Proceedings of the 19th Belgian-Dutch Conference on Artificial Intelligence, 2007, pp.17-24.

[19]Pervouchine, V, Leedham, G, "Extraction and analysis of forensic document examiner features used for writer identification". Pattern Recognition, 2007, pp.1004-1013.

[20]Amaral , A. M. M. M.; Freitas, C. O. A. ; Bortolozzi, F, "Multiple Graphometric Features for Writer Identification as part of Forensic Handwriting Analysis". Proceedings of 17th International Conference on Image Processing, Computer Vision, and Pattern Recognition, 2013, pp.10-17.

Comparaison of Support Vector Machine and Leaky-Integrate & Fire SNN model for Object Recognition

Mrs.S.Chaturvedi[1], **Dr.A.A.Khurshid**[2]
[1]Department of Electronics Engineering, G.H.Raisoni College of Engg., Nagpur, MH, India
[2]Department of Electronics Engineering, RCOEM, Nagpur, MH, India

Abstract *- Classification is récursive an impulse décision making tasks of human quickness. In today's world it is the gaining interest of active research with the areas of neural network. Classification and recognition of objects plays a vital role in computer vision research. In Neural Network, Spiking Neural networks are found to be exquisitely beneficial in impregnable. Classification of data. In this paper we will classify and recognise the object using Support Vector Machine (SVM) and one of the most popular model of SNN which is the Leaky-Integrate and Fire (LIF) model. SVM are supervised learning models that are used for classification of images. SNNs incorporate the concept of time into their operating model. LIF model is easy to implement have good computational efficiency. This paper depicts comparison of SVM and SNN model LIF for object classification and recognition. .*

Keywords: Support Vector Machine (SVM), Spiking Neural Network (SNN), Leaky Integrate and Fire Neuron Model.

1 Introduction

Over the last decades many theories have been developed on how human can recognise. Most of them are based on logical reasoning and on clear abstractions, and sound very plausible. From human object recognition to computerized object recognition is a large rung of ladder. There are various theories on human object recognition which do not touch on the lower plane of vision processing, i.e. assuming that extracting an object from its surroundings is given, and that decomposition of this object into different entities happens naturally.

Humans easily possess the ability to classify the objects. With a simple glance of an object, humans are able to tell its identity or category despite of the appearance variation due to change in pose, illumination, texture, deformation, and under occlusion.

Existing image storing systems limit classification mechanism to describing an image based on metadata such as colour histograms, texture, or shape features. The ability of these systems to retrieve relevant documents based on search criteria could be greatly increased if they were able to provide an accurate description of an image based on the image's content. By using a neural network we can recognise objects from an image.

The difficulties of object recognition are extensive. To avoid too general a discussion we will mainly look at them here in a light that makes sense when working with neural systems. Shapes can differ in appearance for several reasons. The most important reason is the difference in perspective we can have on a shape, i.e. shapes can be viewed from different angles and positions possibly making the shape appear bigger, upside down, tilted etc. In this paper we focus on a supervised learning algorithm called Support Vector Machine (SVM) and Leaky Integrate and Fire neuron model (LIF) which is a well known SNN model for classification of objects.

2 Support Vector Machine (SVM)

Support Vector Machine (SVM) was proposed by Vapnik and has since gained attention a high degree of interest in the machine learning research community. SVM method does not have any limitations of data dimensionality and limited samples. The support vectors, which are critical for classification, are obtained by learning from the training samples.

Support vector machines are supervised learning models with associated learning algorithms that analyze data and recognize patterns, used for classification and regression analysis. The algorithm has scalable memory requirements and can handle problems with many thousands of support vectors efficiently. The SVM constructs a hyper plane or set of hyper planes in a high-or infinite-dimensional space, which is used for classification, regression, or other tasks. Instinctive, a good separation is gained by the hyper plane which has the maximum distance to the nearest training data point of any class, in general the maximum the margin the minimum the generalization error of the classifier.

They belong to a family of generalized linear classification. A special property of SVM is that it simultaneously minimizes the empirical classification error and maximizes the geometric margin. So SVM called Maximum Margin Classifiers. SVM is based on the Structural risk Minimization (SRM). SVM map input vector to a higher dimensional space where a maximal separating hyper plane is constructed. Two parallel hyperplanes are constructed on each side of the hyperplane that separate the data. The separating hyperplane is the hyperplane that maximize the distance between the two parallel hyperplanes. An assumption is made that the larger the margin or distance between these parallel hyperplanes the better the generalization error of the classifier. The support

vector machine (SVM) is superior to all machine learning algorithms which are based on statistical learning theory. There are a number of publications detailing the mathematical formulation and algorithm development of the SVM [6] [7]. The inductive principle behind SVM is structural risk minimization (SRM), which constructs a hyper-plane between two classes, such that the distance between supports vectors to the hyper-plane would be maximum.

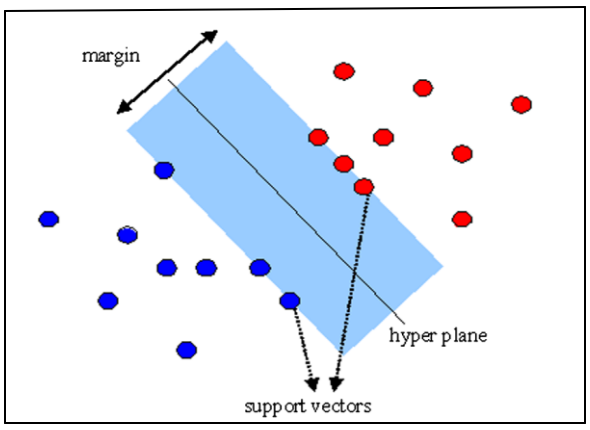


Figure 1. Concept of SVM

The principle of an SVM is to map the input data onto a higher dimensional feature space nonlinearly related to the input space and determine a separating hyperplane with maximum margin between the two classes in the feature space.

3 Leaky Integrate Fire neuron Model

LIF neuron is probably one of the simplest spiking neuron models, but it is still very popular due to the ease with which it can be analyzed and simulated. The LIF model is a popular SNN model which falls into the third generation of neural network models, increasing the level of realism in a neural simulation. SNNs incorporate the concept of time into their operating model.

In this paper, LIF model is used to recognise the objects. LIF model is popular for its simplicity and ease to implement with minimum computational cost. The neuron is called or considered leaky if there is decay in characteristic time constant is summed in contribution to membrane potential when this "leak" is forfeit then he model is considered a perfect integrator. The LIF model is used in computational neuroscience. The leaky integrate-and-fire (LIF) neuron is probably one of the simplest spiking neuron models, but it is still very popular due to the ease with which it can be analyzed and simulated.

The basic circuit of an integrate-and-fire model consists of a capacitor C in parallel with a resistor R driven by a current $I(t)$; The driving current can be split into two components, $I(t) = I_R + I_C$. The first component is the resistive current I_R which passes through the linear resistor R. It can be calculated from Ohm's law as $I_R = u/R$ where u is the voltage across the resistor. The second component I_C charges the capacitor C. From the definition of the capacity as $C = q/u$ (where q is the charge and u the voltage), we find a capacitive current $I_C = C\,\mathrm{d}u/\mathrm{d}t$.

In the leaky integrate-and-fire model, the memory problem is solved by adding a "leak" term to the membrane potential, reflecting the diffusion of ions that occurs through the membrane when some equilibrium is not reached in the cell.

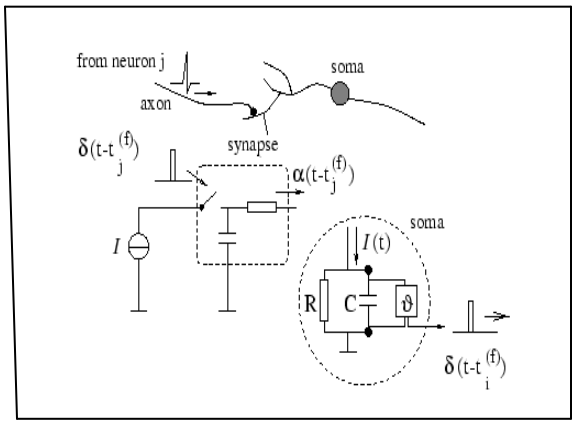


Figure 2. Concept of LIF

4) Steps involved for classification of object

The image of object is scanned which degrades the quality of the image. To enhance or improve the quality of the image, the image undergoes four steps:-

4.1) Pre-processing
4.2) Feature extraction
4.3) Classification of object using SVM
4.4) Object Recognition.

4.1) Pre-processing:

In pre-processing the image is scanned, due to which noise is introduced which leads to degradation of the quality of the image; this parameter can reduce the systems accuracy.

Pre-processing involves following steps:-

4.1.1) Converting the original image to Gray scale image
4.1.2) Use of filter to remove noise
4.1.3) Image enhancement
4.1.4) Segmentation (Thresholding and Edge Detection)

The scanned input image of an object a mug is as shown below:

4.1.1) Converting original image into Gray scale:

The original scanned image is converted to the gray scale. It makes the further processing easy and the image intensity is equal throughout the image.

4.1.2) Reducing noise by using filter

As the image is scanned its quality is reduced and noise is inserted, to remove the noise we use Median filter. It removes the noise from the image.

4.1.3) Image enhancement

It is the process of adjusting digital image for suitable display of the image. It sharpens or brighten the image and makes easy to identify the key features of the image.

4.1.4) Segmentation

Segmentation plays important role in pre-processing, it is the predominant stage in the demonstration. Segmentation involves Thresholding and Edge Detection. Thresholding converts gray scale image to binary image. It extracts the object from the background by selecting certain threshold value. Edge detection detects meaningful discontinuities in intensity value which helps reduce the data to be processed. Edge detection also preserves the efficient properties of the object to be recognized.

4.2) Feature Extraction

The most vital role in classification of object is to extraction of feature. It represents the feature vector which is used for classification purpose. We extract various features from the object such as texture, shape distribution and pair point feature etc. After extracting feature values we get feature vector which is used for the further classification of various objects.

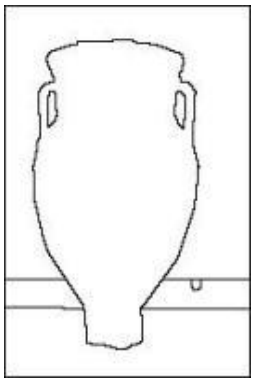
4.3) Classification/Recognition of object

SVM is a supervised learning technique in which the output and input is known.SVM classify efficiently with almost accurate results. We also use LIF model for classification and recognition of object. Among SVM and LIF which classifier technique is accurate and efficient will study by comparing its results.

a)

b)

c)

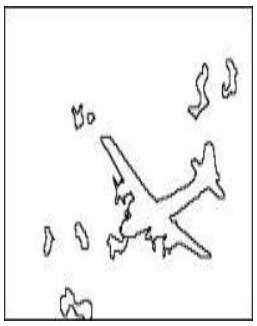


d)

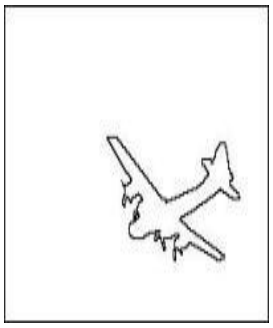
Figure 3. a) Original image b) c) d) Edge detection

a)

b)



c)



d)

Figure 4. a) Original Image b) c) d) Edge Detection

5) SVM Classifier Vs Neural Classifier model

Flexibility of training: The Neural Network classifiers parameters can be adjusted by gradient descent training which aims the global performance. The neural classifier is embedded layout recognizer for object recognition. SVMs can only be trained at the level of holistic patterns.

Complexity of training: Neural classifiers generally adjust their parameters by various algorithms such as Gradient Descent. By giving the training samples a fixed number of sweeps, the training time is proportional with the number of samples. S VMs are trained by quadratic programming (QP), and the training time is generally proportional to the square of number of samples.

Classification accuracy: According to the researchers SVMs has superior classification accuracies to Spiking Neural Classifiers in many experiments.

Storage and execution complexity*:* SVM learning of SVM by QP gives a large number of SVs, which are to be stored and computed in classification. As compared to SVM, Neural classifiers have very less parameters, and the number of parameters is very ease in controlling. In other words, neural classifiers consume less storage and computation than SVMs.

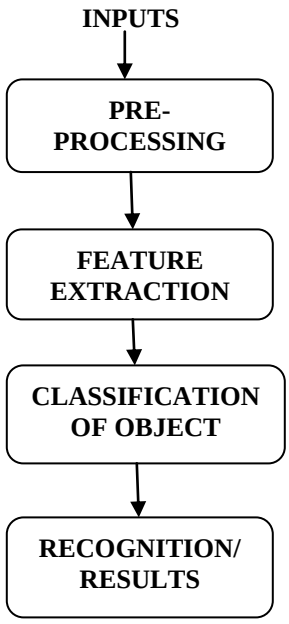


Figure 5. Flowchart of Object Recognition

6) Conclusion

In this paper we will study how to classify object using SVM and recognize using SNN model, Leaky- Integrate fire neuron model (LIF). According to the research, SVM is a good and efficient classifier with high accuracy and better classification as compared to SNN. The LIF model is popular because of its computational efficiency.

7) References

[1] C. Cortes and V. Vapnik "Support-Vector Networks", *Machine Learning*, vol. 20, pp.1 -25 1995.

[2] C.R. De Silva, S. Ranganath, and L.C. De Silva, "Cloud Basis Function Neural Network: A modified RBF network architecture for holistic for holistic Facial Expression Recognition", *The Journal of Pattern Recognition Society*, vol. 41 pp.1241-1253, 2008.

[3]Krishna Mohan Buddhiraju and Imdad Ali Rizvi, Comparison of CBF, ANN and SVM Classifiers for Object based Classification of High Resolution Satellite Images, 978-1-4244-9566-5/10/$26.00 ©2010 IEEE.

[4] M. Pontil and A. Verri "Support Vector Machines for 3D Object Recognition", *IEEE Trans. on Pattern Analysis & Machine Intelligence*, vol. 20, pp.637 -646 1998.

[5] P. Soille, "*Morphological Image Analysis*," Springer-Verlag, Berlin 2003.

[6] S. Arora, D. Bhattacharjee, M. Nasipuri, M.Kundu, D.K.Basu, "Application of Statistical Features in Handwritten Devnagari Character Recognition", International Journal of Recent Trends in Engineering [ISSN 1797-9617], IJRTE Nov 2009.

[7] Sandhya Arora1. Debotosh Bhattacharjee2, Mita Nasipuri2, L. Malik4 , M. Kundu2 and D. K. Basu3Performance Comparison of SVM and ANN for Handwritten Devnagari Character Recognition, IJCSI International Journal of Computer Science Issues, Vol. 7, Issue 3, May 2010.

[8] S.Chaturvedi, N.Sondhiya, R.Titre,"Izhikevich Neuron Model based pattern Classifier for Handwritten Character recognition-Review analysis", ICESC, Nagpur-2014.

[9] Wai-Tak Wong, Sheng-Hsun, Application of SVM and ANN for image retrieval, EJOR, Volume 173, Issue 3, Pages 938–950, Elsevier. 16 September 2006.

Improving the Recall of Context-sensitive Spelling Correction by Loosening the Case-Particle Constraints

Hyun soo Choi[1], **Aesun Yoon**[2], **Hyuk-Chul Kwon**[3]
[1]Dept. of Computer Science, Pusan National University, Busan, Rep. of Korea
[2]Dept. of French, Interdisciplinary Program for Cognitive Science, Pusan National University, Busan, Rep. of Korea
[3]School of Electrical & Computer Engineering, Pusan National University, Busan, Rep. of Korea

Abstract - *The most common method for detection and correction of spelling errors in Korean Spelling and Grammar Checker (KSGC) depends largely on the linguistic rules. Rule-based methods yield very high precision, but extremely low recall because the constituents of the rules should be matched exactly. In this paper, we propose two novel methods that can loosen the case-particle constraints of KSGC's existing rules, in order to improve the recall of Context-Sensitive Spelling Correction: (1) substitution of each case-particle by adding auxiliary particles or particle-compounds; (2) omission of case-particles.*

Keywords: Grammar checker, Context-sensitive Spelling Error, particle, constraint

1 Introduction

Korean Spelling and Grammar Checker (hereafter KSGC) is a system that detects spelling error, syntactical error and semantic error and suggests a replaceable word. Spelling error can be divided into Non-word Spelling Error and Context-sensitive Spelling Error. The former involves the use of words not listed in the dictionary (e.g., teh), and can easily be detected by morphological analysis of the text. On the other hand, the latter should be considered for semantic syntactic relations in context. Therefore, detecting and correcting Context-sensitive Spelling Error significantly affect the overall performance of KSGC.

The ways in which Context-sensitive Spelling Error has been dealt with in the literature can be divided into statistical methods and rule-based methods. The precision of rule-based methods relative to that of statistical approaches is high, but the recall is low. In this paper, we propose a method that loosens the case-particle constraint conditions in existing correction rules to improve the recall of Context-sensitive Spelling Correction.

The method classifies particles used in the constraints of hand-built correction rules according to their semantic/grammatical characteristics. Based on the type of particle, the possibilities of omission and replacement are checked, and the constraint is loosened by omission or addition of replaceable particles to the existing rules.

The following section 2 reviews the Korean and foreign research on Context-sensitive Spelling Error correction and analyzes the existing correction rules and Korean lexical semantic network. Section 3 the possibilities of omission and replacement according to the type of particle constraint to loosen. Section 4 evaluates the performance of the Context-sensitive Spelling Correction rules according to loosening constraint. Section 5 summarizes the paper and outlines the upcoming research.

2 RELATED RESEARCH

2.1 Korean and foreign research on Context-sensitive Spelling Correction

The pertinent foreign research can be divided into two approaches: rule-based methods using parsing, and statistical methods using statistical models such as n-gram.

Parsing with rule-based methods fails when a Context-sensitive Spelling Error occurs[2, 3, 4, 5]. When failure of lexical analysis is detected, parsing with a rule-based parser is conducted. At this point, if parsing error recurs, parsing is performed again, but with replaceable words instead of error words. The replaceable words are selected from a pre-created candidate set (cohort set), a set of similar words and pronunciations. The most important part of parsing with rule-based methods is the meta rules for finding the cause of parsing failure. These meta rules include article omission, synonym spelling errors, selection constraint violations, and others. According to the parser and meta rules used, performance of Context-sensitive Spelling Correction can vary; this makes it difficult to detect either parsing failure by Context-sensitive Spelling Error or the cause of general parsing failure

2.2 Context-sensitive Spelling Correction rules

In our previous work, Context-sensitive Spelling Error was corrected according to hand-built correction rules

formulated by language experts. Table 1 provides a simple example of a correction rule that corrects ″다리다″ (″dalida″; do the ironing), a Context-sensitive Spelling Error incurred as the result of pronunciation similarity, to ″달이다″ (″dal-ida″; boil).

Table 1. Example of Context-Sensitive Spelling Correction Rule

N+P *다리다(dalida; do the ironing)* → ***N+P*** *달이다(dal-ida; boil)*	
(R1)	Context = B1 ;
(R2)	N = [*간장* (*ganjang*; *soy sauce*) | *약* (*yag*; *herbs*) |탕약 (*tang-yag*; *medicinal herbs)* | *한약* (*han-yag*; *herbal medicine*) | *뼈* (*ppyeo*; *bone*) | *사골* (*sagol*; *beef leg bone*) | …]
(R3)	P = [*을*(*eul*) |*를*(*leul*)] (*objective case-particle*)
(R4)	Conjugation = 1001 + 2001

The correction rule is expressed by regular grammar, and, when used, is changed to a finite-state machine. The rules are searched based on keywords (e.g., "dalida"), substitute for the replaceable words (e.g., "dal-ida") to correct. Table 1 describes the selectional constraint and particle constraint (″Noun(N)+Particle(P)″) that appears as context to the right and left of the keyword. Each rule governs ① the context locations relative to the keywords, ② the possibility of the emergence of voluntary components and constraints between the keywords and the context, ③ the conjugation constraints of keywords, ④ the constraints on the components of the context (in this case, N and P), and so on. For example, according to the above rule, if the N+P appears to the left of the keyword ″dalida″ (R1), the change from ″dalida″ to ″dal-ida″ is made. In this case, N is a selectional constraint noun (R2), and P shoud be the objective case-particle ″eul/leul″ (R3). Also, the keyword is conjugated (e.g., ″다렸다″ (dalyeossda; ironed)) (R4).

What can improve recall in existing rules is expanding the selectional constraint of arguments (R2) or loosening the case-particle constraint conditions (R3). Based on our previous research, we have already proposed the expansion of the selectional constraint of arguments through the Korean lexical semantic network. In this paper, we propose loosening the case-particle constraint conditions.

3 Loosening the Case-Particle Constraint Conditions

Correction rule in Table 1, ″다리다″ (dal-ida) is used to ″~을/를 달이다″(~eul/leul dal-ida; boil something) by taking the object as an argument. In this case, if the rule does not constrain the objective case-particle by ″eul / leul″, ″dalida″ can be changed to ″dal-ida″ by mistake in the sentence ″한약이 다린 와이셔츠에 쏟아져...″ (Herbal medicine is poured on ironed shirts...). In this way, particle constraints can be used as a tool to increase the precision of correction rules, but it can decrease the recall.

Particles can be divided into three types: the case particle, the connective particle, and the auxiliary particle [6]. The case particle is the type mainly used in KSGC, and there is a difference in the possibility of replacement and the list of auxiliary particle and combined "case-particle + auxiliary particle" according to the type. However, the rule of the current KSGC chooses between particle constraint and particle removal. Therefore, excessive loosening of the constraints might lead to a drop in precision. This study undertook to determine the possibility of omission and replacement. Presently, with the current Korean information processing technology, it is difficult to distinguish the meanings of isomorphic particles; thus, when reviewing the possibility of omission and replacement, even if the function is the same, we leave the isomorphic particle off of the replaceable words list, which lessens the possibility of omission. For example, the subjective case-particles ″이/가, 께서″ (i/ga, kkeseo) can be replaced by the auxiliary particle or the combined ″case-particle + auxiliary particle.″ But the subjective case-particles ″에서, 서″ (eseo, seo), because they are homographic with the adverbial case, are not included on the replaceable words list.

Table 2 summarizes the possibility of omission and replacement of the case-particle according to the constraints of the existing correction rules. Based on this result, we loosen the constraints in two steps. In step 1, we expand the replaceable particles, and in step 2, we omit the particle constraints. For instance, ″한약도 다린다″ (han-yagdo dalida; Also do the ironing of the herbal medicine) can be changed to ″한약도 달인다″ (han-yagdo dal-ida; Also boil the herbal medicine) in step 1, and ″다린 한약″ (dalin han-yag; ironed herbal medicine) can be changed to ″달인 한약″ (dal-in han-yag; boiled herbal medicine).

Table 2. Possibility of Omission and Replacement

Type of particle		Case-particle constraint	Omit	Replaceable particle
case-particle	subjective	*이/가* (i/ga), *께서* (kkeseo)	+	*은/는* (*eun*/*neun*), (*께서*)+ {auxiliary particle}
	auxiliary	*이/가* (*i*/*ga*)	+	*은/는*, {auxiliary particle}
	objective	*을/를* (*eul*/*leul*)	+	*은/는*, {auxiliary particle}
	prenominal	*의*(*ui*)	+	
	adverbial	*에*(*e*)	+	*에* +[*은/는*, {auxiliary particle}]
		에게 (*ege*), *께* (*kke*), *한테* (*hante*), *더러* (*deoleo*), *보고* (*bogo*)	+	[*에게*, *께*, *한테*, *더러*, *보고*]+[*은/는*, {auxiliary particle}], *에게로* (*egelo*), *한테로* (*hantelo*)
		(*에*)*서*, ((*e*)*seo*) *부터* (*buteo*), (*으*)*로부터* ((*eu*)*lobuteo*)	-	[(*에*)*서*, *부터*, (*으*)*로부터*]+[*은/는*, {auxiliary particle}]
		에게서 (*egeoseo*), *한테서* (*hanteseo*), (*으*)*로부터*	-	[*에게서*, *한테서*, (*으*)*로부터*]+[*은/는*, {auxiliary particle}]
		에로 (*elo*), (*으*)*로* ((*eu*)*lo*)	+	[*에로*, (*으*)*로*]+[*은/는*, {auxiliary particle}]
		(*으*)*로*, (*으*)*로써* ((*eu*)*losseo*)	-	[(*으*)*로*, (*으*)*로써*]+[*은/는*, {auxiliary particle}]
		와/과 (*wa*/*gwa*)	-	*와/과*+[*은/는*, {auxiliary particle}]
		처럼 (*cheoleom*), *만큼* (*mankeum*), *보다* (*boda*)	-	[[*처럼*, *만큼*, *보다*]+[*은/는*, {auxiliary particle}]
{auxiliary particle}		*만*(*man*), *도*(*do*), (*이*)*나*((*i*)*na*), (*이*)*야*((*i*)*ya*), *마저*(*majeo*), *마다*(*mada*), *조차*(*jocha*), *까지*(*kkaji*), *부터*(*buteo*), (*이*)*라도*((*i*)*lado*), (*이*)*나마*((*i*)*nama*), (*이*)*든지*((*i*)*deunji*), *만도*(*mando*), *마저도*(*majeodo*), *마저나마*(*majeonama*), *조차부터*(*jochabuteo*)		

4 Experiment and Evaluation

4.1 Experimental Environment

To evaluate the performance of Context-sensitive Spelling Correction after loosening the case-particle constraint conditions, we used the configurations most commonly used for corpus evaluation in previous studies. For example, to evaluate the rule correcting "dalida" to "dal-ida", we extracted the sentences including "dal-ida" from the "Sejong morph-sense tagged corpus" and changed "dal-ida" to "dalida" in order to form wrong sentences. Then, we built an evaluation corpus by adding correct sentences including "dalida."

To evaluate the performance of the expansion method for contextual information, we configured, as evaluation data (Table 3), 18 word pairs from the most mistakable Context-sensitive Spelling Errors.

Table 3. Evaluation Data

Rule		Correct sentence	Wrong sentence	Total
TR1	*낫다* (*nasda*; *be cured*) => *낳다* (*nahda*; *give birth*)	100	100	200
TR2	*낳다* (*nahda*; *give birth*) => *낫다* (*nasda*; *be cured*)	100	100	200
TR3	*다리다* (*dalida*; *do the ironing*) => *달이다* (*dal*-ida; *boil*)	100	100	200
TR4	*달이다* (dal-ida; *boil*) => *다리다* (*dalida*; *do the ironing*)	100	100	200
TR5	*마치다* (*machida*; *finish*) => *맞히다* (*majhida*; *hit*)	100	100	200
TR6	*맞히다* (*majhida*; *hit*) => *맞추다* (*majchuda*; *guess right*)	100	100	200
TR7	*배다* (*baeda; penetrate*) => *베다* (*beda*; *cut*)	100	100	200
TR8	*베다* (*beda*; *cut*)=> *배다* (*baeda*; *penetrate*)	100	100	200
TR9	*안치다* (*anchida*; *prepare for cooking*) => *앉히다* (*anjhida*; *sit*)	100	100	200
TR10	*앉히다* (*anjhida*; *sit*) => *안치다* (*anchida*; *prepare for cooking*)	100	100	200
TR11	*저리다* (*jeolida*; *numb*) => *절이다* (jeol-ida; *salt*)	100	100	200
TR12	*절이다* (*jeol-ida*; *salt*) => *저리다* (*jeolida*; *numb*)	100	100	200
TR13	*젖히다* (*jeojhida*; *bend back*) => *제치다* (*jechida*; *beat / push*)	100	100	200
TR14	*제치다* (*jechida*; *beat / push*) => *젖히다* (*jeojhida*; *bend back*)	100	100	200
TR15	*집다* (*jibda*; *pick up*) => *짚다* (*jipda*; *place one's hand*)	100	100	200
TR16	*짚다* (*jipda*; *place one's hand*) => *집다* (*jibda*; *pick up*)	100	100	200
TR17	*찢다* (*jjijda*; *tear*) => *찧다* (*jjihda*; *pound*)	100	100	200
TR18	*찧다* (*jjihda*; *pound*) => *찢다* (*jjijdal*; *tear*)	100	100	200

4.2 Experimental Result

Table 4. Performance Evaluation for Loosneing the Case-Particle Constraint Conditions

Rule	Existing method		Step 1 loosening		Step 2 loosening	
	P	R	P	R	P	R
TR1	100.0%	39.81%	97.83%	43.27%	94.55%	49.06%
TR2	100.0%	26.00%	100.0%	34.00%	97.50%	38.61%
TR3	94.29%	32.67%	94.29%	32.67%	83.10%	57.28%
TR4	100.0%	31.00%	100.0%	34.00%	100.0%	48.51%
TR5	95.45%	21.00%	96.00%	24.00%	93.10%	27.00%
TR6	92.59%	25.00%	92.59%	25.00%	89.74%	35.00%
TR7	100.0%	25.00%	100.0%	25.00%	91.43%	32.00%
TR8	100.0%	40.00%	100.0%	41.00%	95.65%	44.00%
TR9	89.09%	49.00%	89.09%	49.00%	84.85%	55.45%
TR10	97.18%	69.00%	97.26%	71.00%	96.15%	73.53%
TR11	100.0%	38.24%	100.0%	40.20%	100.0%	45.10%
TR12	96.88%	31.00%	96.97%	32.00%	84.31%	42.57%
TR13	100.0%	21.00%	100.0%	22.00%	81.58%	31.00%
TR14	100.0%	22.00%	100.0%	23.00%	96.67%	29.00%
TR15	100.0%	27.00%	100.0%	27.00%	100.0%	35.29%
TR16	84.21%	16.00%	85.71%	18.00%	82.61%	19.00%
TR17	100.0%	27.00%	100.0%	27.00%	100.0%	35.29%
TR18	100.0%	13.73%	100.0%	15.69%	84.21%	15.69%
average	97.01%	30.86%	97.02%	32.49%	91.45%	39.28%

Table 4 shows the performance before and after applying loosening constraint proposed in this paper. The "Existing method" results for application of the extended contextual constraint information from the hand-built correction rules are in the forms of Precision (P) and Recall (R). The "Step 1 loosening" results those for the extension of the replaceable words list, and the "Step 2 loosening" results are those for omission of the particle constraint.

As Table 4 indicates, the precision both of Step 1 and of Step 2 was somewhat diminished, but the average of recall was improved 5%. In particular, Step 2 markedly improved the recall. According to user preference, KSGC can be utilized without loosening for perfect detection and correction of error, or with loosening for enhanced recall.

5 Conclusions

In this paper, we propose loosening the case-particle constraint conditions to improve the recall of rule-based Context-sensitive Spelling Correction. We loosen the constraints by omission or addition of replaceable particles to the existing rules, according to the type of particle.

In upcoming research, we will apply the proposed method to additional correction rules to further verify its effectiveness. Additionally, we will review the changes of recall by combining extended contextual information with existing correction rules.

ACKNOWLEDGMENT

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MEST) (No. NRF-2012S1A5A2A03034298)

6 References

[1] Mino Kim, Hyuk-Chul Kwon, Hyun soo Choi , Aesun Yoon "Generalization of Context-sensitive Spelling Correction Rules using Korean WordNet", The Korean Institute of Information Scientists and Enginners 2013 KCC, vol.39, no.1, pp.653~655, 2013.06.

[2] S. D. Richardson, and L. C. Braden-harder, "The experience of developing a larger-scale natural language text processing system: CRITIQUE," Proc. The 2nd Annual Applied Natural Language Conference, pp.195-202, 1988.

[3] R. M. Weischedel, and N. K. Sondheimer, "Meta-rules as a basis for processing ill-formed input," Computational Linguistics, vol.9, no.3-4, pp.161-177, 1983.

[4] L. Z. Suri, "Language transfer: A foundation for correcting the written English of ASL signers," University of Delaware Technical Report TR-91-19, 1991.

[5] J. G. Ccrbonell, and P. J. Hayes, "Recovery strategies for parsing extragrammatical language," Computational Linguistics, vol.9, no.3-4, 1983

[6] Nam, Ki-shim & Ko Young-gun, Syntax · Semantic of Korean Language, TOP publisher, Seoul, 1991.