SESSION

SIMULATION, MODELING, VISUALIZATION METHODS AND RELATED ISSUES

Chair(s)

TBA

Comparison of interval and Monte Carlo simulation for uncertainty propagation in atmospheric dispersion model

El Abed El Safadi, Olivier Adrot, Jean-Marie Flaus
Laboratory: Grenoble–Sciences pour la Conception, l'Optimisation et la Production
(G-SCOP) 46, avenue Félix Viallet - 38031 Grenoble Cedex 1 France,
Email: Abed.Safadi@grenoble-inp.fr, Olivier.Adrot@grenoble-inp.fr, Jean-marie.Flaus@grenoble-inp.fr

Abstract— *In this paper, the problem of tackling uncertainty propagation in the estimation of the atmospheric dispersion of a toxic gas release is analyzed in order to assess the risk at the event of an accident. This estimation is based on an effect model associated with the studied dangerous phenomenon where some input variables and model parameters are known with imprecision. Two simulation approaches, Monte Carlo and interval analysis method, are applied and compared for estimating the confidence interval of risk intensity. Interval analysis method is superior in estimating all the possible values of intensity relative to the Monte Carlo simulation. A sensitivity analysis based on Sobol indices is applied in order to reduce the number of uncertain variables while conserving an acceptable precision of effect model. Furthermore, much less computational time is required for interval analysis method than for Monte Carlo simulation.*

Keywords: Risk assessment, sensitivity analysis, uncertainty propagation, interval analysis, Monte Carlo simulation.

1. Introduction

The risk assessment is a decision aid that aims to rank or quantify risks to human in order to prioritize management actions and the allocation of resources. Science-quality criteria require the assessment to be transparent, repeatable and systematic, and its estimations to be precise and accurate. Intensity estimations of accidental releases of hazardous gases have a significant impact on emergency planning around industrial plants and on the choice of risk prevention and mitigation barriers. This impact has a very high severity in urban areas and may be disastrous for the population [1]. Atmospheric dispersion simulations are dependent on a significant number of input variables (source term, weather conditions) as well as internal parameters of the dispersion model. This effect model includes parameters and variables which may be measured, estimated or deduced from a priori knowledge, but all of them are known with uncertainty [2], [3], [4], [5]. This leads to the inaccuracy in the results when computing the intensity of the dangerous phenomenon i.e the gas concentration. In order to perform this intensity, it is necessary to choose a suitable method able to express the uncertainty associated with parameters and variables of the dispersion model and after that, it is necessary to define a method for estimating the propagation of uncertainties in this model.

In the present study, two simulation approaches, Monte Carlo and interval analysis method are applied for estimating the confidence interval of intensity of the atmospheric dispersion. The obtained results by means of interval analysis method are compared here with Monte Carlo simulation results for uniform probability distributions in order to study the variability of uncertainty propagation in the two approaches and the computation time. A global sensitivity analysis based on Sobol indices is applied in order to determine how uncertainty in the model output can be apportioned to the different uncertain model inputs. This analysis allows reducing the number of uncertain model inputs while conserving an acceptable model precision.

The organization of this paper is as follows. In the next section the problem statement and the global sensitivity analysis are presented. In section 3, the Monte Carlo and interval analysis approaches for uncertainty propagation are explained. The application and the results obtained with the proposed approaches are reported in section 4. Finally, the conclusion is drawn in the section 5.

2. Sensitivity analysis

In this paper, the problem of tackling uncertainty in the estimation of the atmospheric dispersion of a toxic gas release is analyzed. For this reason, two simulation approaches, Monte Carlo and interval method are studied and compared in order to propagate uncertain inputs in a chosen analytical atmospheric dispersion model.

2.1 Uncertainty estimation

The concepts of risk and uncertainty are intimately linked. Risk occurs because the past, present and future are uncertain. A measurement is a process whereby the value of a quantity is estimated. When a measurement is made or when some quantity is calculated from the data, generally it is assumed that some exact or "true value" exists based on how is defined what is being measured (or calculated). A range of values, that should contain this "true value", is then usually specified. The most common way to define this values set is: Measured a calculated value = exact value $\pm$ uncertainty

2.2 Objective of sensitivity analysis

The sensitivity analysis is the study of how uncertainty on the output of a model (numerical or otherwise) can be apportioned to different sources of uncertainty in the model input [6]. This is useful as a guiding tool when the model is under development as well as to understand model behavior when it is used for estimation or for decision support. For mathematical models, sensitivity analysis is closely related to the study of error propagation, i.e. the influence that the lack of precision on model input will have on the output. Sensitivity studies can identify and prioritize the most influential inputs, decide which parameters need more investigations to be precisely determined and simplify the model by removing or making constant the less influential input factors. Two types of sensitivity analysis methods can be selected, local and global methods. The choice depends on the objective of the analysis, the number of uncertain input factors, the degree of regularity of the model, and the computing time for a single model simulation. In this study we are interested in global sensitivity analysis [7], specifically Sobol technique. Sobol method is a global sensitivity analysis (SA) technique which determines the contribution of each input (or group of inputs) to the variance of the output and they take into account the whole field of possible variation of the input variables. The usual Sobol sensitivity indices include the main and total effects for each input, but the method can also provide specific interaction terms [8], [9].

2.3 Estimated Sobol indices by Monte Carlo

In a general manner, the analytical model of atmospheric dispersion can be written in the form of a mathematical relation 1 describing the dangerous phenomenon at a given instant:

$$y = f(x_1, ..., x_p) \quad (1)$$

We assume in this study that the input variables $(x_1, ..., x_p)$ of the model are independent. To appreciate the importance of an input variable x_i on the variance of the output y , we study how the variance of y decreases if the variable x_i is fixed to a value w_i : $V(y|x_i = w_i)$ .The problem with this indicator is the choice of the w_i value of x_i which is solved by considering the expectancy of this quantity for all possible values of w_i : $E[V(y|x_i)]$. Thus, the variable x_i is more influent on the variance of y, when this amount is small. The formula of the total variance
$V(y) = V(E[y|x_i]) + E[V(y|x_i)]$,
leads to use in an equivalent manner the amount $V(E[y|x_i])$, which becomes larger when the variable x_i has a more important contribution to the variance of y. In order to use a standardized indicator, we define the sensitivity indices of y to x_i as [8]:
$S_i = \frac{V(E[y|x_i])}{V(y)}$.

Consider an N-sample $X_{(N)} = (x_{k1}, ..., x_{kp})_{k=1,...N}$ of realizations of the input variables $(x_1, ..., x_p)$, the index k denotes the k^{th} sample. The expectation of y, $E[y] = f_0$ and the variance $V(y) = V$ are classically estimated by:

$$f_0 = \frac{1}{N}\sum_{k=1}^{N} f(x_{k1}, ..., x_{kp}) \ , \ V = \frac{1}{N}\sum_{k=1}^{N} f^2(x_{k1}, ..., x_{kp}) - f_0^2 \quad (2)$$

The estimation of sensitivity indices requires a variance estimation of a conditional expectation. We remind a technique to estimate $V(E[y|x_i])$ due to Sobol [9].
Let us note : $V_i = V(E[y|x_i]) = E[E[y|x_i]^2] - E[E[y|x_i]]^2 = U_i - E[y]^2$ with $U_i = E[E[y|x_i]^2]$.
The variance of y being conventionally estimated by 2, Sobol proposes to estimate the quantity U_i, in other words the expectation of the square of the expectation of y conditional on x_i , as a conventional expectation where, all input variables can vary, except the variable x_i which is fixed. This requires two N samples of input variables, denoted $X^1_{(N)}$ and $X^2_{(N)}$:
$U_i = \frac{1}{N}\sum_{k=1}^{N} f(x^{(1)}_{k1}, ..., x^{(1)}_{k(i-1)}, x^{(1)}_{ki}, x^{(1)}_{k(i+1)} x^{(1)}_{kp}) \times f(x^{(2)}_{k1}, ..., x^{(2)}_{k(i-1)}, x^{(1)}_{ki}, x^{(2)}_{k(i+1)} x^{(2)}_{kp})$,
when the indexes (1) and (2) denote the associated N sample. The sensitivity indices of the first order of the x_i input are then estimated by:
$S_i = \frac{V_i}{V} = \frac{U_i - f_0^2}{V}$

3. General approach on uncertainty propagation

The aim of this approach is to make uncertainty evaluation internationally comparable. This methodology is also proposed by the new draft of the Guide to the expression Uncertainty in Measurement (GUM [10]). The methodology presented can be summarized in the following main steps:

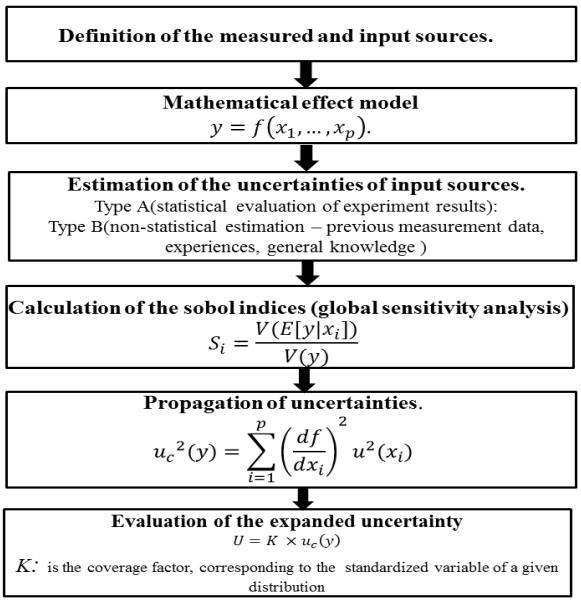


Fig. 1: Methodology for evaluating model uncertainty

3.1 Monte Carlo simulation for uncertainty propagation

Monte Carlo simulation [11], [12] is a computational mathematical technique, which performs model simulation by calculating the model output by substituting each uncertain model input by a particular feasible value. It then calculates outputs over and over, each time using a different set of random input values from the probability functions. Depending upon the number of uncertainties and the ranges specified for them, a Monte Carlo simulation could involve thousands or tens of thousands of recalculations before it is complete.

3.1.1 Monte Carlo simulation process

The Monte Carlo simulation process consists in two steps:

- Generation of a N sample $X_{(N)}$ of size p with uniformly distributed random values, where N is the number of simulations and p is the number of parameters. For each independent sample of size p the resulting model value of y is calculated.
- These N values of y are used to perform the propagation of uncertainties in the model output.

3.1.2 Implementation of the Monte Carlo simulation

Figure 2 present the calculation phase of uncertainty evaluation using Monte Carlo simulation to implement the uncertainty propagation.

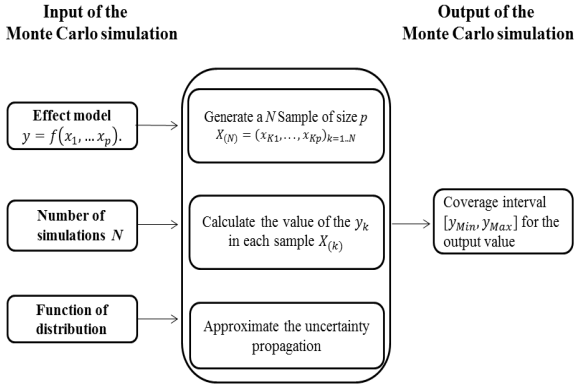


Fig. 2: The calculation phase of uncertainty propagation.

The inputs of the calculation phase of uncertainty propagation are firstly the mathematical effect model, secondly the number N of samples and thirdly the uniform distribution functions of the uncertain model inputs. Three main steps are executed during the implementation of the Monte Carlo simulation : generation of N samples of p input variables, evaluation of the model output for each sample and finally estimation of the model output and the associated uncertainty from the distribution function. The final result of uncertainty propagation is the coverage interval for the model output. Based from the N values $y_1, y_2, ..., y_N$, the uncertainty u is defined as:

$$u = \frac{y_{Max} - y_{Min}}{2} * \frac{100}{y_{NominalValue}}. \quad (3)$$

With $y_{NominalValue}$ is the output value of the model without uncertainty on the model inputs. The y_{Min} and y_{Max} define respectively the minimal and maximal values of $y_{i,i=1,...,N}$.

3.2 Interval analysis approach in uncertainty calculation

The sources of uncertainty are multiple, i.e. mathematical models with uncertain parameters, representation of real numbers on digital computers with finite precision, uncertain initial data. In some applications, it is necessary to know the influence of these uncertainties on the computed solution. To solve such problems, techniques based on interval analysis have been developed in particular by Moore [13], [14]. This tool allows calculating an overestimated interval containing with guarantee the feasible values of the model output. Interval modeling consists in describing a model uncertainty by an unknown bounded variable, whose known support defines its feasible value set. The interval containing a real uncertain variable x, whose value is comprised between a lower bound x^- and an upper bound x^+, is written:
$[x] = [x^-, x^+] = \{x \in \mathbb{R} | x^- \leq x \leq x^+\}$.
Note that no distribution function is required.

3.2.1 Interval Arithmetic

Interval arithmetic in its modern form was introduced by Moore [13], [14] and is based on arithmetic conducted on closed sets of real numbers. Mathematics elementary operations are extended to intervals. The operation result between two intervals is an interval that contains all the results of this operation between the different values contained in these intervals. The operation result on finite intervals is defined by two bounds which are obtained by working only on their bounds. In this way, interval arithmetic is an extension of real arithmetic. For a real arithmetic operation $\circ \in \{+, -, *, /\}$, the corresponding interval operation on intervals $[x]$ and $[y]$ is defined by:

$$[x] \circ [y] = \{x \circ y | x \in [x], y \in [y]\}. \quad (4)$$

Interval arithmetic considers the whole range of possible instances represented by an interval model. In the classic set-theory interval analysis, given a $\mathbb{R}^p$ to $\mathbb{R}$ continuous function $y = f(x_1, ...x_p)$, the interval united extension $[f]$ of f corresponds to the range of f-values on its interval argument $([x_1], ...[x_p])$ in $I(\mathbb{R}^p)$:
$[f]([x_1], ...[x_p]) = \{f(x_1, ...x_p) | x_1 \in [x_1], ...x_p \in [x_p]\} = [min\{f(x_1, ...x_p) | x_i \in [x_i]\}, max\{f(x_1, ...x_p) | x_i \in [x_i]\}] i = 1, ...p.$

3.2.2 Pessimism

Generally, the result of a series of operations between two or more intervals is not minimal; the obtained interval is pessimistic. This problem is mainly due to the dependence problem [15]. Considered a no degenerate interval $[x] = [x^-, x^+]$ and an arithmetic operation $\circ \in \{+, -, *, /\}$, then using the definition 4, we obtain:

$$[x] \circ [x] = \{x \circ y | x \in [x], y \in [x]\}. \quad (5)$$

According to 5, we see that bounded variables x and y are considered different despite the fact that we manipulate the same interval. So, dependency between bounded variables cannot always be taken into account when their interval supports are manipulated and this problem is called dependency phenomenon. For example, let $[x] = [-1, 1]$, then $[x] - [x] = [-1, 1] - [-1, 1] = [-2, 2] \neq \{0\}$, the interval operation overestimates the exact domain $\{0\}$. In a general manner, pessimism depends on the occurrence of interval variables in the expression of $[f]$. It leads to the very interesting guarantee property (reliable computing) of interval tool, but the overestimation may be important if unsuited interval extensions are manipulated. The interval computation can be considered as a semantic extension of f, since it admits the logical interpretation:
$(\forall x_1 \in [x_1])...(\forall x_p \in [x_p])(\exists y \in [f]([x_1], ..., [x_p]))y = f(x_1, ..., x_p)$.
This logical interpretation contains the set of all trajectories that verify the model equation.

3.2.3 The implementation of the interval analysis

Figure 3 presents the calculation phase of uncertainty evaluation using interval analysis method to implement the uncertainty propagation

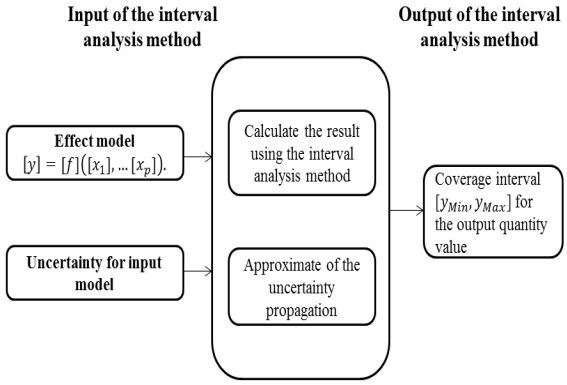


Fig. 3: The calculation phase of uncertainty evaluation

From the result value calculates by the interval analysis method, the propagation uncertainty u is defined as 3. The y_{Min} and y_{Max} are the lower and upper bounds of the caculated of $[y]$.

3.3 The interest of the sensitivity analysis

In one hand, identifying the most influential inputs from the sensitivity analysis aids to decide which uncertain inputs need more investigations in order to reduce their interval supports for the IAM and so to improve model accuracy. On the other hand to reduce the number of uncertain parameters by imposing less influential model inputs to their nominal values leads to reduce the occurrence of some interval variables, and thus the pessimistic in the result value.

4. Results and application

4.1 Mathematical effect model

In order to assess the severity of the risk when an undesirable and unexpected event occurs, a mathematical model can be used to compute physical effects coming from the considered event. In this study, an effect model is used to estimate or to predict the downwind gas concentration emitted from sources such as industrial plants, vehicular traffic or accidental chemical releases. This model represents the relationships between the inputs of the atmospheric dispersion model (wind speed, conditions emission point, release flow ....) and the gas concentration in the air at a specific point [16], [17], [18]. The concentration c_k of the released gas at a position x_k, y_k, z_k from a continuous source is given by the following Gaussian plume model:

$$\begin{aligned} c_k &= f(x_k, y_k, z_k, u_{ref}, z_{ref}, h, q, a_y, a_z, b_y, b_z, c_y, c_z) \\ &= \frac{q z_{ref}^{0.33}}{2\pi u_{ref} h^{0.33} (a_y x_k^{by} + c_y)(a_z x_k^{bz} + c_z)} * exp\left[-\frac{1}{2}\left(\frac{y_k}{a_y x_k^{by} + c_y}\right)^2\right] * \\ &\left\{exp\left(-\frac{1}{2}\left(\frac{z_k - h}{a_z x_k^{bz} + c_z}\right)^2\right) + exp\left(-\frac{1}{2}\left(\frac{z_k + h}{a_z x_k^{bz} + c_z}\right)^2\right)\right\} \end{aligned} \quad (6)$$

Where:
c_k is the concentration of the emission (in micrograms per cubic meter) at any point x_k meters downwind of the source, y_k meters laterally from the centerline of the plume, and z_k meters above ground level. The index k denotes different evaluations of the model output. q is the quantity or mass of the emission (in grams) per unit of time (seconds). u_{ref} is the wind speed (in meters per second) measured at a given altitude z_{ref}. h is the height of the source above ground level (in meters). The terms $a_y x_k^{by} + c_y$ and $a_z x_k^{bz} + c_z$ represent the dispersion parameters and depends on the distance x_k. They represent the standard deviations of a statistically normal plume in the lateral and vertical dimensions, respectively. The values of a_y, a_z, b_y, b_z, c_y and c_z , may be determined for each atmospheric stability class defined by Pasquill, by using the table given in [17].

4.2 Modelling of uncertain model inputs

Instead of representing an uncertain parameter or variable by a constant nominal value, this one can be defined as a bounded variable. In other words, its real value is unknown,

but it belongs to a set of feasible values defined as an interval whose bounds are known. In the following, an imprecision ρ_v means that an uncertain positive variable v is represented by the interval value set

$$[v(1-\rho_v), v(1+\rho_v)]. \quad (7)$$

This study has been applied to an example of accident involving nitric oxide gas. This gas is toxic and has a density relative to air of 1.04, so a Gaussian model is well suited to model dispersion of such a gas.

For a chosen stability class of C, the nominal values of dispersion parameters are: $a_y = 0.105, a_z = 0.066, b_y = b_z = 0.915$, and $c_y = c_z = 0$ [17]. We assume that the height of leakage point is $h = 2m$. The measured wind speed is $u_{ref} = 4.58m/s$, at a height of $z_{ref} = 40m$. The theoretical nominal value of the release flow is $q = 2216g/s$. In the following, the inaccuracy on some parameters and variables are considered: the release flow q with an uncertainty of $\rho_v = 5\%$, the wind speed u_{ref} with $\rho_v = 2.5\%$, the dispersion parameters a_y, a_z with $\rho_v = 2.5\%$ and b_y, b_z with $\rho_v = 1\%$, then for each parameter is defined an interval support of feasible values. These intervals are directly used to calculate the result of the interval analysis method. To compare this result with the Monte Carlo simulation result, we need on the one hand to generate random values for each model input contained in the same bounded support according to a uniform distribution function. On the other hand, we need the same indicator to express the propagation of uncertainties; for this reason, the used indicator for the both approaches is presented in the next section.

4.3 Uncertainty propagation before sensitivity analysis

Let note:

C_{Nom} : Concentration in studied point with the nominal value of the model inputs i.e. without uncertainty on these inputs.

$[MinC, MaxC]_{C-MC}$: Confidence interval of concentration in the studied point with the Monte Carlo simulation (MCS). The bounds $MinC$ and $MaxC$ define respectively the minimal and maximal values of the concentration c_k computed for $N = 100,000$ samples.

$[C_a, C_b]_{C-IA}$: Interval support of concentrations in the studied point computed with the interval analysis method(IAM).

$U - MC$: This indicator defines the uncertainty on the concentration in studied point with the Monte Carlo simulation, $U - MC = \dfrac{MaxC - MinC}{2} * \dfrac{100}{C_{Nom}}$. This indicator expresses the margin value (width) relative to the nominal gas concentration according to uncertainty on the model parameters. More precisely it represents the same quantity in percent than the imprecision ρ_v defined in 7 for uncertain model inputs.

$U - IA$: This indicator defines the uncertainty on the concentration in studied point computed with the modal interval analysis. It is defined in the same way by the following relation:
$U - IA = \dfrac{C_b - C_a}{2} * \dfrac{100}{C_{Nom}}$.

Case study

The objective is to determine the confidence interval of gas concentration in order to determine if it is lower (safety zone) or bigger (danger zone) than a given regulatory threshold that leads to avoid for example lethal or health irreversible effects. In our study the gas concentration is estimated in 5 points placed in the downwind of a source emitting nitric oxide gas. Concentration estimations given by IAM in these 5 points of study are compared with the outcome of a Monte Carlo approach. The number of samples for the latter is increased until no significant changes in the upper and lower bounds are observed. This leads to $N = 100,000$ samples which is a reasonable and classical choice according to the number $p = 6$ of uncertain model inputs. Uncertainties on model inputs in the MCS are represented in terms of uniform probability distributions for comparison with IAM. Multiplicative congruential random generation is used to return successive pseudo-random numbers. A looping program is implemented in java using the class random (). Table 1 illustrates the values of the studied model inputs with the added uncertainties,

Table 1: Model inputs

$q \pm 5\%$	$a_y \pm 2.5\%$	$a_z \pm 2.5\%$	$b_y \pm 1\%$	$b_z \pm 1\%$	$u_{ref} \pm 2.5\%$	x	y	z	z_{ref}	h
2216	0.105	0.066	0.915	0.915	4.58	350	8	2	40	2
						200	10			
						150	5			
						50	5			
						40	5			

Table 2 illustrates the concentration C_{Nom} with the nominal values of the inputs studied, ranges of concentrations with Monte Carlo approaches $[MinC, MaxC]_{C-MC}$ and modal interval analysis $[C_a, C_b]_{C-IA}$, finally the computed indicators $(U - MC, U - IA)$ of the gas concentration on the 5 points studied.

Table 2: Computed confidence intervals and indicators

$Point(x_k, y_k, z_k)$	C_{Nom}	$[MinC, MaxC]_{C-MC}$	$[C_a, C_b]_{C-IA}$	$U - MC$	$U - IA$
(350, 8, 2)	1.22	[1.09, 1.37]	[1.06, 1.41]	11,47%	14.34%
(200, 10, 2)	2.65	[2.39, 2.96]	[2.27, 3.09]	10,75%	15,47%
(150, 5, 2)	5.10	[4.56, 5.65]	[4.40, 5.89]	10,68%	14,63%
(50, 5, 2)	12.04	[10.91, 13.29]	[9.84, 14.64]	9,88%	19,93%
(40, 5, 2)	10.46	[9.25, 11.76]	[8.36, 12.98]	11,99%	22.10%

Figure 4 represents the propagation of uncertainties with MCS and IAM methods for the 5 points of coordinates (x_k, y_k, z_k).

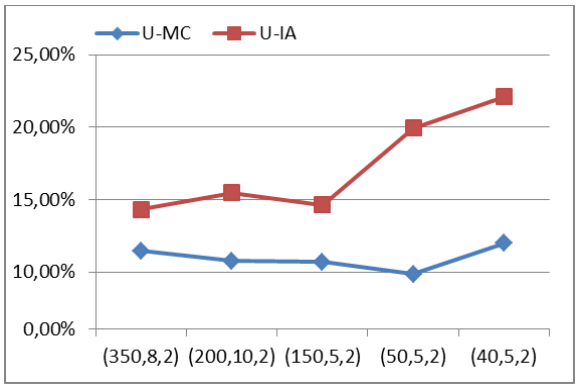


Fig. 4: Uncertainty propagation according to studied points

Interpretation of results

With the Monte Carlo simulation, the obtained results show that the uncertainty on model output varies between 9.88% and 11.99% in the different points studied. With the interval analysis method, the indicator varies between 14.34% and 22.10% and the uncertainty increases when the distance decreases between the source and the point studied. Concerning the execution time with the IAM, the calculation script needs 1.5 ms as execution time to obtain the concentration at a given point. With the MCS the execution time is 128 ms, so it can be deduced that the reduction time with the IAM is almost 98.8% compared to the MCS. These results show that the IAM provides larger confidence intervals relative to the MCS when some model inputs are uncertain. Several reasons explain the difference between the both approaches. The first one is due to the problem of pessimism of IAM because of multiple occurrences of some uncertain model inputs such as a_y, a_z, b_y, b_z. The second reason is that the Monte Carlo simulation needs to randomly generate each model input in its interval support. On one hand the MCS does not guarantee to take all the values in these bounded supports and on the other hand to take all the possible combinations of model input values. For comparison, the IAM takes into account all the feasible combinations which guarantees the results. In others word, IAM and MCS leads respectively to outer and inner approximations of the exact confidence interval on gas concentration. Concerning the execution time, the principal reason of the difference is the large number N of samples used by MCS.

4.4 Sensitivity analysis

Table 3 presents the results of the global sensitivity analysis for the studied uncertain model inputs. The first order indices are computed for 100 repetitions and $N = 100,000$ samples:
The result shows, that the less influential model inputs on the model output are a_z, b_y and b_z.

Table 3: Sensitivity analysis using Sobol indices

	Si :Sobol index of the first order	Confidence interval of Si
u_{ref}	0.14	[0.09, 0.23]
q	0.50	[0.44, 0.54]
a_y	0.33	[0.29, 0.41]
a_z	0.07	[-0.02, 0.12]
b_y	0.02	[-0.05, 0.10]
b_z	0.02	[-0.05, 0.10]

4.5 Uncertainty propagation after the sensitivity analysis

Based on the results of the sensitivity analysis, uncertainty on a_z, b_y and b_z has been removed, in other terms these model inputs are fixed on their nominal values. All the other model inputs q, a_y and u_{ref} are considered uncertain and can vary on their respective bounded supports.
Table 4 illustrates the values of the studied model inputs with the added uncertainties only on q, a_y and u_{ref}. Table 5 represents the obtained results for uncertainty propagation.

Table 4: Model inputs

$q \pm 5\%$	$a_y \pm 2.5\%$	a_z	b_y	b_z	$u_{ref} \pm 2.5\%$	x	y	z	z_{ref}	h
2216	0.105	0.066	0.915	0.915	4.58	350	8	2	40	2
						200	10			
						150	5			
						50	5			
						40	5			

Table 5: Computed confidence intervals and indicators

$Point(x,y,z)$	C_{Nom}	$[MinC, MaxC]_{C-MC}$	$[C_a, C_b]_{C-IA}$	$U-MC$	$U-IA$
(350, 8, 2)	1.22	[1.11, 1.34]	[1.10, 1.35]	9,42 %	10.24%
(200, 10, 2)	2.65	[2.44, 2.88]	[2.36, 2.97]	8,30 %	11,47%
(150, 5, 2)	5.10	[4.66, 5.58]	[4.58, 5.66]	9,01 %	10.60%
(50, 5, 2)	12.04	[10.96, 13.14]	[10.40, 13.87]	9,05 %	14,41%
(40, 5, 2)	10.46	[9.34, 11.63]	[8.82, 12.31]	10,94 %	16,69%

Figure 5 presents the comparison of the uncertainty propagation with MCS and IAM.

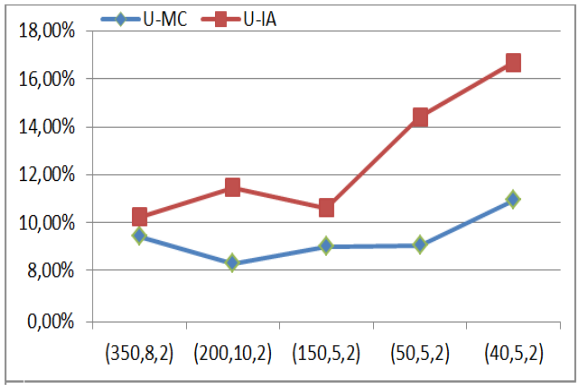


Fig. 5: Uncertainty propagation according to studied points

Interpretation of results

With the Monte Carlo simulation, the obtained result shows that the uncertainty on output model varies between 8.30% and 10.94% for the different points studied. With the interval analysis method, the indicator varies between 10.24% and 16.69% and the uncertainty increases when the distance decreases between the source and the point studied. With the IAM, the calculation script needs 0.7 ms as execution time to obtain the concentration of a given point. With the MCS the execution time is 105 ms, so the reduction time with the IAM is almost 99.3% compared to the MCS. The sensitivity analysis helps in fixing the less influential parameters as constant values. This in turn reduces the computation time which leads to a faster treatment. While some solutions may be loosed in the MCS method, the sensitivity analysis carried out is pertinent because this loss is reasonable. For the interval method, the number of lost solutions is greater than those in the MCS method, i.e. the reduction of the uncertainty on model output is more important. This is an expected result because some of the multi-occurrent variables (e.g. a_z, b_y, b_z) are fixed to constant and nominal values. Accordingly, this leads to a reduction in the dependence phenomenon between uncertain model inputs, so the reduction of the pessimism in the interval method and produces more accurate results. It is worth noting that the results obtained with IAM are almost equal to the confidence interval of MCS before carrying out the sensitivity analysis (see Table 2). Therefore, the sensitivity analysis may lead to an interesting simplification by improving the precision of the IAM model. In the context of risk assessment in the transport of hazardous materials, it is better to get all the possible estimations of gas concentration as with the method of analysis interval, instead of getting a part of the values as in the Monte Carlo simulation. An inner estimation of the interval confidence may lead to an inadequate and insufficient evacuation operation from the danger area, and then leads to serious injuries.

5. Conclusion

From this study we can conclude that the interval analysis method is a significant tool for estimating the propagation of uncertainties. In this study where several model inputs of the analytical model studied are uncertain , we find that the IAM provides larger confidence intervals relative to the MCS. Moreover the computation time is smaller with IAM than with the Monte Carlo simulation. The sensitivity analysis helps in fixing the less influential parameters as constant values. This in turn reduces the computation time which leads to a faster treatment and on the other hand leads to a reduction of the pessimism in the interval method and produces more accurate results. At last, the notion of reliable or guaranteed computation is crucial for risk assessment. The next objective is to extend the proposed approach which may be also used to determine all the geographical region in which gas concentration is less than a given regulatory threshold or used for other types of dangerous phenomenon like the explosion of dangerous goods.

Acknowledgment

This study is part of the Geofencing DG research project, and was initiated under the program Transport System of the French competitiveness cluster Lyon Urban Truck & Bus (LUTB). The authors would like to thank the Region Rhône-Alpes for funding this work in the Geofencing project.

References

[1] A-M. Tomasoni, "Models and methods of risk assessment and control in dangerous goods transportation systems, using innovative information and communication technologies" Sophia Antipolis, France, 2010.

[2] L. Abramson, "Model uncertainty from a regulatory point of view." Model Uncertainty : its Characterization and Quantification Workshops, Anapolis, (Maryland,USA), Tech. Rep., 1993.

[3] W. Oberkampf, S. DeLand, B. Rutherford, K. Diegert and K. Alvin, "Error uncertainty in modeling and simulation," *Reliability Engineering and System Safety*,2002.

[4] U. Pulkkinen and T. Huovinen, "Model uncertainty in safety assessment." Technical Report STUKYTO- TR 95, Finnish Center for Radiation and Nulcear Safety, 1996.

[5] E. Zio and G. Apostolakis, "Two methods for the structured assessment of model uncertainty by experts in performance assessments of radioactive waste repositories," *Reliability Engineering and System Safety*,1996.

[6] A. Saltelli, S. Tarantola, F. Campolongo and M. Ratto, *Sensitivity Analysis in Practice: A Guide to Assessing Scientific Models*, Wiley, 2004.[Online]. Available: http://books.google.fr/books?id=NsAVmohPNpQC

[7] B. Iooss, "Review of global sensitivity analysis of numerical models" *Journal de la Société Française de Statistique*,2011.

[8] J. Jacques, "Contributions à l'analyse de sensibilité et à l'analyse discriminante généralisée" Grenoble,France, 2005.

[9] I.M. Sobol, "Sensitivity estimates for nonlinear mathematical models" *Mathematical Modelling and Computational Experiments*,1993.

[10] Comité commun des guides en métrologie, Bureau international des poids et mesures, IFCC., CEI., ILAC., ISO.,UICPA., OIML. , *Guide pour l'expression de l'incertitude de mesure (GUM 1995 avec des corrections mineures): évaluation des données de mesure*, JCGM, 2008.[Online]. Available: http://books.google.fr/books?id=KkdRMwEACAAJ

[11] C. ROBERT and G. CASELLA. , *Monte Carlo Statistical Methods*, ser. Springer Texts in Statistics. Springer-Verlag GmbH,1999. [Online]. Available: http://books.google.fr/books?id=nlqVQgAACAAJ

[12] G. Fishman. , *Monte Carlo:*, ser. Springer Series in Operations Research and Financial Engineering. Springer 1996. [Online]. Available: http://books.google.fr/books?id=jK8TAhUaK9wC

[13] R. Moore, *interval analysis*, ser. Prentice-Hall series in automatic computation. Prentice-Hall 1960.

[14] R. Moore and F. Bierbaum, *Methods and applications of interval analysis*, ser. SIAM studies in applied mathematics. Siam 1979. [Online]. Available: http://books.google.fr/books?id=3_JQAAAAMAAJ

[15] T. Raissi, "Méthodes ensemblistes pour l'estimation d'état et de paramètres" Paris, France, 2004.

[16] Committee the Prevention of Disasters, *Methods for the calculation of physical effects due to releases of hazardous materials-yellow book* Voorburg, Netherlands, 2005.

[17] Documentation INERIS, "Méthodes pour l'évaluation et la prévention des risques accidentels (DRA-006)" Tech. Rep., 2006.[Online]. Available: http://www.ineris.fr

[18] Documentation INERIS, " Emissions accidentelles substances chimiques dangereuses dans l'atmosphére seuils de toxicité aigue" INERIS –DRC –08–94398–12846A

Flow Simulation Models for WinDAM

Mitchell L. Neilsen
Dept. of Computing and Info. Sciences
Kansas State University
234 Nichols Hall
Manhattan, KS, USA

Abstract

Windows™ Dam Analysis Modules (WinDAM) is a set of modular software components that can be used to analyze overtopped earthen embankments and internal erosion of embankment dams. These software components are being developed in stages. The initial computational modules address routing of floods through the reservoir with dam overtopping and evaluation of the potential for vegetation or riprap to delay or prevent failure of the embankment. Subsequent modules incorporate dam breach analysis. Current work is underway to include analysis of internal erosion, non-homogeneous, zoned embankments, and the analysis of various other forms of embankment protection.

The focus of this paper is on the development of new computational fluid dynamics (CFD) models from existing WinDAM models, and to determine the resulting flows using OpenFOAM. The next step will be to use these models to conduct coupled analysis at the particle-fluid level by coupling new erosion models based on the existing WinDAM erosion models with these CFD models.

Keywords: Computational fluid dynamics, erosion, hydraulic modeling, hydrology, simulation.

1. Introduction

Windows™ Dam Analysis Modules (WinDAM) is a set of modular software components that can be used to analyze overtopped earthen embankments and internal dam erosion. The development of WinDAM is staged. The initial computational model addresses routing of the flood through the reservoir with dam overtopping and evaluation of the potential for vegetation or riprap to delay or prevent failure of the embankment. The first module, WinDAM A+, also incorporates the auxiliary spillway erosion technology used in SITES. However, unlike SITES, it allows a user to simultaneously analyze up to three auxiliary spillways and embankment erosion on the dam. The next computational model, WinDAM B, incorporates dam breach analysis; i.e., the breach failure of a homogeneous embankment through overtopping and drainage of stored water in the reservoir. In addition, work is currently underway to include analysis of internal erosion, analysis of non-homogeneous embankments, and analysis of other forms of embankment protection in WinDAM C. The two most common causes of earthen embankment and levee failure are overtopping and internal erosion [14]. Example of overtopping analysis in the lab and internal erosion in the field is shown below in Figure 1.

WinDAM is designed to address the dam safety concerns facing the national legacy infrastructure of over 11,000 small watershed dams constructed with US Federal involvement over a seventy-year period. The US Department of Agriculture -Agricultural Research Service (USDA-ARS), US Department of Agriculture-Natural Resources Conservation Service (USDA-NRCS), and Kansas State University are working jointly to develop and refine this software. Public Law 78-534 – Flood Control Act of 1944 started the small watershed program, and it was followed by Public Law 83-566 – Watershed Protection and Flood Prevention Act of 1954. Starting in 1958, an average of one dam per day was constructed over a period of twenty years.

Figure 1. Overtopping and internal erosion of dams

Most flood routing of dams before the middle 1960's was computed manually. Then, routing software on computers began to replace manual methods. In 1983, the USDA-SCS-ARS Emergency Spillway Flow Study Task Group (ESFSTG) was formed to develop better technology for earth spillway analysis. The ESFSTG

collected data on dams that experienced either emergency spillway flow at least three feet deep or significant damage during a storm event. Approximately one hundred sites were selected for more in-depth evaluation and data collection, and data analysis began in 1990 from the field spillway data initially collected. Tests were conducted at the USDA-ARS outdoor Hydraulic Engineering Research Unit (HERU) Laboratory near Stillwater, Oklahoma, to further understand spillway performance processes such as flow concentration, vegetal cover failure, surface detachment, and headcut migration as shown in Figure 2.

Figure 2. Large flume test at USDA HERU

These findings were incorporated into the DAMS2 software, and then into Stability and Integrity Technology for Earth Spillways (SITES) software in 1994. The bulk length concept was replaced by SITES spillway erosion modeling technology in other USDA-NRCS references. Although SITES may be used to analyze existing dams and spillways, it was developed primarily for design and was developed over a period in which computational capability was much more limited than today. The legacy infrastructure of aging structures also means a transition from design of new structures to analysis of existing structures. For example, existing structures may overtop as a result of watershed changes or sediment deposition within the flood pool leading to inadequate spillway capacity. WinDAM builds on and extends the existing technology in SITES to provide the needed capability for these types of analyses.

Windows™ Dam Analysis Modules (WinDAM) is a collection of modular software components that can be used to design and analyze the performance of earthen dams. The focus of the initial collection of computational modules is to evaluate earth dams subjected to flooding that may result in overtopping of the dam embankment and auxiliary spillway(s) [1]. The reservoir routing model incorporated into the software includes outflow from a principal spillway, up to three auxiliary spillways, and over the top of the dam embankment. For conditions where overtopping of the embankment is predicted, the hydraulic attack on the downstream face can also be evaluated using the initial software modules in WinDAM A+. The downstream face of a dam is typically protected using vegetation or riprap. WinDAM A+ has been extended to include erosion and breach computations for conditions where the hydraulic attack exceeds that which can be withstood by the vegetal or riprap lining, and the resulting modules are in WinDAM B. The next version, WinDAM C, will incorporate analysis of failures caused by internal erosion or piping failures. To evaluate erosion in each auxiliary spillway, the SITES Spillway Erosion Analysis module with Latin Hypercube Sampling (SSEA+LHS) is integrated with WinDAM A+. The Embankment Erosion Module is extended to include a Breach Analysis Module. The current model assumes the dam has a homogeneous embankment. It is most applicable for the analysis or design of embankments constructed from cohesive soil materials. It is anticipated that the model will be expanded to handle zoned embankments in WinDAM D. The breach technology enabling this expansion is currently under development.

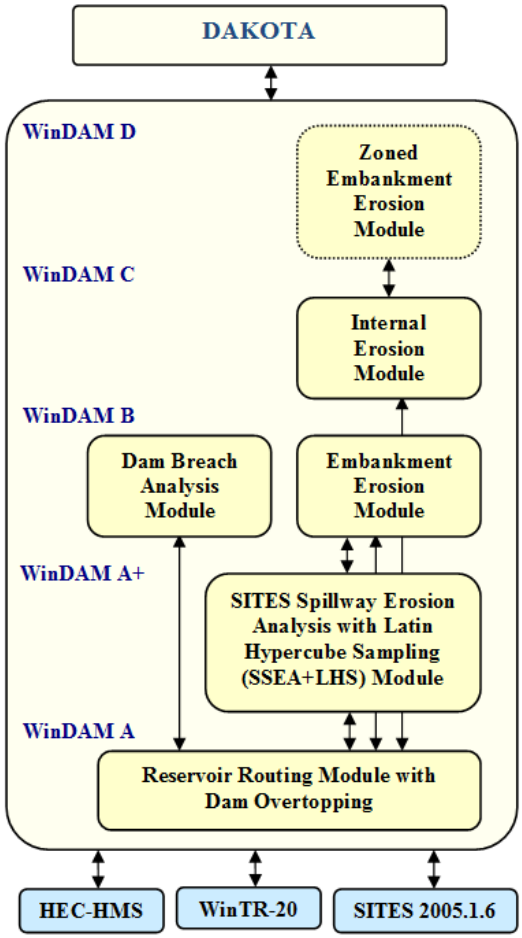


Figure 3. WinDAM software architecture

Inputs to WinDAM include a description of the reservoir inflow hydrograph, reservoir storage capacity, all spillway properties, the dam cross section and profile, properties of the embankment, and input parameters for the breach analysis module. Inflow hydrographs can also be obtained automatically from other reach routing software, such as SITES 2005.1.6, SSEA+LHS [2], HEC-HMS [3], HEC-RAS, or WinTR-20 as shown in Figure 3.

Outputs include a description of the reservoir water surface variation with time, the hydrographs associated with outflow through each of the spillways and over the top of the embankment, and a description of the attack on the dam embankment and downstream embankment face. Output hydrographs can be directed to external reach routing software. Output information is generated in both text and graphical format. The software generates ASCII text and/or XML control files for the model simulator which performs the model calculations. Output from the simulator is written to intermediate XML and/or fixed-format ASCII text files that can be read by a Graphical User Interface (GUI) to display results in both text and graphical format. Due to the well-defined interfaces that automatically convert data to and from different forms, it is easy for software developers to interface the system with existing analysis software and with software under development. Templates that can be used in conjunction with DAKOTA are also automatically generated.

In the DAKOTA system, a strategy is used to create and manage iterators and models [4]. A model contains a set of variables, an interface, and a set of responses, and an iterator operates on the model to map the variables into responses using the interface. The WinDAM system is used to automatically generate DAKOTA input files. For parameter studies, the user indirectly specifies these components through strategy, method, model, variables, interface, and responses keywords. Then, DAKOTA is invoked to iterate on the WinDAM simulation models, or vice versa, as needed to generate output. Instead of having WinDAM drive the analysis, we can also allow DAKOTA to be used to drive the analysis in an iterative fashion [6].

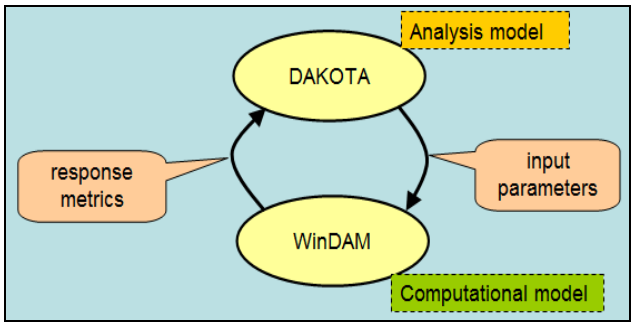


Figure 4. Iterative analysis

Dakota supports several different options for the Design and Analysis of Computer Experiments (DACE):

- *Sensitivity Analysis (SA)* - determine which inputs have the most influence on the output.
- *Uncertainty Analysis (UA)* - compare the relative importance of model input uncertainties on output.
- *Response Surface Approximation (RSA)* - use sample input and output to create an approximation to the simulation output; e.g., neural net, etc.
- *Uncertainty Quantification (UQ)* - take a set of distributions on the inputs and propagate them through the model to obtain distributions on the outputs.

At the same time, we are working to refine the models used within WinDAM to leverage advances in fluid flow models. The focus of this paper is on the development of new computational fluid dynamics (CFD) models from existing WinDAM models, and to determine the resulting flows using OpenFOAM. The next step will be to use these models to conduct coupled analysis at the particle-fluid level by coupling refined erosion models based on the existing WinDAM erosion models with these new CFD models as shown in Figure 5.

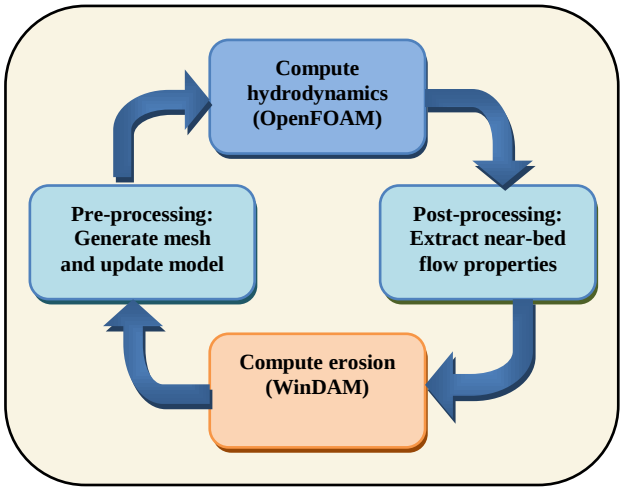


Figure 5. Coupled iterative analysis

In what follows, Section 2 describes WinDAM and the development of computational fluid dynamics models from existing WinDAM models that are used for overtopping/breach and internal erosion analysis; e.g., WinDAM B and WinDAM C models, and the integration of WinDAM with other tools to perform coupled analysis. Section 3 covers the use of OpenFOAM to compute the resulting hydrodynamics and visualization of the results using ParaView. Finally, Section 4 concludes the paper.

2. Computational Fluid Dynamics Models

In WinDAM, hydraulic flow is routed through the reservoir by balancing inflow, outflow, and storage under the assumptions of a level reservoir surface with all outflow being a function of reservoir water surface elevation. Stage-storage properties of the reservoir are entered in tabular format with elevation in feet and the corresponding surface area in acres or storage volume in acre-feet. Reservoir inflow hydrographs are entered into WinDAM as series of time-discharge pairs with time in hours and flow in cubic feet per second (cfs) or cubit meters per second (cms).

Inflow hydrographs are normally computed using other software that is capable of generating a rainfall-runoff hydrograph. The time increment used for entry of the hydrograph is normally used in performing the routing and erosive attack computations.

The computational model currently incorporated into the WinDAM software assumes stepwise steady-state

flow and a level water surface in the reservoir. The mass balance equation governing flow through the reservoir for any given time step may be obtained by averaging conditions over the time step. The inflow to the reservoir is a known function of time only, and is obtained through application of appropriate hydrologic models such as SITES 2005.1.6, HEC-HMS [3], or WinTR-20. The outflow from the reservoir is the sum of the outflow from all spillways and the outflow over the top of the dam. Using the assumptions of a level water surface in the reservoir and stepwise steady flow, each of the individual outflows may be treated as a unique function of the reservoir water surface elevation. Likewise, the storage volume in the reservoir becomes a unique function of the reservoir water surface elevation.

WinDAM B

The primary purpose of WinDAM B is threefold:

- Hydraulically route one input hydrograph through, around, and over a single earthen dam.
- Estimate auxiliary spillway erosion in up to three earthen or vegetated auxiliary spillways.
- Estimate erosion of the earthen embankment caused by overtopping of the dam embankment.

Since WinDAM B does not include any specific hydrology component, the user must create the input hydrograph using other software. This allows the user the flexibility to choose the hydrologic software most suitable for analysis of site conditions; e.g., HEC-HMS, etc.

WinDAM B assumes the embankment of the dam is a homogenous earthen material. Many USDA-NRCS dams are homogenous earthen fill, so the WinDAM B model applies. Future versions of WinDAM will address zoned embankments where each zone exhibits different erosion resistance from other zones.

Most existing USDA-NRCS dams are built with a single earthen auxiliary spillway. In rehabilitation of old USDA-NRCS-designed dams, it is more common to also utilize additional auxiliary spillways. As a result, WinDAM B allows the user to input up to three auxiliary spillways, each spillway with a zoned embankment and different physical characteristics.

Computation of the discharge through the area of the breach, if any, is unit discharge based on the effective width. If breach is to be evaluated, the associated erosion is assumed to be initiated in an area corresponding to maximum unit discharge over the top of the dam.

Following breach initiation, the unit discharge is computed assuming negligible energy loss from the reservoir to the hydraulic control and critical flow conditions with hydrostatic pressure at the hydraulic control. The processes that determine the erosion during embankment breach are dependent on the breach geometry and the breach area discharge.

The way in which the erosion will progress depends on the local geometry and discharge. Initially, the headcut (local vertical) may not be sufficiently high to generate the plunging action that is associated with typical headcut advance. Likewise, during latter stages of the process, the headcut may become submerged.

WinDAM C

The primary purpose of WinDAM C is to extend prior models to include internal erosion models developed as a result of empirical analysis at the USDA-ARS HERU as shown in Figure 6.

Figure 6. Internal erosion analysis at USDA-ARS HERU

Next, we turn our attention to the construction of computational fluid dynamics models to represent the flow through dams.

Computational Fluid Dynamics Models

Several different tools are used to facilitate the efficient development of computational fluid dynamics (CFD) models for OpenFOAM to model the hydraulic flow over dams and through auxiliary spillways. The ultimate goal is to efficiently extend the analysis of WinDAM models to incorporate coupled analysis as shown in Figure 5. For this paper we only focus on the development of CFD models from existing WinDAM models.

In the simplest case, WinDAM models the dam as a simple two-dimensional trapezoidal cross section by specifying the crest width, upstream and downstream slopes, and height as shown in Figure 7.

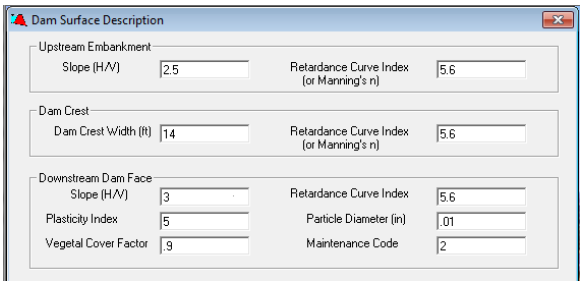


Figure 7. WinDAM B dam surface description input

The base width can be easily computed from the other inputs.

A user can use computer-aided design (CAD) software to generate an STL (STereoLithography) file or manually generate an OpenFOAM blockMeshDict ASCII file for a WinDAM model using a tool such as HexBlocker.

HexBlocker

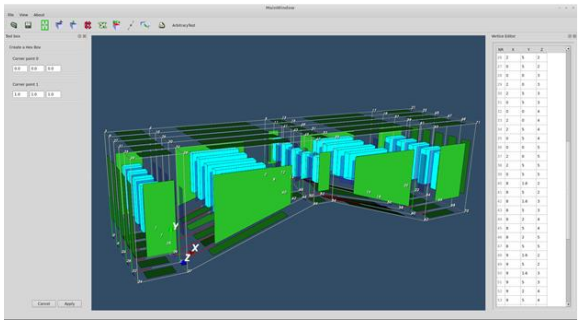


Figure 8. WinDAM dam model in HexBlocker

HexBlocker is an unofficial tool created by Nicholas Edh and is released with a GPL license [15]. While it does not support all the features that a blockMeshDict file can contain, it allows a user to visually create the file one hex block at a time. Figures 8 and 9 show examples of the dams and spillway models constructed using HexBlocker. A parser is developed to automatically construct these models from the input WinDAM model. The next step will be to refine the parser to generate an input file for snappyHexMesh to enable refinement of the mesh so that more elements are assigned to critical regions near the crest and toe. HexBlocker can be used to verify the mesh file, blockMeshDict, generated by the parser.

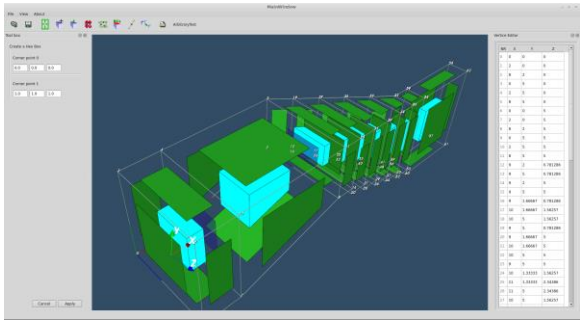


Figure 9. Stepped converging wall spillway

ParaView

Both input models and output results can be visualized using another widely used tool called ParaView. After creating a mergePatchPair for each step and executing the command blockMesh, the mesh is ready for use and can be viewed using Paraview as shown in Figure 10.

OpenFOAM includes a custom version of ParaView with readers to import output generated by OpenFOAM. ParaView can also be used to check the mesh generated as shown in Figure 10.

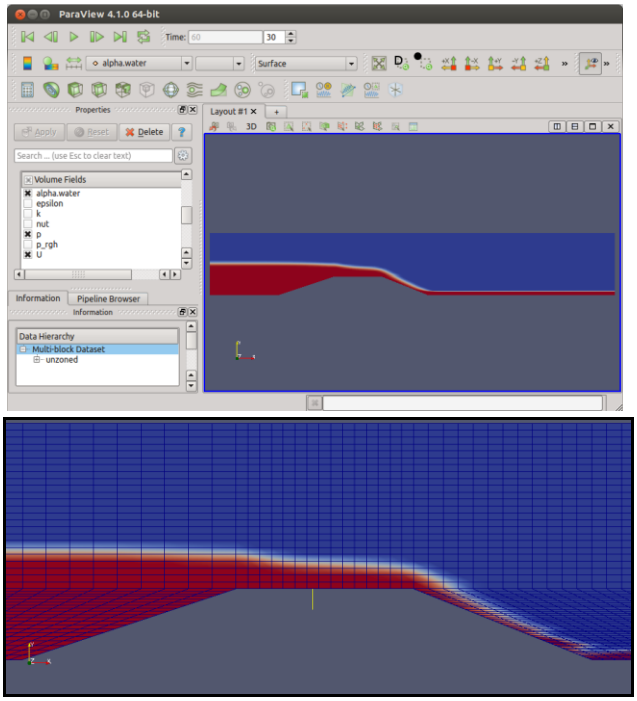


Figure 10. View CFD model and flow in ParaView

In addition to simple dam cross sections, WinDAM allows for zoned embankments for auxiliary spillway erosion analysis as shown in Figure 11.

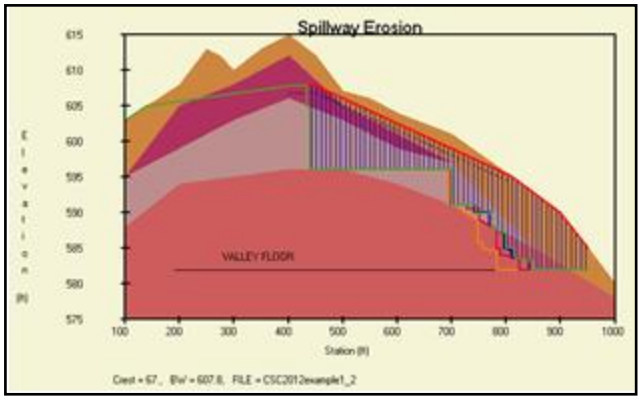


Figure 11. Auxiliary spillway erosion analysis

The next step will be to automatically generate the corresponding CFD models for zoned embankments.

3. OpenFOAM Analysis

In order to perform the fluid dynamics calculations on the dam, a computational fluid dynamics program is needed. OpenFOAM (Open Source Field Operation and Manipulation) is chosen for this task [16]. It is a "free, open source CFD software package which has a large user base across most areas of engineering and science, from both commercial and academic organizations.

OpenFOAM has an extensive range of features to solve anything from complex fluid flows involving chemical reactions, turbulence and heat transfer, to solid dynamics

and electromagnetics [16]. The OpenFOAM package contains fourteen solvers for incompressible flows, fifteen solvers for multiphase flows, ten solvers for combustion, seven solvers for buoyancy-driven and heat-driven flows, eleven solvers for compressible flows, and several others for particles and electromagnetics. Since OpenFOAM is open source, users have complete freedom to modify these solvers to fit their needs and even create new solvers. Complementing the solvers, there are over 170 utility applications for performing pre- and post-processing tasks.

The most important parts of pre-processing are creating the geometry of the dam and setting physical parameters. OpenFOAM uses simple ASCII text files to allow users to input this information. These text files all contain either assignments or dictionaries of assignments and other dictionaries. To define the geometry, a blockMeshDict file must be created or generated. A blockMeshDict file contains 1 dictionary and 6 assignments.

Next, the user needs to set the physical parameters. Our goal is to automatically generate these files based on input from WinDAM.

Once the simulation has been created the user can run the solver and begin post-processing. OpenFOAM comes packaged with a custom version of ParaView as its default visualization tool. However, OpenFOAM supports several other third-party post-processing products as well.

Analysis

All models were processed on a quad-core Intel® Core™ i7-3740QM CPU running at 2.7 GHz, and a desktop machine with 12 cores running at 3.4 GHz using Message Passing Interface (MPI). Models can be decomposed in OpenFOAM using decomposePar, and the solution can be reconstructed using reconstructPar. For the 3-d models, a decomposition of x=3, y=2, and z=2 was chosen because the x axis had the longest length, and for the 2-d models a decomposition of x=4, y=1, and z=1 was used on the quad-core machine. OpenFOAM's interFoam solver was then run on each dam to produce thirty seconds of data, producing the results shown in Figures 10, 12-15. The 2-d model shown in Figure 10 only took 5 minutes to generate thirty seconds of output data.

The simple dam in Figure 12-13 took 52 minutes to create thirty seconds of data with the interFoam solver. The water line was placed 0.2 meters below the top of the dam at 1.8 meters.

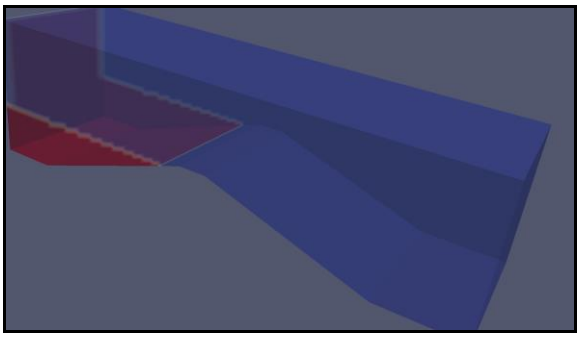


Figure 12. Simple dam at initial time

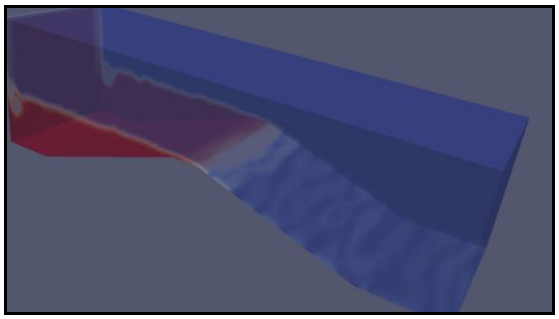


Figure 13. Simple dam initially and after 7 seconds

The stepped dam with converging wall took 94 minutes to create thirty seconds of data with the interFoam solver.

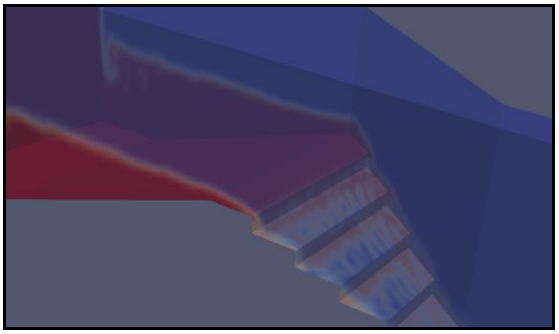


Figure 14. Stepped converging wall spillway after 7 secs

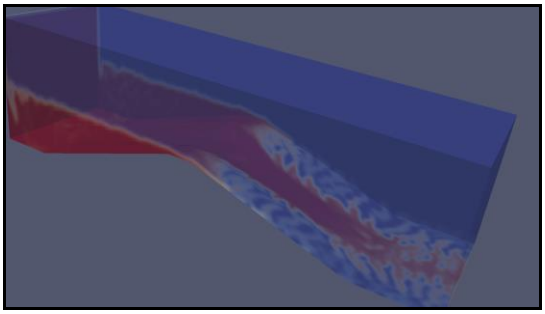


Figure 15. Simple auxilliary spillway after 30 secs

Uncertainty Analysis

The goal of uncertainty analysis is to obtain a better understanding of the probable range of outputs given that there is a certain amount of uncertainty in the input. In particular, based on uncertain inputs, determine the distribution function (uncertainty) of the outputs and probabilities of failure (reliability metrics); identify the statistical measures (mean, variance, etc.) of the outputs; and identify the inputs whose variance contribute most to variance in the outputs (global sensitivity analysis) [4].

Spillway designs are compared by determining both the stability and integrity of the spillway when it is subjected to a given design storm. In a typical design, three types of hydrographs are used: principal spillway hydrographs, stability design hydrographs, and freeboard hydrographs. A *principal spillway hydrograph* is used to size the principal spillway and set the elevation of the crest of the emergency or auxiliary spillway. The principal spillway is typically a conduit through the dam

used to pass low flows, whereas the auxiliary spillway is often an open channel capable of passing infrequent large flows. Earth auxiliary spillways are typically wide trapezoidal channels vegetated as appropriate for the local area. A *stability design hydrograph*, when routed through a reservoir, generates the maximum auxiliary spillway outflow that the reservoir will be expected to pass without erosion damage. For the design to be stable, erosion thresholds must bound hydraulic stresses that lead to the initiation of erosion. For flows larger than the stability design hydrograph, spillway erosion may occur, and the spillway may require some maintenance. A *freeboard hydrograph* represents the maximum flow for which the structure is designed. The integrity of the auxiliary spillway, as represented by its resistance to breach, is evaluated for the spillway outflow associated with this hydrograph. Naturally, this is the most important consideration in designing an earth (soil, rock, or both) spillway. Even though extremely large discharges may cause significant erosion, the spillway must not breach during passage of the *freeboard hydrograph*.

Figure 16. Empirical analysis of breach widening

Breach potential is a function of the spillway system, the characteristics of the spillway outflow hydrograph, the erodibility of the earth materials, the spillway layout, bottom width, and maintenance. Integrity analysis is based on the idea that some erosion is allowable if its occurrence is infrequent, maintenance is provided [7]. Models are developed in conjunction with ongoing empirical analysis as shown in Figure 16.

4. Conclusions

WinDAM is being developed in stages to evaluate the performance of earth dams. Existing modules with well-defined interfaces enable efficient integration of existing legacy software and future enhancements. The system provides tools that can be used to better understand the structure, function, and dynamics of such structures. This paper describes how WinDAM models can be converted into CFD models using OpenFOAM. The next step will be to couple these models with DEM models to model the erosion that results from the given flows. Finally, uncertainty quantification and sensitivity analysis can be incorporated by linking the development environment with DAKOTA.

Acknowledgements

We would like to thank the USDA-ARS and USDA-NRCS for use of images used in this paper.

References

[1] D.M. Temple, G.J. Hanson, and M.L. Neilsen, "WinDAM -- Analysis of overtopped earth embankment dams", In *Proc. of the ASABE Annual Conference*, Paper Number 062105, 2006.

[2] M.L. Neilsen, D.M. Temple, and J.L. Wibowo, "A distributed hydrologic simulation environment with latin hypercube sampling", In *Proc. of the Intl. Conf. on Env. Modelling and Simulation*, No. 432-032, St. Thomas, USVI, Nov. 22-24, 2004.

[3] United States Army Corps of Engineers, "Hydrologic modeling system HEC-HMS User's Manual", CPD-74A, Ver. 3.5, USACE, HEC, 2010.

[4] B.M. Adams, W.J. Bohnhoff, K.R.Dalbey, J.P. Eddy, M.S. Eldred, D.M. Gay, K. Haskell, P.D. Hough, and L.P. Swiler, "DAKOTA, A Multilevel Parallel Object-Oriented Framework for Design Optimization, Parameter Estimation, Uncertainty Quantification, and Sensitivity Analysis: Version 5.0 User's Manual," Sandia Technical Report SAND2010-2183, Dec. 2009. Updated Dec. 2010 (Ver. 5.1) Updated Nov. 2011 (Ver. 5.2)

[5] D.M. Temple and G. J. Hanson, "Earth dam overtopping and breach outflow", In *Proc. of the World Water and Environmental Resources Congress*, Anchorage, Alaska, ASCE, 8 pp., 2005.

[6] M.L. Neilsen, "Global sensitivity analysis of dam erosion models", in Proceedings of the 10th International Conference on Scientific Computing, Paper No. CSC-3502, July 22-25, 2013.

[7] United States Department of Agriculture, Natural Resources Conservation Service, "Earth spillway erosion model", Ch. 51, Part 628, Dams, *National Engineering Handbook*, 210-VI-NEH, 1997.

[8] D.M. Temple, J. Wibowo, M.L. Neilsen, "Erosion of earth spillways", In *Proc. of 23rd United States Society on Dams (USSD) Annual Meeting and Conference*, pp. 331-339, 2003.

[9] M.L. Neilsen and D.M. Temple, "A concurrent simulation model for analysis of water control structures at the watershed scale", In *Proc. of the Intl. Conf. on Par. and Dist. Proc. Tech. and Apps.*, (PDPTA 2010), pp. 1565-1570, June 26-29, 2000.

[10] M. D. McKay, W.J. Conover, and R. J. Beckman, "A comparison of three methods for selecting values in the analysis of output from a computer code", *Technometrics*, 21(2):239-245, 1979.

[11] N. Edh, "HexBlocker". Software retrieved 3/2014, from https://github.com/nicolasedh/hexBlocker.

[12] OpenFOAM - software and documentation retrieved from the www.openfoam.org, 2014.

Periodic, Aperiodic, and Partly Periodic Clocks in Scientific Simulations

Clarence Lehman[1] **and Adrienne Keen**[2]
[1]University of Minnesota, 123 Snyder Hall, 1475 Gortner Avenue, Saint Paul, MN 55108, USA
[2]London School of Hygiene and Tropical Medicine, Keppel Street, London WC1E 7HT, UK

"Professor Einstein says that time differs from place to place...
If time is not true, what purpose have watchmakers?"
—Allen Moore, Dave Gibbons, 1986

Abstract— *In microscale simulations that forecast stochastic times for future events, most new events develop internally, either within a simulated entity or from interactions among entities in the simulation. Births, deaths, infections, migrations, and other events can arise internally in this way. However, some events arise exogenously, entirely outside the system, while others are triggered routinely by calendar times rather than by internal conditions. For example, in simulating the population of a region without its surrounding world, immigration of new individuals into the region would be exogenous, occurring at fixed or random intervals. For individuals in the simulation, the timing of medical checkups or other appointments could similarly occur at regular or irregular intervals, independently of other conditions in the simulation. Here we describe a way to implement clocks for such events, inspired by work on a large-scale epidemiological simulation program* [1]. *The clocks can tick deterministically or randomly following any probability distribution. Two forms of clocks, periodic and aperiodic, simulate natural processes such as oscillatory signals or radioactive decay. A third form, which we call partly periodic, does not typically occur in nature, but is devised to match empirical counts exactly. The design we describe is general and can be applied to any individual-based, agent-based, discrete event, or other microscale simulation model that stochastically schedules future events* [2].

Keywords: simulation clocks, microscale modeling, individual-based modeling, periodic events, aperiodic events, waiting-time paradox

1. Introduction

Microscale models simulate individuals directly rather than combining them into continuous fluids, probability distributions, or populations [3]. Individual-based and agent-based models are examples. Advances in computational power and methods now make microscale models competitive in simulating macroscale models defined as ordinary, partial, or integro-differential equations, as well as in simulating systems that cannot readily be formulated in macroscale terms. An efficient approach to microscale modeling schedules all events into the future rather than testing for events at each time step, and the existence of constant-time algorithms for managing schedules and groups of individuals [2] [4] [5] now allows hundreds of millions of individuals—from molecules to hayseeds to orca whales—to be tracked.

In this paper we describe our approach to "clocks" for events in large-scale simulations. We assume that a global timeline measures the flow of all events in the simulation. Any number of clocks may tick concurrently on this global timeline, with a new event scheduled at each tick of each clock. We explain the structure and dynamics of simple clocks, with algorithms for the clocks given in detail, and how collections of simple clocks may be combined into more elaborate custom clocks. The methods we describe may prove useful to scientists in ecology, epidemiology, economics, and other disciplines that employ individual-based, agent-based, discrete-event, or other forms of microscale modeling.

2. Events

Most events scheduled in microscale simulations arise endogenously from actions of individuals, conditions of individuals, or the environment of the system. For example, births in an ecological population model arise repeatedly as individual plants or animals reach appropriate age and condition. Infections in an epidemiological model arise when infectious individuals encounter susceptible individuals.

Yet some events are exogenous, not caused directly by actions within the system. Examples are seeds arriving on a simulated island from a mainland source, mutations triggered by external radioactive decay, and unpredictable fluctuations in the stock market. Other events are internal but more routine and temporal, occurring with little or no reference to the states of individuals or to interactions among individuals. Examples are scheduled checkups at a medical center, periodic events such as anniversaries, and random events like winning the lottery.

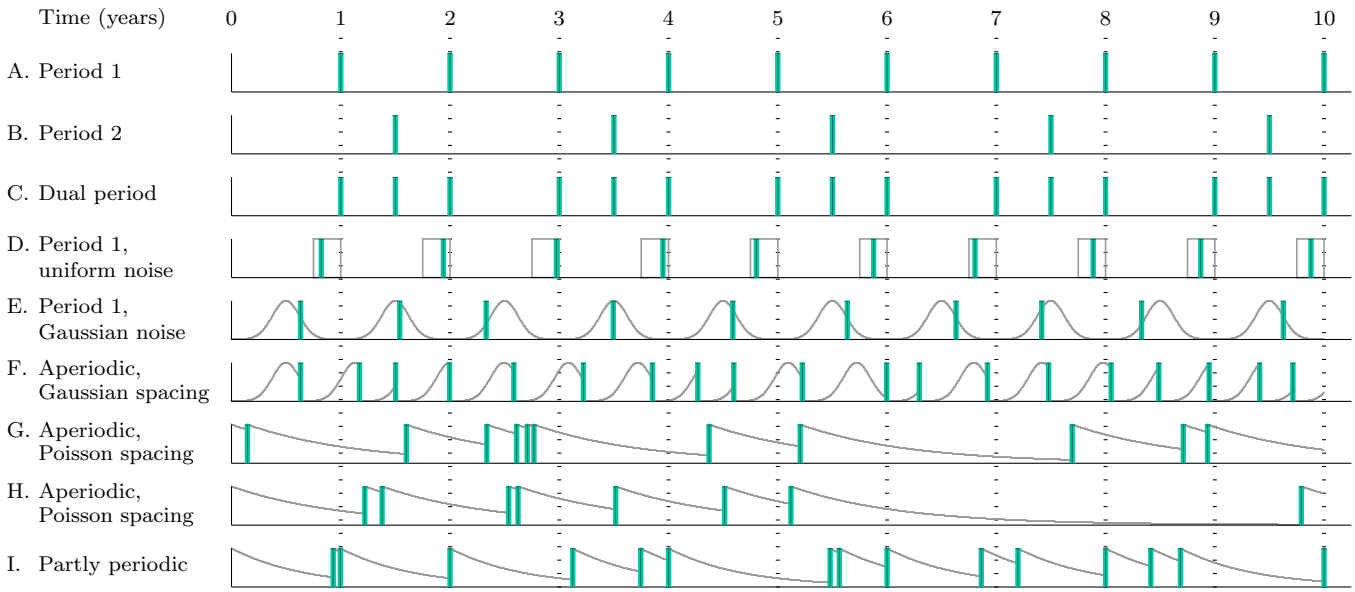


Figure 1. Illustration of clock patterns. *Clock ticks are shown with vertical bars on each graph for ten successive years.* **(A)** *Simple periodic clock with period 1 year.* **(B)** *Simple periodic clock with period 2, phase-shifted backwards 6 months.* **(C)** *The union of A and B above. Any combination of simpler clocks can form composite clocks.* **(D)** *Periodic clock with uniform random noise, with the tick anywhere in the last 3 months of the period. All points in that interval are equally likely, as outlined by the probability density function in gray.* **(E)** *Periodic clock with truncated Gaussian random noise, allowing the clock to tick anywhere within the period, but with the beginning and end of the period unlikely, as outlined by the probability density function in gray. Here* $\sigma = 1/8$ *year and* $\mu = 1/2,\ 3/2,\ 5/2, \ldots$ *years.* **(F)** *Aperiodic clock with the truncated Gaussian distribution of E. The probability evaluation restarts following each tick, making the aperiodic clock tick faster.* **(G)** *Simple aperiodic clock with mean time per tick of one year, where 11 ticks occurred in the time allotted to 10, due purely to random chance. A uniform random spacing of points (Poisson spacing) results from the exponential density function in gray.* **(H)** *Independent run of the same clock as in D, but where only 8 ticks occurred by random chance.* **(I)** *Partly periodic clock with 3 ticks distributed aperiodically every 2 years, the last of the 3 constrained to occur periodically at the end of the 2-year period.*

3. Periodic clocks

The simplest periodic clock ticks with complete regularity each time a fixed interval has elapsed. Starting at some initial time, t_0, it ticks at times $t_0+\varphi+\tau$, $t_0+\varphi+2\tau$, $t_0+\varphi+3\tau$, $\ldots$, $t_0+\varphi+n\tau$, and so forth ad infinitum, where τ is the period and φ is the phase.

Simple periodic clocks are established by *ClockPeriodic*, defined in the appendix, and provided with a clock number that identifies the data structure controlling the clock, an identification number denoting the event that is controlled by the clock, a period, and an optional phase shift. Such clocks can be established for any purpose—for example, to periodically examine conditions within the simulation like population growth rates, or to count out time intervals for a macroscale simulation of differential equations embedded within the microscale model.

The behavior of a simple periodic clock with period 1 is shown in Figure 1A. A similar simple periodic clock with period 2 is Figure 1B, shifted back from the end of its period by $1/2$ year ($\varphi = -0.5$). The union of these two clocks is Figure 2C. It ticks in the last three half-year intervals within the two-year period. Any collection of clocks can be combined into a single clock by assigning them the same identification number.

Figures 1D and 1E are periodic clocks with random fluctuations superimposed. Such fluctuations can arise from any probability distribution, which is supplied when the clock is started. Figure 1D is uniform random noise that allows a tick anywhere in the last quarter-period. That noise is supplied by the probability distribution shown in Figure 3A, where the cumulative probability P is 0 until $t = 9$ months, then rises linearly to 1 at $t = 12$ months. Figure 1E is similar, but with truncated Gaussian noise centered at the middle of the period. That noise is supplied by a cumulative probability distribution that rises sigmoidally from $P = 0$ at $t = 0$ to $P = 1$ at $t = 12$ months, as shown in Figure 3B. These cumulative distributions are represented by piecewise linear or higher-order approximations and passed to a random number generator that can work with arbitrary probability distributions [6].

Periodic example, birthday party clock. For an initial intuitive example, think about a "birthday party clock." A birthday begins regularly at midnight on a certain day of the year—a simple periodic clock. However, the celebration may be a few days early or a few days late. Nonetheless, the timing of the celebration does not change the timing of the birthday for subsequent years. That is, the celebration will always be synchronized with the calendar, not drifting

over time. This is therefore a periodic clock, with random noise for the timing of the celebration. (Note, however, that accounting for leap years considerably complicates the clock.)

Periodic example, macroscale simulation clock. Suppose a population is being simulated for which the microscale dynamics are not known, or for which there is not precise empirical data. Or suppose the dynamics are well known but the population is large enough that stochasticity tends to cancel out, so a microscale simulation of it is unnecessary. In such cases a macroscale population model may be embedded within a microscale simulation. For example, suppose a microscale model of the bacteria and individual parasitic ticks on populations of wild field mice is used to understand wildlife epidemiology. Also suppose that the behavior of the mouse population is not of interest, but is only necessary to provide background for the epidemiological part of the simulation. In that case the mouse population could be simulated with a known differential equation model embedded within the microscale model. A periodic clock can be used to count the time intervals of a macroscale differential equation solver—by Euler's method, Runge–Kutta, or other integration technique [7]. See Figure 2.

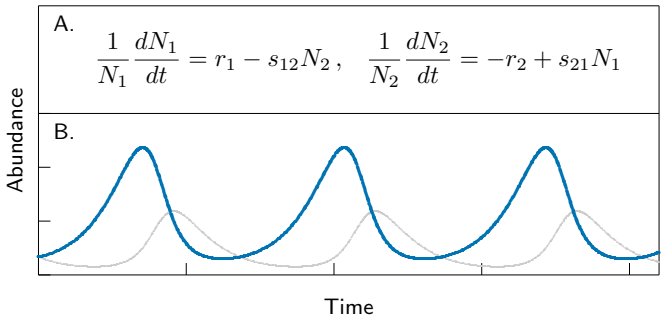


Figure 2. *A macroscale model embedded within a microscale model. Here a simple periodic clock supplies the time steps, Δt_i, to drive a differential equation simulator for population dynamics that do not need to be simulated by individuals within the microscale model.* **(A)** *The macroscale model to be embedded. This is an ecological Lotka–Volterra predator–prey model.* **(B)** *Simulated trace of the macroscale model, with $r_1{=}r_2{=}1$, $s_{12}{=}1$, and $s_{21}{=}2$. The abundance of prey $N_1(t)$, bold curve, can be used to drive individual births and deaths of a microscale prey species. The abundances of predator $N_2(t)$, light curve, are only to induce the bold curve and would not be needed further in the microscale simulation.*

4. Aperiodic clocks

The simplest aperiodic clock is the exact opposite of periodic. It ticks with complete irregularity, with all instants equally likely to see a tick (exponential delay), and with the probability of a tick in an instant defined by an average number of ticks per time unit. Starting at some initial time, t_0, it ticks at times $t_0+\tau_1$, $t_0+\tau_2$, $t_0+\tau_3$, $\ldots$, $t_0+\tau_4$, and so forth ad infinitum, where the random τ_i are uniformly distributed across time.

The behavior of a simple aperiodic clock ticking on average once per time unit is shown in Figures 1G and 1H. The first example ticks eleven times and the second ticks only eight, due solely to random variation. In the simplest aperiodic clock, the number of ticks per time unit follows a Poisson distribution and the time between ticks follows a corresponding exponential distribution [8]. In this case, the probability of exactly the expected number of ticks, 10, is only about 1 out of 8. (From the Poisson density function, $p(k) = k^\lambda \cdot e^{-\lambda}/k! = 10^{10} \cdot e^{-10}/10! \approx 0.125$.) Notice in Figures 1G and 1H how irregularly such a clock behaves.

The behavior of a related aperiodic clock appears in Figure 1F. It has the same probability distribution governing its ticks as the periodic clock in 1E above it. However, that probability distribution restarts at each tick. Notice how much more regular the ticks are in Figure 1F than in 1G and 1H, even though they are still aperiodic.

Aperiodic clocks are established by *ClockAperiodic*, defined in the appendix, and provided with a mean time between ticks or an optional probability distribution. Such clocks can simulate purely random processes such as radioactive decay, but also can approximate other events such as successive times of transmission in a population with infectious individuals.

Aperiodic example, hair appointment clock. Suppose that an individual is scheduled for a haircut every month, but that the haircut is occasionally delayed, due to negligence or other causes. If for some reason three months have elapsed between haircuts, most certainly is not necessary for the individual to have three haircuts in rapid succession to make up for the haircuts that were missed. The timing of the next haircut restarts at the time of the last, and occurs again at some average time in the future. That is an aperiodic clock.

5. Partly periodic clocks

Starting at some initial time, t_0, a partly periodic clock ticks at times $t_0+\tau$, $t_0+2\tau$, $t_0+3\tau$, $\ldots$, $t_0+n\tau$, and so forth, where τ is the period. In addition, however, it ticks aperiodically $k-1$ times in between each periodic tick, giving an average time between ticks of τ/k every period.

Figure 1I shows a partly periodic clock of period 2, with a mean time between ticks of $2/3$. One periodic tick and two aperiodic ticks occur in each period of the clock, with the spacing between all three ticks matching an exponential distribution. That exponential distribution decays more rapidly in Figure 1I than in 1F and 1H, because the mean time between ticks in 1F and 1H is 1 time unit, while in 1I it is only $2/3$ time unit.

Partly periodic example, immigration clock. Suppose a local population is being simulated and the number of immigrants each year is taken from known historical accounts. Suppose new individuals can arrive in the population at any random time of the year, but the number of immigrants each

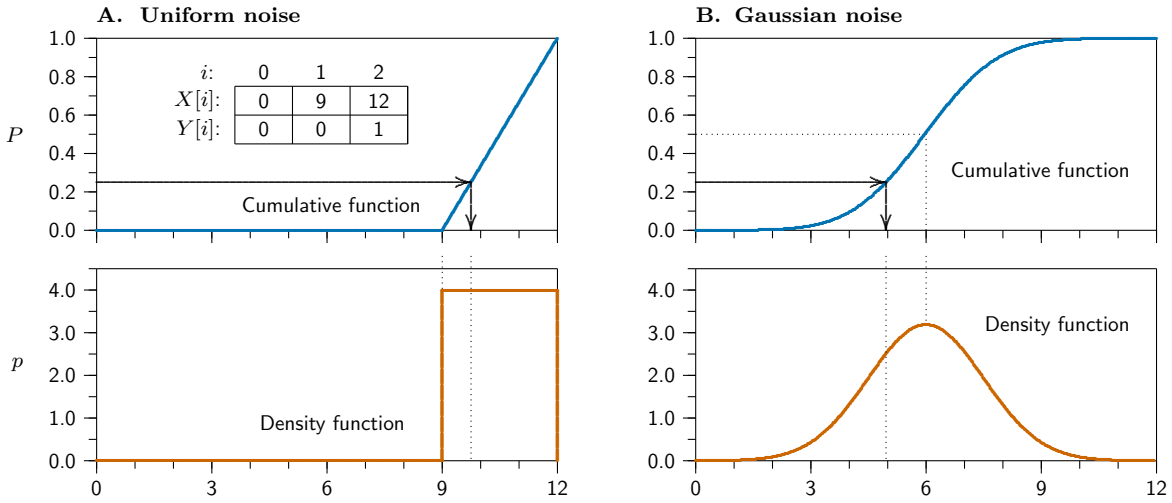


Figure 3. Random noise specifications. *Horizontal axes represent time, here marked in months. The vertical axis for cumulative functions represents the probability that the random variable is less than or equal to the corresponding value on the horizontal axis. The vertical axis for density functions, multiplied by the width of a small interval on the horizontal axis, represents the probability that a random number falls within that interval. Arrows demonstrate how random noise is generated, from a uniform random number located on the vertical axis, then followed to the right to the cumulative function, then down to the axis to pick a random number from the desired distribution* [6]. **(A)** *Uniform noise in the last 4 months of the year. Inset shows a tabular form of the cumulative distribution, in this case where three months on the horizontal axis,* $X[i] = \{0, 9, 12\}$*, correspond to three probabilities on the vertical axis,* $Y[i] = \{0, 0, 1\}$*.* **(B)** *Truncated Gaussian noise with mean 6 and variance* $9/4$*. The tabular form would be similar to that of Part A, but with several hundred small steps to approximate a continuous distribution, and optionally with nonlinear smoothing* [6].

year is fixed to exactly match historical records. This is not a periodic clock because immigrants can appear randomly throughout the year. However, it is not an aperiodic clock either, for in an aperiodic clock, the number of ticks in a given time interval cannot be specified exactly. This is a partly periodic clock. It is not intended to be an analog of a natural process, but rather to match empirical data, for which precise values are known at regular intervals.

6. Algorithms

Each clock is defined and tracked as one entry in an array $A[n]$ that stores the information associated with every individual in the simulation, as described in the appendix and elsewhere [2]. That array ordinarily records all future events for each individual, with the earliest of those events recorded in a global list of future events. Adding, deleting, and accessing events uses a constant amount of time regardless of how many events are in the global list [2].

However, for clocks, which act as "pseudo-individuals" in the simulation, data elements within the entry are used differently than they are used for individuals. Future times for each clock are recorded not explicitly but algorithmically, as described above and detailed in the appendix.

For periodic clocks, the algorithm records the period τ, the phase shift φ, and the number of ticks n that have occurred since the time t_0 that the clock started ticking. That allows the next time to be calculated as $t_0+(n+1)\tau+\varphi$. Multiplying the period by the number of elapsed ticks avoids cumulative drift due to rounding error, as would occur if the time of the previous tick were incremented by the period.

Aperiodic clocks are easier, since there is no starting time, phase shift, or past number of ticks to be tracked. These clocks have no memory of what has happened in the past. That makes their implementation simpler, even though understanding the dynamics of their behavior is more difficult (see discussion section below).

Partly periodic clocks are the most difficult. They require an entire sequence of k ticks to be remembered, so that all k can be rescaled precisely to fit within the period τ, with the last tick occurring at the last moment of the period. This could be accomplished with a quantity of memory proportional to k, to record the ticks in advance, but it can also be accomplished with a constant quantity instead. Deterministic random number generators have a state variable [9] that allows any subsequence of pseudo-random numbers to be regenerated. The algorithm for partly periodic clocks first runs through k ticks to determine how much total time $\mathcal{K}$ they would take. It then reruns the sequence one tick at a time, at each tick rescaling by $\tau/\mathcal{K}$.

All three kinds of clocks are proportional to n for speed and independent of n for memory. They are fully defined in the appendix, embodied in four algorithms.

1. *ClockPeriodic* Starts a periodic clock.
2. *ClockAperiodic* Starts an aperiodic clock.
3. *ClockPartlyPeriodic* Starts a partly periodic clock.
4. *ClockTick* Schedules another tick of any clock.

In addition, *T1*, *T2*, and *T3* are subroutines of *ClockTick* to implement the three kinds of clocks.

7. Discussion

Within the dynamics of periodic and aperiodic clocks, a curious observational paradox arises that affects measurements by individuals or agents within the simulation. It is helpful to understand this paradox when working with simulation clocks.

Suppose you have a simple periodic clock ticking precisely every ten simulated minutes and a matching simple aperiodic clock ticking on average at the same rate. Over the course of time both of these clocks tick equally often. Suppose each clock represents some service that individuals in the simulation occasionally wait for, such as catching a bus or being served at a medical emergency center. What will be the average interval between events, with an event occurring on average every ten simulated minutes, as measured by individuals within the simulation?

For a periodic clock, if a simulated individual arrives at a completely random time, independent of the ticking of the clock, that individual will observe an average of ten minutes from the time of the previous tick until the clock ticks again, since the clock is perfectly periodic. It would seem at first glance that it should be similar for an aperiodic clock—that an individual within the simulation would observe an average interval between ticks of ten minutes, since that is the average rate of the clock. But that is incorrect. For a completely random aperiodic clock ticking once every ten minutes on average, all individuals will observe an average time between ticks of not ten but of twenty minutes—half the speed of the actual clock! This effect also doubles the average time for each individual waiting for an aperiodic versus a periodic clock.

Such dynamics must be taken into account in auditing the performance of simulations, and even in designing real systems for use by living individuals. The inflated waiting time is not an illusion nor a property of computer simulation; it occurs in all aperiodic events, tangible or abstract. Upon first learning of this phenomenon, people are usually incredulous. Feller calls it the "waiting time paradox." He assures that although you may be shocked when you first encounter it, "after due reflection the difference becomes intuitively obvious." [8]

The resolution of the paradox lies in a "time line" that extends back to the past. If you picked a time at random on that line, you will have been more likely to have found yourself in a longer interval than a shorter one, simply because longer intervals contain a greater measure of time points than shorter ones. The intervals of a simple aperiodic clock vary widely, as illustrated in Figures 1G and 1H. When the exponential density function of 1G and 1H is integrated over all possible interval lengths and probabilities, the result of twice the average time between ticks emerges. With aperiodic clocks whose ticks are more aggregated than random, the observed interval between ticks can be arbitrarily long.

An understanding of this phenomenon and a review of how it applies in simulations is important so that aggregation of aperiodic events can be controlled in real systems to reduce actual waiting times there.

8. Conclusions

Many events arising within microscale simulations are simple enough to be handled by "clocks" of standard design. Three kinds of clocks—periodic, aperiodic, and partly periodic—cover a diversity of situations. All three require computing time proportional to the number of ticks and memory independent of the number of ticks.

9. Acknowledgements

We are grateful to Tendai Mugwagwa and Peter White for initial discussions leading to the design of these clocks, and to Todd Lehman, Shelby Williams, Katie Hoffman, and the anonymous reviewers for help with the presentation. This project was supported in part by a resident fellowship grant to C. Lehman from the UMN Institute on the Environment, by grants of computer time from the Minnesota Supercomputing Institute, and by doctoral research funding to A. Keen from the Modelling and Economics Unit at Public Health England, formerly Health Protection Agency, London.

References

[1] A. Keen, "Understanding tuberculosis dynamics in the United Kingdom using mathematical modelling," *Doctoral Thesis, London School of Hygiene and Tropical Medicine*, p. 493pp, 2013.

[2] C. Lehman, A. Keen, and R. Barnes, "Trading space for time: Constant-speed algorithms for managing future events in scientific simulations," *Proceedings, International Conference on Scientific Computing*, vol. CSC12, p. 8 pp, 2012.

[3] L. Gustafsson and M. Sternad, "Consistent micro, macro and state-based population modelling," *Mathematical Bioscience*, vol. 225, pp. 94–107, 2010.

[4] R. Brown, "Calendar queues: A fast O(1) priority queue implementation for the simulation event set problem," *Communications of the ACM*, vol. 31, pp. 1220–1227, 1988.

[5] A. Keen and C. Lehman, "Trading space for time: Constant-speed algorithms for grouping objects in scientific simulations," *Proceedings, International Conference on Scientific Computing*, vol. CSC12, pp. 146–151, 2012.

[6] C. Lehman and A. Keen, "Efficient pseudo-random numbers from any probability distribution," *Proceedings, International Conference on Modeling, Simulation, and Visualization Methods*, vol. MSV12, pp. 121–127, 2012.

[7] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, "Numerical recipes: The art of scientific computing, Third Edition," *Cambridge University Press, New York*, 2007.

[8] W. Feller, "An introduction to probability theory and its applications, Volume II, Second Edition," *John Wiley & Sons, New York*, 1971.

[9] D. E. Knuth, "The art of computer programming, volume 2: Seminumerical algorithms, third edition," *Addison-Wesley, Reading, MA*, 1997.

10. Appendix

To use the algorithms described in this paper, it is only necessary to understand the entry and exit conditions that appear at the beginning of each subroutine, not the code itself. Nonetheless, to allow complete evaluation of the algorithms, and to encourage further development of them, we present them as pseudo-code inspired by and simplified from the programming languages C, R, and Python. The algorithms are defined with sufficient precision that they can be run, tested, timed, modified, or translated to other languages. Familiarity with a few operators* and with the syntax of flow control (if, for, while, etc.), is sufficient to follow the algorithms. External routine *EventSchedule*(n, τ) schedules a future event for $A[n]$ at time τ [2], *Rand* generates uniform pseudo-random numbers $0 \leq \rho < 1$, *RandF* is an interface for other distributions, *RandSeed* retrieves the current state of the random sequence, *RandSeq*(q) resets the sequence to a new or previous state q, and *Cinverse* converts to any probability distribution [6]. These algorithms translated into operational C are available free from the authors upon request.

DATA ELEMENTS

Clock parameters are stored in array $A[n]$, which is also used for individuals in the simulation. The main simulation program keeps track of which values of n represent clocks, either by assigning them to fixed positions of the array or by placing them in a group [5] reserved for clocks. The elements of $A[n]$ must be large enough to represent both clocks and individuals.

Variables in $A[n]$ are assigned names beginning with V, as follows.

Vid	$\equiv A[n][0]$	Identification for this specific clock.
Vtype	$\equiv A[n][1]$	Type of clock (periodic, aperiodic, etc).
Vper	$\equiv A[n][2]$	Time units per tick or set of ticks.
Vbase	$\equiv A[n][3]$	Base time (periodic or aperiodic).
Vstep	$\equiv A[n][4]$	Number of steps presently beyond the base.
Vticks	$\equiv A[n][5]$	Number of ticks per period (partly periodic).
Vk	$\equiv A[n][6]$	Ticks thus far in the period (partly periodic).
Vscale	$\equiv A[n][7]$	Scaling factor for the period (partly periodic).
Vseed	$\equiv A[n][8]$	Random number seed (partly periodic).
Vx	$\equiv A[n][9]$	Cumulative distribution, 'x' variable.
Vy	$\equiv A[n][10]$	Cumulative distribution, 'y' variable.
Vnxy	$\equiv A[n][11]$	Number of elements in 'x' and 'y'.

CLOCK TICK

Routine *ClockTick* is called after each tick of a clock, typically from a dispatching routine in the main simulation program, to schedule the next tick. It is also called once a new clock is started, to schedule the first tick. **Upon entry, (1)** $A[n]$ records the state of the clock, as defined in the exit conditions of *ClockPeriodic*, *ClockAperiodic*, and *ClockPartlyPeriodic* below. In particular, **(2)** $Vtype$ defines the type of clock, 1=periodic, 2=aperiodic, 3=partly periodic. **On exit,** the next tick is scheduled.

```
ClockTick(n)   integer n;
  choose from Vtype:
    case 1: T1(n); return;          Periodic clock.
    case 2: T2(n); return;          Aperiodic clock.
    case 3: T3(n); return;          Partly periodic clock.
    other:  ExitMsg(1); return;     Improper clock type.
```

* *In the pseudo-code here, indentation defines the nested structure. Variables and function names are italicized and flow control and reserved words are bolded. Assignment is left to right, represented by '$\rightarrow$'. Individual parts of any compound assignments also operate left to right, so that '$a + 1 \rightarrow a \rightarrow b \rightarrow W[i][j]$' operates by first incrementing a and placing the results back in a, then in b, and then in the i, jth element of array W. Array indexing starts with 0. Any logical operators such as 'and' and 'or' are preemptive, terminating a chain of logical operations as soon as the result is known. Permanent global assignments are rendered '$\alpha \equiv \beta$'.*

Algorithm 1. PERIODIC CLOCK

Routine *ClockPeriodic* establishes a new periodic clock or alters an existing one. **Upon entry, (1)** $A[n]$ is an entry available for controlling the clock. **(2)** id contains the identification number for the clock. **(3)** $period$ is the period of the clock, greater than 0. **(4)** $phase$ is phase shift—the position of the periodic tick within the period. **(5)** x, y, and nxy optionally define a probability distribution for the noise, in the form of $RandF$. If x is null, no probability distribution is supplied and no noise is applied. **(6)** $Anull$ is an entry of $A[n]$ that has all variables zero. **(7)** t is the current time. **On exit, (1)** $A[n]$ is prepared so that clock ticks can be scheduled by a call to $ClockTick$. In particular, the following elements are set. **(2)** $Vtype$ is 1, meaning a periodic clock. **(3)** Vid carries the value of id on entry. **(4)** $Vperiod$ contains the value of $period$ on entry. **(5)** $Vbase$ contains the base time for the clock, equal to t on entry plus the value of $phase$ on entry. **(6)** Vx, Vy, and $Vnxy$ contain the values of x, y, and nxy on entry, respectively. These may be null if no distribution was supplied. **(7)** All other elements are zero.

ClockPeriodic(n, *id*, *period*, *phase*, $x[\,]$, $y[\,]$, *nxy*) **integer** n, *id*, *nxy*; **real** *period*, *phase*, $x[\,]$, $y[\,]$;

Anull $\rightarrow A[n]$; 1 $\rightarrow$ *Vtype*;	1. Prepare a new table entry.
id $\rightarrow$ *Vid*; *period* $\rightarrow$ *Vper*; $t +$ *phase* $\rightarrow$ *Vbase*;	2. Save entry parameters.
$x \rightarrow Vx$; $y \rightarrow Vy$; *nxy* $\rightarrow$ *Vnxy*;	3. Save any custom random function.

Routine *T1* is called each time a periodic clock has ticked, to schedule the next tick. It is also called immediately after *ClockPeriodic* has been called, to start the clock ticking. **Upon entry, (1)** $A[n]$ defines the clock structure. In particular, the following elements are relevant. **(2)** Vid contains the identification number for the clock. **(3)** $Vper$ contains the period. **(4)** $Vbase$ contains the base time for the clock, equal to the starting time plus the phase shift. **(5)** $Vstep$ contains the number of ticks thus far. **(6)** Vx, Vy, and $Vnxy$ optionally define a probability distribution for the noise, in the form of $RandF$. If Vx is null, no probability distribution is supplied and no noise is applied. **(7)** t is the current time. **On exit, (1)** The next tick of the clock is scheduled. **(2)** $Vstep$ is advanced by the 1.

T1(n) **integer** n;

if *Vx*: *RandF*(*Vx*, *Vy*, *Vnxy*) $\rightarrow w$, (w/*Vx*[*Vnxy* − 1])∗*Vper* $\rightarrow w$;	1. If random noise has been specified, select a tick from within that noise.
else *Vper* $\rightarrow w$;	2. Otherwise generate a precise tick.
Vbase + (*Vstep*∗*Vper*) + $w \rightarrow v$;	3. Compute the time of the tick.
Vstep + 1 $\rightarrow$ *Vstep*;	4. Advance the tick counter.
if $v < t$: $t \rightarrow v$;	5. Control any rounding error.
EventSchedule(n, v);	6. Schedule the next tick.

Algorithm 2. APERIODIC CLOCK

Routine *ClockAPeriodic* establishes a new aperiodic clock or alters an existing one. **Upon entry, (1)** $A[n]$ is an entry available for controlling the clock. **(2)** id contains the identification number for the clock. **(3)** $mean$ optionally defines the time between ticks in an exponential probability distribution, if a custom probability distribution is not supplied. **(4)** x, y, and nxy optionally define a custom probability distribution for the waiting time to the next tick, in the form of $RandF$. If x is null, no custom distribution is supplied and an exponential distribution is used instead. **(5)** $Anull$ is an entry of $A[n]$ that has all variables zero. **On exit, (1)** $A[n]$ is prepared so that clock ticks can be scheduled by a call to $ClockTick$. In particular, the following elements are set. **(2)** $Vtype$ is 2, meaning an aperiodic clock. **(3)** Vid carries the value of id on entry. **(4)** $Vper$ contains the value of $mean$ on entry. **(5)** Vx, Vy, and $Vnxy$ contain the values of x, y, and nxy on entry, respectively. These may be null if no custom distribution was supplied. **(6)** All other elements are zero.

ClockAperiodic(n, *id*, *mean*, $x[\,]$, $y[\,]$, *nxy*) **integer** n, *id*, *nxy*; **real** *mean*, $x[\,]$, $y[\,]$;

Anull $\rightarrow A[n]$; 2 $\rightarrow$ *Vtype*;	1. Prepare a new table entry.
id $\rightarrow$ *Vid*; *mean* $\rightarrow$ *Vper*;	2. Save entry parameters.
$x \rightarrow Vx$; $y \rightarrow Vy$; *nxy* $\rightarrow$ *Vnxy*;	3. Save any custom random function.

Routine *T2* is called each time an aperiodic clock has ticked, to schedule the next tick. It is also called immediately after *ClockAperiodic* has been called, to start the clock ticking. **Upon entry, (1)** $A[n]$ defines the clock structure. In particular, the following elements are relevant. **(2)** Vid contains the identification number for the clock. **(3)** $Vper$ contains the mean time between ticks in an exponential distribution if Vx, Vy, and $Vnxy$ do not specify a custom distribution. **(4)** Vx, Vy, and $Vnxy$ optionally define a custom probability distribution for the waiting time to the next tick, in the form of $RandF$. If Vx is null, no custom distribution is supplied and an exponential distribution is used instead. **(5)** t is the current time. **On exit,** The next tick of the clock is scheduled.

T2(n) **integer** n;

if *Vx*: *RandF*(*Vx*, *Vy*, *Vnxy*) $\rightarrow w$;	1. Generate the time until the next tick, from a custom distribution or an exponential.
else *Exponential*(*Vper*) $\rightarrow w$;	
EventSchedule(n, $t + w$);	2. Add current time to schedule the next tick.

Algorithm 3. PARTLY PERIODIC CLOCK

Routine *ClockPartlyPeriodic* establishes a new partly periodic clock or alters an existing one. **Upon entry, (1)** $A[n]$ is an entry available for controlling the clock. **(2)** id contains the identification number for the clock. **(3)** $period$ defines the period of the periodic part of the clock. **(4)** $ticks$ defines the number of ticks that will occur aperiodically during that period, spaced according to the dictates of the relevant probability distribution. **(5)** x, y, and nxy optionally define a custom probability distribution for the waiting time to the next tick, in the form of $RandF$. If x is null, then no custom distribution is supplied and an exponential distribution is used instead. **(6)** $Anull$ is an entry of $A[n]$ that has all variables zero. **(7)** t is the current time. **On exit, (1)** $A[n]$ is prepared so that clock ticks can be scheduled by a call to $ClockTick$. In particular, the following elements are set. **(2)** $Vtype$ is 3, meaning a partly periodic clock. **(3)** Vid carries the value of id on entry. **(4)** $Vper$ contains the value of $period$ on entry. **(5)** $Vticks$ contains the number of ticks to occur in that period. **(6)** $Vbase$ contains the value of t on entry. **(7)** Vx, Vy, and $Vnxy$ contain the values of x, y, and nxy on entry, respectively. **(8)** All other elements are zero.

ClockPartlyPeriodic(n, *id*, *period*, *ticks*, $x[\,]$, $y[\,]$, *nxy*) **integer** n, *id*, *nxy*, *ticks*; **real** *period*, $x[\,]$;	
Anull $\rightarrow$ *A*[*n*]; 3 $\rightarrow$ *Vtype*;	1. Prepare a new table entry.
id $\rightarrow$ *Vid*; *period* $\rightarrow$ *Vper*; *ticks* $\rightarrow$ *Vticks*; *t* $\rightarrow$ *Vbase*;	2. Save entry parameters.
x $\rightarrow$ *Vx*; *y* $\rightarrow$ *Vy*; *nxy* $\rightarrow$ *Vnxy*;	3. Save any custom random function.

Routine *T3* is called each time a partly periodic clock has ticked, to schedule the next tick. It is also called immediately after *ClockPartlyPeriodic* has been called, to start the clock ticking. **Upon entry, (1)** $A[n]$ defines the clock structure. In particular, the following elements are relevant. **(2)** Vid contains the identification number for the clock. **(3)** $Vper$ contains the $period$. **(4)** $Vticks$ contains the number of ticks to occur in that period. **(5)** $Vbase$ contains the starting time. **(6)** $Vstep$ contains the number of ticks thus far. **(7)** Vx, Vy, and $Vnxy$ optionally define a custom probability distribution for the waiting time to the next tick, in the form of $RandF$. If x is null, then no custom distribution is supplied and an exponential distribution is used instead. **(8)** t is the current time. **On exit, (1)** The next tick of the clock is scheduled. **(2)** $Vstep$ is advanced by the 1.

T3(n) **integer** n;	
if *Vk* $= 0$:	1. At the beginning of an aperiodic section, run through the sequence to determine how much it has to be up or down-scaled to match the specific period.
Vseed $\leftarrow$ *RandSeed*(); 0 $\rightarrow$ *Vscale*;	
loop $Vticks$ **times**:	
if *Vx*: *Vscale* + *RandF*(*Vx*, *Vy*, *Vnxy*) $\rightarrow$ *Vscale*;	
else *Vscale* + *Exponential*(1) $\rightarrow$ *Vscale*;	
Vk + 1 $\rightarrow$ *Vk*; **if** *Vk* $\geq$ *Vticks*: 0 $\rightarrow$ *Vk*;	2. Advance the subperiod counter.
RandSeed() $\rightarrow$ *q*; *RandSeq*(*Vseed*);	3. Step again through the previous random sequence to generate the next tick, without disturbing the current sequence.
if *Vx*: *RandF*(*Vx*, *Vy*, *Vnxy*) $\rightarrow$ *w*;	
else *Exponential*(1) $\rightarrow$ *w*;	
RandSeed() $\rightarrow$ *Vseed*; *RandSeq*(*q*);	
Vbase + (*Vstep*$*$*Vper*) $\rightarrow$ *base*;	4. Locate the beginning of the period.
if *base* $<$ *t*: *t* $\rightarrow$ *base*;	5. Control any rounding error.
Vstep + 1 $\rightarrow$ *Vstep*;	6. Locate the next periodic point.
EventSchedule(n, $t + (w/Vscale)*Vper$);	7. Schedule the next tick.

RANDOM NUMBER INTERFACE

Routine *RandF* connects the internal routines represented here with the external random number routine *Cinverse* [6], which generates random numbers from any probability distribution. It uses two arrays to define the distribution, one for the x-axis and one for the y-axis. For example, the distribution of Figure 3A would have $x[0,1,2] = \{0, 9, 12\}$, $y[0,1,2] = \{0, 0, 1\}$, and $n = 3$, meaning the cumulative distribution remains 0 between $x = 0$ and $x = 9$, then rises linearly to 1 at $x = 12$. **Upon entry, (1)** x is an array of values in the set of random numbers to be generated. **(2)** y is an array of cumulative probabilities, each being the probability that a random value will be less than or equal to the corresponding value in x. **(3)** n is the number of entries in tables x and y. **On exit,** $RandF$ contains a random value drawn from the given distribution.

RandF(x, y, n) **integer** n; **real** $x[\,]$, $y[\,]$;
return *Cinverse*(1, *Rand*(), $x[0]$, n, x, y, 0);

Application of SolidWorks® & AMESim® – based Updated Simulation Technique to Back-flow Analysis of Trochoid Hydraulic Pump for Lubrication

Seung Won Jeong[#], Won Jee Chung[#], Man Su Kim[#], Myung Sik Kim*

[#] *School of Mechatronics, Changwon National University, South Korea*

[1]dongnegosu@gmail.com

[2]wjchung@changwon.ac.kr

[3]subek@naver.com

* *Flutek, 6 Gongdan-ro 98beon-gil, Seongsan-gu, Changwon-si, Gyeongsangnam-do, South Kore*

[4] mskim@flutek.co.kr

Abstract—

In this paper, we deal with the Gerotor pump which will be referred to as "trochoid pump," hereinafter since it has been widely used for the transmission part of a automobile. Usually, the simulation of trochoid pump has utilized a professional analysis program or language such as CFD® (specifically CFX®) or C-code (e.g. C++) which is not an easy tool for a field engineer. But the field engineer based on his experience has a hurdle in modifying the flow model according to the need of a customer. To cope with this problem, in the paper of Kim et al. [3] in WORLDCOMP 2012 (CSC 2012), we have presented how to establish the flow model of a trochoid pump by using the most popular 3-dimensional (or 3D) modeling tool SolidWorks® and a hydraulic analysis program AMESim®. For this model, we have figured out the clearance problem which was neglected in the paper of Kim et al. [3]. In this paper, we will update Kim's flow model in order to solve the clearance problem by incorporating Jang's formulation [4] into the SolidWorks® model. The practical excellence of updated simulation technique proposed in this paper will be verified to be compared with experimental results.

Keywords— *lubrication pump , trochoid pump, clearance, simulation technique, viscosity of hydraulic fluid, experimentation*

I. INTRODUCTION

As for the environmental issues of transportation equipment, research and development has been recently focused on high fuel efficiency. The control of the engine (or transmission) of vehicle, as a method for sustaining high fuel efficiency during the stable running of a vehicle, is required. In order to maintain normal performance and durability of the engine (or transmission), a lubrication device for preventing internal friction which can result in overheating is necessary. Usually the output power of engine is used to the lubrication device in the range of about 25-30%. Nowadays a (fixed displacement) internal gear pump using the modified curve of sphere gear is used for a lubrication device in the viewpoint of market. The internal gear pump can be divided into two kinds, *i.e.*, Gerotor pump and Partition pump. In this paper, we deal with the Gerotor pump which will be referred to as "*trochoid* pump," hereinafter since it has been widely used for the transmission part of a automobile. The terminology of *trochoid* is derived from a trochoidal tooth profile as shown in Fig.3. A sample of a trochoid pump is shown in Fig.1, which can be used for the transmission of a commercial vehicle.

Fig. 1 Trochoid Pump

The trochoid pump is composed of outer rotor and inner rotor as shown in Fig. 2. This means that it has simple structure. Owing to the development of sintering technology, any profile of rotor can be manufactured and then can be easily assembled. Especially the relative motion between outer and inner rotors is small for a trochoid pump. Pump efficiency can be maintained for long operation. These points explain the wide use of trochoid pump as a hydraulic source for the lubrication of transmission or engine.

In previous papers including Yang *et al*. [1] and Nam *et al*. [2], the simulation of trochoid pump has utilized a professional analysis program or language such as CFD® (specifically CFX®) or C-code (e.g. C++) which is not an easy tool for a field engineer. This software tool can realize mesh and then make a flow model. But the field engineer

based on his experience has a hurdle in modifying the flow model according to the need of a customer. To cope with this problem, in the paper of Kim *et al*. [3] in WORLDCOMP 2012 (CSC 2012), we have presented how to establish the flow model of a trochoid pump by using the most popular 3-dimensional (or 3D) modeling tool SolidWorks® and a hydraulic analysis program AMESim®. For this model, we have figured out the clearance problem which was neglected in ref. [3], so as to be explained in detail in Section 2. In this paper, we have updated Kim's flow model in order to solve the clearance problem by incorporating Jang's formulation [4] into the SolidWorks® model. The practical excellence of updated simulation technique proposed in this paper will be verified to be compared with experimental results.

II. Theoretical Design of Trochoid Pump[4]

In Kim *et al*. [3], the "clearance" (specifically the distance between inner rotor and outer rotor as depicted in Fig. 2) can be turned out to '0 ', which can be accepted in theoretical sense. But, in practical sense, the rotation between inner rotor and outer rotor cannot occur because there is no hydraulic fluid in the clearance and thereby the rotors get jammed between each other. Thus we have utilized Jang's theory [4] to treat with this clearance problem. Figure 2 shows a schematic diagram of the trochoid pump mainly used, where this pump is composed of the outer rotor and inner rotor.

The outer rotor is designed from a trochoid shape of the tooth of inner rotor and thereby can have an arc/oval/sinusoidal shape. Specifically the inner rotor is drawn by using an arc with almost the same radius. Moreover the number of robes for the outer rotor is one more than the that of inner rotor. A coaxial drive is used by inserting the shaft of the pump into the center of inner rotor. Thus the trochoid pump has the constant eccentric distance of e between the inner rotor and the outer rotor, which can result in pumping by changing the intaking volume of hydraulic fluid as shown in Fig. 2.

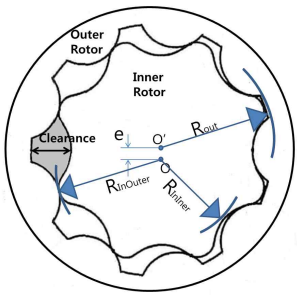


Fig. 2 drawing of inner and outer rotor

Basically we have exploited the design formulation of trochoid profile proposed by Jang *et al.* [4]. Referring to Fig. 3, R_r is the radius of rolling circle, while R_c is the radius of the basic circle. R_z denotes the radius of idle circle for a trochoid curve.

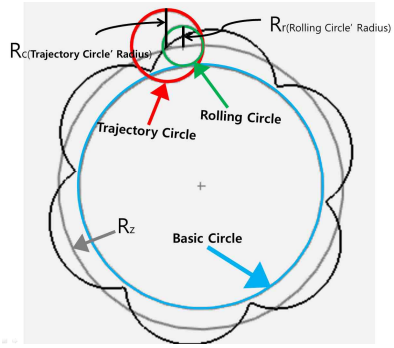


Fig. 3 Profile of trochoid curve

Defining N by Eq. (1),

$$N = \frac{R_c}{R_r} \tag{1}$$

then the basic trochoid curve in [4] can be given by Eqs. (2) and (3) :

$$x = R_r(N+1)\cos\theta - e\cos((N+1)\theta) \tag{2}$$

$$y = R_r(N+1)\sin\theta - e\sin((N+1)\theta) \tag{3}$$

where ϕ is the rotation angle of rolling circle and θ the rotation angle of basic circle, resulting in the following relationship:

$$R_c\theta = R_r\phi \tag{4}$$

Defining K as

$$K = \frac{-R_r\sin\theta + e\sin((N+1)\theta)}{R_r\cos\theta + e\cos((N+1)\theta)} \tag{5}$$

then we have the final (modified) trochoid trajectory (X, Y) by Eq. (6) :

$$X = x + \frac{R_z}{\sqrt{1+K^2}}$$

$$X = x - \frac{R_z K}{\sqrt{1+K^2}}$$

$$Y = y - \frac{R_z}{\sqrt{1+K^2}}$$

$$Y = y + \frac{R_z K}{\sqrt{1+K^2}} \tag{6}$$

Unfortunately, Kim *et al.* [3] has based on only Eq. (6), which cannot treat with the actual problem of clearance '0.' Specifically, clearance '0' means that there is no hydraulic fluid in the clearance and thereby both the inner and outer rotors get jammed. In the meanwhile, to cope with clearance '0' problem in this paper, we have used the eccentricity denoted by e as follows:

$$e = \frac{R_{Inouter} - R_{InInner}}{2} \tag{7}$$

where $R_{Inouter}$ and $R_{InInner}$ are defined in Fig. 2, respectively. Then R_z can be determined by Eq. (8) :

$$R_z = (N+1)R_r - e - R_{InInner} \tag{8}$$

where R_r is usually given by a design engineer.

Figure 3 means that field engineers who are familiar with SolidWorks® can draw easily the trochoid model by using SolidWorks® based on Eqs. (6) and (8) not by virtue of C-code which is difficult for the engineers. Specifically, using the motion analysis module of SolidWorks®, the rolling circle can be rotated based on the basic circle in order to generate trochoid curve as shown in Fig. 3. Only the basic condition, R_r > e, should be satisfied. Next, when the circle of trace of wheel is made to be rotated according to the trochoid curve through the motion analysis module of SolidWorks®, the shape of inner rotor can be designed as shown in Fig. 4. Finally the pump configuration which is designed by using a trochoid tooth profile is shown in Fig. 5.

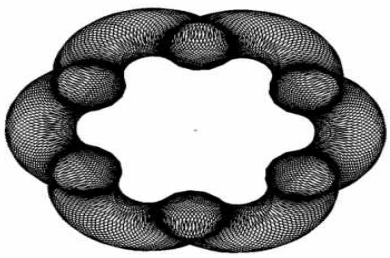


Fig. 4 Generation of Inner Rotor Shape

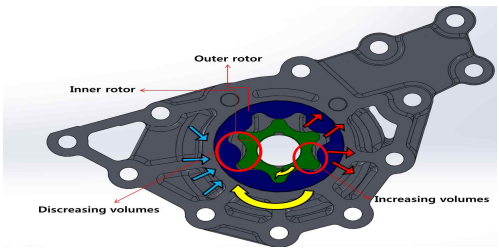


Fig. 5 Designed trochoid pump

III. Flow Field Area Modelling[3] Using SolidWorks®

The flow field area modeling technique using SolidWorks® is almost similar to that of Kim *et al.* [3]. In this section we will briefly explain this technique. As shown in Eq. (10), the dispensing flow rate Q_{pump} of the positive displacement pump can be obtained by using the change of flow volume *displ* per one rotation and the rotational angular speed ω_{pump}. Especially *displ* can be expressed by Eq. (11) because the thickness H of trochoid pump is constant. In Eq. (11), Area denotes the cross-sectional area of flow rate while θ indicates rotation angle. Now we need the data of area change according to rotation angle, *i.e.* $\frac{dArea}{d\theta}$ by using SolidWorks® as a CAD (Computer Aided Design) tool.

Then, how to obtain the flow rate of the trochoid pump is as follows.

$$Q_{pump} = displ \cdot \omega_{pump} \tag{10}$$

$$displ = \frac{dV}{d\theta} = H \cdot \frac{dArea}{d\theta} \left[\frac{m^3}{rad}\right] \tag{11}$$

The data of area change ($dArea$) can be obtained by using angular velocity ratio of trochoid pump. It can be performed by changing the area through the rotation of each rotor according to the rotation ratio of inner and outer rotors. The speed ratio can be determined by Eq. (12) according to the number of outer rotor lobes (or teeth) (N_{lobe}):

$$\frac{\omega_{out}}{\omega_{in}} = \frac{N_{lobe} - 1}{N_{lobe}} \tag{12}$$

where ω_{out} (ω_{in}) denotes the angular velocity of outer (inner) rotor.

When the drawings of the inner and outer rotors are completed, flow field area modeling can be done through the element conversion technique of SolidWorks® as shown in Fig. 6 where inlet and outlet are simplified. In specific, the sketch of 3 parts, i.e., inner rotor, outer rotor and inlet/outlet, are first designated as 'BLOCKs' by using BLOCK technique of SolidWorks®. Then the flow field area (to be explained later) can be modeled by performing the PROTRUSION BASE (similar to PAD technique of Solidworks®) technique of SolidWorks®. Consequently 7 models of flow field area are designated as shown in Fig. 6.

Now, the area change of one flow field should be investigated according to Eq. (12) (in other words, according to the rotation of inner rotor BLOCK). Specifically, while making inlet and outlet BLOCK fixed, the outer rotor BLOCK is rotated based on Eq. (12) as the inner rotor BLOCK is rotated, by using FORMULA EDIT technique of SolidWorks®. In every rotation, the area of one flow field can be changed in

shape according to the rotation angle of inner rotor as shown in Fig. 7. Thus the area change of one flow field between inlet and outlet can be depicted as a graph of Fig. 8 which will be used later for the simulation of trochoid pump using AMESim®. In a similar manner to Fig. 8, the area change of outlet and inlet flow field can be also obtained as shown in Fig. 9.

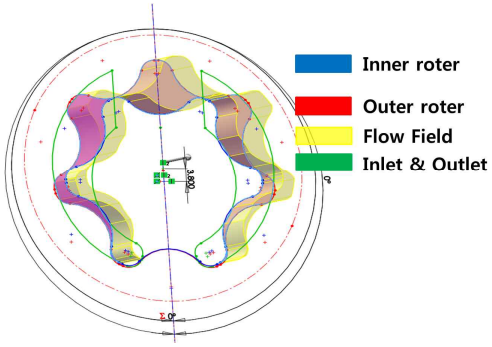


Fig. 6 Flow field modeling using SolidWorks®

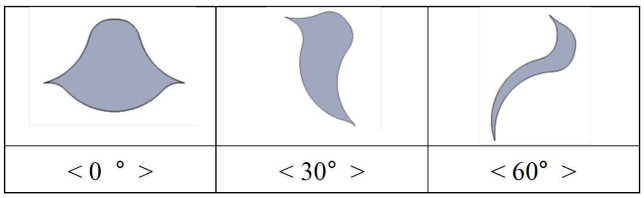


Fig. 7 Area change of one flow field according to rotation angle of inner rotor (shape)

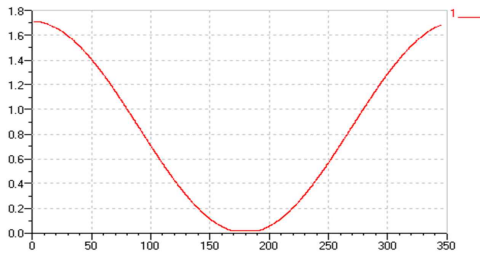


Fig. 8 Area change of one flow field according to rotation angle of inner rotor

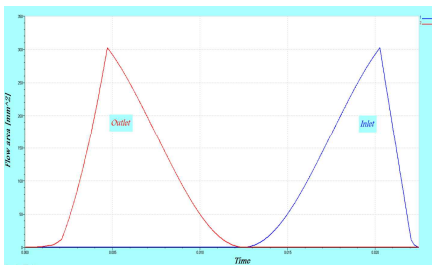


Fig. 9 Area change of outlet and inlet flow field

IV INTERNAL FLOW ANALYSIS

The objective of hydraulic circuit modeling for trochoid pump using AMESim® is to realize flow and then simulate back-flow so as to control the flow rate control of trochoid pump. In case of hydraulic circuit modeling of only one flow field for trochoid pump, the area change data of Fig. 8 and the thickness of rotor can be made equivalent with an 1-dimensional piston model as shown in Fig. 10.

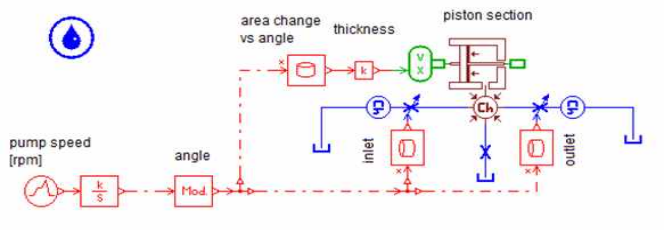


Fig. 10 1-dimensional piston model of one flow field using AMESim®

Hydraulic circuit modeling of trochoid pump is required to show the volume generated at the location of each particular angle when the inner rotor comes in contact with the outer rotor. This can be carried out at the location of each phase change, i.e., $360°/N_{lobe}$. Finally N_{lobe} models are generated similarly as Fig.10 and connected each other as shown in Fig. 11.

The factors that can be simply controlled in real time through the formation of the AMESim® hydraulic circuit of a trochoid pump include: the rotation speed of pump, the shape angle of inlet and outlet, gap between outer rotor and pump casing, and gap between inner rotor and outer rotor. In back-flow simulation, these control factors are important for the flow rate control of trochoid pump. Especially, from the viewpoint of a field engineer, AMESim® is more uselful for this back-flow simulation, compared with a traditional analysis software of fluid mechanics, *i.e.*, CFD®, because CFD® needs the renewal of mesh modeling every time each control factor has a different value while AMESim® needs only the input values of control factors without any change of hydraulic circuit modeling.

In some cases, cavitation occurs at low velocity resulting from decreasing suction force due to the characteristics of lubricating hydraulic fluid, not high-velocity revolution. However, as the difference at low velocity is negligible, the present study compares a simulation-based method with an actual operation case to do research needed in the field. This paper is intended to reduce the budget for experimentation as well as trial and error via the simulation-based estimation, provided the confidence level of the case compared falls within the error range. The simulation for the outlet part shows the hydraulic variation from the inside of the stabilized domain excluding the transitional condition at the time of initial response as in Fig. 12.

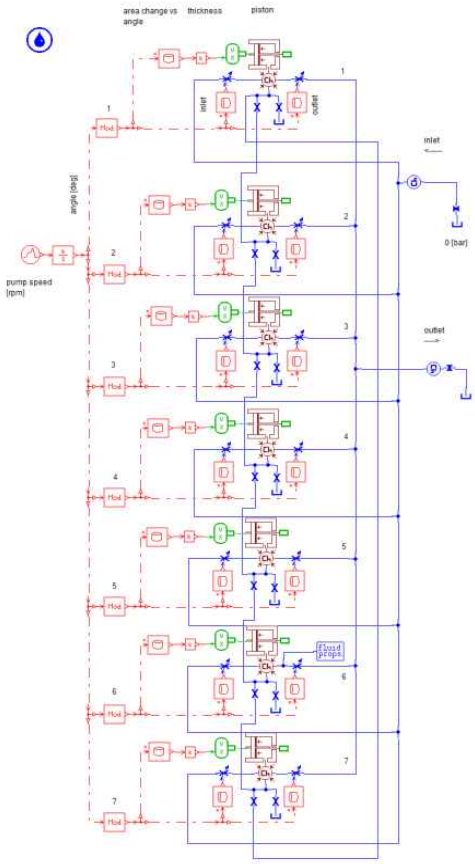


Fig. 11 Connected N_{lobe} hydraulic circuit modeling of trochoid pump using AMESim®

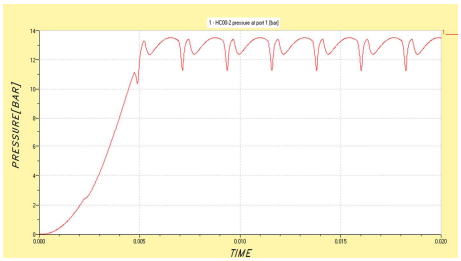


Fig. 12 Hydraulic pressure change at the outlet

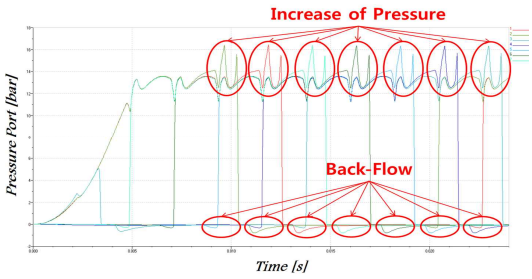


Fig. 13 Flow loss occurred at back-flow

As in Fig. 13, upon the rotor rotating, the pressure surges right before the lubricating fluid moves out. Then, the fluid pressure falls below "0" due to the momentary pressure drop upon the fluid going out. This moment is considered a back-flow, implying the loss of flow. In Fig. 13, compared with that of Kim *et al.* [3], the back-flow is more observable, where the pressure rises due to the hydraulic fluid stuck in the rotor slots and it sharply drops upon the hydraulic fluid moving out.

Normally, the following two approaches have been used to reduce the back-flow by decreasing the pulsation in existing pumps. The first approach is to apply a notch to the rotor, causing a little leakage to occur in advance in a closed space, to lessen the sharp increase in pressure and reduce the back-flow. The second approach is to decrease the flow loss by varying the discharge angles of the hydraulic fluid touching the pump cover where the rotor meets the outlet housing.

For the notch-related approach, it has been proved that it is hard to take a formulaic approach to the area representing the notch part. Moreover, it is hard to apply a notch to an actual trochoid pump, which has no room for a notch. As this approach is not consistent with the objective of seeking a method for enabling field engineers to access with ease, the second approach is used for the present study.

With the second approach, to reduce the flow loss, the angle of the hydraulic fluid discharged from the rotor to the pump cover has been varied by 1° from 0°, *i. e.,* 1°, 2°, 3°, 4° and 5°. As shown in Fig. 14, the back-flow effect is smallest at 4°. The variation of the discharge angle where the pump cover comes into contact with the hydraulic fluid lessens the back-flow, and thus effectively decreases the overall leakage between the inner and outer rotors. This simulation-based reduction of the flow loss is applied to the pump design and then verified with experimentation.

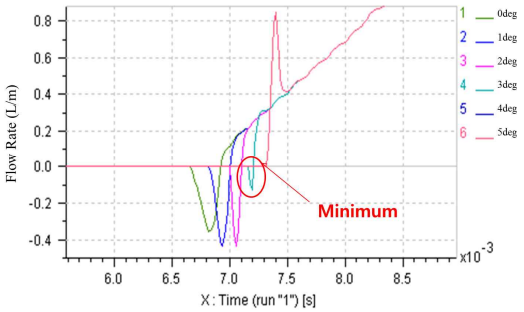
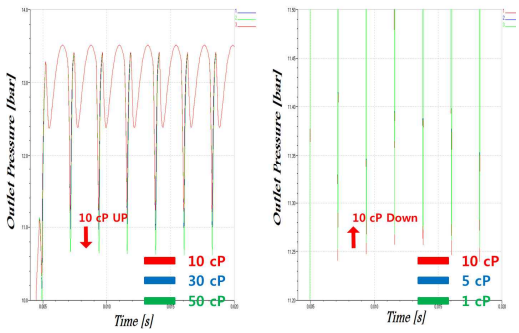


Fig. 14 Flow rate according to discharging angle of outlet

The effects of varying viscosity of hydraulic fluid on efficiency in overall flow is also analyzed with AMESim® for the hydraulic circuit model of Fig. 11. In general, SAE (Society of Automative Engineers)-defined 80W~90W (W denotes the unit of viscosity at -35 °C) lubricating fluid is used as the hydraulic fluid for the trochoid pump. Another normal hydraulic fluid, ISO (International Organization Standardization) VG32 46 is also used, which is more of a gear hydraulic fluid with high viscosity than a lubricating hydraulic fluid. The hydraulic fluid with high viscosity can be used at a high temperature, *e.g.* 80~90 °C maximum, whereas

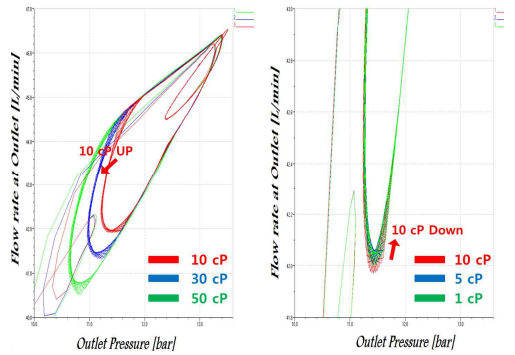
the lubricating hydraulic fluid for the trochoid pump can be used only at 60C° and below. Yet, there is no need to use the gear hydraulic fluid with high viscosity as lubrication is the main function of the hydraulic fluid. Figure 15 shows the simulation result of pressure according to the varying viscosity of the lubricating hydraulic fluid. Figure 16 represents the varying outlet flow in line with the outlet pressure as the viscosity changes. It is possible to infer from the graphs the internal flow loss according to the viscosity effect.

Figure 15 shows the varying pressure over time with the viscosity of hydraulic fluid being 20cP (centi Poise) above (see Fig. 15-(a)) and 5cP below (see Fig. 15-(b)) the reference level of 10cP. No difference in the variation of pressure is found in line with the difference in viscosity in Fig. 15. But in Fig.16-(a) the higher the viscosity, the bigger the difference in flow, leading to loss. In Fig. 16-(b), the lower the viscosity, the less the difference in flow loss below 10cP. Yet, according to Won's *et al.* [7], it would be better not to use a lubricating hydraulic fluid with viscosity below 30cP considering rust protection and performance of the hydraulic fluid.



(a) (b)

Fig. 15 Graph of outlet pressure according to viscosity



(a) (b)

Fig. 16 Graph of outlet flow according to outlet pressure

V EXPERIMENTATION

An experimentation is conducted in collaboration with *Yongdong Tech*, Ltd. in Korea, in order to find out how the optimal effects found in the simulation work in an actual pump in comparison with existing pumps. Figures 17 and 18 show the actual experimental testing set-up and the separate parts after the experimental testing, respectively, so as to check any breakage and leakage from inner and outer rotors. Especially the checking list designated by KIMM (Korea Institute of Machinery and Materials, a governmental research institute in Korea) are tested in the experiment, which lasts for a total of 1,500 hours at 1500RPM and 55~60C° .

Table 1 compares the actual measurements of varying flows between the reference pump and the one whose outlet structure is modified via simulation and thereby manufactured. The flow in the closed section inside the pump and that in the open path connected to the tank are marked. The measurements for the open flow path are marked in brackets. As for the varied angles at the outlet, efficiency changes over time. That is, the longer the duration of use, the higher the benefits. The flow increases by a small margin, which is negligible.

Fig. 17 Experimental testing set-up

Fig. 18 Separate parts after the experiment

Table 1 Experimental values of different progression

Accu-mulated time	Flow rate(cc)		Efficiency (%)		Noise (dB)
	Before	After	Change difference	Compa-rison	
4	25.1 (22.4)	25.3 (22.6)	+0.893	97	78
199	22.5 (25.8)	25.4 (22.7)	-0.439	98	71.3
271	25.3 (22.6)	25.5 (22.9)	+1.327	98	77.1
343	25.4 (20.0)	25.2 (19.9)	-0.5	97	77.3
439	25.6 (20.1)	25.5 (20.0)	-0.498	98	77.4
511	25.2 (19.9)	25.6 (20.1)	-1.005	98	77.1
607	25.0 (19.9)	25.1 (19.8)	-0.503	96	76.8
751	24.9 (19.6)	25 (19.8)	+1.02	96	77
823	25.3 (20.0)	25.1 (19.9)	-0.5	96	78
967	24.0 (19.1)	24.5 (19.5)	+2.094	94	78.8
1087	24.1 (19.2)	24.3 (19.5)	+1.016	93	78.8
1159	23.8 (18.7)	24.11 (19)	+1.604	93	78.8
1255	23.7 (18.7)	23.9 (18.7)	+0.5	93	78.9
1303	23.6 (18.6)	23.6 (18.6)	0	92	78.9
1351	23.6 (18.6)	23.5 (18.5)	-0.538	92	79
1502	23.5 (18.6)	23.7 (18.5)	-0.538	91	79

Table 2 compares the result of 1500-hour experiment of the prototype trochoid pump with the pump acceptance criteria of KIMM. Overall, the flow efficiency of the trochoid pump decreases over time. Still, its efficiency is 6% higher than the criteria. Also, noise is lower than the criteria by 6dB, which indicates the stability of the design.

Table 2 Evaluation results of experimental testing

Item	Acceptance Criteria	Testing Data of *Yongdong Tech*	Pass/Unpass
Flow rate	25.894(cc)	23.7(cc)	Pass
Noise	85dB	79dB	Pass
Effcie ncy	85%	91%	Pass

VI CONCLUSIONS

This paper provides a methodology for extracting the design data throughout a design software SolidWorks®, based on the existing trochoid pump design equations which are used by hydraulic field engineers. In the paper of Kim *et al*. [3] in WORLDCOMP 2012 (CSC 2012), we have presented how to establish the flow model of a trochoid pump by using the most popular 3-dimensional modeling tool SolidWorks® and a hydraulic analysis program AMESim®. For this model, we have figured out the clearance problem which was neglected in ref. [3]. In this paper, we have updated Kim's flow model in order to solve the clearance problem by incorporating Jang's formulation [4] into the SolidWorks® model. Through simulation results, this paper propose how to reduce the flow loss based according to the angle adjustment of the outlet of the trochoid pump. This proposal has been verified by using actual experimental results, which confirms that the design through the adjustment of outlet angle can increase flow rate. Hence this paper can be contributed to the prototyping of a trochoid pump by reducing the cost that can result from a trial-and-error design.

VII ACKNOWLEDGEMENT

This research is financially supported by Changwon National University in 2013 ~ 2014.

VIII REFERENCE

[1] S. Y. Yang, S. J. Cha, "Simulation of Gavitating Flow in a Gerotor Oil Pump," Journal of Flud Machinery, The Korean Society of Automotive Engineers, Autumn Conference, pp. 152 ~ 158, 2006.

[2] K. W. Nam, S. H. Jo and J. I. Park, "Numerical simulation in the engine lubricating gerotor oil pump," The Korea Society of Mechanical Engineers, Vol. 30, pp. 1019 ~ 1025, 2006

[3] M. S, Kim, W. J. Chung, S. W. Jeong, and J. Y. Jeon, "Methodology for simulation of trochoid pump," Journal of the KSMTE, Vol. 22, No. 3, pp. 465~471, 2013

[4] J. S. Jang, J. W. Lee, D. C. Han, and M. R. Jo, "A Study on Tooth Design Program Development of Gerotor Pump/Motor," Journal of KSTLE, Vol. 12, No. 3, pp. 100~106, 1996.

[5] B. J. Kim, S. H. Seung, and S. H. Yoon, "Flow Analysis and Port optimization of Gerotor Pump Using Commercial CFD Code," Journal of KSME, Vol. 60, No. 11, pp. 709~714, 2005.

C. S. Won, N. G. Hur, and S. H. Kwon, "Flow Analysis of Automotive Oil Pump of Gerotor Type," Journal of KSFM, Vol. 6, No. 4, pp. 7~13, 2003.

SESSION

COMPUTATIONAL SCIENCE, SYSTEMS, HPC AND APPLICATIONS

Chair(s)

TBA

Nonlinear Free Vibration of Nanobeams Subjected to Magnetic Field Based on Nonlocal Elasticity Theory

Tai-Ping Chang[1] **and Quey-Jen Yeh**[2]

[1] *Department of Construction Engineering, National Kaohsiung First University of Science and Technology, Kaohsiung, Taiwan*

[2] *Department of Business Administration, National Cheng-Kung University, Tainan, Taiwan*

Abstract *- In the present study, nonlinear free vibration behavior of nanobeam subjected to magnetic field is investigated based on Eringen's nonlocal elasticity and Euler–Bernoulli beam theory. The Hamilton's principal is adopted to derive the governing equations together with Euler–Bernoulli beam theory and the von-Kármán's nonlinear strain–displacement relationships. An approximate analytical solution is obtained for the nonlinear frequency of the nanobeam under magnetic field by using the Galerkin method and He's variational method. In the numerical results, the ratio of nonlinear frequency to linear frequency is presented. The effect of nonlocal parameter on the nonlinear frequency ratio is studied; furthermore, the effect of magnetic field on the nonlinear free vibration behavior of nanobeam is investigated.*

Keywords: Nonlinear vibration; Nanobeams; Magnetic field; Nonlocal elasticity theory; Variational method.

1 Introduction

Carbon nanotubes (CNTs) have attracted a great attention due to their superior electrical and mechanical properties and their potential applications in nanotechnology, electronics, optics and other fields of materials science. Both experimental and atomistic simulation studies show that when the dimensions of structures become very small, the size effect is important. Due to this fact, the size effect plays an important role on the mechanical behavior of micro- and nanostructures, and it cannot be ignored. Since controlled experiments in nanoscale are both difficult and expensive, the development of appropriate mathematical models for nanostructures is an important issue concerning approximate analysis of nanostructures. The nonlocal elasticity theory, which was introduced by Eringen [1] to account for scale effect in elasticity, was used to study lattice dispersion of elastic waves, wave propagation in composites, dislocation mechanics, fracture mechanics and surface tension fluids. According to the Eringen's nonlocal elasticity theory, the stress at a reference point is considered to be a function of the strains at all other points in the body. The study of Peddieson et al. [2] can be considered to be a pioneering work which first applied the nonlocal continuum theory to the nanotechnology to obtain the static deformations of beam structures by using a simplified nonlocal beam model based on the nonlocal elasticity theory of Eringen [1]. After this study, a great deal of attention has been focused on studying the static, buckling and vibration [3] analysis of nanostructures. Previous theoretical and experimental studies show that the mechanical behavior of nanostructures is nonlinear in nature when they are subjected to large external loads [4]. It is also known that when a beam with immovable end supports undergoes transverse vibration, an axial tension is produced, which is nonlinearly proportional to the amount of lateral deflection of the beam, and the beam vibrates in the nonlinear regime. Relatively little attention has been paid to the nonlinear analysis of nanostructures. For example, Yang et al. [4] studied nonlinear free vibration of single-walled carbon nanotubes (SWCNTs) based on von-Kármán's geometric nonlinearity and Eringen's nonlocal elasticity theory by using differential quadrature method. Ke et al. [5] performed the nonlinear free vibration analysis of double-walled carbon nanotubes (DWCNTs) by considering the nonlocal effect in conjunction with Timoshenko beam theory. More recently, Fang et al. [6] have examined size-dependent nonlinear vibration of DWCNTs by using the harmonic balance method and Davidon–Fletcher–Powell method. The study of nonlinear oscillators, which are described by nonlinear differential equations, is of great importance in areas of engineering, physics and other disciplines. The solution of nonlinear differential equations is more complex and therefore very time consuming. Furthermore, in most cases, the exact solution of the nonlinear equations is not possible, and therefore some approximate solution methods are needed. In this context, significant advances have been made recently in developing various analytical and numerical techniques to solve different type of nonlinear problems in structural mechanics, i.e., the multiple scales method, the elliptic Lindstedt–Poincare method, the harmonic balance method [7], the variational iteration method [8], the homotopy perturbation method [9,10]. More recently, He [11] has proposed a novel variational method to obtain a simple and efficient approximate closed form solution for nonlinear differential equations. This new method, which is Ritz-like method, can be easily extended to any nonlinear oscillator without any difficulty. For example, Fallah and Aghdam [12,13] used He's method to examine nonlinear free vibration

of functionally graded beams on nonlinear elastic foundation in the framework of the classical (local) elasticity theory. In the present paper, He's variational method is applied to nonlinear free vibration of nanobeams subjected to magnetic field based on the Eringen's nonlocal elasticity. The nonlinearity of the problem is introduced by the axial force due to the stretching. The Hamilton's principal is adopted to derive the governing equations together with Euler–Bernoulli beam theory and the von-Kármán's nonlinear strain–displacement relationships. An approximate analytical expression is obtained for the nonlinear frequency of the nanobeam by utilizing the Galerkin method and He's variational method. Some numerical examples are presented for the nonlinear frequency ratio for nanobeams with magnetic field.

2. The governing equation based on the nonlocal elasticity theory

Based on the Euler–Bernoulli beam theory, the displacement field of any point of the beam is given as

$$u_x(x,z,t) = u(x,t) - z\frac{\partial w(x,t)}{\partial x} \quad (1)$$

$$u_y(x,z,t) = 0 \quad (2)$$

$$u_z(x,z,t) = w(x,t) \quad (3)$$

where u and w are the axial and the transverse displacement of any point on the neutral axis. The von-Kármán's nonlinear strain–displacement relationship based on assumptions of large transverse displacements, moderate rotations and small strains for a straight beam are given by

$$\varepsilon_{xx} = \varepsilon_{xx}^0 - z\kappa_x \quad (4)$$

$$\varepsilon_{xx}^0 = \frac{\partial u(x,t)}{\partial x} + \frac{1}{2}\left(\frac{\partial w(x,t)}{\partial x}\right)^2,\ \kappa_x = \frac{\partial^2 w(x,t)}{\partial x^2} \quad (5)$$

where ε_{xx} is the longitudinal strain, ε_{xx}^0 is the nonlinear membrane strain, κ_x is the curvature of the beam. In this study, the equations of motion are derived by using Hamilton's principle. This principle can be expressed as

$$\delta\int_0^t \left[K - (U - W)\right]dt = 0 \quad (6)$$

where K is the kinetic energy, U is the strain energy and W is the work done by the external applied forces. Based on nonlocal elasticity theory, the nonlocal governing equations in terms of the displacements can be obtained as follows:

$$EI\frac{\partial^4 w}{\partial x^4} + \left[\frac{EA}{2L}\int_0^L\left(\frac{\partial w}{\partial x}\right)^2 dx\right]\left[(e_0a)^2\frac{\partial^4 w}{\partial x^4} - \frac{\partial^2 w}{\partial x^2}\right] + \rho A\frac{\partial^2}{\partial t^2}\left[w - (e_0a)^2\frac{\partial^2 w}{\partial x^2}\right] = q_w - (e_0a)^2\frac{\partial^2 q_w}{\partial x^2} \quad (7)$$

where q_w is distributed transverse load, e_0 is a constant appropriate to each material, a is an internal characteristic length (e.g., length of C–C bond, lattice parameter, and granular distance). Now let us consider the nanobeam is subjected to an externally applied longitudinal magnetic field, then the governing equation of motion of the system can be expressed as follows:

$$EI\frac{\partial^4 w}{\partial x^4} + \left[\frac{EA}{2L}\int_0^L\left(\frac{\partial w}{\partial x}\right)^2 dx\right]\left[(e_0a)^2\frac{\partial^4 w}{\partial x^4} - \frac{\partial^2 w}{\partial x^2}\right] + \rho A\times\frac{\partial^2}{\partial t^2}\left[w - (e_0a)^2\frac{\partial^2 w}{\partial x^2}\right] - f(x,t) + (e_0a)^2\frac{\partial^2 f}{\partial x^2} = q_w - (e_0a)^2\frac{\partial^2 q_w}{\partial x^2} \quad (8)$$

where $f(x,t) = \int_A \bar{f}_z dz = \xi A H_x^2\frac{\partial^2 w}{\partial x^2}$, ξ is the magnetic field permeability and H_x is the longitudinal magnetic field. In order to obtain general results, the following non-dimensional quantities can be defined:

$$\bar{x} = \frac{x}{L}, \bar{w} = \frac{w}{r}, t = \bar{t}\sqrt{\frac{\rho AL^4}{EI}}, \beta = \frac{e_0a}{L} \quad (9)$$

where $r = \sqrt{I/A}$ is the radius of gyration of the cross-section. Using Eq. (9) and neglecting the distributed load q_w for free vibration analysis, the equation of motion can be written in the non-dimensional form as follows:

$$\frac{\partial^4\bar{w}}{\partial\bar{x}^4} + \left[\frac{1}{2}\int_0^1\left(\frac{\partial^2\bar{w}}{\partial\bar{x}^2}\right)^2 d\bar{x}\right]\left(\beta^2\frac{\partial^4\bar{w}}{\partial\bar{x}^4} - \frac{\partial^2\bar{w}}{\partial\bar{x}^2}\right) + \frac{\partial^2}{\partial\bar{t}^2}\left[\bar{w} - \beta^2\frac{\partial^2\bar{w}}{\partial\bar{x}^2}\right] - H_1\frac{\partial^2\bar{w}}{\partial\bar{x}^2} + \beta^2 H_2\frac{\partial^4\bar{w}}{\partial\bar{x}^4} = 0 \quad (10)$$

where $H_1 = \xi H_x^2\frac{L^2A}{EI}$, $H_2 = \xi H_x^2\frac{L^4}{E}$, both H_1 and H_2 are dimensionless magnetic field intensity. According to the usual Galerkin method, an approximate solution for $\bar{w}(\bar{x},\bar{t})$ is assumed as

$$\bar{w}(\bar{x},\bar{t}) = q(\bar{t})N(\bar{x}) \quad (11)$$

where $q(\bar{t})$ is the unknown time-dependent coefficient to be determined and $N(\bar{x})$ is the basis function which must satisfy the kinematic boundary conditions.

Substituting the approximate solution in Eq. (11) into Eq. (10), then multiplying both sides of the resulting equation with $N(x)$ and integrating it over the domain (0, 1) yields

$$\ddot{q}(\bar{t}) + K_1 q(\bar{t}) + (K_2 + K_3) q^3(\bar{t}) = 0 \tag{12}$$

Here $\ddot{q}$ is the second derivative of q with respect to time. The coefficients K_1, K_2 and K_3 in Eq. (12) can be expressed as

$$\begin{aligned}
&K_1 = \frac{\bar{K}_1}{\bar{K}_0},\ K_2 = \frac{\bar{K}_2}{\bar{K}_0},\ K_3 = \frac{\bar{K}_3}{\bar{K}_0},\\
&\bar{K}_0 = \int_0^1 N^2 d\bar{x} - \beta^2 \int_0^1 N'' N d\bar{x},\\
&\bar{K}_1 = \int_0^1 N^{(4)} N d\bar{x} - H_1 \int_0^1 N'' N d\bar{x} + \beta^2 H_2 \int_0^1 N^{(4)} N d\bar{x},\\
&\bar{K}_2 = -\frac{1}{2}\int_0^1 (N')^2 d\bar{x} \int_0^1 N'' N d\bar{x},\\
&\bar{K}_3 = \frac{1}{2}\beta^2 \int_0^1 (N')^2 d\bar{x} \int_0^1 N^{(4)} N d\bar{x}
\end{aligned} \tag{13}$$

where $N(x)$ is the fourth derivative of N with respect to $\bar{x}$. It should be noted that the midpoint of the nanobeam is subjected to the following initial conditions

$$q(0) = \alpha,\ \dot{q}(0) = 0 \tag{14}$$

where $\alpha = w_{max}/r$ is the dimensionless maximum vibration amplitude of the nanobeam.

3. Analytical solution based on He's variational method

By using the semi-inverse method, Eq. (12) can be expressed as

$$J(q) = \int_0^{T/4} \left(-\frac{1}{2}\dot{q}^2 + K_1 \frac{q^2}{2} + (K_2 + K_3)\frac{q^4}{4} \right) d\bar{t} \tag{15}$$

Here T is the period of the nonlinear oscillator. When the approximate solution $q(t) = \alpha \cos \omega t$ that satisfies the initial conditions in Eq. (14) is considered with the transformation $\theta = \omega t$, one can obtain

$$J(\alpha, \omega) = \frac{1}{\omega}\int_0^{\pi/2} \left(\frac{1}{2}\alpha^2\omega^2 \sin^2\theta + \frac{K_1}{2}\alpha^2 \cos^2\theta + \frac{(K_2+K_3)}{4}\alpha^4 \cos^4\theta \right) d\theta \tag{16}$$

The stationary condition dJ/dα = 0 results in

$$\frac{dJ}{d\alpha} = \frac{1}{\omega}\int_0^{\pi/2} \left(-\alpha\omega^2 \sin^2\theta + K_1 \alpha \cos^2\theta + (K_2 + K_3)\alpha^3 \cos^4\theta \right) d\theta = 0 \tag{17}$$

After some amendment, Eq. (17) takes the following form

$$\omega^2 = \frac{\int_0^{\pi/2} \left(K_1 \cos^2\theta + (K_2 + K_3)\alpha^2 \cos^4\theta \right) d\theta}{\int_0^{\pi/2} \sin^2\theta d\theta} \tag{18}$$

The nonlinear natural frequency ω can be found by performing the integral expression in Eq. (18) as follows:

$$\omega^2 = K_1 + \frac{3}{4}(K_2 + K_3)\alpha^2 \tag{19}$$

Then the following approximate solution can be found for $q(t)$

$$q(\bar{t}) = \alpha \cos\left(\sqrt{K_1 + \frac{3}{4}(K_2 + K_3)\alpha^2}\, \right) \bar{t} \tag{20}$$

4. Numerical results and discussion

First of all, nonlinear free vibration of nanobeams without magnetic field is obtained based on the nonlocal elasticity and Euler–Bernoulli beam theory. For comparison with the previously published results, the ratio of nonlinear frequency to linear frequency is presented. It is seen from Eq. (19) that if α is set to zero, the linear frequency ω_L is obtained, namely $D_1 = \omega_L^2$. In the light of this information, the nonlinear frequency ratio is expressed as

$$\omega_{ratio} = \frac{\omega_{NL}}{\omega_L} = \sqrt{1 + \frac{3}{4}\frac{(K_2 + K_3)}{K_1}\alpha^2} \tag{21}$$

In the numerical results, the nonlinear frequency ratio is presented for various values of the dimensionless nonlocal parameter ($\beta = e_0 a / L$) and the dimensionless amplitude (α). In this study, the boundary conditions of the nanobeam are considered as simply supported at both ends. In order to validate the present results, some comparisons with previous studies have been carried out. The formulation and solution method proposed herein is validated against the available results regarding to the nonlinear frequency ratio of beams based on the classical beam theory [12] and [17]. In Table 1, the nonlinear frequency ratio values obtained by the present method are compared with the results of [12] and [17].

According to this table it is observed that the nonlinear frequency ratio values generated from this study agree well with the results from Refs. [12] and [17].
Figs. 1 presents the variation of the nonlinear frequency ratio with the dimensionless amplitude for the selected values of the dimensionless nonlocal parameter (i.e., $\beta = 0$, 0.1, 0.2, 0.3 and 0.4) without magnetic field or with magnetic field ($H_1 = 5, H_2 = 0.1$). The figure reveals that the nonlinear frequency ratio increases as the dimensionless amplitude increases.

Table 1
Comparison of the nonlinear frequency ratio for various values of dimensionless amplitude.

Dimensionless amplitude (α)	Present	Ref. [12]	Ref. [17]
1	1.0897	1.0897	1.0897
2	1.3228	1.3229	1.3228
3	1.6393	1.6393	1.6393
4	2.0000	1.9999	1.9999

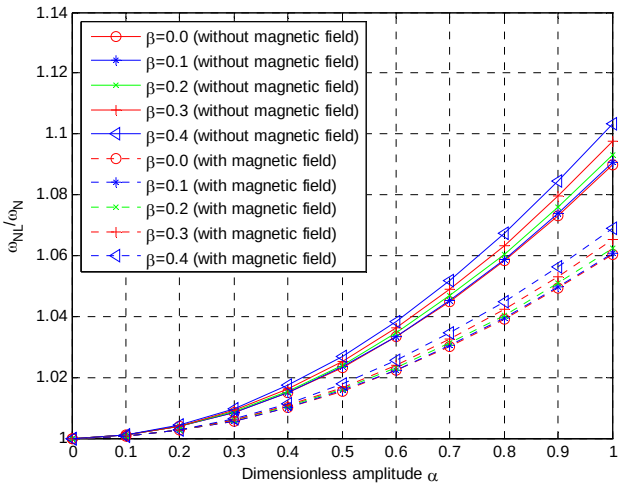


Fig.1. Variation of nonlinear frequency ratio with dimensionless amplitude without magnetic field or with magnetic field.

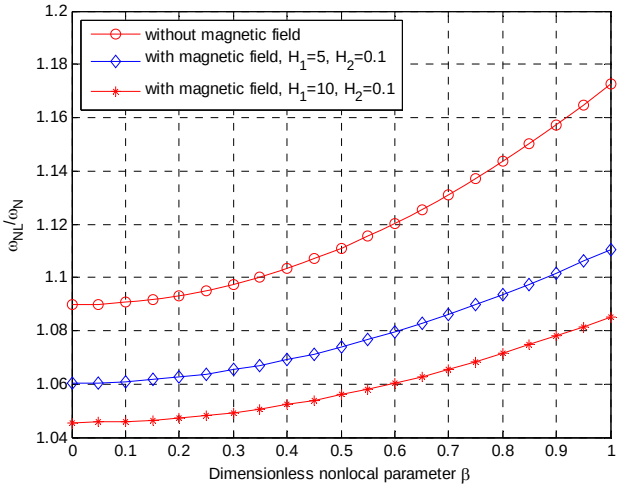


Fig. 2. Variation of nonlinear frequency ratio with dimensionless nonlocal parameter without magnetic field or with magnetic field.

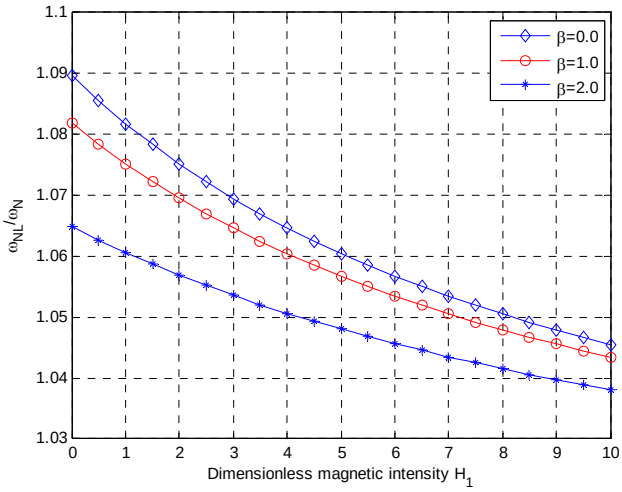


Fig. 3. Variation of nonlinear frequency ratio with dimensionless magnetic intensity H_1 for various dimensionless nonlocal parameter.

This behavior is known as "hardening spring" behavior. The reason of this behavior is that increasing the dimensionless amplitude implies increasing the axial stretching due to the large deflection, and that leads to a stiffer structure and a larger nonlinear frequency. Furthermore, the nonlinear frequency ratio decreases when the nanobeam is subjected to the magnetic field.

Fig. 2 displays the variation of the nonlinear frequency ratio with the dimensionless nonlocal parameter at a given value of the dimensionless amplitude ($\alpha = 1$). Obviously, the nonlinear frequency ratio increases with the increase of the dimensionless nonlocal parameter, moreover, the nonlinear frequency ratio decreases when the magnetic field intensity increases. Based on the results in Figs. 1-2, it is observed both the dimensionless nonlocal parameter and the dimensionless amplitude cause a rise in the nonlinear frequency. Also, note that the nonlinear frequency is nonlinearly dependent on the dimensionless nonlocal parameter and vibration amplitude. Fig. 3 presents the variation of nonlinear frequency ratio with dimensionless magnetic intensity for various dimensionless nonlocal parameters. As it can be detected from the figures, the nonlinear frequency ratio decreases with the increase of the magnetic field intensity, which is reasonable.

5. Conclusions

Nonlinear free vibration behavior of nanobeam subjected to magnetic field is investigated based on the Eringen's nonlocal elasticity and Euler–Bernoulli beam theory. The governing equations are derived using the Hamilton's principal. The nonlinearity of the problem stems from the von-Kármán's nonlinear strain–displacement relationships. In the present paper, He's variational method is applied to the problem of nonlinear free vibration of nanobeams. An approximate analytical solution is obtained for the nonlinear frequency of the nanobeam by using He's variational method. The effects of the dimensionless nonlocal parameter on the nonlinear frequency ratio are discussed. Numerical results show that the nonlocal effects play an important role on the nonlinear responses of nanobeams. The new nonlocal beam model produces larger nonlinear frequency ratio than the classical (local) beam model. Therefore, the nonlocal effects should be considered in the analysis of mechanical behavior of nanostructures. It is concluded that the nonlinear frequency ratio of nanobeam with magnetic field decreases with the increase of the magnetic field intensity.

Acknowledgments This research was partially supported by the National Science Council in Taiwan through Grant NSC-99-2221-E-327-020. The authors are grateful for this support.

References

[1] A.C. Eringen, On differential equations of nonlocal elasticity and solutions of screw dislocation and surface waves, J Appl Phys, 54 (1983) 4703–4710.
[2] J. Peddieson, G.R. Buchanan, R.P. McNitt, Application of nonlocal continuum models to nanotechnology, Int J Eng Sci, 41 (2003) 305–312.
[3] M. Aydogdu, A general nonlocal beam theory: its application to nanobeam bending, buckling and vibration, Physica E, 41 (2009) 1651–1655.
[4] J. Yang, L.L. Ke, S. Kitipornchai, Nonlinear free vibration of single-walled carbon nanotubes using nonlocal Timoshenko beam theory, Physica E, 42 (2010) 1727–1735.
[5] L.L. Ke, Y. Xiang, J. Yang, S. Kitipornchai, Nonlinear free vibration of embedded double-walled carbon nanotubes based on nonlocal Timoshenko beam theory, Comput Mater Sci, 47 (2009) 409–417.
[6] B. Fang, Y.X. Zhen, C.P. Zhang, Y. Tang, Nonlinear vibration analysis of double-walled carbon nanotubes based on nonlocal elasticity theory, Appl Math Model, 37 (2013) 1096–1107.
[7] J.H. He, Homotopy perturbation technique, Comput Meth Appl Mech Eng, 178 (1999. 257–262.
[8] J.H. He, Some asymptotic methods for strongly nonlinear equations, Int J Modern Phys B, 20 (2006. 1141–1199.
[9] D.D. Ganji, A. Sadighi, Application of He's homotopy-perturbation method to nonlinear coupled systems of reaction-diffusion equations, Int J Nonlinear Sci Numer Simul, 7 (2006) 411–418.
[10] T. Ozis, A. Yildirim, Traveling wave solution of Korteweg-de Vries equation using He's homotopy perturbation method, Int J Nonlinear Sci Numer Simul, 8 (2007) 239–242.
[11] J.H. He, Variational approach for nonlinear oscillators, Chaos, Solitons and Fractals, 34 (2007) 1430–1439.
[12] A. Fallah, M.M. Aghdam, Nonlinear free vibration and post-buckling analysis of functionally graded beams on nonlinear elastic foundation, Eur J Mech/A Solid, 30 (2011) 571–583.
[13] A. Fallah, M.M. Aghdam, Thermo-mechanical buckling and nonlinear free vibration analysis of functionally graded beams on nonlinear elastic foundation, Composites: Part B, 43 (2012) 1523–1530.
[14] Q. Wang, Wave propagation in carbon nanotubes via nonlocal continuum mechanics, J Appl Phys, 98 (2005) 124301.

[15] S.A. Emam, A static and dynamic analysis of the postbuckling of geometrically imperfect composite beam, Compos Struct, 90 (2009) 247–253.
[16] S.A. Emam, A.H. Nayfeh, Postbuckling and free vibrations of composite beams, Compos Struct, 88 (2009), pp. 636–642.
[17] T. Pirbodaghi, M.T. Ahmadian, M. Fesanghary, On the homotopy analysis method for non-linear vibration of beams, Mech Res Commun, 36 (2009) 143–148.

Queuing Network Approximation Technique for Evaluating Performance of Computer Systems with Memory Resource used by Multiple job types

Mizuki Takaya, Mie Ogiwara, Noorafiza Matrazali,
Chiaki Itaba, Itaru Koike, Toshiyuki Kinoshita

School of Computer Science, Tokyo University of Technology
1404-1 Katakura, Hachioji Tokyo, 192-0982, Japan

Abstract *Queuing network techniques are effective for evaluating the performance of computer systems. We discuss a queuing network technique for computer systems with multiple types of jobs and a single type of memory resource. When a job arrives from outside the network, it occupies one of the memory resources and executes CPU and I/O processing in the network occupying the memory. When the job completes the CPU and I/O processing, it releases the memory and leaves the network. However, because the memory resource is considered to be a secondary resource for the CPU and I/O equipment, and because the queuing network model of computer systems with memory resources is an open one, we cannot calculate its exact solutions.*

We propose here an approximation queuing network technique for calculating the performance measures of computer systems on which multiple types of jobs compete for a single type of memory resource. This technique involves dividing the queuing network into two parts; one is a "processing part" in which a job executes CPU and I/O processing, and the other is a "memory part" that indicates how the memory resources are used by jobs. By dividing the network into two parts, we can prevent the number of network states from increasing and can approximately calculate the performance measures of the network. We evaluated the proposed approximation technique using numerical experiments.

Keywords *performance of computer systems, central server model, memory resource, memory resource requirements*

1. Introduction

Queuing network techniques are effective for evaluating the performance of computer systems. In computer systems, two or more jobs are generally executed at the same time, which causes delays due to conflicts in accessing hardware or software resources such as the CPU, I/O equipment, or data files. We can evaluate how this delay affects the computer system performance by using a queuing network technique. Some queuing networks have an explicit exact solution, which is called a product form solution [1]. With this solution, we can easily calculate the performance measures of computer systems, for example the busy ratio of hardware and the job response time. However, when the exclusion controls are active or when a memory resource exists, the queuing network does not have a product form solution. To calculate an exact solution of a queuing network that does not have a product form solution, we have to construct a Markov chain that describes the stochastic characteristics of the queuing network and numerically solve its equilibrium equations. The number of unknown quantities in the equilibrium equations is the same as the number of states of the queuing network. Since the number of states of the queuing network drastically increases when the number of jobs or the amount of hardware in the network increases, the number of unknown quantities in the equilibrium equations also drastically increases. Therefore, we cannot numerically calculate the exact solution of the queuing network. Moreover, when the queuing network is an open model where jobs arrive from or depart for the outside of the network, the number of states of the network can become infinite (the number of jobs can be infinite.), and we cannot actually calculate an exact solution.

We discuss the queuing network technique for computer systems with memory resources. When a job arrives from outside the network, it occupies a portion of the memory resources and executes CPU and I/O processing in the network with the memory. When the job completes CPU and I/O processing, it releases the memory and leaves the network. Therefore, memory can be considered as a secondary resource for the CPU and I/O equipment. When a queuing network includes a secondary resource, it does not have product form solutions. Moreover, since the queuing network technique of computer systems with memory resources is an open model, we cannot calculate its exact solutions.

We propose an approximation technique for calculating the performance measures of computer systems with memory resource requirements. We previously reported the results for a single job type case in [6], and proposed the approximation technique for computer systems with memory resource requirements by multiple types of jobs in [7]. In this paper, we consider them for the case when the multiple types of jobs compete for a single type of memory resource (see Figure 1). Similarly in [6] and [7], we divide the network into two parts in order to prevent the number of states of the Markov chain from increasing. One part is called the "processing part," in which jobs execute CPU and I/O processing, and the other is the "memory part," which indicates how the memory resources are used by the jobs. As with the behavior of CPU and I/O processing, the memory usage behavior also differs for each job class, such as how much memory is allocated to the job. When there is a single job class, both the processing and memory parts have a product form solution, but when there are multiple job classes, only the processing part has a product form solution. Therefore, it is not sufficient to divide the queuing network into two parts; an approximation is also needed to analyze the memory part. Dividing the model into primary and secondary resources is a two-layer queuing network techniques [3][4]. Our proposed technique is also a two-layer technique for computer systems with memory resources.

In our previous study [5], we reported about approximation technique for evaluating performance of computer systems with file resources. Meanwhile, heterogeneous parallel computer systems with distributed memory is researched in [8], and the Markov chain involving two dimensional state transition similar to our proposed model in this paper was discussed in [9].

2. Model Description

The processing part is equivalent to the ordinary central server model with multiple job types (each of which is called a job classe). In this model, K job classes exist, and each of them is numbered k =1, 2, … , K by affixing a k. The processing part consists of a single CPU node and multiple I/O nodes. We denote M as the number of I/O nodes. The I/O nodes are numbered m =1, 2, … , M by affixing m, and the CPU node is numbered m =0 by also affixing m. The service rate of job class k at the CPU node is μ_0^k, and the service rate of job class k at an I/O node m is μ_m^k. The service time at each node is a mutually independent random variable subject to common exponential distributions. Jobs are scheduled on a first come first served (FCFS) principle at all nodes. At the end of

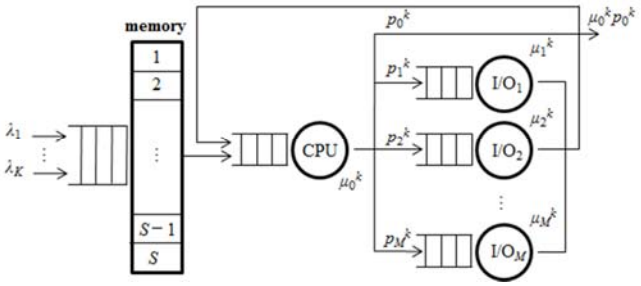


Fig. 1 Central server model with memory resource requirements

CPU processing, a job probabilistically selects a I/O node and moves to it, or completes CPU and I/O processing and departs from the network. The selection probability of I/O node m of job class k is p_m^k (k =1, 2, ... , K; m =1, 2, ... , M), and the completion probability of job class k is p_0^k (k =1, 2, ... , K). Therefore, $\sum_{m=0}^{M} p_m^k = 1$ (k =1, 2, ... , K).

Memory resources are added to this ordinary central server model (Figure 1). We denote S as the number of memory resources. A job from job class k arrives from the outside at random at an arrival rate λ_k, and requests and acquires a memory resource before entering the processing part. If all the memory resources are occupied when the job arrives, the job joins the memory-waiting queue and waits for one of the memory resources to be released by another job. When the job completes CPU and I/O processing, it releases the memory resource and leaves the network. Because the job has to occupy a memory resource upon entering the processing part, the number of jobs occupying a memory resource is always equal to the number of jobs in the processing part. Therefore, at most S jobs can enter the processing part. That is, the maximum job multiplicity in the processing part is S. When the number of jobs of job class k in the processing part is denoted by n_k, $\sum_{k=1}^{K} n_k \le S$.

By replacing "CPU → outside transition" with "CPU → CPU transition," the processing part is modified to a closed central server model in which the number of jobs is constant (Figure 2). In this closed model, when "CPU → CPU transition" occurs, the job terminates and a new job starts. Therefore, the mean job response time is the mean time between two successive "CPU → CPU transitions." This means job response time can be considered as a job lifetime.

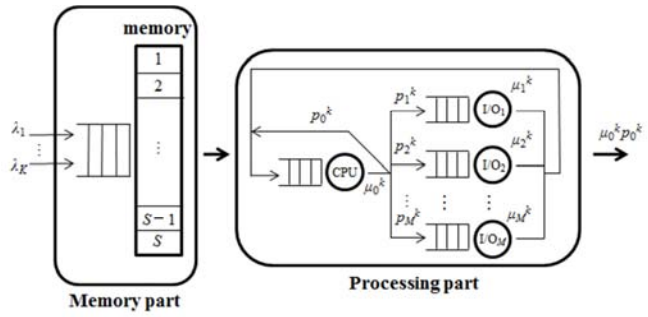


Fig. 2 Concept of approximation

3. Approximation Model

To obtain the exact solution of the central server model with memory resource requirements, we have to describe the entire model with a single Markov chain for each job class. However, this causes the number of states of the Markov chain to drastically increase when the number of nodes and the number of jobs in the network increase. By dividing the central server model into two parts (processing part and memory part), and describing each part with two Markov chains, we can prevent the number of states of the model from increasing (Figure 2). We set the following notations.

τ_{km} : total mean service time at node-m in a job lifetime of job class k

n_{km} : number of jobs in job class k at node-m (k=1, 2, ... , K; m =0, 1, ... , M)

$\boldsymbol{n}^* = (n_1, n_2, \dots, n_K)$: vector of number of jobs (n_k=0, 1, 2, ...)

$\boldsymbol{n} = (n_{10}, n_{11}, \dots, n_{1M}, n_{20}, n_{21}, \dots, n_{2M}, \dots, n_{K0}, n_{K1}, \dots, n_{KM})$: state vector of the processing part

$$\boldsymbol{F}(\boldsymbol{n}) = \{\boldsymbol{n} \mid \sum_{m=0}^{M} n_{km} = n_k,\ n_{km} \geq 0\ (m=0, 1, \dots, M)\} \qquad (n_1+n_2+\dots+n_K \leqq S)$$

: set of all feasible states of the processing part when the number of jobs of job class k is n_k

$P_s(\boldsymbol{n})$: steady-state probability of state $\boldsymbol{n}$

The processing part is equivalent to the ordinary central server model with multiple job classes, and therefore, the Markov chain describing the processing part has a product form solution. Then the steady-state probability $P_s(\boldsymbol{n})$ is represented by the following formula [1][2].

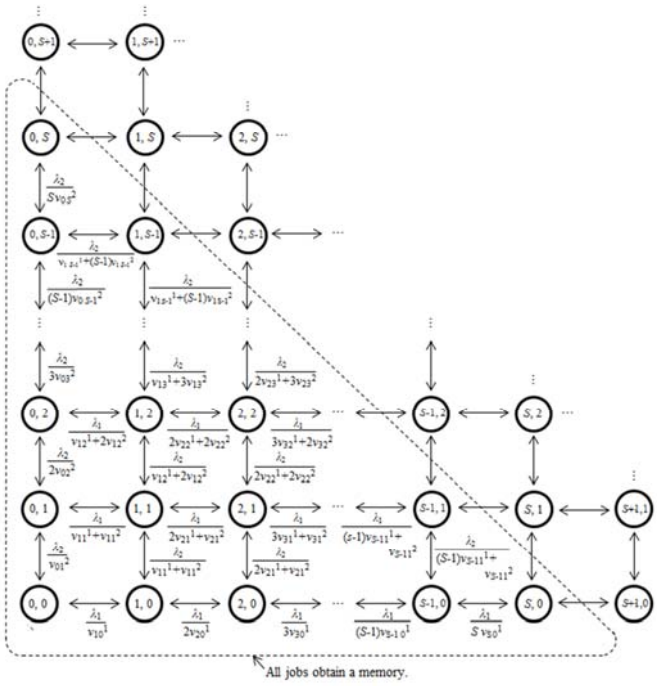


Fig. 3 State Transition diagram (two job classes)

$$P_s(\boldsymbol{n}) = \frac{\prod_{k=1}^{K}\prod_{m=0}^{M} \tau_{km}^{\,n_{km}}}{\varphi(n_1, n_2, \cdots, n_K, M)}$$

where $\varphi(n_1, n_2, \cdots, n_K, M) = \sum_{n \in F(n)} \prod_{k=1}^{K}\prod_{m=0}^{M} \tau_{km}^{\,n_{km}}$ is the normalizing constant of steady-state probabilities when the number of jobs of job class k in the processing part is n_k (k =1, 2, ... , K). From these steady-state probabilities, we can calculate the mean job response time $T_{\boldsymbol{n}}^{\,k}$ of job class k in the processing part, when the number of jobs is n_k (k =1, 2, ... , K), as follows.

$$T_n^{\,k} = \frac{n_k \cdot \varphi(n_1, \cdots, n_k, \cdots, n_K, M)}{\varphi(n_1, \cdots, n_k - 1, \cdots, n_K, M)}$$

The memory part can be considered as an M/M/S queuing model with S servers. In an ordinary M/M/S queuing model, the service rate at a server is constant, regardless of the number of guests in the service. In the memory part, however, the service rate changes depending on the number of occupied memory resources. The mean job response time $T_{\boldsymbol{n}}^{\,k}$ of job class k (k =1, 2, ... , K) in the processing part, when the number of jobs in the processing part is $\boldsymbol{n}^* = (n_1, n_2, \dots, n_K)$, is equal to the mean time of occupied memory. Since the service rate from the processing part $v_{\boldsymbol{n}}^{\,k}$ is denoted as $v_{\boldsymbol{n}}^{\,k} = \frac{1}{T_{\boldsymbol{n}}^{\,k}}$, $v_{\boldsymbol{n}}^{\,k}$ depends on the number of occupied

memory resources n_k. The state transition of the M/M/S queuing model with two job classes is shown in Figure 3, where the memory service rates change depending on the number of occupied memory resources. This is a two dimensional birth-death process. The equilibrium equations with the steady-state probability $Q_S(\boldsymbol{n}^*)=Q_S(n_1, n_2)$, when the number of memory resources is S and the number of occupied memory resources is $\boldsymbol{n}^*=(n_1, n_2)$, are as follows (similar to the case with higher dimensions).

(1) $n_1=0, n_2=0$

$$(\lambda_1+\lambda_2)\cdot Q_S(0,0) = v_{10}^{1}\cdot Q_S(1,0)+v_{01}^{2}\cdot Q_S(0,1)$$

(2) $n_1=1, 2, \ldots, S-1, n_2=0$

$$(\lambda_1+\lambda_2+n_1 v_{n_1 0}^{1})\cdot Q_S(n_1,0) = \lambda_1\cdot Q_S(n_1-1,0) + (n_1+1)\, v_{n_1+1 0}^{1}\cdot Q_S(n_1+1,0)+(n_1 v_{n_1 1}^{1}+v_{n_1 1}^{2})\cdot Q_S(n_1,1)$$

(3) $n_1=S, S+1, \ldots, n_2=0$

$$(\lambda_1+\lambda_2+S v_{S0}^{1})\cdot Q_S(n_1,0) = \lambda_1\cdot Q_S(n_1-1,0)+ S v_{S0}^{1}\cdot Q_S(n_1+1,0)+(S v_{S1}^{1}+v_{S1}^{2})\cdot Q_S(n_1,1)$$

(4) $n_1=0, n_2=1, 2, \ldots, S-1$

$$(\lambda_1+\lambda_2+n_2 v_{0 n_2}^{2})\cdot Q_S(0,n_2) = \lambda_2\cdot Q_S(0,n_2-1)+ (v_{1 n_2}^{1}+n_2 v_{1 n_2}^{2})\cdot Q_S(1,n_2)+(n_2+1)v_{0 n_2+1}^{2}\cdot Q_S(0,n_2+1)$$

(5) $n_1=0, n_2=S, S+1, \ldots$

$$(\lambda_1+\lambda_2+S v_{0S}^{2})\cdot Q_S(0,n_2) = \lambda_2\cdot Q_S(0,n_2-1)+ (v_{1S}^{1}+n_2 v_{1S}^{2})\cdot Q_S(1,n_2)+S v_{0S}^{2}\cdot Q_S(0,n_2+1)$$

(6) $n_1+n_2 \leqq S-1, n_1=1, 2, \ldots, S-2, n_2=1, 2, \ldots, S-2$

$$(\lambda_1+\lambda_2+n_1 v_{n_1 n_2}^{1}+n_2 v_{n_1 n_2}^{2})\cdot Q_S(n_1,n_2) = \lambda_1\cdot Q_S(n_1-1,0)+\lambda_2\cdot Q_S(0,n_2-1)+ ((n_1+1)\, v_{n_1+1 n_2}^{1}+n_2 v_{n_1+1 n_2}^{2})\cdot Q_S(n_1+1,n_2)+ (n_1 v_{n_1 n_2+1}^{1}+(n_2+1)\, v_{n_1 n_2+1}^{2})\cdot Q_S(n_1,n_2+1)$$

(7) $n_1+n_2=S, n_1=1, 2, \ldots, S-1, n_2=1, 2, \ldots, S-1$

$$(\lambda_1+\lambda_2+n_1 v_{n_1 n_2}^{1}+n_2 v_{n_1 n_2}^{2})\cdot Q_S(n_1,n_2) = \lambda_1\cdot Q_S(n_1-1,0)+\lambda_2\cdot Q_S(0,n_2-1)+ (n_1 v_{n_1 n_2}^{1}+n_2 v_{n_1 n_2}^{2})\cdot Q_S(n_1+1,n_2)+ (n_1 v_{n_1 n_2}^{1}+n_2 v_{n_1 n_2}^{2})\cdot Q_S(n_1,n_2+1)$$

(8) $n_1+n_2>S, n_1=1, 2, \ldots, n_2=1, 2, \ldots$

When the lattice point (m_1, m_2) such as $m_1+m_2=S$ is on the shortest route from (0, 0) to (n_1, n_2),

$$(\lambda_1+\lambda_2+m_1 v_{m_1 m_2}^{1}+m_2 v_{m_1 m_2}^{2})\cdot Q_S^{m}(n_1,n_2) = \lambda_1\cdot Q_S^{m}(n_1-1,0)+\lambda_2\cdot Q_S^{m}(0,n_1-1)+ (m_1 v_{m_1 m_2}^{1}+m_2 v_{m_1 m_2}^{2})(Q_S^{m}(n_1+1,n_2)+Q_S^{m}(n_1,n_1+1)).$$

(a) $n_1+n_2>S, n_1=1, 2, \ldots, S, n_2=1, 2, \ldots, S$

$$\Rightarrow Q_S(n_1,n_2) = \sum_{m=S-n_2}^{n_1} Q_S^{m}(n_1,n_2)$$

(b) $n_1+n_2>S, n_1=S+1, S+2, \ldots, n_2=1, 2, \ldots, S$

$$\Rightarrow Q_S(n_1,n_2) = \sum_{m=S-n_2}^{S} Q_S^{m}(n_1,n_2)$$

(c) $n_1+n_2>S, n_1=1, 2, \ldots, S, n_2=S+1, S+2, \ldots$

$$\Rightarrow Q_S(n_1,n_2) = \sum_{m=0}^{n_1} Q_S^{m}(n_1,n_2)$$

(d) $n_1+n_2>S, n_1=S+1, S+2, \ldots, n_2=S+1, S+2, \ldots$

$$\Rightarrow Q_S(n_1,n_2) = \sum_{m=0}^{S} Q_S^{m}(n_1,n_2)$$

For the state (n_1, n_2) of the Markov chain, all jobs obtain the memory resource and are in the processing part when $n_1+n_2 \leqq S$, and n_1+n_2-S jobs are in the memory waiting queue and waiting for one of the memory resources to be released when $n_1+n_2>S$. The transition diagram of the two dimensional birth-death process is shown in Figure 3. However, the steady-state equation does not have a product form solution. Therefore, some approximation is required to solve it.

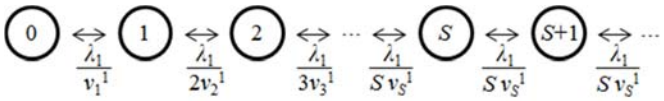


Fig. 4 State transition diagram (single job class)

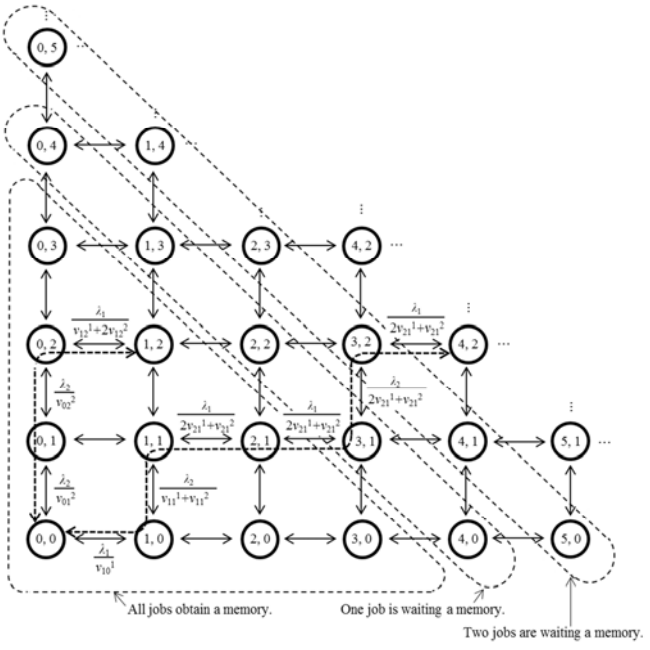


Fig. 5 Calculating state probability for two job classes

When a single job class and a single memory resource exist in the model, it can be described with a one dimensional birth-death process. Its transition diagram is shown in Figure 4, and the steady-state equation is as follows:

$$\lambda_1 \cdot Q_S(0) = v_1^1 \cdot Q_S(1)$$
$$(\lambda_1 + n_1 v_{n_1}^1) \cdot Q_S(n_1) = \lambda_1 \cdot Q_S(n_1 - 1) + (n_1+1) v_{n_1+1}^1 \cdot Q_S(n_1+1) \quad (n_1 = 1, 2, \ldots, S-1)$$
$$(\lambda_1 + S v_S^1) \cdot Q_S(n_1) = \lambda_1 \cdot Q_S(n_1 - 1) + S v_S^1 \cdot Q_S(n_1+1) \quad (n_1 = S, S+1, \ldots)$$

Solutions for the steady-state equation are in the following product form.

$$Q_S(n_1) = Q_S(0) \cdot \frac{1}{n_1!} \prod_{i=1}^{n_1} \left(\frac{\lambda_1}{v_i^1} \right) \qquad (n_1 = 1, 2, \ldots, S-1)$$
$$= Q_S(0) \cdot \frac{1}{S! S_1^{n_1 - S}} \prod_{i=1}^{S} \left(\frac{\lambda_1}{v_i^1} \right) \cdot \left(\frac{\lambda_1}{v_S^1} \right)^{n_1 - S} \qquad (n_1 = S, S+1, \ldots)$$

In this formula, for the state transition 1at $i = 1, 2, \ldots, S_1 - 1$, multiply by factor $\frac{\lambda_1}{i \cdot v_i^1}$ while for the state transition at $i = S_1, S_1+1, \ldots$, multiply by factor $\frac{\lambda_1}{S_1 \cdot v_{S_1}^1}$. For two dimension case, we consider a route from lattice point (0, 0) to (n_1, n_2) shown in Figure 5, and for the horizontal state transition at the lattice point (i_1, i_2) such as $i_1+i_2 \leqq S$ on the route, multiply by factor $\frac{\lambda_1}{i_1 \cdot v_{i_1}^1 + i_2 \cdot v_{i_2}^2}$, and multiply by factor $\frac{\lambda_2}{i_1 \cdot v_{i_1}^1 + i_2 \cdot v_{i_2}^2}$ for the vertical state transition. When the lattice point (i_1, i_2) such as $i_1+i_2 > S$, for the state transition outside of the lattice point (m_1, m_2) such as $m_1+m_2=S$ on the route (between (m_1, m_2) and (i_1, i_2)), multiply by factor $\frac{\lambda_1}{m_1 \cdot v_{m_1}^1 + m_2 \cdot v_{m_2}^2}$ or $\frac{\lambda_2}{m_1 \cdot v_{m_1}^1 + m_2 \cdot v_{m_2}^2}$.

Thus, the coefficient of $Q_S(n_1, n_2)$ related to $Q_S(0, 0)$ is represented as the summation of the multiplication based on all the routes from (0, 0) to (n_1, n_2). For example, for the route from (0, 0) to (1, 2) when S=3, which is the case of $n_1+n_2 \leqq S$, the multiplication along the route of broken line (i) in Figure 5 is $Q_S(0,0) \cdot \left(\frac{\lambda_2}{v_{01}^2} \right) \left(\frac{\lambda_2}{2 v_{02}^2} \right) \left(\frac{\lambda_1}{v_{12}^1 + 2 \cdot v_{12}^2} \right)$. For

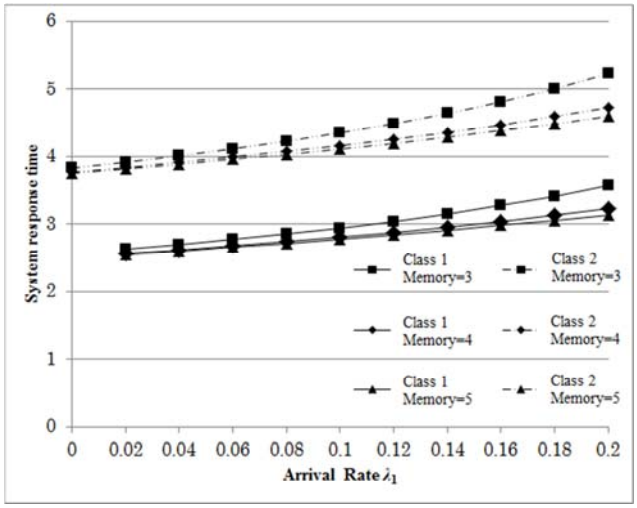


Fig. 6 Mean system response time (λ_2=0.2)

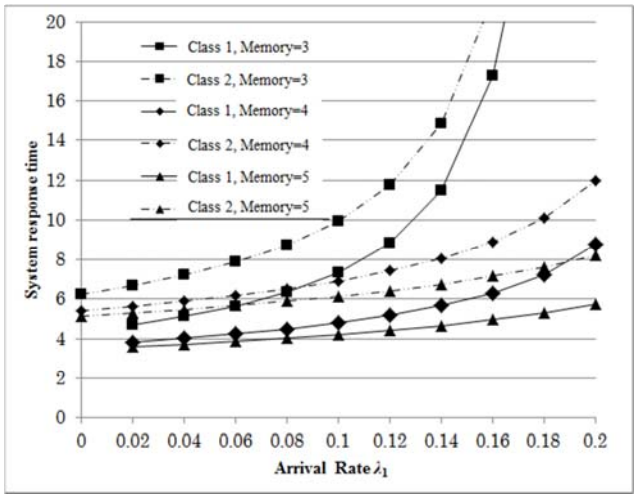


Fig. 7 Mean system response time (λ_2=0.4)

the route from (0, 0) to (4,2), which is the case of $n_1+n_2 > S$, the multi-plication along the (ii) is $Q_S(0,0) \cdot \left(\frac{\lambda_2}{v_{01}^2} \right) \times \left(\frac{\lambda_2}{v_{11}^1 + v_{11}^2} \right) \left(\frac{\lambda_1}{2 \cdot v_{21}^1 + v_{21}^2} \right) \left(\frac{\lambda_1}{2 \cdot v_{21}^1 + v_{21}^2} \right)^3$ Since there are multiple routes from (0, 0) to (n_1, n_2), the coefficient of $Q_S(n_1, n_2)$ related to $Q_S(0, 0)$ is approximately represented as the total of the multiplication based on all routes. Similarly to the case above, we can approximately calculate the state probability of a queuing network with multiple memory resource requirements when K >2.

4. Numerical Experiments

We evaluated the proposed approximation technique through numerical experiments. We used the following parameters.

1. Number of memory resources: S=3, 4, 5
2. Arrival rate: λ_1=0.0, 0.02, ... , 0.20, λ_2 =0.2, 0.4
3. Number of I/O nodes: M =2
4. Total service time at each node
   τ_{10}=1.0, τ_{11}=τ_{12}=0.5
   τ_{20}=1.0, τ_{21}=τ_{22}=1.0
   where τ_{km} is the total service time of job class k at node m.

Figures 6 and 7 show the mean system response times of job classes 1 and 2 as T_1, T_2 respectively, when λ_2 is fixed at 0.2 and 0.4, and λ_1 changes from 0.0 to 0.2. The mean system response time is the mean time from job arrival to departure from the system. Similarly to the case of a single job class, the mean system response times for both job classes increase monotonically in a convex curve. Moreover maximum arrival rate will exist and the system will overflow and the stable solution does not exist when the arrival rate exceeds the maximum arrival rate. As the arrival rate λ_1 is increased and λ_2 is fixed (the only workload of job class 1 is increased), not only the mean response time of job class 1 but also job class 2 increases. We can see that the mean response time of job class 2 is increasing higher than that of job class 1, and the reason for this is presumed to be that job class 2 has a longer I/O time than job class 1.

5. Conclusion

We proposed an approximation technique for evaluating the performance of computer systems with multiple memory resource requirements using a queuing network and analyzed its performance measures through numerical experiments. The concept of the approximation is based on separately analyzing the processing part and the memory part of the queuing network model.

The numerical experiments clarified the characteristics of the mean job response time.

In the future we plan to examine the accuracy of the proposed approximation technique by comparing it with exact solutions or simulation results.

REFERENCES

[1] F. Baskett, K. M. Chandy, R. R. Muntz and F. G. Palacious, "Open, Closed, and Mixed Networks of Queues with Different Classes of Customers," J. ACM, Vol.22, No.2, pp.248--260, April 1975.

[2] H. Kobayashi, "Modeling and Analysis," Addison-Wesley Publishing Company, Inc. 1978.

[3] T. Kurasugi and I. Kino, "Approximation Method for Two-layer Queueing Models," Performance Evaluation 36--37, pp.55--70, 1999.

[4] J. A. Rolia and K. C. Sevcik, "The Method of Layers," IEEE Trans. on Software Engineering, Vol.21, No.8, pp.689--700, Aug. 1995.

[5] T. Kinoshita and Y. Takahashi, "A Queuing Network Modeling and Performance Evaluation Method for Computer Systems with Resource Requirement," IEICE D-I, Vol. J 82-D-I, No.6, pp.701--710, Jun. 1999.

[6] T. Kinoshita and X. Gao, "Queuing Network Approximation Technique for Evaluating Performance of Computer Systems with Memory Resources," PDPTA2010, pp.640, July 2010

[7] A. Razali, T. Kinoshita and A. Tanabe, "Queuing Network Approximation Technique for Evaluating Performance of Computer Systems with Multiple Memory Resource Requirements," PDPTA2012, pp.758-763, July 2012

[8] O. E. Oguike, M. N. Agu and S. C. Echezona,"Modeling Variation of Waiting Time of Distributed Memory Heterogeneous Parallel Computer System Using Recursive Models," International Journal of Soft Computing and Engineering, vol. 2, Issue 6, Jan 2013

[9] A. Gandhi, S. Doroudi, M. Harchol-Balter and A. Scheller-Wolf, "Exact Analysis of the M/M/k/setup Class of Markov Chains via Recursive Renewal Reward," SIGMETRICS'13, pp.153-166, June 2013

Particle-Wave Unification - An Object-Oriented Approach to Equilibrium-Based Computing

Wen-Ran Zhang
Department of Computer Science, College of Engineering and IT
Georgia Southern University, Statesboro, GA, USA

Abstract – ***E****quilibrium-based computing is introduced and distinguished from truth-based computing. It is shown that object-oriented languages can be used for both equilibrium-based and truth-based programming. This observation supports the claim that any physical being must exist in certain dynamic equilibrium. The applicability of this approach is discussed and illustrated with Java and C++ code. It is shown that the equilibrium-based computing paradigm is applicable in both digital and quantum computing for particle-wave unification of matter and antimatter atoms.*

Keywords: Bipolarity; Equilibrium-Based Programming; Object Orientation; Particle-Wave Unification

1 Introduction

Traditionally, computer languages are classified into four different programming paradigms including imperative paradigm, functional paradigm, object oriented paradigm, and logical paradigm. These paradigms are all based on being and truth. Therefore, we classify them into truth-based programming paradigms. Scientific computing has been being-centered and truth-based.

It can be argued that truth-based programming is inadequate in modeling equilibrium and non-equilibrium conditions of beings. Based on this argument an equilibrium-based programming paradigm is introduced in this work. Philosophically, the new paradigm claims that any physical being must exist in certain dynamic equilibrium where bipolar dynamic equilibrium is the most fundamental form of any multidimensional equilibrium or non-equilibrium.

This work is organized into five sections: Introduction, Mathematical Foundation, Equilibrium-Based Programming with Object-Orientation, Application, and Conclusion.

2 Mathematical Foundation

2.1 Bipolar Sets and Bipolar Dynamic Logic

Truth-based mathematical abstraction follows Aristotle's "being qua being" metaphysics that asserts truth as the essence of being. The principle claims that the concept of an element in a set is self-evident without the need for proof of any kind and the properties of the set are independent of the nature of its elements. Classical logic is based on this principle. However, the principle may fail in the natural, biological and social worlds. For instance, the identity law A=A may not be able to hold in the quantum world due to quantum entanglement. And the independence of a set from the nature of its elements excludes any possibility of a formal definition for the ultimate being of all beings. This has led to nihilism and the indefinability of causality.

While classical set theory is based on truth and singularity, bipolar set theory is based on dynamic equilibrium and bipolarity [Zhang 1998, Zhang 2011]. The equilibrium-based approach follows the ancient Chinese YinYang cosmology and asserts bipolar dynamic equilibrium (including equilibrium, quasi-equilibrium, and non-equilibrium states) as the essence of being. Thus, bipolar sets present a major challenge to the principle of truth-based mathematical abstraction.

Ontologically, YinYang bipolarity is observable. While ether, monad, monopoles and strings are imaginable but so far not testable, dipoles are everywhere in the universe; particle-antiparticle pairs (-q,+q) and action-reaction (-f,+f) are believed the most fundamental elements of the universe; negative and positive energies form the regulating force of multiple universes [Hawking & Mlodinow 2010 p179-180]; competition and cooperation exist in any biological society; the Yin Yang 1 (YY1) genomic regulator protein with bipolar repression-activation functions is found ubiquitous in the cells of all living species [Shi, *et al.* 1991; Ai, Narahari & Roman, 2000; Palko *et al.* 2004**;** Zhou & Yik 2006; Wilkinson, *et al.* 2006; Kim, Faulk & Kim, 2007; Santiago, *et al.* 2007; Liu, *et al.* 2007; Vasudevan, Tong & Steitz 2007]; self-negation and self-assertion bipolar emotional equilibrium or disorder is a psychiatric reality [Zhang, Pandurangi and Peace 2007; Zhang et al 2011]; it is becoming scientifically evident that brain *bioelectromagnetic field is crucial for neurodynamics and different mental states* [Carey 2007] *where bipolarity is unavoidable.* In one word, any being or agent has to exist in certain dynamic equilibrium and bipolar dynamic equilibrium is shown to be the most basic type of equilibrium.

YinYang bipolar set theory leads to bipolar dynamic logic (BDL) which presents an equilibrium-based approach to mathematical abstraction [Zhang 1998a; Zhang & Zhang 2004a; Zhang 2011]. In bipolar sets the elements are bipolar agents such as dipoles, particle-antiparticle pairs, nature's action-reaction objects, genomic repression-activation capacities, social competition-cooperation relations, input-output of any system, self-negation and self-assertion abilities

in mental health, in general, the negative and positive energies of nature (Fig. 1). This ontological claim positioned BDL in the context of logically definable causality for ubiquitous quantum computing and quantum intelligence.

BDL is defined on $B_1 = \{-1,0\} \times \{0,+1\} = \{(0,0), (0,+1), (-1,0), (-1,+1)\}$ – a bipolar quantum lattice in the YinYang bipolar geometry as shown in Fig. 2. The new geometry is background independent [Smolin 2005]. The background independent property makes quadrant irrelevant. The four values of B_1 form a bipolar causal set which stand, respectively, for eternal equilibrium (0,0), non-equilibrium (-1,0), non-equilibrium (0,+1); equilibrium (-1,+1). Evidently, each bipolar element can be used to code two bits of binary information (or one bit with a $\vee$ or $\wedge$ operation on the two poles in absolute values). Fig. 3 illustrates bipolar interaction and entanglement.

Equations (1)-(12) in Table 1 provide the basic operations of BDL. The laws in Table 2 hold on BDL. Bipolar universal modus ponens (BUMP) is listed in Table 3 which logically defines equilibrium-based bipolar causality.

An equilibrium-based axiomatization is shown in Table 4 which has been proven sound [Zhang 2011 Ch.3]. In BDL $\oplus$ and $\oplus^-$ are "balancers" that can, at the most basic level, be used as nuclear fusion operators; $\varnothing$, $\otimes$, $\varnothing^-$ and $\otimes^-$ are intuitive and counter-intuitive "oscillators" that leads to particle wave unification; & and $\&^-$ are "minimizers" that can, at the most basic level, be used as particle-antiparticle annihilation operators. The linear, cross-pole, bipolar fusion, fission, oscillation, interaction and entanglement properties are depicted in Figure 3.

The propositional BDL has been extended to a 1st order formal system [Zhang 2011 Ch. 3] in which equilibrium-based bipolar predicates can be used similarly as truth-based predicates. For instance, given bipolar agent A and let the bipolar functor (f^-,f^+) be self-negation and self-assertion abilities, $(f^-,f^+)(A)$ can denote the mental equilibrium or non-equilibrium of A; given bipolar agents A and B, and let the bipolar functor (r^-,r^+) be competition and cooperation relations, $(r^-,r^+)(A,B)$ can denote the relation between A and B.

Thus, BDL presents a causal logic for both digital and quantum computing. While the causal set quest for quantum gravity stopped short of going beyond classical truth-based set theory to reach logically definable quantum causality, bipolar sets and BDL as a formal bipolar equilibrium-based system presents a major step toward logically definable causality.

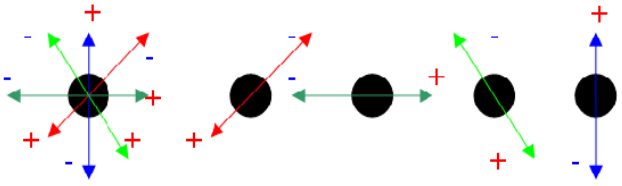


Figure 1. Multidimensional equilibrium or non-equilibrium deconstructed to bipolar equilibria/non-equilibria

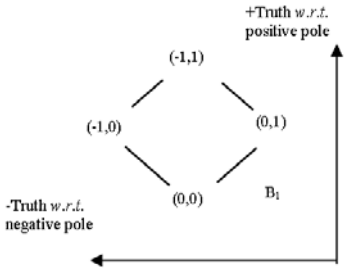


Figure 2. Hasse diagram of B_1 in bipolar geometry

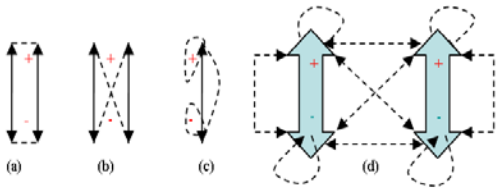


Figure 3 Bipolar relativity: (a) Linear interaction; (b) Cross-pole non-linear interaction; (d) Oscillation; (e) Bipolar entanglement

Table 1. YinYang Bipolar Dynamic Logic (BDL)
(Note: The use of |x| in this paper is for explicit bipolarity only)

Bipolar Partial Ordering: $(x,y)\geq\geq(u,v)$, iff $\lvert x\rvert\geq\lvert u\rvert$ and $y\geq v$;	(1)
Complement: $\neg(x,y)\equiv(-1,1)-(x,y)\equiv(\neg x,\neg y)\equiv(-1-x,1-y)$;	(2)
Implication: $(x,y)\Rightarrow(u,v)\equiv(x\rightarrow u,y\rightarrow v)\equiv(\neg x\vee u),\neg y\vee v)$;	(3)
Negation: $-(x,y) \equiv (-y,-x)$;	(4)
Bipolar least upper bound (blub):	
$blub((x,y),(u,v))\equiv(x,y)\oplus(u,v)\equiv(-(\lvert x\rvert\vee\lvert u\rvert),y\vee v)$;	(5)
-blub: $blub^-((x,y),(u,v))\equiv (x,y)\oplus^- (u,v) \equiv (-(y\vee v),(\lvert x\rvert\vee\lvert u\rvert))$;	(6)
Bipolar greatest lower bound (bglb):	
$bglb((x,y),(u,v)) \equiv (x,y)\&(u,v) \equiv (-(\lvert x\rvert\wedge\lvert u\rvert),y\wedge v))$;	(7)
-bglb: $bglb^-((x,y),(u,v)) \equiv (x,y)\&^-(u,v) \equiv (-(y\wedge v),(\lvert x\rvert\wedge\lvert u\rvert)))$;	(8)
Cross-pole greatest lower bound (cglb):	
$cglb((x,y),(u,v))\equiv(x,y)\otimes(u,v)\equiv(-(\lvert x\rvert\wedge\lvert v\rvert\vee\lvert y\rvert\wedge\lvert u\rvert),(\lvert x\rvert\wedge\lvert u\rvert\vee\lvert y\rvert\wedge\lvert v\rvert))$;	(9)
-cglb: $cglb^-((x,y),(u,v)) \equiv (x,y)\otimes^-(u,v) \equiv -((x,y)\otimes(u,v))$;	(11)
Cross-pole least upper bound (cglb):	
$club((x,y),(u,v))\equiv(x,y)\varnothing(u,v)\equiv(-1,1)-(\neg(x,y)\otimes\neg(u,v))$;	(10)
-club: $club^-((x,y),(u,v))\equiv(x,y)\varnothing^-(u,v)\equiv -((x,y)\varnothing(u,v))$.	(12)

Table 2. Laws of bipolar equilibrium/non-equilibrium

Excluded Middle	$(x,y)\oplus\neg(x,y) \equiv (-1,1)$; $(x,y)\oplus^-\neg(x,y) \equiv (-1,1)$;
No contradiction	$\neg((x,y)\&\neg(x,y))\equiv(-1,1)$; $\neg((x,y)\&^-\neg(x,y))\equiv(-1,1)$;
Linear Bipolar DeMorgan's Laws	$\neg((a,b)\&(c,d)) \equiv \neg(a,b)\oplus\neg(c,d)$; $\neg((a,b)\oplus(c,d)) \equiv \neg(a,b)\&\neg(c,d)$; $\neg((a,b)\&^- (c,d)) \equiv \neg(a,b)\oplus^-\neg(c,d)$; $\neg((a,b)\oplus^- (c,d)) \equiv \neg(a,b)\&^-\neg(c,d)$;
Non-Linear Bipolar DeMorgan's Laws	$\neg((a,b)\otimes (c,d)) \equiv \neg(a,b)\ \varnothing\ \neg(c,d)$; $\neg((a,b)\ \varnothing\ (c,d)) \equiv \neg(a,b)\otimes\neg(c,d)$; $\neg((a,b)\otimes^-(c,d)) \equiv \neg(a,b)\ \varnothing^-\ \neg(c,d)$; $\neg((a,b)\varnothing^- (c,d)) \equiv \neg(a,b)\otimes^-\neg(c,d)$

Table 3. Bipolar Universal Modus Ponens (BUMP)

$\forall\phi=(\phi^-,\phi^+)$, $\varphi=(\varphi^-,\varphi^+)$, $\psi=(\psi^-,\psi^+)$, and $\chi=(\chi^-,\chi^+) \in B_1$,

$$[(\phi \Rightarrow \varphi)\ \&(\psi\Rightarrow\chi)] \Rightarrow [(\phi*\psi) \Rightarrow (\varphi*\chi)];$$
$$OR\ (\phi \Leftrightarrow \varphi) \Rightarrow (\phi*\psi \Leftrightarrow \varphi*\psi)$$

Two-fold universal instantiation:

1) Operator instantiation: $*$ as a universal operator can be bound to &, $\oplus$, $\&^-$, $\oplus^-$, $\otimes$, $\varnothing$, $\otimes^-$, $\varnothing^-$. $(\phi \Rightarrow \varphi)$ is designated (bipolar true (-1,+1)); $((\phi^-,\phi^+)*(\psi^-,\psi^+))$ is undesignated.
2) Variable instantiation:
$\forall x, (\phi^-,\phi^+)(x) \Rightarrow (\varphi^-,\varphi^+)(x)$; $(\phi^-,\phi^+)(A)$; $\therefore (\varphi^-,\varphi^+)(A)$.

Table 4. From Truth-Based to Equilibrium-Based Axiomatization

Unipolar Axioms (UAs): UA1: $\phi\rightarrow(\varphi\rightarrow\phi)$; UA2: $(\phi\rightarrow(\varphi\rightarrow\chi))$ $\rightarrow((\phi\rightarrow\varphi)\rightarrow(\phi\rightarrow\chi))$; UA3: $\neg\phi\rightarrow\varphi)\rightarrow((\neg\phi\rightarrow\neg\varphi)\rightarrow\phi)$; UA4: (a) $\phi\wedge\varphi\rightarrow\phi$; (b) $\phi\wedge\varphi\rightarrow\varphi$; UA5: $\phi\rightarrow(\varphi\rightarrow\phi\wedge\varphi)$;	Bipolar Linear Axioms: BA1: $(\phi^-,\phi^+)\Rightarrow((\varphi^-,\varphi^+)\Rightarrow(\phi^-,\phi^+))$; BA2: $((\phi^-,\phi^+)\Rightarrow((\varphi^-,\varphi^+)\Rightarrow(\chi^-,\chi^+)))$ $\Rightarrow(((\phi^-,\phi^+)\Rightarrow(\varphi^-,\varphi^+))\Rightarrow((\phi^-,\phi^+)\Rightarrow(\chi^-,\chi^+)))$; BA3: $(\neg(\phi^-,\phi^+)\Rightarrow(\varphi^-,\varphi^+))$ $\Rightarrow((\neg(\phi^-,\phi^+)\Rightarrow\neg(\varphi^-,\varphi^+))\Rightarrow(\phi^-,\phi^+))$; BA4: (a) $(\phi^-,\phi^+)\&(\varphi^-,\varphi^+)\Rightarrow(\phi^-,\phi^+)$; (b) $(\phi^-,\phi^+)\&(\varphi^-,\varphi^+)\Rightarrow(\varphi^-,\varphi^+)$; BA5: $(\phi^-,\phi^+)\Rightarrow((\varphi^-,\varphi^+)\Rightarrow((\phi^-,\phi^+)\&(\varphi^-,\varphi^+)))$;
Inference Rule – Modus Ponens (MP): UR1: $(\phi\wedge(\phi\rightarrow\varphi))\rightarrow\varphi$.	Non-Linear Bipolar Universal Modus Ponens (BUMP) ($*$ can be bound to any bipolar operator in Table 1) BR1: IF $((\phi^-,\phi^+)*(\psi^-,\psi^+))$, $[((\phi^-,\phi^+)\Rightarrow(\varphi^-,\varphi^+))\&((\psi^-,\psi^+)\Rightarrow(\chi^-,\chi^+))]$, THEN $[(\varphi^-,\varphi^+)*(\chi^-,\chi^+)]$;
Predicate axioms and rules UA6: $\forall x,\phi(x)\rightarrow\phi(t)$; UA7: $\forall x,(\phi\rightarrow\varphi)\rightarrow(\phi\rightarrow\forall x,\varphi)$; UR2–Generalization: $\phi\rightarrow\forall x,\phi(x)$	Bipolar Predicate axioms and Rules of inference BA6: $\forall x,(\phi^-(x),\phi^+(x))\Rightarrow(\phi^-(t),\phi^+(t))$; BA7: $\forall x,((\phi^-,\phi^+)\Rightarrow(\varphi^-,\varphi^+))\Rightarrow((\phi^-,\phi^+)\Rightarrow\forall x,(\varphi^-,\varphi^+)$; BR2-Generalization: $(\phi^-,\phi^+)\Rightarrow\forall x,(\phi^-(x),\phi^+(x))$

2.2 Bipolar Relations and Equilibrium Relations

As a causal set, a bipolar relation is characterized with bipolar values such as (0,0) for no relation, (-1,0) for conflict relation, (0,+1) for coalition, and (-1,+1) for harmonic relation, respectively. The *bipolar transitive closure* of a bipolar relation R is the smallest transitive bipolar relation containing R [Zhang 2003a; Zhang 2011, Ch. 3], denoted by $\Re$ and

$$\Re = R^1 \oplus R^2 \oplus R^3 \oplus .... \quad (13)$$

It is found that, let $X=\{x_1,x_2,...,x_n\}$ be a finite bipolar set, the $\oplus$–$\otimes$ bipolar transitive closure (denoted $\Re$) of R in X exists, is unique, and

$$\Re = R^1 \oplus R^2 \oplus R^3 \oplus ... \oplus R^{2n}. \quad (14)$$

Bipolar transitive closure is a causal structure. Bipolar reflexivity, symmetry and transitivity lead to the generalizations of equivalence relations to bipolar equilibrium relations [Zhang 2003a; Zhang 2011 Ch. 3] and fuzzy similarity relations to bipolar fuzzy or quasi-equilibrium relations [Zhang 2006; Zhang 2011 Ch. 5]. Based Eq. (14), algorithms have been devised for bipolar clustering from equilibrium relations. While an equivalence relation induces partitions of equivalence sets; an equilibrium relation induces partitions of coalition sets, conflict sets, and harmonic sets [Zhang 2006; Zhang 2011 Ch. 5]. Thus, the partitions from an expected equilibrium state could be used as predictions for decision support [Zhang 2003a,b].

2.3 Bipolar Quantum Linear Algebra

The bipolar lattice $B_1=\{-1,0\}\times\{0,1\}$ and the bipolar fuzzy lattice $B_F = [-1,0] \times [0,1]$ can be naturally extended to the real valued bipolar lattice $B_\infty = [-\infty, 0] \times [0, +\infty]$. B_1 and B_F are bounded and complemented unit square lattices, respectively; B_∞ is unbounded. $\forall(x,y),(u,v)\in B_\infty$, Eqs. 15-16 define two algebraic operations.

Bipolar Multiplication: $(x,y) \times (u,v) \equiv (xv+yu, xu+yv);$ (15)

Bipolar Addition: $(x,y) + (u,v) \equiv (x+u, y+v).$ (16)

In Eq. (15), $\times$ is a cross-pole multiplication operator with the infused non-linear bipolar causal semantics --=+, -+=+-=1, and ++=+; + in Eq. (16) is a linear bipolar addition or fusion operator. With the two basic operations, classical linear algebra is naturally extended to an equilibrium-based *causal algebra* named *bipolar quantum linear algebra* (BQLA) with bipolar fusion, fission, diffusion, interaction, oscillation, annihilation, and quantum entanglement properties [Zhang et al 2009; Zhang 2011 Ch. 7; Zhang 2012a]. These properties enable physical or biological agents to interact through bipolar quantum fields such as atom-atom, cell-cell, heart-heart, heart-brain, brain-brain, organ-organ, and genome-genome bio-electromagnetic quantum fields as well as biochemical pathways in energy equilibrium or non-equilibrium. These properties lead to the inception of YinYang bipolar atom [Zhang 2012a], bipolar quantum logic gates and quantum cellular combinatorics [Zhang 2013]. Quantum cellular combinatorics provides a modular graph theory for multidimensional bipolar cause-effect modeling (Fig. 1) of YinYang-N-element cellular automata [Zhang 2011 Ch. 8; Zhang 2012a].

3 An Object-Oriented Approach to Equilibrium-Based Programming

Logically, bipolar dynamic logic (BDL) presents an equilibrium-based non-linear dynamic generalization of Boolean logic from the truth-based domain or bivalent lattice {0,1} to the equilibrium-based domain or bipolar lattice B_1 = {(0,0), (-1,0), (0,+1), (-1,+1)}. This generalization provides a logical basis for equilibrium-based dynamic programming (EDP). Now our question is whether EDP can be realized with object-orientated languages such as in C++ and Java. Interestingly, the answer is positive.

3.1 C++ Class for Bipolar Variable

Bipolar variable, the basic concept of BDL and BQLA, is defined in the following C++ class with object-orientation:

```
class nppair {  // specify bipolar variable and its operations
  float lower_weight; // negative pole
  float upper_weight; // positive pole
public:
  // constructors
  nppair() { }
  nppair(float l,float u)
    {lower_weight = l; upper_weight = u; }
  void update(float l,float u)
    {lower_weight = l; upper_weight = u; }
  // getter and setters
  float& lower() { return lower_weight; }
  float& upper() { return upper_weight; }
  float lower() const { return lower_weight; }
  float upper() const { return upper_weight; }
  float& left() { return lower_weight; }
  float& right() { return upper_weight; }
  float leftt() const { return lower_weight; }
  float rightt() const { return upper_weight; }
  void  new_lower(float x) { lower_weight=x; }
  void  new_upper(float y) { upper_weight=y; }
  // operators
  void operator =(nppair& p) {
```

```
    lower_weight = p.lower_weight;
    upper_weight = p.upper_weight; }
  void operator *=(float d);
  void operator /=(float d);
  void operator *=(nppair& p);
  void operator +=(nppair& p);
  void operator -=(nppair& p);
  friend nppair operator *(nppair& p1,nppair& p2);
  friend nppair operator *(nppair& p1,float d);
  friend nppair operator /(nppair& p1,float d);
  friend nppair operator +(nppair& p1,nppair& p2);
  friend nppair operator |(nppair& p1,nppair& p2);
  friend npinterval operator ||(nppair& p1,nppair& p2);
  friend istream& operator>> (istream& ci, nppair& c){
    ci >> c.lower_weight;
    ci >> c.upper_weight;
    return ci; }
  friend ostream& operator<< (ostream& co, const nppair& c) {
      co << '(' << c.lower_weight << ' ' << c.upper_weight << ')';
          return co; }
  friend int contain1(const nppair& p1,float d);
  friend int contain2(const nppair& p1,const nppair& p2);
  friend npinterval;
};
```

3.2 C++ Class for Bipolar Vector or Matrix

Based on the class of nppair, a bipolar vector or matrix and its operations can be defined.

```
class npmatrix {  // bipolar matrix
  nppair* m;    // pointer to matrix in 1-d storage
  int rows;     // number of rows
    int cols;        // number of col
  float* rowEnergy; // pointer to row bipolar energy
  float* colEnergy;  // pointer to col bipolar energy
public:
  // constructor
       npmatrix() {}
  npmatrix(int r,int c){
  rows=r; cols=c; m = new nppair[r*c];
  rowEnergy = new float[rows];
  colEnergy = new float[cols];}
  npmatrix(int r,int c,nppair* m1) { rows=r; cols=c; m = m1; }
  // member operators and functions
  void clear();
  nppair& operator ()(int x) { return m[x]; } // 1-d getter
  nppair& operator ()(int i,int j) { return m[i*cols+j]; } //2-d getter
  float negativeEnergy();
  float positiveEnergy();
  float totalEnergy();
  float localImbalance();
  float globalImbalance();
  float localStability();
  float globalStability();
  nppair harmonyLevel();
  void operator *=(float d);  // multiply by d
  void operator /=(float d);   // divid by d
  void operator +=(npmatrix& m); // matrix addition
  void operator -=(npmatrix& m); //matrix subtraction
  void closure(int Tnorm);    // bipolar transitive closure with Tnorm
  friend npmatrix operator *(npmatrix& m1,float d);
  friend npmatrix operator /(npmatrix& m1,float d);
  friend npmatrix operator +(npmatrix& m1,npmatrix& m2);
  friend npmatrix operator -(npmatrix& m1,npmatrix& m2);
  friend npmatrix operator *(npmatrix& m1,npmatrix& m2);
  friend istream& operator>> (istream& ci, npmatrix& m);
  friend ostream& operator<< (ostream& co, const npmatrix& m);
  friend istream& inner_outer_i(istream& ci, npmatrix& m1, npmatrix&
m2);
  friend ostream& inner_outer_o(ostream& co, const npmatrix& m1, const
npmatrix& m2);
  friend istream& inner_outer_i(istream& ci, npmatrix& m1, npmatrix& m2,
npmatrix& m3, npmatrix& m4);
  friend ostream& inner_outer_o(ostream& co, const npmatrix& m1, const
npmatrix& m2, const npmatrix& m3, const npmatrix& m4);
  friend ostream& linguistic(ostream& co, const npmatrix& m1, const
npmatrix& m2);
  friend ostream& linguistic(ostream& co, const npmatrix& m1, const
npmatrix& m2, const npmatrix& m3, const npmatrix& m4);
  void randomize();   // assign random bipolar weights
  void normalizeRow();  // normalize row energy
  void normalizeCol(); // normalize row energy
  int normalized(); // check normalization
  void row_Energy(); // row energy
  void col_Energy(); // col energy
};
```

3.3 Equilibrium-Based but Object-Oriented

The C++ program examples for bipolar variables and vectors show that a bipolar equilibrium can be coded as an object class and a multidimensional equilibrium can be coded as a set of bipolar objects. Therefore, object-oriented languages can be used for equilibrium-based programming.

4 Applications

4.1 Bipolar Complementarity

Niels Bohr - a father figure of quantum mechanics - was the first to bring YinYang into quantum theory for his particle-wave complementarity or duality principle. When Bohr was awarded the Order of the Elephant by the Danish government, he designed his own coat of arms which featured in the center a YinYang logo (or Taijit symbol) and the Latin motto "contraria sunt complementa" or "opposites are complementary" (Fig. 4).

While Bohr's quantum mechanics recognized particle-wave complementarity, it stopped short of identifying the essence of YinYang bipolar coexistence. It is argued that without bipolarity any complementarity is less fundamental due to the missing "opposites" (Fig. 5) [Zhang 2011; Zhang 2013]. If bipolar equilibrium is the most fundamental form of equilibrium, any multidimensional model of spacetime such as string theory and superstring theory cannot be most fundamental.

In brief, action-reaction, particle-antiparticle, negative-positive energies, input and output, or the Yin and Yang of nature in general could be the most fundamental opposites, but man and woman, space and time, particle and wave, truth and falsity are not exactly bipolar opposites. This could be the reason why Bohr found causal description of a quantum process unattainable and we have to content ourselves with particle-wave complementary descriptions [Bohr 1948]. Since then, particle and wave as a YinYang duality has not reached unification. Now, equilibrium-based computing provides a basis for such a unification.

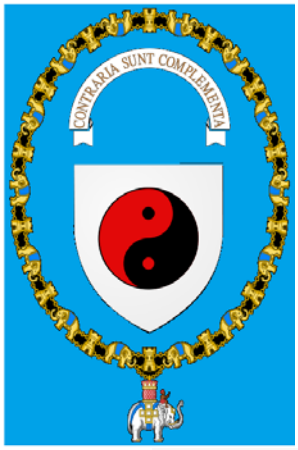


Figure 4. Bohr's Coat of Arms (Creative Commons file by GJo, 3/8/2010, Source: File:Royal Coat of Arms of Denmark.svg (Collar of the Order of the Elephant) + File:Yin yang.svg)

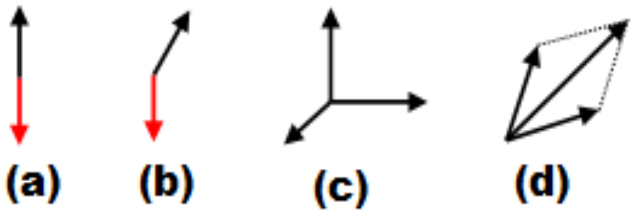


Figure 5. Fundamental and non-fundamental complementarities: (a) Fundamental; (b)-(d) Non-fundamental

4.2 Bipolar Quantum Logic Gates

Bipolar Energy Conservational Quantum Logic Gate. If the energy of every row and every column of a bipolar decimal matrix M of a quantum agent in Eq. (24) always adds up to 1.0, we call M a bipolar energy conservational quantum logic gate matrix.

Law of Bipolar Equilibrium or Symmetry: With Eq. (24), if M is bipolar energy conservational, any quantum agent's energy vector $E(t_1)=M\times E(t_0)$ at t_0 and t_1 can be characterized as satisfies Eq. (17).

$$|\varepsilon|\,(E(t_1))= |\varepsilon|\,(M \times E(t_0)) \equiv |\varepsilon|\,(E(t_0)) \qquad (17)$$

Evidently, any integer unitary quantum logic gate matrix must be energy conservational. Therefore, a unitary quantum logic gate in quantum computing can be deemed part of energy conservation and the concepts of equilibrium, symmetry, unitarity and reversibility in quantum computing are generalized to the equilibrium or non-equilibrium condition of any agent. This generalization illustrates the quantum nature of all agents in multidimensional bipolar equilibrium or non-equilibrium. [Zhang 2013]

4.3 Equilibrium-Based Algorithm

Two equilibrium-based algorithms are shown in C++ language in Table 5 for quantum cellular automata using bipolar quantum logic gates.

Table 5. Two C++ algorithms for testing energy equilibrium [Zhang *et al.* 2009]

```
//Algorithm A: Normalization of the Random Connectivity Matrix M
// (1) normalize the energy of each row and column of M to |ε|=1.0 as an equilibrium condition;
// (2) normalize the energy to |ε|>1.0 as a non- equilibrium condition for energy increase;
// (3) normalize the energy to |ε|<1.0 as a non- equilibrium condition for energy decrease;
//-------------------------------------------------------------------------
YinYangMatrix M(N,N); // create an N×N bipolar connectivity matrix
M.randomize();        // assign random weights to the elements of M
M.normalizeRows();  // normalize each row |ε|(M_k*)
M.normalizeCols().   // normalize each column  |ε|(M_*j)
//-------------------------------------------------------------------------
//Algorithm B: Test the three conditions :
//(1) ∀t , Y(t+1) = M(t)Y(t) – equilibrium
//(2) ∀t , Y(t+1) > M(t)Y(t) – energy increase
//(3) ∀t , Y(t+1) < M(t)Y(t) – energy decrease
//-------------------------------------------------------------------------
YinYangMatrix M(N,N); // create an N×N bipolar connectivity matrix
YinYangMatrix Yt0(1,N); // create column vector at t0
YinYangMatrix Yt1(1,N); // create column vector at t1
M.randomize();        // assign random link weights to M
M.normalizeRows();  // normalize each row |ε|(M_k*)
M.normalizeCols().   // normalize each column  |ε|(M_*j)
file1 >> Yt0;         // input col vector from file1
int times;
cin >> times;         // enter number of iteration
for (int i = 0; i<times;  i++){
Yt1 = M*Yt0;          // M multiply column vector
file2 << Yt1;          // output result vector to file2
file2 << Yt1.totalEnergy() << "\n";   // output energy to file2
Yt0 = Yt1;            // reassign for next iteration
}
```

4.4 Particle-Wave Unification

The object-oriented approach to equilibrium-based programming can be used to demonstrate particle-wave unification of both matter and antimatter atoms. At the most fundamental level we have (-1,0) ⊗ (-1,0) = $(-1,0)^2$ = (0,1) and (-1,1) ⊗ (-1,1) = $(-1,1)^2$ = (-1,1). $(-1,0)^n$ defines a bipolar oscillation. Such property provides a unifying logical representation for particle-wave duality of both matter and antimatter particles.

Generic Bipolar Agent: (1) $\phi(P)(f) = (-1,0)^n(3\times10^{12})$ can denote the fact "Particle P changes polarity three trillion times per second from particle to antiparticle or vice versa." P is a subatomic particle named B-sub-s meson discovered at the Fermi National Accelerator Laboratory [Fermi National Accelerator Laboratory, 2006]. Fig. 6a shows the graphical representation of P as a YinYang-1-Element with a negative reflexivity. (2) φ(A) = (-1,+1) ⊗ (-1,0) = (-1,+1) ⊗ (-1,0) = (-1,+1) ⊗ (-1,+1) can denote the fact "Agent A is a has strong mental equilibrium of self-negation and self-assertion abilities who can bear with negative event, positive event as well as harmonic event." Fig. 6b shows the graphical representation of the strong equilibrium.

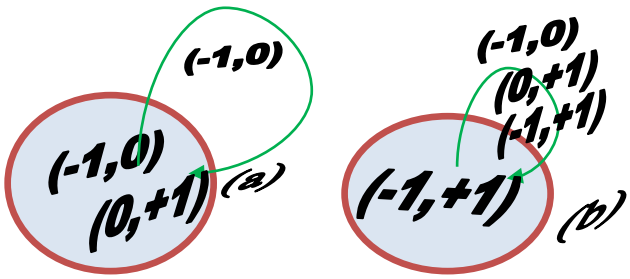


Fig. 6. (a) Oscillating particle-antiparticle; (b) Agent with strong mental equilibrium

Composite Bipolar Agents: Fig. 7 shows a YinYang-N-Element cellular automaton where each element is bipolar such as electron-positron as in matter atom or antimatter atom. In either case the bipolar energy can be characterized as $E(t_1)=M\times E(t_0)$ in equilibrium or non-equilibrium. Dramatically, each bipolar wave form is actually a bipolar element (object) in a collection of different bipolar states in an ordered sequence as shown in Figs. 7, 8 and 9. Thus, an N-electron matter atom or N-positron antimatter atom can be represented as the superposition of N such wave forms. The particle-wave forms are generated with Java programs in object-orientation. Thus particle-wave unification is realized.

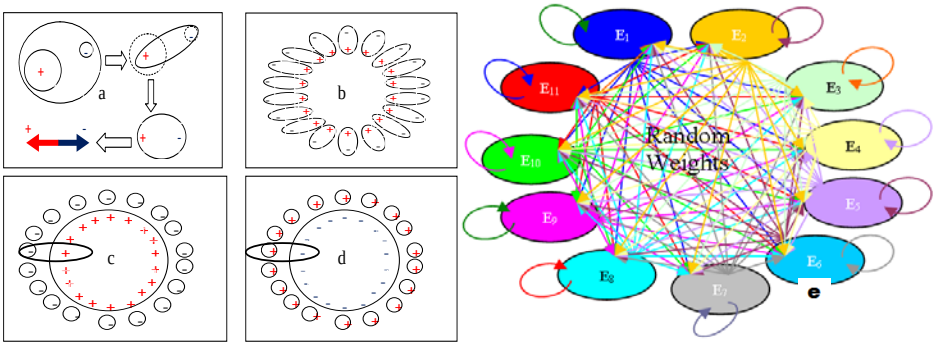


Figure 7. Particle-wave unification as a bipolar cellular automaton: (a) bipolar representation of a hydrogen; (b) bipolar representation of YinYang-n-elements; (c) Matter atom; (d) Antimatter atom; (e) Bipolar Quantum Cellular Automaton

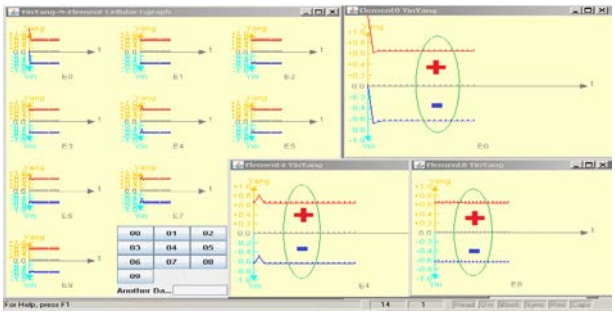


Fig. 8. Bipolar energy rebalancing to equilibrium after a disturbance to one element

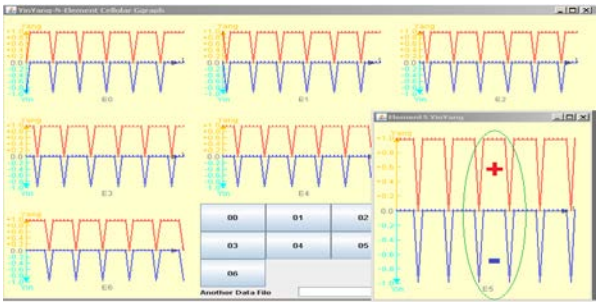


Fig. 9. Elementary bipolar energy oscillation under global equilibrium

5 Conclusions

Equilibrium-based programming has been introduced and distinguished from truth-based programming. It has been shown that object-oriented languages can be used for both equilibrium-based and truth-based programming. This observation supports the claim that any physical being or object must exist in certain dynamic equilibrium.

The applicability of equilibrium-based programming has been discussed and illustrated in scientific computing with C++ code and particle-waveforms from Java code to illustrate matter-antimatter and particle-wave unification. It is further expected that, based on BDL and BQLA, imperative, functional, object-oriented and logical programming paradigms can all be extended from truth-based to equilibrium-based programming paradigms.

From a different perspective, bipolar dynamic equilibrium as holistic truth does not exclude but generalizes truth from the bivalent domain or lattice {0,1} to the bipolar domain or lattice {(-1,0) × (0,+1)}. Since the universe (or multiple universes) can be deemed a dynamic equilibrium of negative-positive energies, particle-antiparticles, and action-reaction forces, the equilibrium-based programming paradigm is expected to bridge a gap between digital computing and quantum computing for programming the universe [Lloyd 2006] with applications in physical, social, biological, and mental worlds [Ball 2011][Zhang and Zhang et al 1989-2013].

References

[1] Ai, W., Narahari, J. & Roman A. (2000). Yin Yang 1 Negatively Regulates the Differentiation-Specific E1 Promoter of Human Papillomavirus Type 6. *J. of Virology, vol. 74, no. 11*, 5198-5205.

[2] Ball, P. (2011), Physics of Life : The Dawn of Quantum Biology. Published online 15 June 2011 | *Nature* 474, 272-274 (2011) | doi:10.1038/474272a

[3] Bohr, N. (1948), On The Notions of Causality and Complementarity. *Dialectica*, Vol. 2, Issue 3-4, 312–319, 1948.

[4] Carey, B. (2007). Man regains speech after brain stimulation. *The New York Times – Health*, Aug. 1, 2007. http://www.nytimes.com/2007/08/01/health/01cnd-brain.html.

[5] Fermi National Accelerator Laboratory (2006). *Press Release 06-19*, September 25, 2006. http://www.fnal.gov/pub/presspass/press_releases/CDF_meson.html

[6] Hawking, S. and Mlodinow, L. (2010), The Grand Design. Random House Digital, Inc., New York, 2010.

[7] Kim, J. D., Faulk, C. & Kim J. (2007). Retroposition and evolution of the DNA-binding motifs of YY1, YY2 and REX1. *Nucleic Acids Research, vol. 35, no. 10,* 2007, 3442-52.

[8] Liu, H., Schmidt-Supprian, M., Shi, Y., Hobeika, E., Barteneva, N., Jumaa, H., Pelanda, R., Reth, M., Skok, J., Rajewsky, K. & Shi, Y (2007). Yin Yang 1 is a critical regulator of B-cell development. *Genes Dev.* 21:1179-1189 (2007).

[9] Lloyd, S. (2006), *Programming the Universe*. Alfred A. Knopf, Inc. New York, 2006.

[10] Palko, L., Bass, H. W., Beyrouthy, M. J., & Hurt, M. M. (2004). The Yin Yang-1 (YY1) protein undergoes a DNA-replication-associated switch in localization from the cytoplasm to the nucleus at the onset of S phase. J of Cell Science**,** 117, 465-476.

[11] Santiago, F. S., Ishii, H., Shafi, S., Khurana, R., Kanellakis, P., Bhindi, R., Ramirez, M. J., Bobik, A., Martin, J. F., Chesterman, C. N., Zachary, I. C. & Khachigian, L. M. (2007). Yin Yang-1 Inhibits Vascular Smooth Muscle Cell Growth and Intimal, Thickening by Repressing p21WAF1/Cip1 Transcription and p21WAF1/Cip1-Cdk4-Cyclin D1 Assembly. *Circ. Res.* 101:146-155 (2007).

[12] Shi, Y., Seto, E., Chang, L.-S. & Shenk, T., (1991). Transcriptional repression by YY1, a human GLI-Kruppel-related protein, and relief of repression by adenovirus E1A protein. *Cell*, vol. 67, no. 2, 1991, 377-388.

[13] Smolin, L. (2005). The case for background independence. 2005. arXiv:hep-th/0507235v1.

[14] Vasudevan, S., Tong, Y., Steitz, J. A. (2007). Switching from Repression to Activation: MicroRNAs Can Up-Regulate Translation. *Science*, Vol. 318. no. 5858, 2007, 1931-1934

[15] Wilkinson, F. H. Park, K. & Atchison**,** M. L. (2006). Polycomb recruitment to DNA in vivo by the YY1 REPO domain. *Proc. Natl. Acad. Sci. USA*, 103:19296-19301 (2006).

[16] Zhang, W. -R., S. Chen & J. C. Bezdek (1989). POOL2: A Generic System for Cognitive Map Development and Decision Analysis. *IEEE Trans. on SMC., Vol. 19, No. 1*, 1989, 31-39.

[17] Zhang, W. -R., Chen, S., Wang, W. & King, R. (1992). A Cognitive Map Based Approach to the Coordination of Distributed Cooperative Agents. *IEEE Trans. on SMC, Vol. 22, No. 1*, 1992, 103-114.

[18] Zhang, W. -R., Wang, W. & King, R. (1994). An Agent-Oriented Open System Shell for Distributed Decision Process Modeling. *J. of Organizational Computing, Vol. 4, No. 2*, 1994, 127-154.

[19] Zhang, W. -R. (1996). NPN Fuzzy Sets and NPN Qualitative Algebra: A Computational Framework for Bipolar Cognitive Modeling and Multiagent Decision Analysis. *IEEE Trans. on SMC., Vol. 16*, 1996, 561-574.

[20] Zhang, W. –R. (2003a), Equilibrium Relations and Bipolar Cognitive Mapping for Online Analytical Processing with Applications in International Relations and Strategic Decision Support. *IEEE Transactions on Sys., Man and Cybern.*, Part B, Vol. 33. No. 2, 2003, 295-307.

[21] Zhang, W. -R. (2003b), Equilibrium Energy and Stability Measures for Bipolar Decision and Global Regulation. *Int'l J. of Fuzzy Sys*. Vol. 5, No. 2, 2003, 114-122.

[22] Zhang, W. -R. & Zhang, L. (2004a). YinYang Bipolar Logic and Bipolar Fuzzy Logic. *Information Sciences. Vol. 165, No. 3-4*, 2004, 265-287.

[23] Zhang, W. -R. (2005). YinYang Bipolar Lattices and L-Sets for Bipolar Knowledge Fusion, Visualization, and Decision. *Int'l J. of Inf. Technology and Decision Making, Vol. 4, No. 4*, 2005, 621-645.

[24] Zhang, W. -R. (2006). YinYang Bipolar Fuzzy Sets and Fuzzy Equilibrium Relations for Bipolar Clustering, Optimization, and Global Regulation. *Int'l J. of Inf. Technology and Decision Making*, Vol. 5 No. 1, 2006, 19-46.

[25] Zhang, W. –R. (2011), YinYang Bipolar Relativity: A Unifying Theory of Nature, Agents and Causality with Applications in Quantum Computing, Cognitive Informatics and Life Sciences. IGI Global, Hershey and New York, 2011.

[26] Zhang, W.-R. (2012a), YinYang Bipolar Atom – An Eastern Road toward Quantum Gravity. *J. of Modern Physics*, Vol. 3, No. 9, 2012, pp. 1261-1271. DOI: 10.4236/jmp.2012.329163 (Open Access)

[27] Zhang, W.-R. (2012b), Beyond Spacetime Geometry – The Death of Philosophy and Its Quantum Reincarnation. *J. of Modern Physics*, Vol. 3, No. 9, 2012. pp. 1272-1284. DOI: 10.4236/jmp.2012.329164 (Open Access)

[28] Zhang, W.-R. (2013), Bipolar Quantum Logic Gates and Quantum Cellular Combinatorics – A Logical Extension to Quantum Entanglement. *J. of Quantum Information Science*, Vol. 3, No. 2, 2013. pp. 93-105. DOI: 10.4236/jqis.2013.32014 (Open Access)

[29] Zhang, W.-R. and Peace, K. E. (2013), Revealing the Ubiquitous Effects of Quantum Entanglement – Toward a Notion of God Logic. *J. of Quantum Information Science*, Vol. 3, No. 4, 2013. pp. 143-153. DOI: 10.4236/jqis.2013.34019 (Open Access)

[30] Zhang, W. -R., A. Pandurangi & K. Peace (2007). YinYang Dynamic Neurobiological Modeling and Diagnostic Analysis of Major Depressive and Bipolar Disorders. *IEEE Trans. on Biomedical Engineering*, Oct. 2007 54(10):1729-39 (2007).

[31] Zhang, W. -R., Pandurangi, K. A., Peace, K., E., Zhang, Y. & Zhao, Z. (2011), MentalSquares − A Generic Bipolar Support Vector Machine for Psychiatric Disorder Classification, Diagnostic Analysis and Neurobiological Data Mining. Int'l *J. on Data Mining and Bioinformatics*. Vol. 17, No. 4, 2011, 547-576

[32] Zhou, Q. & Yik, J. H. N. (2006). The Yin and Yang of P-TEFb Regulation: Implications for Human Immunodeficiency Virus Gene Expression and Global Control of Cell Growth and Differentiation. *Microbiol Mol Biol Rev., vol. 70, no. 3*, 646–659.

Parallel Algorithm for Symmetric Positive Definite Banded Linear Systems: A Divide and Conquer Approach

S. Chandra Sekhara Rao[1] **and Sarita**[1]
[1]Department of Mathematics, Indian Institute of Technology Delhi, Hauz Khas, New Delhi- 110 016, India

Abstract— *The WZ factorization for the solution of symmetric positive definite banded linear systems when combined with a partitioned scheme, renders a divide and conquer algorithm. The WZ factorization of the coefficient matrix in each block has the properties: the vector $[a_1, \ldots, a_\beta, 0, \ldots, 0, a_{n-\beta+1}, \ldots, a_n]^T$ is invariant under the transformation W where β is the semibandwidth of the coefficient matrix and the solution process with the coefficient matrix Z proceeds from the first and the last unknowns to the middle. These properties of WZ factorization help us to decouple the partitioned system for the parallel execution once the 'reduced system' is solved.*

Keywords: WZ factorization, banded linear systems, parallel computing.

1. Introduction

Consider the parallel solution of the linear system

$$Ax = f \tag{1}$$

where A is an $N \times N$ symmetric positive definite matrix with $A = (a_{i,j}),\ \ i, j = 1, 2, \ldots, N$ and $a_{ij} = 0$ if $|i - j| > \beta$, where β is an integer such that $\beta << N$, called semi bandwidth of A. x, f are N-component unknown and known vectors given by $x = x^T_{1\to N}$, $f = f^T_{1\to N}$. They occur frequently in the numerical solution of partial and ordinary differential equations. To solve narrow banded systems in parallel divide and conquer, single width separator and double width separator approaches are available in the literature ([15],[11],[3],[16],[2]). Divide and conquer ([15]) and single width separator ([12],[4]) approaches are suitable for diagonally dominant and positive definite matrices, while double width separator approach ([16],[2]) is suitable for arbitrary (nonsingular) matrices. With the increasing availability and use of parallel computers much effort had been spent on the development of algorithms for the solution of banded systems. Survey on parallel solution of linear systems was given in [5]. Conroy [2] discussed the generalization of Wang's partition method [15] and one way dissection was applied to band matrices. These algorithms are comparable with Gaussian elimination and cyclic reduction. Wright [16] described and analyzed partitioned Gaussian elimination algorithm which was based on Dongarra and Johnsson [3]. Polizzi and Sameh [13] gave narrow banded system solvers based on SPIKE algorithm.

For the parallel algorithm based on modified Q.I.F. which was given in [8] the number of processors required for the algorithm was in terms of semibandwidth and the size of the system; on the other hand the number of processors required for the present parallel algorithm is in terms of number of blocks into which the system is partitioned. In the present work, the WZ factorization for the solution of symmetric positive definite banded linear systems is combined with a partitioned scheme. This renders a divide and conquer algorithm. The WZ factorization of the coefficient matrix in each block has the properties: the vector $[a_1, \ldots, a_\beta, 0, \ldots, 0, a_{n-\beta+1}, \ldots, a_n]^T$ is invariant under the transformation W where β is the semibandwidth of the coefficient matrix and the solution process with the coefficient matrix Z proceeds from the first and the last unknowns to the middle. These properties of WZ factorization help us to decouple the partitioned system for the parallel execution once the 'reduced system' is solved.

The outline of the paper is as follows. Section 2 describes the WZ factorization, partitioning of the symmetric banded linear system and the method of solution. The Algorithm is presented in section 3. Section 4 contains numerical experiments

2. The Present Method

First we describe the WZ factorization, then consider the partitioning of the system and decoupling the partitioned subsystems. Finally, the method is discussed. The resulting algorithm is given in section 3.

2.1 The WZ Factorization

Consider an $n \times n$ symmetric positive definite matrix A with $n = 2m-2$. Then there exists a matrix W (see[14])such that

$$A = WW^T \tag{2}$$

where

$$W = \begin{bmatrix} w_{1,1} & w_{1,2} & & & \cdots & \cdots & & w_{1,n} \\ 0 & w_{2,2} & \ddots & & \cdots & \cdots & w_{2,n-1} & 0 \\ & 0 & \ddots & & & \iddots & 0 & \\ & & & w_{m-1,m-1} & w_{m-1,m} & 0 & & \\ \vdots & \vdots & 0 & 0 & w_{m,m} & 0 & \vdots & \vdots \\ & & \iddots & & & \ddots & 0 & \\ & 0 & w_{n-1,3} & \cdots & \cdots & w_{n-1,n-1} & 0 & \\ 0 & w_{n,2} & & \cdots & \cdots & & w_{n,n} & \end{bmatrix}.$$

Note that the structure of W here, appears as transpose of W structure of Evans [6].

When A is a symmetric band matrix with semibandwidth β, for example, for $n = 10, \beta = 2$; the matrix W in (2) is given by

$$W = \begin{bmatrix} w_{1,1} & w_{1,2} & w_{1,3} & 0 & 0 & 0 & 0 & 0 & w_{1,9} & w_{1,10} \\ 0 & w_{2,2} & w_{2,3} & w_{2,4} & 0 & 0 & 0 & w_{2,8} & w_{2,9} & 0 \\ 0 & 0 & w_{3,3} & w_{3,4} & w_{3,5} & 0 & w_{3,7} & w_{3,8} & 0 & 0 \\ 0 & 0 & 0 & w_{4,4} & w_{4,5} & w_{4,6} & w_{4,7} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & w_{5,5} & w_{5,6} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & w_{6,6} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & w_{7,5} & w_{7,6} & w_{7,7} & 0 & 0 & 0 \\ 0 & 0 & 0 & w_{8,4} & w_{8,5} & w_{8,6} & w_{8,7} & w_{8,8} & 0 & 0 \\ 0 & 0 & w_{9,3} & w_{9,4} & 0 & 0 & w_{9,7} & w_{9,8} & w_{9,9} & 0 \\ 0 & w_{10,2} & w_{10,3} & 0 & 0 & 0 & 0 & w_{10,8} & w_{10,9} & w_{10,10} \end{bmatrix}. \quad (3)$$

In order to design a multiprocessor algorithm for symmetric banded linear system when solved by divide and conquer technique, we introduce the WZ factorization in which the inner $(n-2\beta) \times (n-2\beta)$ submatrices of W and Z are same as that of W and W^T in (3) respectively, and the vector $[a_1, \ldots, a_\beta, 0, \ldots, 0, a_{n-\beta+1}, \ldots, a_n]^T$ is invariant under the transformation W.

The factorization of A into WZ is defined as follows ($n = 2m-2$);

$$A = WZ.$$

Let w_i and z_i be the i^{th} column of W and Z respectively. Each $w_i,\ i = 1, 2, \ldots, n$ is of the following form.

$$w_i = \begin{cases} [\underbrace{0, \ldots, 0}_{i-1}, 1, \underbrace{0, \ldots 0}_{n-i}]^T, \\ \text{for } i = 1 \text{ to } \beta \text{ and for } i = n-\beta+1 \text{ to } n. \\ [0, \ldots, 0, w_{i-\beta,i}, \ldots, w_{i,i}, 0, \ldots, 0, w_{n-i+2,i}, \ldots, \\ w_{n-i+\beta+1,i}, 0, \ldots, 0]^T, \text{ for } i = \beta+1 \text{ to } m-1. \\ [0, \ldots, 0, w_{n-i-\beta+2,i}, \ldots, w_{n-i+1,i}, 0, \ldots, 0, \\ w_{i,i}, \ldots, w_{i+\beta,i}, 0, \ldots, 0]^T, \text{ for } i = m \text{ to } n-\beta. \end{cases}$$

and each $z_i,\ \ i = 1, 2, \ldots, n$ is of the following form.

$$z_i = \begin{cases} [w_{1,i}, \ldots, w_{i,i}, w_{i+1,i}, \ldots, w_{\beta,i}, w_{i,\beta+1}, \ldots, w_{i,i+\beta}, 0, \\ \ldots, 0, w_{i,n-i-\beta+2}, \ldots, w_{i,n-\beta}, w_{n-\beta+1,i}, \ldots, w_{n,i}]^T, \\ \quad \text{for } i = 1 \text{ to } \beta. \\ [0, \ldots, 0, w_{i,i}, \ldots, w_{i,i+\beta}, 0, \ldots, 0, w_{i,n-i-\beta+2}, \ldots, \\ w_{i,n-i+1}, 0, \ldots, 0]^T, \quad \text{for } i = \beta+1 \text{ to } m-1. \\ [0, \ldots, 0, w_{i,n-i+2}, \ldots, w_{i,n-i+\beta+1}, 0, \ldots, 0, \\ w_{i,i-\beta}, \ldots, w_{i,i}, 0, \ldots, 0]^T, \quad \text{for } i = m \text{ to } n-\beta. \\ [w_{1,i}, \ldots, w_{\beta,i}, w_{i,\beta+1}, \ldots, w_{i,n-i+\beta+1}, 0, \ldots, 0, \\ w_{i,i-\beta}, \ldots, w_{i,n-\beta}, w_{n-\beta+1,i}, \ldots, w_{n,i}]^T, \\ \quad \text{for } i = n-\beta+1 \text{ to } n. \end{cases}$$

The first β and the last β columns of W contain unit element on the diagonal position and zero elsewhere and W is transpose of Z, except for the first β and the last β columns. For example, for n=10, $\beta = 2$

$$W = \begin{bmatrix} 1 & 0 & w_{1,3} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & w_{2,3} & w_{2,4} & 0 & 0 & 0 & w_{2,8} & 0 & 0 \\ 0 & 0 & w_{3,3} & w_{3,4} & w_{3,5} & 0 & w_{3,7} & w_{3,8} & 0 & 0 \\ 0 & 0 & 0 & w_{4,4} & w_{4,5} & w_{4,6} & w_{4,7} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & w_{5,5} & w_{5,6} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & w_{6,6} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & w_{7,5} & w_{7,6} & w_{7,7} & 0 & 0 & 0 \\ 0 & 0 & 0 & w_{8,4} & w_{8,5} & w_{8,6} & w_{8,7} & w_{8,8} & 0 & 0 \\ 0 & 0 & w_{9,3} & w_{9,4} & 0 & 0 & w_{9,7} & w_{9,8} & 1 & 0 \\ 0 & 0 & w_{10,3} & 0 & 0 & 0 & 0 & w_{10,8} & 0 & 1 \end{bmatrix}$$

and

$$Z = \begin{bmatrix} w_{1,1} & w_{1,2} & 0 & 0 & 0 & 0 & 0 & 0 & w_{1,9} & w_{1,10} \\ w_{2,1} & w_{2,2} & 0 & 0 & 0 & 0 & 0 & 0 & w_{2,9} & w_{2,10} \\ w_{1,3} & w_{2,3} & w_{3,3} & 0 & 0 & 0 & 0 & 0 & w_{9,3} & w_{10,3} \\ 0 & w_{2,4} & w_{3,4} & w_{4,4} & 0 & 0 & 0 & w_{8,4} & w_{9,4} & 0 \\ 0 & 0 & w_{3,5} & w_{4,5} & w_{5,5} & 0 & w_{7,5} & w_{8,5} & 0 & 0 \\ 0 & 0 & 0 & w_{4,6} & w_{5,6} & w_{6,6} & w_{7,6} & w_{8,6} & 0 & 0 \\ 0 & 0 & w_{3,7} & w_{4,7} & 0 & 0 & w_{7,7} & w_{8,7} & w_{9,7} & 0 \\ 0 & w_{2,8} & w_{3,8} & 0 & 0 & 0 & 0 & w_{8,8} & w_{9,8} & w_{10,8} \\ w_{9,1} & w_{9,2} & 0 & 0 & 0 & 0 & 0 & 0 & w_{9,9} & w_{9,10} \\ w_{10,1} & w_{10,2} & 0 & 0 & 0 & 0 & 0 & 0 & w_{10,9} & w_{10,10} \end{bmatrix}.$$

2.2 Partitioning of the System

Consider the linear system $Ax = f,\ \ A \in \mathbb{R}^{N \times N},\ \ x^T Ax > 0,\ f \in \mathbb{R}^N$.

$$A = (a_{ij}),\ i, j = 1, 2, \ldots, N \text{ and } a_{ij} = 0 \text{ if } |i-j| > \beta \quad (4)$$

where β is an integer such that $\beta << N$, called semibandwidth of A. Partition A along the diagonal into r blocks each of size $n \times n$ (i.e. $N = rn$, we assume for simplicity that all blocks are of same size). Assume that $2\beta < n$. Partition the vectors x and f accordingly. Now each diagonal block has the same structure and the same bandwidth of A.

$$B^{(j)} x^{(j-1)} + A^{(j)} x^{(j)} + C^{(j)} x^{(j+1)} = f^{(j)}\ \ 1 \le j \le r \quad (5)$$

with $B^{(1)} = 0$, $C^{(r)} = 0$; $B^{(j+1)} = {C^{(j)}}^T$; $j = 1, \ldots r-1$; $x^{(0)} = 0$, $x^{(r+1)} = 0$; and

$$B^{(j)} = \begin{bmatrix} 0 & \hat{B}^{(j)} \\ 0 & 0 \end{bmatrix},\ \ C^{(j)} = \begin{bmatrix} 0 & 0 \\ \hat{C}^{(j)} & 0 \end{bmatrix};$$

$$x^{(j)} = [x_1^{(j)}, \ldots, x_n^{(j)}]^T,\ \ f^{(j)} = [f_1^{(j)}, \ldots, f_n^{(j)}]^T.$$

All $B^{(j)}, A^{(j)}, C^{(j)}$ are $n \times n$ matrices and $x^{(j)}$ and $f^{(j)}$ are $n \times 1$ vectors. $\hat{B}^{(j)}$ and $\hat{C}^{(j)}$ are $\beta \times \beta$ upper and lower triangular matrices respectively.

From (5) it follows that

$$A^{(j)}x^{(j)} = f^{(j)} - \begin{bmatrix} \hat{B}^{(j)}{x_L}^{(j-1)} \\ 0 \\ \hat{C}^{(j)}{x_F}^{(j+1)} \end{bmatrix} = f^{*(j)} \text{ say}, j = 1, \ldots, r. \quad (6)$$

where

$${x_F}^{(j)} = [x_1^{(j)}, \ldots, x_\beta^{(j)}]^T, {x_L}^{(j)} = [x_{n-\beta+1}^{(j)}, \ldots, x_n^{(j)}]^T.$$

Note that $f^{*(j)}$ in (6) differs from $f^{(j)}$ only in its first β and last β components. For the purpose we need a factorization of submatrices $A^{(j)}$ into $W^{(j)}Z^{(j)}$; $1 \leq j \leq r$, with the properties that vector $[a_1, \ldots, a_\beta, 0, \ldots, 0, a_{n-\beta+1}, \ldots, a_n]^T$ is invariant under the transformation $W^{(j)}$ and the solution process with coefficient matrix $Z^{(j)}$ proceeds from the first and the last unknowns to the middle. The WZ factorization which we introduced has these properties.

2.3 The Method

We now consider the solution of the systems (5). This consists of finding ${y^*}^{(j)}$ from

$$W^{(j)}{y^*}^{(j)} = f^{*(j)}, \qquad j = 1, \ldots, r$$

and then solving for $x^{(j)}$

$$Z^{(j)}x^{(j)} = {y^*}^{(j)}, \qquad j = 1, \ldots, r. \quad (7)$$

Let $y^{(j)} = [y_1^{(j)}, \ldots, y_n^{(j)}]^T$ and consider

$$W^{(j)}y^{(j)} = f^{(j)}, \qquad j = 1, \ldots, r.$$

Because the vector $[a_1, \cdots, a_\beta, 0, \cdots, 0, a_{n-\beta+1}, \cdots, a_n]^T$ is invariant under the transformation $W^{(j)}$, and from the definition of $f^{*(j)}$, it immediately follows that

$${y^*}^{(j)} = y^{(j)} - \begin{bmatrix} \hat{B}^{(j)}x_L^{(j-1)} \\ 0 \\ \vdots \\ 0 \\ \hat{C}^{(j)}x_F^{(j+1)} \end{bmatrix}, \quad j = 1, \ldots, r. \quad (8)$$

Once $y^{(j)}$ are determined, the subsystem (7) can be replaced by

$$Z^{(j)}x^{(j)} = y^{(j)} - \begin{bmatrix} \hat{B}^{(j)}x_L^{(j-1)} \\ 0 \\ \vdots \\ 0 \\ \hat{C}^{(j)}x_F^{(j+1)} \end{bmatrix}, \quad j = 1, \ldots, r. \quad (9)$$

From the definition of $W^{(j)}$ it is now clear that among the subsystems in (9) we can extract a relatively small (for $r << N$) subsystem involving only the first β and the last β unknowns of each block. Accordingly, collecting together the first and the last β equations of each block, we obtain a linear system called 'reduced system' of order $2\beta r \times 2\beta r$ of semi bandwidth $2\beta - 1$.

Let

$$W_1^{(j)} = \begin{bmatrix} w_{1,1}^{(j)} & \cdots & w_{1,\beta}^{(j)} \\ \vdots & \vdots & \vdots \\ w_{\beta,1}^{(j)} & \cdots & w_{\beta,\beta}^{(j)} \end{bmatrix},$$

$$W_2^{(j)} = {W_3^{(j)}}^T = \begin{bmatrix} w_{1,n-\beta+1}^{(j)} & \cdots & w_{1,n}^{(j)} \\ \vdots & \vdots & \vdots \\ w_{\beta,n-\beta+1}^{(j)} & \cdots & w_{\beta,n}^{(j)} \end{bmatrix},$$

$$W_4^{(j)} = \begin{bmatrix} w_{n-\beta+1,n-\beta+1}^{(j)} & \cdots & w_{n-\beta+1,n}^{(j)} \\ \vdots & \vdots & \vdots \\ w_{n,n-\beta+1}^{(j)} & \cdots & w_{n,n}^{(j)} \end{bmatrix},$$

where $W_2^{(j)}$ is symmetric.

Then the reduced system can be written as,

$$\begin{bmatrix} W_1^{(1)} & W_2^{(1)} & & & & & & & \\ W_3^{(1)} & W_4^{(1)} & \hat{C}^{(1)} & & & & & & \\ & \hat{B}^{(2)} & W_1^{(2)} & W_2^{(2)} & & & & & \\ & & W_3^{(2)} & W_4^{(2)} & \hat{C}^{(2)} & & & & \\ & & & \hat{B}^{(3)} & \ddots & \ddots & & & \\ & & & & \ddots & \ddots & \hat{C}^{(r-1)} & & \\ & & & & & \hat{B}^{(r)} & W_1^{(r)} & W_2^{(r)} & \\ & & & & & & W_3^{(r)} & W_4^{(r)} & \end{bmatrix} \begin{bmatrix} x_F^{(1)} \\ x_L^{(1)} \\ x_F^{(2)} \\ x_L^{(2)} \\ \vdots \\ \vdots \\ x_F^{(r)} \\ x_L^{(r)} \end{bmatrix} = \begin{bmatrix} y_F^{(1)} \\ y_L^{(1)} \\ y_F^{(2)} \\ y_L^{(2)} \\ \vdots \\ \vdots \\ y_F^{(r)} \\ y_L^{(r)} \end{bmatrix}. \quad (10)$$

The reduced system (10) may be represented as

$$Rx_R = y_R \quad (11)$$

where

$$x_R = [x_F^{(1)}, x_L^{(1)}, \ldots, x_F^{(r)}, x_L^{(r)}]^T, y_R = [y_F^{(1)}, y_L^{(1)}, \ldots, y_F^{(r)}, y_L^{(r)}]^T,$$

and R is the coefficient matrix of (10).

For a matrix $S = (s_{i,j})_{i,j=1}^n$ and a vector $q = (q_1, \ldots\ldots, q_n)^T$, we introduce the notation:

$$S_{2\to n-1} = (s_{i,j})_{i,j=2}^{n-1} \text{ and } q_{2\to n-1} = (q_2, \ldots\ldots, q_{n-1})^T.$$

Once the reduced system of order $2\beta r$ is solved for $x_F^{(j)}$ and $x_L^{(j)}$ $j = 1, \ldots, r$; the subsystems in (9), for $j = 1, \ldots, r$, are uncoupled into

$$Z_{\beta+1\to n-\beta}^{(j)} x_{\beta+1\to n-\beta}^{(j)} = y_{\beta+1\to n-\beta}^{(j)} - \begin{bmatrix} U_1^{(j)}x_F^{(j)} + U_2^{(j)}x_L^{(j)} \\ 0 \\ \vdots \\ 0 \\ L_1^{(j)}x_F^{(j)} + L_2^{(j)}x_L^{(j)} \end{bmatrix}. \quad (12)$$

where

$$U_1^{(j)} = \begin{bmatrix} w_{1,\beta+1} & \cdots & w_{\beta,\beta+1} \\ 0 & \ddots & \vdots \\ 0 & 0 & w_{\beta,2\beta} \end{bmatrix},$$

$$U_2^{(j)} = \begin{bmatrix} w_{n-\beta+1,\beta+1} & \cdots & w_{n,\beta+1} \\ \vdots & \ddots & 0 \\ w_{n-\beta+1,2\beta} & 0 & 0 \end{bmatrix},$$

$$L_1^{(j)} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & w_{\beta,n-2\beta+2} \\ 0 & \ddots & \ddots & \vdots \\ 0 & w_{2,n-\beta} & \cdots & w_{\beta,n-\beta,} \end{bmatrix},$$

$$L_2^{(j)} = \begin{bmatrix} w_{n-\beta+1,n-2\beta+1} & 0 & 0 \\ \vdots & \ddots & 0 \\ w_{n-\beta+1,n-\beta} & \cdots & w_{n,n-\beta} \end{bmatrix}.$$

3. Algorithm

Step 1. For $j = 1, \ldots, r$ factorize in parallel

$$A^{(j)} = W^{(j)} Z^{(j)}.$$

Step 2. For $j = 1, \ldots, r$ compute $y^{(j)}$ in parallel

$$W^{(j)} y^{(j)} = f^{(j)}.$$

Step 3. Solve the reduced subsystem (10) for $x_1^{(j)}, \cdots, x_\beta^{(j)}, x_{n-\beta+1}^{(j)}, \cdots, x_n^{(j)}, j = 1, \ldots, r$ sequentially by band cholesky factorization.

Step 4. For $j = 1, \ldots, r$ compute $x_{\beta+1 \to n-\beta}^{(j)}$ in parallel from (12).

Speedup, S_p of the parallel algorithm on r processor machine is given by

$$S_p = \frac{\text{Time taken by the best sequential ([7]) algorithm}}{\text{Time taken by the parallel algorithm on r processor machine}}$$

$$S_p = \frac{N\beta^2+7N\beta+3N}{4n\beta^2+12n\beta+2n-10+8r\beta^3+24r\beta^2-6r\beta-(8\beta^3+30\beta^2+40\beta)/3}$$

As $N = nr$, if N is large enough and β is fairly large, speedup is approximately $0.25r$. If β is 2, the speedup is $0.5r$ and if β is 10 the speedup is approximately $0.33r$.

4. Numerical Experiments

Numerical experiments are conducted on a parallel machine; the Ultra SPARC III technology based **Sunfire 6800** having 16 processors (each of 800 MHZ and has 1 GB RAM) shared memory server. Full hardware redundancy and a variety of advances mainframe-class availability features such as CPU upgrades and dynamic reconfiguration, provide maximum uptime. The hard disk memory is 192 GB and the operating system is Solaris 8.0 .

Total (computation and communication) time (in seconds) statistics of the proposed algorithm for the solution of symmetric positive definite banded linear systems of different orders with coefficient matrices having semibandwidth $\beta = 10$ against different number of processors is given in Table 1 . In this table time for p=1, corresponds to the time taken by the best sequential algorithm (band Cholesky [7]). From Table 1, we observe that the total time for the solution (for p=2 onwards) in each column decreases from top to bottom where as total time for the solution each row increases from left to right.

Speedup against different number of processors for symmetric positive definite banded linear systems of different orders with coefficient matrices having semibandwidth $\beta = 10$ is plotted in Figure 1. We observe from the Figure 1 that the speedup increases with the increase in the number of processors. Communication time against different number of processors in solving banded linear systems of different orders with coefficient matrices having semibandwidth $\beta = 10$ is plotted in Figure 2. We observe from the Figure 2 that the communication time on and after 6 processors is almost the same. This is because the semibandwidth β is same though the orders of symmetric positive definite banded linear systems are different and the dimension of the reduced system is $2\beta r$ where r is the number of processors. This is a main point of the algorithm which is illustrated through the leveling of the communication costs as the number of processors increases.

Table 1: Total (Computation and Communication) time for the solution of banded symmetric positive definite linear systems of different orders with coefficient matrices having semibandwidth $\beta = 10$

'N'→ 'P' ↓	84000	168000	252000	336000	420000
1	0.524443	1.128744	1.692446	2.242299	2.855724
2	0.859492	1.745133	2.624939	3.502575	4.238210
4	0.454786	0.904346	1.327487	1.836458	2.251653
6	0.337217	0.612587	1.000377	1.246517	1.568865
8	0.275847	0.516232	0.739026	1.030361	1.295398
10	0.274412	0.468705	0.625528	0.886914	1.086321
12	0.254802	0.450342	0.614491	0.701791	0.948445
16	0.243192	0.431932	0.592732	0.632783	0.773912

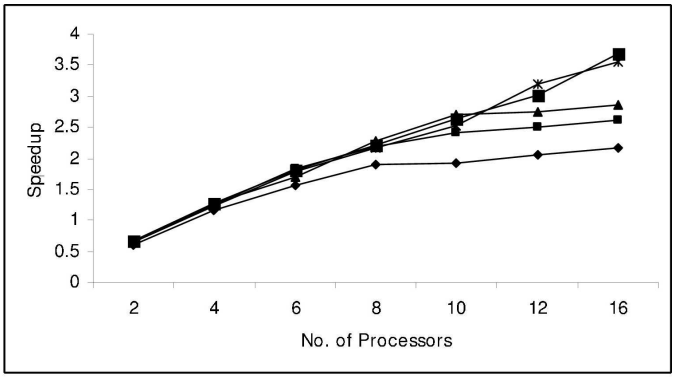


Fig. 1: Speedup against different number of processors in solving banded linear systems of different orders with coefficient matrices having semibandwidth $\beta = 10$. The symbols ♦, ■, ▲, ∗, ■ correspond to linear systems of order $84000, 168000, 252000, 336000, 420000$ respectively.

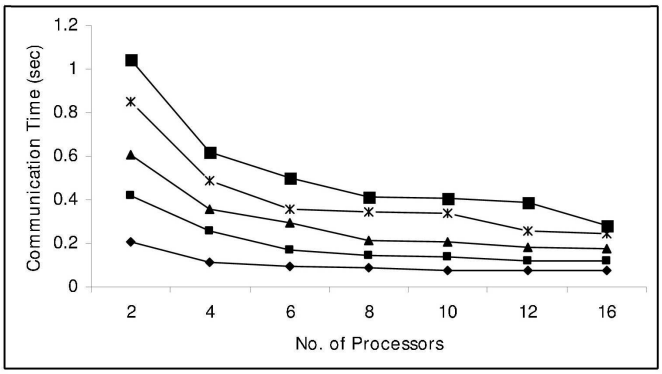


Fig. 2: Communication time against different number of processors in solving banded linear systems of different orders with coefficient matrices having semibandwidth $\beta = 10$. The symbols ♦, ■, ▲, ∗, ■ correspond to linear systems of order $84000, 168000, 252000, 336000, 420000$ respectively.

References

[1] I. Baron, "A practical parallel algorithm for solving band symmetric positive definite systems of linear equations," *ACM Trans. Math. Software*, vol. 13, pp. 323–332, 1987.

[2] J. M. Conroy, "Parallel algorithms for the solution of narrow banded systems" *Appl. Numer. Math.*,vol. 5, pp. 409-421, 1989.

[3] J .J. Dongarra, and S. L. Johnsson, "Solving banded systems on a Parallel Processor," *Parallel Comput.*,vol. 5, pp. 219–246, 1987.

[4] J. J. Dongarra,and A. Sameh, "On some parallel banded system solvers," *Parallel Comput.* , vol. 1, pp. 223–235, 1984.

[5] I. S. Duff, and H. A. Van der Vorst, "Developments and trends in the parallel solution of linear systems," *Parallel Comput.*, vol. 25, pp. 1931–1970, 1999.

[6] D. J. Evans, "The Choleski Q.I.F. algorithm for solving symmetric linear systems,"*Intern. J. Computer Math.*, vol. 72 pp. 283–288, 1999.

[7] G. H. Golub, and C. F. Van Loan, *Matrix computations*, 3rd ed., The Johns Hopkins University Press, 1995.

[8] D. J. Evans, A. Hadjidimos, and D. Noutsos, "The parallel solution of banded linear equations by the new quadrant interlicking factorization (Q.I.F.) method," *Intern. J. Computer Math.*, vol. 9, pp. 151–161, 1981.

[9] I. N. Haji, and S. Skelboe, " A multilevel parallel solver for block tridiagonal and banded linear systems," *Parallel Comput.*, vol. 15,pp. 21–45, 1990.

[10] L. Halada, "A parallel algorithm for solving banded systems and matrix inversion," *Computer Arti. Intell.*, vol. 2, pp. 373–383, 1983.

[11] S. L. Johnsson, "Solving narrow banded system on ensamble architectures," *ACM Trans. Math. Software.*, vol. 11, pp. 271–288, 1985.

[12] S. L. Johnsson, " Communication efficient basic linear algebra computations on hypercube architectures," *J. Par. Dist. Comput.*, vol. 4, pp. 133–172, 1987.

[13] E. Polizzi, and A. H. Sameh, "A Parallel hybrid banded system solver, the SPKIE algorithm," *Parallel Comput.*, vol. 32, pp. 177–194, 2006.

[14] S. C. S. Rao, and Sarita, " A symmetric linear system solver," *Appl. Math. Comput.*, vol. 203, pp. 368–379, 2008.

[15] H. H. Wang, "A parallel method for tridiagonal equations," *ACM Trans. Math. Software.*, vol. 7, pp. 170–183, 1981.

[16] S. J. Wright, "Parallel algorithms for banded linear systems," *SIAM J. Sci. Stat. Comput.*, vol. 12, pp. 824–842, 1991.

SESSION

SCIENTIFIC COMPUTING + FINITE ELEMENT METHODS + ODE + KALMAN FILTER

Chair(s)

TBA

Finite Element Analysis of Five-Transmission Lines Embedded in Four-Layered Dielectric Media

Sarhan M. Musa and Matthew N. O. Sadiku
Roy G. Perry College of Engineering, Prairie View A&M University
Prairie View, TX 77446

Abstract- **Development of very high speed integrated circuits is currently of great interests for today's technologies. This paper presents the quasi-TEM approach for the accurate parameters extraction of multiconductor transmission lines interconnect in four-layered dielectric region using the finite element method (FEM). We illustrate that FEM is accurate and effective for modeling multilayered multiconductor transmission lines in strongly inhomogeneous media. We mainly focus on designing of five-transmission lines embedded in four-layered dielectric media. We computed the capacitance matrices for this configuration. Also, we determine the quasi-TEM spectral for the potential distribution of the multiconductor transmission lines in multilayer dielectric media.**

Keywords-capacitance per unit length; multiconductor transmission lines;finite element method; multilayer dielectric media

I. INTRODUCTION

Nowadays, the designing of fast electronics circuits and systems with increase of the integration density of integrated circuits led to wide use and cautious analysis of multilayer and multiconductor interconnects. As the transversal size multiple-conductor transmission lines are reduced, adjacent conductors are electromagnetically coupled so that they must be considered as multimode waveguides [1]. The matrices of capacitances per unit length of multilayered multiconductor quasi-TEM transmission lines are known as the essential parameters in designing of package, lossless transmission line system, microwave circuits, and very large scale integration circuits. Therefore, the improvement of accurate and efficient computational method to analyze the multiconductor quasi-TEM transmission lines structure becomes an important area of interest. Also, to optimize the electrical properties of the integrated circuits, the estimate of the capacitance matrix of multilayer and multiconductor interconnects in very high-speed integrated circuit must be investigated. The computational values of self and coupling capacitance can also help engineers and designers to optimize the layout of the circuit.

Previous attempts at the problem include using the analytical modelization of multiconductor quasi-TEM transmission lines [2], spectral domain method [3], the method of moments (MoM) [4,5], spectral domain approach (SDA) [6], Green's function approach [7,8], the method of lines (MoL) [9,10], domain decomposition method (DDM), finite difference methods (FDM) [10], and finite element method (FEM) [11-17]

In this work, we design five-transmission lines interconnect in four-layered dielectric region using COMSOL, a finite element package. Many industrial applications depend on different interrelated properties or natural phenomena and require multiphysics modeling and simulation as an efficient method to solve their engineering problems. Moreover, superior simulations of microwave integrated circuit applications will lead to more cost-efficiency throughout the development process. We specifically calculate the self and mutual capacitances and the potential distribution of the configurations.

II. RESULTS AND DISCUSSIONS

The models are designed in 2D using electrostatic environment in order to compare our results with some of the other available methods. In the boundary condition of the model's design, we use ground boundary which is zero potential ($V=0$) for the shield. We use port condition for the conductors to force the potential or current to one or zero depending on the setting. Also, we use continuity boundary condition between the conductors and between the conductors and left and right grounds.

The quasi-static models are computed in form of electromagnetic simulations using partial differential equations. The quasi-static analysis

is valid under the assumption that $\frac{\partial \mathbf{D}}{\partial t} = 0$, where $\mathbf{D}$ is the electric displacement. Thus Maxwell's equations can be written in the following forms:

$$\nabla \times \mathbf{H} = \mathbf{J} = \sigma\left(\mathbf{E} + \mathbf{v} \times \mathbf{B}\right) + \mathbf{J}^e \quad (1)$$

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t} \quad (2)$$

$$\nabla \times \mathbf{B} = 0 \quad (3)$$

$$\nabla \times \mathbf{D} = \rho \quad (4)$$

$$\nabla \times \mathbf{J} = 0 \quad (5)$$

where $\mathbf{H}$ is the magnetic field, σ the electrical conductivity, $\mathbf{E}$ is the electric field, $\mathbf{v}$ is the velocity of the conductor, $\mathbf{B}$ is the magnetic flux density, $\mathbf{J}^e$ is an externally generated current density, ρ is the charge density, $\mathbf{J}$ is the current density. However, the essential criterion for the quasi-static approximation to be valid is that the currents and the electromagnetic fields vary slowly.

Using magnetic potential $\mathbf{A}$ definition we get:

$$\mathbf{B} = \nabla \times \mathbf{A} \quad (6)$$

and

$$\mathbf{E} = -\nabla V - \frac{\partial \mathbf{A}}{\partial t} \quad (7)$$

where V is the electric potential. And by using the constitutive relation,

$$\mathbf{B} = \mu_0(\mathbf{H} + \mathbf{M}) \quad (8)$$

where $\mathbf{M}$ is the magnetization, the Ampere's law now can be rewritten as

$$\sigma \frac{\partial \mathbf{A}}{\partial t} + \nabla \times \left(\mu_0^{-1} \nabla \times \mathbf{A} - \mathbf{M}\right) - \sigma \mathbf{v} \times \left(\nabla \times \mathbf{A}\right) + \sigma \nabla V = \mathbf{J}^e \quad (9)$$

where μ_0 is the permeability of vacuum. Thus, the continuity equation can be written as

$$-\nabla \cdot \left(\sigma \frac{\partial \mathbf{A}}{\partial t} - \sigma \mathbf{v} \times \left(\nabla \times \mathbf{A}\right) + \sigma \nabla V - \mathbf{J}^e\right) = 0 \quad (10)$$

Equations (9) and (10) provide the system of the two equations for $\mathbf{A}$ and V .

Now, we illustrate the modeling of five-transmission lines interconnect in four-layered dielectric media. We focus on the calculation of self and mutual capacitances per unit length and determine the quasi-TEM spectral for the potential distribution of the model.

In Figure 1, we show the cross section for five-conductor transmission lines in four-layered dielectric region and its parameters.

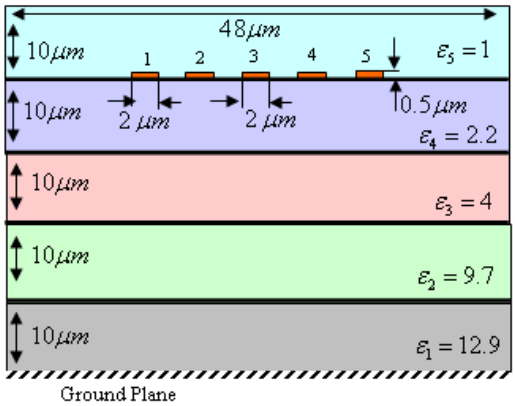


Fig. 1. Cross-section of five-conductor transmission lines interconnect in four-layered dielectric region.

From the model, we generate the finite element mesh plot as in Figure 2. Table I shows the statistical properties of the model mesh.

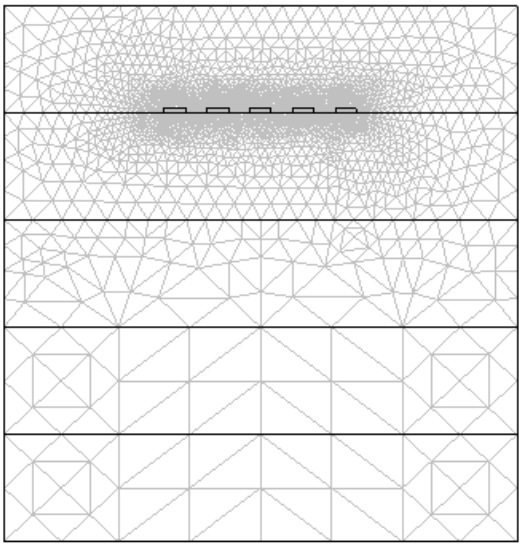


Fig. 2. Mesh plot of five-conductor transmission lines interconnect in four-layered dielectric region.

Table I. Mesh statistics of the five-conductor transmission lines interconnect in four-layered dielectric region

Items	Value
Number of degrees of freedom	10596
Total number of mesh points	2579
Total number of elements	5068
Triangular elements	5068
Quadrilateral elements	0
Boundary elements	296
Vertex elements	36

Figure 3 shows the 2D surface for electrical potential (V) distribution of the transmission lines, while the contour of electric potential (V) and streamline of electric field plots of the model are presented in Figures 4 and 5, respectively.

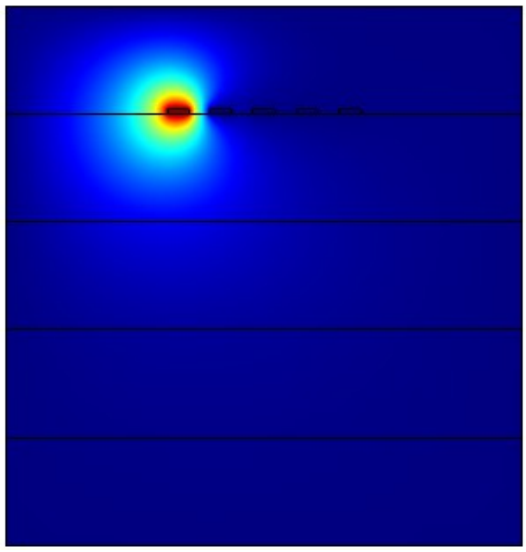


Fig. 3. 2D surface electrical potential distribution of the five-conductor transmission lines in four-layered dielectric region.

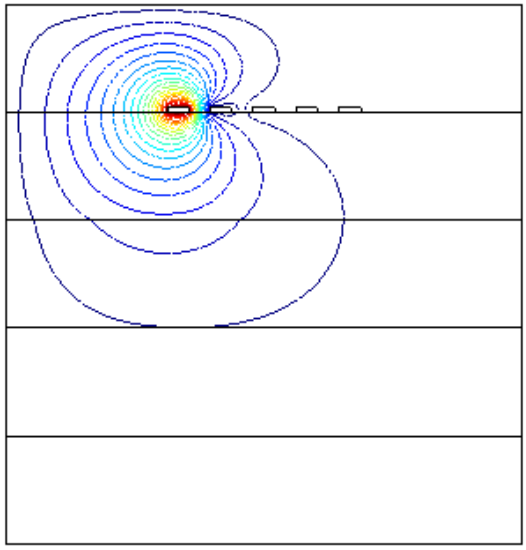


Fig. 4. Contour plot of the five-conductor transmission lines in four-layered dielectric region.

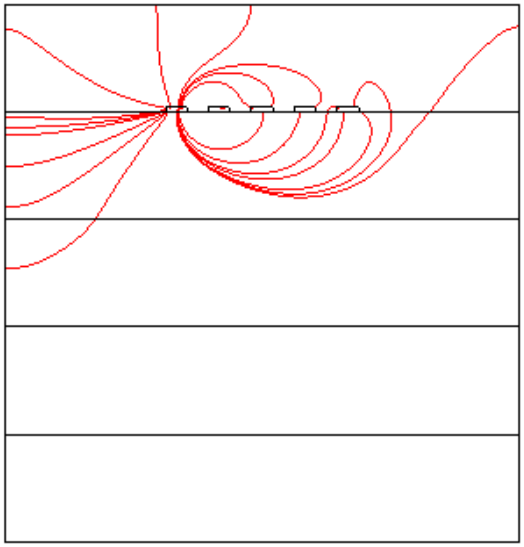


Fig. 5. Streamline plot of the five-conductor transmission lines interconnect in four-layered dielectric region.

Fig. 6 presents the Potential distribution of five-conductor transmission lines interconnect in single layered dielectric region from (x,y) = (0,0) to (x,y) = (48,50) μ m, using port 1 as input. Fig. 7 shows the comparison analysis of potential distribution of the model with and without dielectric substrate. It observed that the peak value of electric potential is decreased as the dielectric is placed in the substrate.

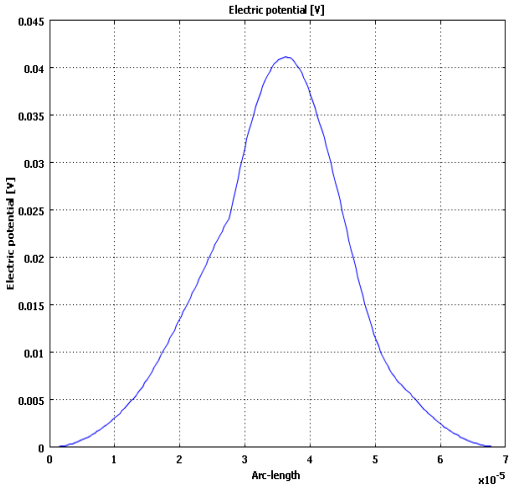


Fig. 6. **E**lectric potential plot (V) as a function of arc-length for the model.

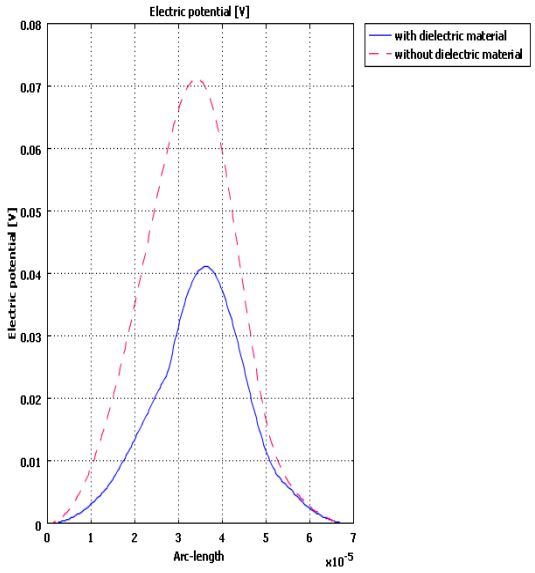


Fig. 7. Comparison analysis of potential distribution of the model with and without dielectric substrate.

Recently, with the advent of integrated circuit technology, the coupled microstrip transmission lines consisting of multiple conductors embedded in a multilayer dielectric medium have led to a new class of microwave networks. Multiconductor transmission lines have been utilized as filters in microwave region which make it interesting in various circuit components. For coupled multiconductor microstrip lines, it is convenient to write [18-19]:

$$Q_i = \sum_{j=1}^{m} C_{sij} V_j \quad (\text{i} = 1, 2, \ldots, \text{m}) \quad , \qquad (11)$$

where Q_i is the charge per unit length, V_j is the voltage of j th conductor with reference to the ground plane, C_{sij} is the short circuit capacitance between i th conductor and j th conductor. The short circuit capacitances can be obtained either from measurement or from numerical computation. From the short circuit capacitances, we obtain

$$C_{ii} = \sum_{j=1}^{m} C_{sij} \quad , \qquad (12)$$

where C_{ii} is the capacitance per unit length between the i th conductor and the ground plane. Also,

$$C_{ij} = -C_{sij}, \quad j \neq i \quad , \qquad (13)$$

where C_{ij} is the coupling capacitance per unit length between the i th conductor and j th conductor. The coupling capacitances are illustrated in Fig. 8.

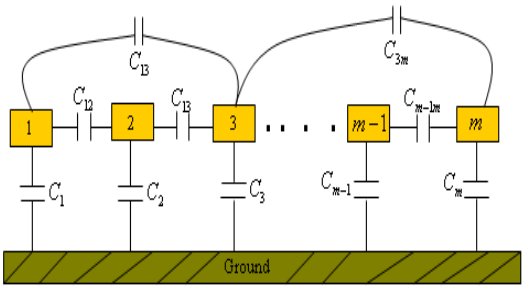


Fig. 8. The per-unit length capacitances of a general m -conductor transmission line.

For m -strip line, the per-unit-length capacitance matrix [C] is given by [20]

$$[C]=\begin{bmatrix} C_{11} & -C_{12} & \mathsf{L} & -C_{1m} \\ -C_{21} & C_{22} & \mathsf{L} & -C_{2m} \\ \mathsf{M} & \mathsf{M} & & \mathsf{M} \\ -C_{m1} & -C_{m2} & \mathsf{L} & C_{mm} \end{bmatrix} . \quad (14)$$

The finite element results for the self and mutual capacitances per unit length of the five-transmission lines interconnect in four-layered dielectric medium are:

$$[C]=\begin{bmatrix} 36.0 & -15.2 & -2.3 & -0.8 & -0.4 \\ -15.2 & 42.6 & -14.3 & -2.0 & -0.8 \\ -2.3 & -14.3 & 42.8 & -14.3 & -2.3 \\ -0.8 & -2.0 & -14.3 & 42.6 & -15.2 \\ -0.4 & -0.8 & -2.3 & -15.2 & 36.0 \end{bmatrix} pF/m .$$

III. CONCLUTIONS

In this paper we have presented the modeling in 2D of designing of quasi-TEM five-transmission lines interconnect in lines interconnect in four-layered dielectric region using FEM. We computed the capacitance-per-unit length matrix of the model. Also, we determine the quasi-TEM spectral for the potential distribution of the multiconductor transmission lines in multilayer dielectric media. The results obtained in this research are encouraging and motivating for further study.

REFERENCES

[1]. C. Seguinot, E. Paleczny, F. Huret, J. F. Legier, and P. Kennis, "Experimental determination of the characteristic impedance matrix of multiconductor quasi-TEM lines," *Microwave Theory and Optical Technology Letters,* vol. 22, no. 6, pp. 429-431, Sep. 1999.

[2]. X. Pannier, E. Paleczny, C. Seguinot, F. Huret, P. Kennis, "analytical and full-wave characterization of multimode waveguide discontinuities," *27th European Microwave Conference*, vol. 1, pp. 485 – 489, 1997.

[3]. R. Schwindt and C. Nguyen, "Spectral domain analysis of three symmetric coupled lines and application to a new bandpass filter", *IEEE Transmissions on Microwave Theory and Techniques*, Vol. 42, No. 7, July 1994, pp. 1183-1189.

[4]. C. Wei, R. F. Harrington, J. R. Mautz, and T. K. Sarkar, "Multiconductor transmission lines in multilayered dielectric media, "*IEEE Transmissions on Microwave Theory and Techniques*, Vol. 32, No. 4, April 1984, pp. 439-45.

[5]. C. Wei and R. F. Harrington, "Computation of the parameters of multiconductor transmission lines in two dielectric layers above a ground plane, "Depar t. Electrical Computer Eng., Syracuse University, Rep. TR-82-12, Nov. 1982.

[6]. G. Plaza, F. Mesa, and M. Horno," Quick computation of [G], [L], [G], and [R] matrices of multiconductor and multilayered transmission systems," *IEEE Transmissions on Microwave Theory and Techniques*, Vol. 43, No. 7, July 1995, pp. 1623-1626.

[7]. W. Shu and S. Xu," Capacitance extraction for multiconductor transmission lines in multilayered dielectric media using the numerical green's function," *Microwave and*

Optical Technology Letters, Vol. 40, No. 6, March 2006, pp. 529-531.

[8]. W. Delbare and D. De Zutter," Space-domain Green's function approach to the capacitance calculation of multiconductor lines in multilayered dielectrics with improved surface charge modeling," *IEEE Transmissions on Microwave Theory and Techniques*, Vol. 37, No. 10, October 1989, pp. 1562-1568.

[9]. A. Papachristoforos," Method of lines for analysis of planar conductors with finite thickness, "*IEEE Proc. Microwave Antennas & Propagation*, Vol. 141, No. 3, June 1994, pp. 223-228.

[10]. L. Shujing and Z. Hanqing," An efficient algorithm for the parameter extraction of multiconductor transmission lines in multilayer dielectric media," *Proceeding of IEEE Antennas and Propagation Society International Symposium,* July 2005, Vol. 3A, pp.228-231.

[11]. S. M. Musa, M. N. O. Sadiku and P. H. Obiomon "Integrated Circuit Interconnect Lines on Lossy Silicon Substrate with Finite Element Method," *Int. Journal of Engineering Research and Applications*, vol. 4, issue 1 (version 5), pp.17-21, January 2014.

[12]. S. M. Musa and M.N.O. Sadiku, "Finite Element Approach for Coupled Striplines Embedded in Dielectric Material," *TELKOMNIKA (Telecommunication, Computing, Electronic and Control)*, vol. 11, no. 1, pp. 47-54, March 2013.

[13]. S. M. Musa and M.N.O. Sadiku, "Finite Element Approach of Shielded, Suspended and Inverted, Microstrip Lines," *Bulletin of Electrical Engineering and Informatics (BEEI)*, vol. 2, no. 1, pp. 1-10, March 2013.

[14]. S. M. Musa and M.N.O. Sadiku, "Finite Element Approach of Unshielded Multiconductor Transmission Lines Embedded in Layered Dielectric Region for VLSI Circuits," International *Journal of Computing and Digital Systems (IJCDS)*, Vol. 1, No. 1, pp. 25-30 October , 2012.

[15]. B. ElJaafari, M. A. Gonzalez de Aza, and J. Zapata, "An approach based on finite element method for CAD of printed antennas," *IEEE Antennas and Wireless Propagation Letters,* vol. 11,pp. 1238-1241, 2012.

[16]. Z. J. Mancic and V. V. Petrovic, "Quasi-Static analysis of the shielded microstrip line with bi-isotropic substrate by the strong FEM formulation," 11th International conference on Telecommunication in Modern Satellite, Cable and Broadcasting Service (TELSIKS), pp. 513-516, Serbia, Nis, vol. 2, October 16-19, 2013.

[17]. M. A. Kolbehdari, "Quasi-Satic schematistic of a shield cylindrical coupled microstrip transmission line by finite element method," IEEE *Proceedings of Southeastcon,* pp. 170-174, 1994.

[18]. M. S. Lin, "Measured capacitance coefficients of multiconductor microstrip lines with small dimensions," *IEEE Transactions on Microwave Theory and Techniques*, vol. 13, no. 4, pp. 1050-1054, Dec. 1990.

[19]. F. Y. Chang, "Transient analysis of lossless coupled transmission lines in a nonhomogeneous dielectric media," *IEEE Transactions on Microwave Theory and Techniques*, vol 18, no 9, pp. 616-626, Aug. 1970.

[20]. P. N. Harms, C. H. Chan, and R. Mittra, "Modeling of planar transmission line structures for digital circuit applications," *Arch. Eleck. Ubertragung.,* vol. 43, pp. 245-250, 1989.

Kalman Filter Based Safety Application

Rawa Adla[1], Nizar Al-Holou[1], and Youssef A.Bazzi[2]
[1]Department of Electrical and Computer Engineering, University of Detroit Mercy, Detroit, Michigan, USA
[2]Department of Electrical and Computer Engineering, Lebanese University, Beirut, Lebanon

Abstract: *This article examines the basic concepts of the Kalman filter. The origination of the Kalman filter and the theory behind it has been presented, along with an explanation of the Kalman filter algorithm and its equations. In order to study the Kalman filter and the method of implementation, a case study using the Kalman filter to estimate the speed of a preceding vehicle in a variety of scenarios is discussed. Through simulation it is observed that the Kalman filter is an optimal filter and a very helpful mathematical tool to cancel out noise when dealing with a noisy environment, but only under some restrictions that are discussed in the paper.*

Keywords: Sensor fusion, Kalman Filter, speed estimation.

1 Introduction

In 1960 Rudolf Kalman published his famous articles about the Kalman filter [1] [2]. The Kalman filter essentially is an analytical form that makes an estimation of the state of interest for any linear system. Kalman filter can determine a system's condition over time from a number of indirect measurements or on the basis of inaccurate measurements of the system itself. These measurements can be, for example, disturbed by random measurement noise or noise within the system [3].The Kalman filter idea is depicted in Fig (1) [3].

The Kalman filter consists of a series of equations that calculate a system's mode which is not possible to observe [3]. In order to understand the Kalman filter as well as its method of implementation, however, it is necessary to know the theory behind the Kalman filter and its equations. The Kalman filter is considered very effective when dealing with a noisy system. Its basic idea is to obtain less noisy output data from a system, which will result in more accurate data.

The Kalman filter has provided innovations in the field of control systems [4]. First, in 1960 it was used for navigation purposes in the Apollo project. Recently it has been used in many other applications and has become common in a wide range of engineering systems [4] [5]. The strength of the Kalman filter that made it a practical tool in a vast array of applications lies in its ability to combine all available information, errors, and system dynamics [4].

This article seeks to present an understanding of the Kalman filter theory, combined with a real life application. Section II presents the background of the Kalman filter and the derivation of its equation, while Section III presents the Kalman filter algorithm, and Section V discusses the limitation and assumptions of the Kalman filter. Then, a real life application of the Kalman filter is performed in estimating the speed of a vehicle in front of a host vehicle. We conclude the paper with our summary and results.

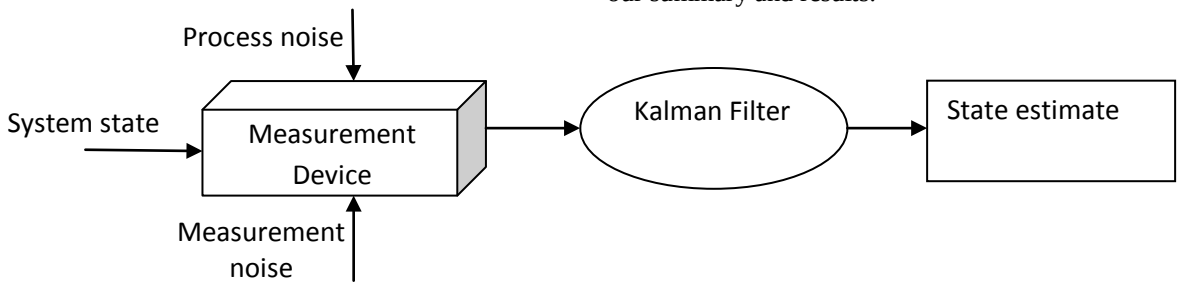


Fig. 1. General model of The Kalman filter role

2 The Theory Behind the Kalman Filter

The Kalman filter is built on some essential aspects [5], as shown in Fig (2). The main role of Kalman filter is to estimate the state of interest for a linear system.

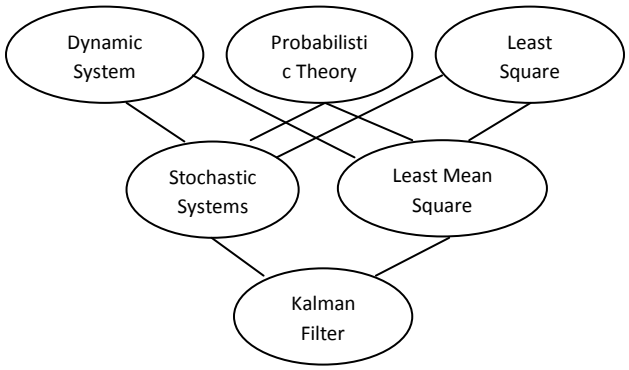


Fig. 2. The essential aspects of the Kalman filter

When the process and measurement models are considered as a discrete-time linear system, we can model the system using these equations [4]:

$$x_t = Ax_{t-1} + Bu_{t-1} + v_{t-1} \quad (1)$$
$$z_t = Hx_t + w_t \quad (2)$$

A: *n x n* matrix, which associates the system state at time t-1 to the system state at the time step t. Matrix *A* might be changed with each time step, but in most cases, it is possible to assume it as a constant.
B: *n x 1* matrix, relates the control input *u* at time t-1 to the system state at the current time step t.
H: m x n matrix, which relates the system state at time t to the observed measurement *z* at time t. Also, this matrix might change with each measurement. In some cases, depending on the system modeled, it is practical to assume that it is a constant.
v: Represents the process error, with a covariance *Q*.
w: Represents the measurement error, with a covariance *R*.
Kaman filter assumes that both, the process noise and the measurement noise are white noise with Gaussian distributions:

$$P(v) \approx N(0, Q)$$
$$P(w) \approx N(0, R)$$

At each time step there are two state estimates: a priori and a posteriori. Therefore, we need to define two error estimates [4]:

The a priori error state estimate is the a priori state estimate at time step t, based on the process at time step t-1.

$$e_t^- = x_t - x_t^- \quad (3)$$

The a posteriori error state estimate is the a posteriori state estimate at time step t, given the current measurement z at the same time step t.

$$e_t = x_t - x_t^+ \quad (4)$$

The general process of Kalman filter involves two stages, first make a prediction of the state, and then correct the estimated state based on the difference between that prediction and the observed measurement. This can be interpreted by the equation (5) [5],

$$x_t = x_t^- - K(z_t - z_t^-) \quad (5)$$

Where,
x_t^- : The a priori state estimate that was predicted in the time gap between t-1 and t and before observing the new measurement at time t. That is, it is the predicted state estimate.
x_t : The a posteriori state estimate that was updated after observing the new measurement at time t. That is, it is the updated/filtered state estimate (6).

$$z_t^- = H\, x_t^-$$
$$z_t = H\, x_t + w_t \quad (6)$$

Where, $E[v_t] = 0$, $E[w_t] = 0$, $E[v_t v_t^T] = Q_t$, $E[w_t w_t^T] = R_t$
K: The "Kalman Gain", was created to reduce the a posteriori error covariance. Typically, any gain matrix should be chosen to satisfy optimal conditions. One of these, the stochastic process is used by the Kalman filter based on the nature of the dynamics of the measurements. These conditions have been taken into account when deriving the Kalman gain, in such a way to minimize the square error of the estimated state of the system [5].

Based on the Kalman filter's assumptions, we start with the equation (5); then, by applying a limited number of math concepts, we obtain the Kalman filter equations [5]. First, to find the a posteriori error state estimate, we substitute the a posteriori error state estimate in (5):

$$e_t = e_t^- - K_t(H\, e_t^- + w_t) \quad (7)$$

The a priori estimate error covariance is given by:

$$P_t^- = E[e_t^- e_t^{-T}] \quad (8)$$

and the a posteriori estimate error covariance is given by:

$$P_t = E[\ e_t\, e_t^T\ ] \qquad (9)$$

From the linear model of the system with no noise, the a priori state estimate is of the form:

$$x_t^- = Ax_t + u_t \qquad (10)$$

And the a priori state estimate error is:

$$e_t^- = Ae_t + v_t \qquad (11)$$

To find the a priori state error covariance, substituting (11) in (8):

$$P_t^- = E[\ (Ae_t^- + v_t)(Ae_t^- + v_t)^T]$$
$$P_t^- = A_t P_t^- A_t^T + E[v_t v_t^T]$$
$$P_t^- = A_t P_t^- A_t^T + Q_t \qquad (12)$$

Substituting (4) in (9) we find the a posteriori state error covariance is:

$$P_t = E[\ (e_t^- - K_t(H\, e_t^- + w_t))(e_t^- - K_t(H\, e_t^- + w_t))^T]$$
$$P_t = P_t^- - H^T P_t^- K_t^T - HK_t P_t^- + K_t(HP_t^- H^T + R)K_t^T \qquad (13)$$

Where, K is the gain matrix, which is responsible for minimizing the a posteriori error covariance [4]. To obtain the optimal gain, we take the derivative of (13) and solve it for K, and we find the famous Kalman gain equation:

$$K_t = P_t^- H^T (HP_t^- H^T + R)^{-1} \qquad (14)$$

By substituting (14) in (13), we get the a posteriori state estimate error covariance:

$$P_t = P_t^- - K_t H P_t^- \qquad (15)$$

From the previous equations, we notice that the Kalman gain depends on the uncertainty in the state estimate, and depends inversely on the measurements' uncertainty [5]. The more accurate the observed measurement, the less measurement error covariance, R; this means that when R comes up to zero, then the observed measurement comes up to the precise value. So, the value of K is getting higher, and the correction would be based on the sensor measurement. On the other hand, when the value of R is increasing, and the state prediction is almost accurate, then K has no much effect on correcting the system state [6].

3 Kalman Filter Algorithm:

Basically, the Kalman filter theory is based on feedback control to make an estimation of the system state: At time t-1 the filter makes a prediction about the state at time t; thus, the measurements are fed into the filter as feedback to estimate the state at the next time step t+1 [7].

For each time step, the Kalman filter can be evolved in two stages: prediction stage (time update) and correction stage (measurement update) [4].

The prediction stage is responsible for projecting the a priori state estimate and the a priori state estimate error. This stage consists of two equations:

- State estimation:
$$x_t^- = Ax_{t-1} + Bu_{t-1}$$
- Error covariance estimation:
$$P_t^- = A_t P_t^- A_t^T + Q_t$$

The correction stage is responsible for combining the current observed measurement at time step t, with the a priori state estimate, in order to acquire the a posteriori state estimate. This stage consists of three equations:

- Kalman Gain calculation:
$$K_t = P_t^- H^T (HP_t^- H^T + R)^{-1}$$
- State estimation update:
$$x_t = x_t^- + K_t(z_t - Hx_t^-)$$
- Error covariance update :
$$P_t = (1 - AK_t H)P_t^-$$

A consummate representation of the Kalman filter procedure is given in Fig (3).

4 Kalman Filter Assumptions

The Kalman filter theory is formulated on some expectations which relate to the form of the estimator: It is both a linear estimator, and a recursive estimator. This requires the state variable to be Gaussian, and the sensor's output must be Gaussian, with the knowledge of the variances and covariance of sensor output [7].

The Kalman filter is only affective for the linear dynamic systems. There are many extensions to the Kalman filter theoy that can be used for non-linear systems [8] [9], such as, the Extended Kalman filter, which works only for systems that are not highly non-linear because it depends on a first order approximation. Another version of the Kalman filter is the Unscented Kalman filter which is more powerful for extremely non-linear systems.

While the Kalman filter is an optimal recursive estimator, this is only true for the linear case. A non-linear Kalman Filter cannot be proven to be optimal. The Kalmna filter requires that both the sensor input

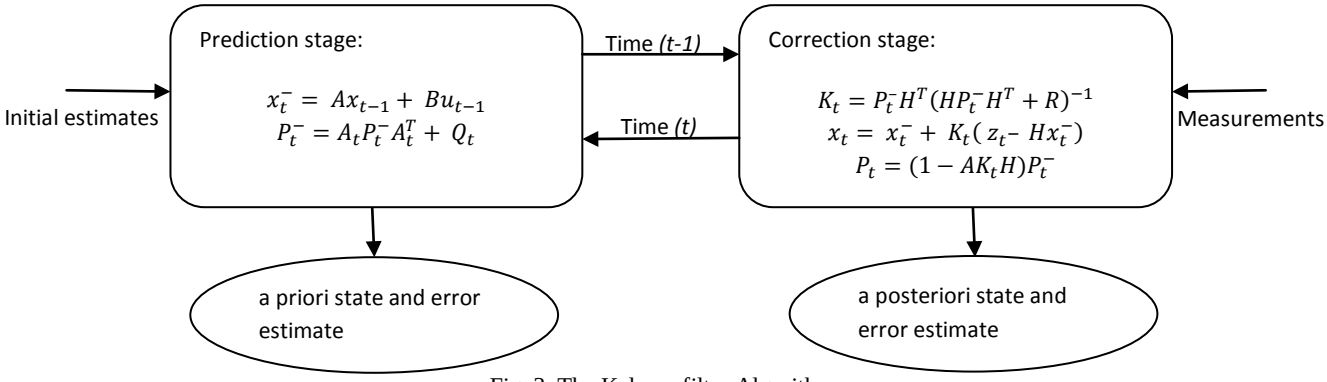


Fig. 3. The Kalman filter Algorithm

and the prediction of state variables have to be Gaussian. Otherwise, sometimes the filter will be unable to produce the correct estimates.

5 Kalman Filter Implementation for a Collision Avoidance System

The Kalman Filter applications in the real world are very wide. A real-life application that could be used in a vehicle's safety system is studied here. In our application, we measure the speed of the vehicle in front using two sensors mounted on a host vehicle. The sensor's measurement techniques do not vary in the same way because the sources of noise are unrelated, and the amount of noise is typical of a measurement system, as it is Gaussian [8] [9]. The solution to obtain the precise speed is by implementing the Kalman filter to provide a real time filtered information about the velocity of a preceding vehicle. The process of cancelling the noise of the signal and obtaining the true value of the measurement makes it possible for the host vehicle to adjust its speed in order to avoid any potential forward collision.

5.1 Integrating Two Speed Sensors Tracking a Moving Object with a Constant Speed:

We are using the Kalman filter to cancel the noise from the sensor readings, and approximate the true value of the velocity. We assume the vehicle is cruising at a specific velocity, where v=65mph. This is a simple case because we have two identical sensors measuring the speed of the vehicle in front. We want to integrate these two readings, trying to reduce the error and get closer to the actual speed.

In our model we don't need to consider control inputs, unless we're interested in characterizing the vehicle. In our particular case user input would be irrelevant, anyway, because even with a good vehicle dynamics model, in the real life, we cannot reliably estimate the vehicle speed based on acceleration/brake pedal position, due to transitory states, variable road/terrain, vehicle load conditions, etc. Our simulation is based on these preliminary values:

The standard deviation error for the first sensor is (sd1=1). The standard deviation error for the second sensor is (sd2=2). Also, we assume the process error is (Q=0.005), and the measurements are ready every (dt=0.1s), for a total time of (T=30s) in our simulation:

Then, we define the matrices:

$$Q = \begin{bmatrix} 0.005 & 0 \\ 0 & 0.005 \end{bmatrix}, R = \begin{bmatrix} sd1^2 & 0 \\ 0 & sd2^2 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, H = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

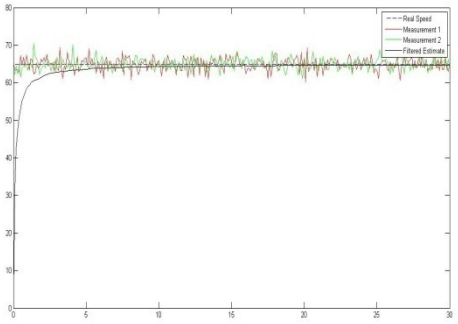


Fig. 4. The filtered estimate of a constant speed

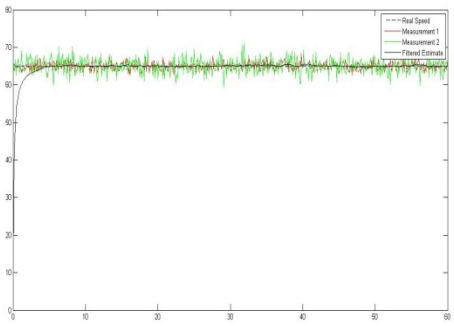


Fig. 5. The filtered estimate of a constant speed with more cycles

If we put the total time T=60 with the same time step dt=0.1, then we have the filtered estimate smoother. Fig (4) shows the two sensor measurements and the filtered estimate. The P matrix shows the variances of the state variables in its diagonals. When the variances of the state variables are then the state estimate is very close to the actual value. In general, as the Kalman filter performs more cycles, the variances of the state variables get smaller. That is, whatever the sensor output, the filtered estimate will get more and more confident of that speed estimation, which is represented in Fig (5), as we change the total time to (T=60). It shows that the kalman filter can reduce the error state estimate, and by that time, the state estimate of the system is getting closer to the actual value.

5.2 Integrating Two Speed Sensors Tracking a Moving Object with a Variable Speed:

Suppose we have two measurements of the same thing, say velocity in our application, and the preceding vehicle is driving without any cruise control (the speed is changing from 0 up to 60 mph). Then, we need to integrate these two readings to obtain the accurate speed by reducing error variances. We assume the model is exactly as in the previous application. Fig (6) shows two sensor measurements and the filtered estimate. We notice that the error rate of the speed estimation is greater than the error rate when measuring a constant speed. To improve the speed estimation value and reduce the error rate, it is better to design the system with a speed sensor and an accelerometer (instead of another speed sensor). Fig (7) shows the filtered speed estimate of the system modeled with a speed sensor and accelerometer.

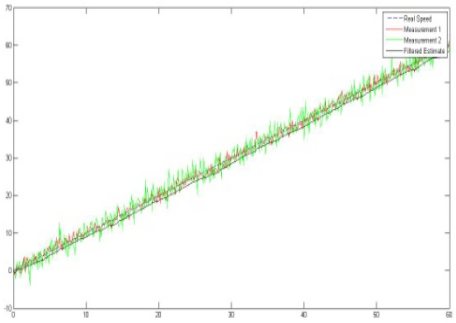


Fig. 6. The filtered estimate of two speed sensors

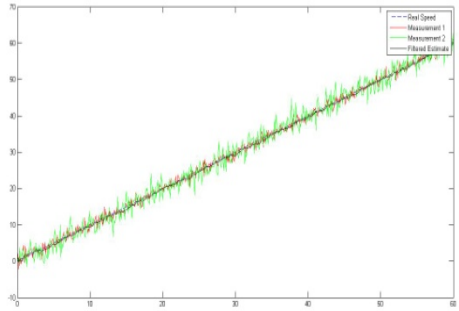


Fig. 7. The filtered estimate of a speed sensor and accelerometer

From Fig(6) and Fig(7), there is a trade-off when using the Kalman filter for slow varying, and fast varying processes. In the first example (cruise control), we can see the filter's output is very smooth. But if we vary the speed, then the filter's estimation will significantly lag behind. So, for time-varying processes we need to consider the derivative of the input to help predict the next measurement; in our case this derivative was the acceleration. We can notice that the filter's output in this case matches well with the actual speed, but it will not be as smooth as in the first example, when the speed was constant.
Sometimes the preceding vehicle velocity is changing up and down, as in the regular driving scheme; Fig(8) shows the filtered estimate of the preceding vehicle's speed. The filter is able to re-adapt to any changing in the system state.

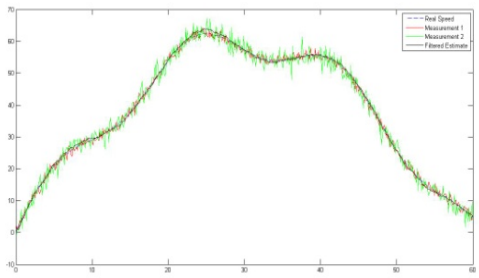


Fig. 8. The filtered estimate of a speed sensor and

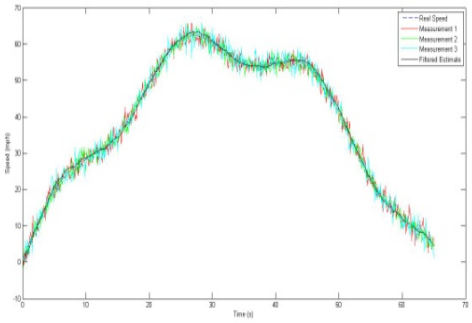


Fig. 9. The filtered estimate of a three speed sensor and accelerometer

5.3 Integrating Three Speed Sensors Tracking a Moving Object with a Variable Speed:

The last case discussed in this paper is using the Kalman filter to integrate three speed sensor readings, trying to obtain the actual speed estimate and reduce the error rate. The velocity is changing between 0 and 60 mph, and our system assumptions are:

$$A = \begin{bmatrix} 1 & dt & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, H = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

$$R = \begin{bmatrix} sd1\text{^}2 & 0 & 0 \\ 0 & sd2\text{^}2 & 0 \\ 0 & 0 & sd3\text{^}2 \end{bmatrix}$$

$$Q = \begin{bmatrix} 0.005 & 0 & 0 \\ 0 & 0.005 & 0 \\ 0 & 0 & 0.005 \end{bmatrix}$$

The speed estimate is depicted by fig(9). The error rate is almost the same when using two or more sensors measuring the same state of the system.

6 Conclusion:

This paper discussed the Kalman Filter theory that was promoted for linear discrete-time systems. While most of the filters are formulated in the frequency domain, the Kalman filter is a purely time domain filter. The assumptions and the limitations of Kalman theory have been studied and presented. Integrating sensor data through the Kalman filter is very easy to apply and significantly improves the estimation by reducing the effects of sensor noise and bias. The results show that the Kalamn filter is the optimal filter, but only under some restrictions. The filter divergence happens when the filter seems to perform well, having low error variance, but the estimate is far away from the actual value. This is due to errors in the system modeling or where the system has bias errors.

7 References:

[1] Kalman, R. E. 1960. "A New Approach to Linear Filtering and Prediction Problems," Transaction of the ASME—Journal of Basic Engineering, pp. 35-45 (March 1960).

[2] R. E. Kalman and R. S. Bucy. New results in linear filtering and prediction theory. Transactions of the ASME. Series D, Journal of Basic Engineering, 83:95–107, 1961.

[3] Maybeck, P. S. "The Kalman filter: An introduction to concepts." Autonomous Robot Vehicles. I. J. Cox and G. T. Wilfong. New York, Springer-Verlag: 194-204, 1990.

[4] Greg Welch, and Gary Bishop. "An Introduction to the Kalman Filter," UNC-Chapel Hill, TR 95-041, July 24, 2006

[5] J. Andrade-Cetto and A. Sanfeliu. Environment Learning for Indoor Mobile Robots. A Stochastic State Estimation Approach to Simultaneous Localization and Map Building, volume 23 of Springer Tracts in Advanced Robotics. Springer, 2006.

[6] Simon, D. " Kalman Filtering". Embedded Systems Programming, 14, 6, 72-79. 2001

[7] Garry Einicke (2012). Discrete-Time Minimum-Variance Prediction and Filtering, Smoothing, Filtering and Prediction - Estimating The Past, Present and Future, (Ed.), ISBN: 978-953-307-752-9, InTech, DOI: 10.5772/39252.

[8] Michael Athans. The Control Handbook, chapter Kalman Filtering, pages 589–594. CRC Press, 1996

[9] Terejanu, G.A., Discrete Kalman filter tutorial, University at Buffalo Department of Computer Science and Engineering, Buffalo, NY 14260

A finite difference-extrapolation method for solving ordinary differential equations

Adel N. Boules
Department of Mathematics and Statistics
University of North Florida
1 UNF Drive
Jacksonville, Fl. 32224

ABSTRACT

This paper presents an algorithm for the numerical approximation of initial value problems for ordinary differential equations. It rests on approximating the second derivative by a central difference and extends previous work by the author, and overcomes the limitations that existed in previously published papers. Extrapolation is then used to improve the accuracy of the approximation. The included numerical examples illustrate the efficiency of the method

Keywords: Initial value problems; ordinary differential equations; finite differences; extrapolation; **variable step methods**

1. Background, scope and organization

This work is an extension of previously published papers by the author. In [1] an ODE solver was developed based on finite difference approximations and knowledge of the values of higher order derivatives of the solution function obtained from the ODE by repeated differentiation, much in the same way as in Taylor methods. The number of the finite difference approximations was dependent on the order of the equation. The method was later extended in [2] where the need for computing higher order derivatives as well as the need for complicated finite difference formulas were eliminated. However, the scheme developed in [2] applies only to higher order equations with no obvious way to extend it to include first order systems. Both schemes allowed for step variability and used local extrapolation to improve the order of the method and to provide a mechanism for error estimation and step size determination. This feature is retained on this paper.

In this work we develop the ideas further and derive a method for first order systems of equations, thus vastly extending the applicability of the method. The core idea in this paper is to approximate the second derivative of the solution function using a finite difference formula, then using the quadratic approximation to advance the solution, as will be presented in the next section. The numerical examples included in this paper show that the scheme is extremely accurate and robust.

Section 2 describes the basic scheme for advancing the solution one step, without extrapolation. Section 3 contains a brief description of the use of extrapolation to improve the order of the method and its role in error estimation and the choice of step size, as developed in [1,2]. Section 4 contains numerical examples that demonstrate the accuracy and efficiency of the proposed scheme. The last section contains an outline of the stability of the method.

2. The Basic scheme

In this section we describe the finite difference approximation that allows us to advance the solution only one step, without extrapolation. we first state the scheme and explain the motivation behind it later.

Consider the initial value problem

$$\begin{aligned} y'(x) &= f(x,y), \quad x_0 \le x \le b \\ y(x_0) &= y_0 \end{aligned} \tag{1}$$

Where y is a vector function of x.

$$k_1 = y_0' = f(x_0, y_0) \tag{2}$$

$$y_h = y_0 + \alpha h y_0' \tag{3}$$

$$y_{-h} = y_0 - h\alpha y_0' \tag{4}$$

$$k_2 = y_h' = f(x_0 + \alpha h, y_h) \tag{5}$$

$$k_3 = y_{-h}' = f(x_0 - \alpha h, y_{-h}) \tag{6}$$

$$y(x+h) \approx y_1 = y_0 + h\left[k_1 + \frac{1}{4\alpha}(k_2 - k_3) \right] \tag{7}$$

Equation (2) uses the differential equation to compute (an approximation of) the first derivative. Since equation (2) is exact, the only error committed in using it to approximate y_0' is of the same order of error contained in y_0. The motivation behind the rest of the scheme is rooted in the need for approximating the second derivative of the solution function y. Equations (3) and (4) provide linear approximations of $y(x+\alpha h)$ and $y(x-\alpha h)$, respectively, where $0 < \alpha \le 1$. Equations (5) and (6) give approximations of the first derivatives $y'(x+\alpha h)$ and $y'(x-\alpha h)$ respectively. The second derivative can be approximated by the central difference $y_0'' = \frac{y_h' - y_{-h}'}{2\alpha h}$. Now, using the second degree Taylor polynomial, we approximate $y(x_0 + h)$ by $y_0 + h y_0' + \frac{h^2}{2} y_0''$. Substituting the above finite difference approximation of y_0'' and the approximation of the first derivative from equation (2), one arrives at equation (7). The central difference approximation we used (see above) has $O(h^2)$ accuracy. However since the truncation error in equations (3) and (4) is $O(h^2)$, the approximation $y_0'' = \frac{y_h' - y_{-h}'}{2\alpha h}$ is accurate of order only $O(h)$. This is enough for our purpose since when y_0'' is multiplied by h^2, the total error is of $O(h^3)$, consistent with the error in the second degree Taylor approximation of y. One can clearly choose any value of α between 0 and 1. This is because we have the freedom of approximating the derivatives (equations 3-6) at any point between x_0 and $x_0 + h$. One can very well use $\alpha = 1$. However, $\alpha = 1/8$ was used and produces a slight improvement in performance.

The approximation described above has a local truncation error of $O(h^3)$, which is clearly a low order approximation. Extrapolation is used to improve the order in the same way developed in [1] and [2],

which will be briefly outlined in the next section. The interested reader can see the details in the cited references.

3. Extrapolation, error estimation and step acceptance

Let $\bar{\bar{y}}_1 = \bar{y}(h)$, $\bar{\bar{y}}_2 = \bar{\bar{y}}(h/2)$, $\bar{\bar{y}}_3 = \bar{y}(h/4)$ denote, respectively the approximations obtained by applying the current scheme to the initial value problem (1), in one, two and four steps. The second level of extrapolation, namely $\bar{\bar{y}}_{12}=(4\bar{\bar{y}}_2-\bar{\bar{y}}_1)/3$, $\bar{\bar{y}}_{23}=(4\bar{\bar{y}}_3-\bar{\bar{y}}_2)/3$, produce approximations accurate of $O(h^4)$, and the final approximation in a 3-level local extrapolation scheme $\bar{\bar{y}}_{123}=(8\bar{\bar{y}}_{23}-\bar{\bar{y}}_{12})/4$, produces an approximation where the local truncation error is of $O(h^5)$.

We use $e\equiv\left|\bar{\bar{y}}_{123}-\bar{\bar{y}}_{23}\right|$ as an estimate of the error in $\bar{\bar{y}}_{23}$, although we use $\bar{\bar{y}}_{123}$ to advance the solution. This is customary practice in the implementation of imbedded methods (see [5]).

The criterion for accepting the step is as follows:

$$h\ \max\left|\frac{e(k)}{\max\left\{1,\left|\bar{\bar{y}}\ (k)\right|\right\}}\right|<\varepsilon$$

Where $\bar{\bar{y}}$ the approximation in the last accepted step and ε is a small tolerance chosen by the user. In the examples included in this paper we used $\varepsilon=10^{-13}$. If a step size is rejected twice in a row, the potential step size h is halved and $\bar{\bar{y}}_{123}$ is recomputed until the criterion is met.

The maximum in the criterion runs over all the components of the solution vector. Therefore the criterion for accepting a step size is quite stringent.

4. Examples

We believe the examples below provide ample evidence that the method is both accurate and economical. It must be emphasized, however, that we do not claim that the examples provide a rigorous performance study, but rather strong evidence of the viability and robustness of the scheme presented in this paper.

Example 1
Consider the logistic equation

$\frac{dP}{dt}=k\,P(P_\infty-P)$, $P(0)=P_0$, where k, P_0 and P_∞ are positive constants.

The equation has a stable equilibrium solution at P_∞, thus $\lim_{t\to\infty}P(t)=P_\infty$ for all solutions with $P_0>0$.

The exact solution of the above IVP is $P(t)=\frac{P_0\,P_\infty}{P_0+(P_\infty-P_0)\exp(-k\,P_\infty\,t)}$.

By choosing k and P_∞ such that kP_∞ is large, one can construct exact solutions that approach the limiting population (P_∞) very rapidly. In this example we use $P_0 = 2$, $k = 2^{-10}$, $P_\infty = 2^{14}$. The graph of the exact solution is shown in Figure 1. With these parameters, the maximum value of the first derivative of the exact solution is 65536, which is rather challenging to capture numerically.

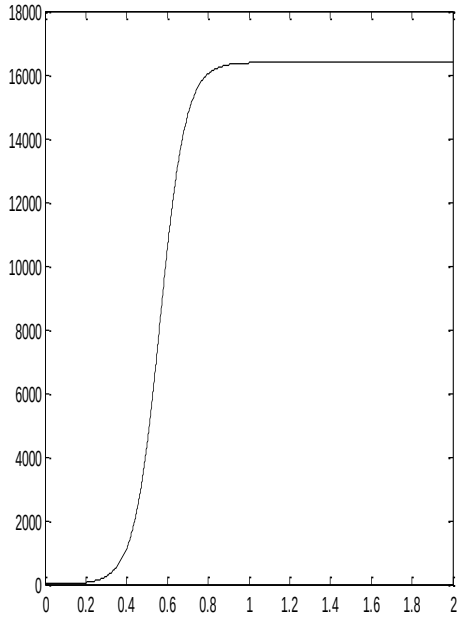


Figure1.
The exact solution

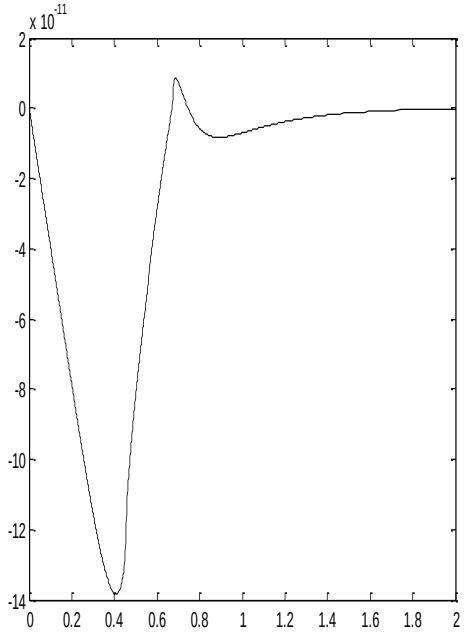


Figure2.
The relative error in the computed solution

The current scheme was implemented with three levels of extrapolation and the numerical approximation was computed. It took 942 successful steps and 7 failed steps to compute the solution. Figure 2 shows the relative error in the computed solution, $(P_{computed} - P_{exact}) / P_{exact}$. The approximation is clearly extremely accurate.

Example 2
Consider the van der Pol equation

$$\frac{dy_1}{dt} = y_2$$

$$\frac{dy_2}{dt} = -y_1 + \mu\, y_2 (1 - y_1^2)$$

$$y_1(0) = 2,\ y_2(0) = 0$$

The equation has a unique stable limit cycle and all other solutions except (0,0) converge to the limit cycle. Approximating the solution of the van der Pol equation is a problem of considerable computational difficulty, especially for large values of μ. This equation was considered in [3] and we use it here for comparison purposes.

The current numerical scheme is applied to the problem with $\mu = 300$ and the solution is computed on the interval $0 \le t \le 1500$. The same problem was solved using an implementation of the scheme in [4] and good care was exercised in making the implementations as close as possible to guarantee a reasonable basis for comparison. Thus the step size determination and error control mechanisms follow those described in [2].The scheme in [4] is the basis for the MATLAB function ode45 [5]. Table 1 shows a comparison of the results delivered by the current scheme with four extrapolation levels vs. the scheme described in [4], abbreviated DORPRI. The current scheme requires less than one-third the number of steps although it has a slightly elevated (but still small) number of failed steps. It is clear that the current scheme is quite competitive. Figure 3 shows the tip of the solution component y_2 near t=1455.2581.

	Number of successful steps	Number of failed steps	CPU time
Current scheme	118, 433	119	7.5 sec
DORPRI	381,317	32	9.8 sec

Table 1. Comparison statistics

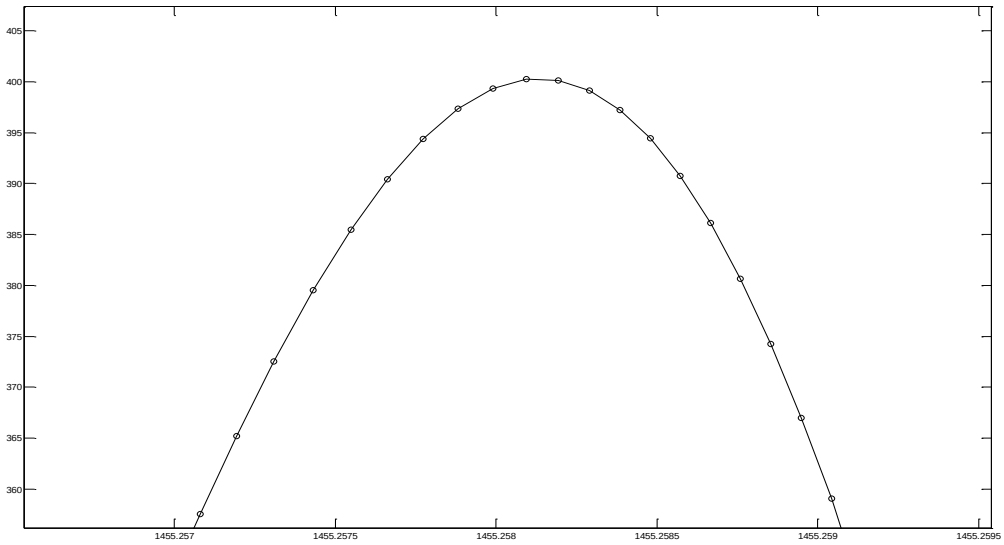


Figure3. The computed solution component, y_2. DORPRI _____ , current scheme *ooo*

Finally, MATLAB's ode45 was used to compute the solution at the same sequence of time steps generated by the current scheme. The maximum difference between the two approximations for the solution components (y_1 and y_2) is 1.5×10^{-5} and 4.8×10^{-3}, respectively. Figure 5 shows the plot of the following quantity for the second component y_2 : $Q = \dfrac{\left| y_{current} - y_{45} \right|}{\max\left(1, \left| y_{current} \right|\right)}$. Thus the absolute difference is used when $\left| y_{current} \right| \le 1$, otherwise Q measures the relative difference between the two approximations. The difference between the two computed solutions is clearly miniscule.

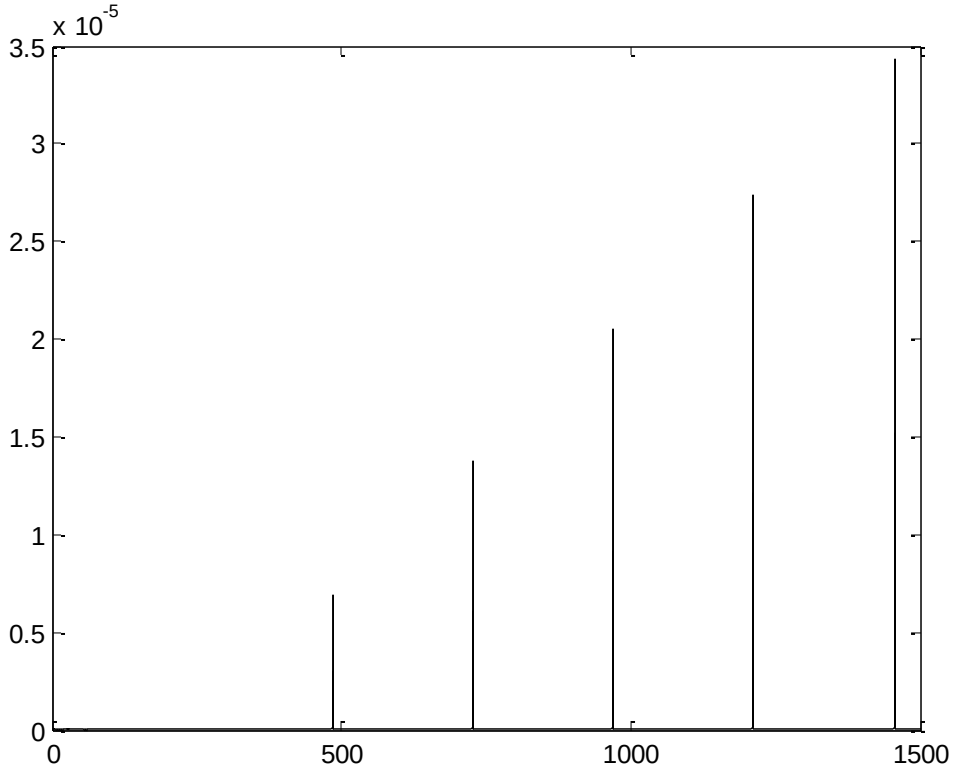


Figure 4. The difference between the current scheme and ode45

5. Stability

We follow the classical definition of stability to study the stability of the method. Let y_1 be the approximation of $y_1 = y(h)$ obtained by applying the scheme to the initial value problem $y' = \lambda y$, $y(0) = y_0$ It is quite straightforward to verify that the application of the scheme in section 2 to this problem yields $y_1 = (1 + z + z^2/2) y_0$

This is the same result obtained for the algorithm previously studied in [2]. The stability region for the current method is therefore identical to that in [2]. For the purpose of self containment, we reproduce the plot of the stability regions of the 3- and 4-level extrapolation schemes as well as the stability region for the classical RK method of order 4 for comparison. The interested reader can consult [2], where more details can be found, including the stability polynomial of the 3-step extrapolation method.

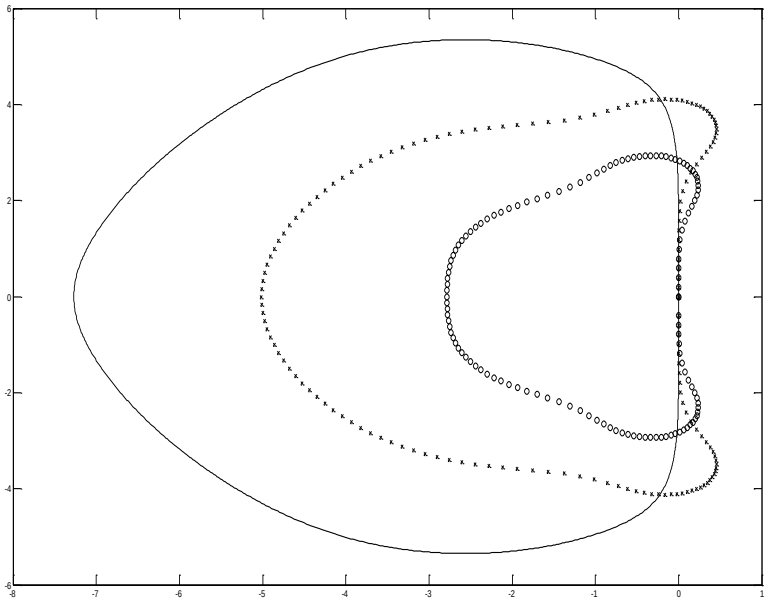


Figure 5. The stability regions. Current (4 levels) ____ , Current (3 level) *xxx* , RK4 *ooo*

REFERENCES

[1] Adel N. Boules, A variable step scheme for solving ordinary differential equations. Proceedings of the 2008 *International Conference on Scientific Computing*, 2007 Las Vegas: 361-367

[2] Adel N. Boules, A new algorithm for the approximation of initial value problems. *International Journal of Mathematics and Computation*, 2012 Vol. 16, Issue No. 3

[3] J.C. Butcher, Numerical methods for ordinary differential equations, 2008, J. Wiley

[4] Dormand, J. R.; Prince, P. J., "A family of embedded Runge-Kutta formulae", *Journal of Computational and Applied Mathematics*, 1980 **6** (1): 19–26

[5] MATLAB version 7. *The Mathworks*

Quasi-TEM Analysis of Multilayer Coplanar Waveguide Broadside Coupled Lines Balun

Sarhan M. Musa and Matthew N. O. Sadiku,
Roy G. Perry College of Engineering, Prairie View A&M University
Prairie View, TX, USA

Abstract- **The accurate estimates of values of electromagnetic parameters are essential to determine the final circuit speeds and functionality for designing of high-performance integrated circuits and integrated circuits packaging. In this paper, the quasi-TEM analyses of multilayer coplanar waveguide (CPW) broadside coupled-line balun are successfully demonstrated using Finite Element Method (FEM). We specifically determine the capacitance and inductance matrices and the quasi-static spectral for the potential distribution of the developed integrated circuit.**

Keywords- finite element method; capacitance;, inductance; IC interconnect; CPW transmission lines

I. INTRODUCTION

The designing of fast electronics circuits and systems with increase of the integration density of integrated circuits has led to wide use and cautious analysis of CPW broadside coupled-line balun. For example, a triple coupled CPW can be used for microwave applications as couplers for combining two independent signals [1] and as basis building blocks [2, 3]. The matrices of capacitances per unit length of CPW transmission line are known as the essential parameters in designing of package, lossless transmission line system, microwave circuits, printed circuit board (PCB), multichip modules (MCM) design and high speed very large scale integration (VLSI) circuits. Therefore, the improvement of accurate and efficient computational method to analyze quasi-TEM transmission lines structure becomes an important area of interest. Also, to optimize the electrical properties of the integrated circuits, the estimate of the capacitance matrix of multilayer and multiconductor interconnects in VLSI circuit must be investigated. The computational values of self and coupling capacitance can also help engineers and designers to optimize the layout of the circuit [4]. There are previous attempts at the problem. These include using the conformal mapping method [5], the spectral domain method [6], the potential integral formalization method [7].

In this work, we design electrostatic model CPW broadside coupled-line balun using the FEM. Many industrial applications depend on different interrelated properties or natural phenomena and require multiphysics modeling and simulation as an efficient method to solve their engineering problems. Moreover, superior simulations of microwave integrated circuit applications will lead to more cost-efficiency throughout the development process. In this article, we specifically calculate the capacitance and inductance matrices and the potential distribution of the configuration.

II. RESULTS AND DISCUSSIONS

The models are designed in two-dimensional (2D) using electrostatic environment in order to compare our results with some of the other available methods. In the boundary condition of the model's design, we use ground boundary which is zero potential ($V=0$) for the shield. We use port condition for the conductors to force the potential or current to one or zero depending on the setting. Also, we use continuity boundary condition between the conductors and between the conductors and left and right grounds.

The quasi-static models are computed in form of electromagnetic simulations using partial differential equations. For coupled multiconductor transmission lines, it is convenient to write:

$$Q_i = \sum_{j=1}^{m} C_{sij} V_j \quad (\text{i} = 1, 2, \ldots, \text{m}) \ , \qquad (1)$$

where Q_i is the charge per unit length, V_j is the voltage of j th conductor with reference to the ground plane, C_{sij} is the short circuit capacitance between i th conductor and j th conductor. The short circuit capacitances can be obtained either from measurement or from numerical computation. From the short circuit capacitances, we obtain

$$C_{ii} = \sum_{j=1}^{m} C_{sij} \qquad (2)$$

where C_{ii} is the capacitance per unit length between the i th conductor and the ground plane. Also,

$$C_{ij} = -C_{sij}, \qquad j \neq i \qquad (3)$$

where C_{ij} is the coupling capacitance per unit length between the i th conductor and j th conductor. The coupling capacitances are illustrated in Fig. 1.

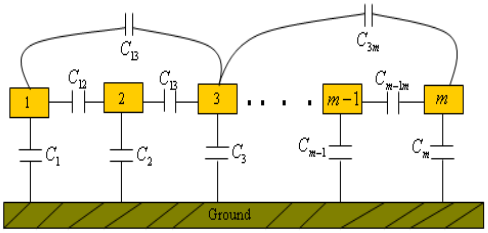


Fig. 1. The per-unit length capacitances of a general m -conductor transmission line.

For m -strip line, the per-unit-length capacitance matrix [C] is given by

$$[C] = \begin{bmatrix} C_{11} & -C_{12} & \text{L} & -C_{1m} \\ -C_{21} & C_{22} & \text{L} & -C_{2m} \\ \text{M} & \text{M} & & \text{M} \\ -C_{m1} & -C_{m2} & \text{L} & C_{mm} \end{bmatrix} \qquad (4)$$

For a triple coupled CPW lines, the capacitance matrix can be defined as:

$$\begin{bmatrix} Q_1 \\ Q_2 \\ Q_3 \end{bmatrix} = \begin{bmatrix} C_{11} & -C_{12} & -C_{13} \\ -C_{12} & C_{22} & -C_{12} \\ -C_{13} & -C_{12} & C_{11} \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \\ V_3 \end{bmatrix} \qquad (5)$$

In any electromagnetic field analysis, the placement of far-field boundary is an important concern, especially when dealing with open solution regions. It is necessary to take into account that the natural boundary of a line at an infinity and presence of remote objects and their potential influence on the field [8]. In all our simulations, the open models are surrounded by a $W \times H$ shield, where W is the width and H is the thickness.

We illustrate in this article the modeling of multilayer CPW broadside coupled lines balun which is recently developed by the authors. Balun is a device which converts balanced to unbalanced transmission lines that join balanced structures and unbalanced structures [9]. Indeed, multilayer CPW broadside coupled lines balun is widely applied on microwave integrated circuit of wireless communication systems. Therefore, we focus here on the calculation of self and mutual (coupling) capacitances per unit length and determine the quasi-TEM spectral for the potential distribution of the model.

In Fig. 2, we show the cross-section of the developed multilayer CPW broadside coupled lines balun. The geometry of the model has the following parameters values:

ε_r = dielectric constant = 4.4;

w_1 = width of the lower middle conductor = 1.8 mm;

w_2 = width of the upper middle conductor = 1 mm;

w_3 = width of the upper corners conductors = 19.3 mm;

w_4 = width of the lower corners conductors = 18.9 mm;

t = thickness of the conductors = 0.01mm;

h_1 = height of the lower conductors from the ground = 1.6 mm;

h_2 = height of the upper conductors from the ground = 2 mm;

$s_1 = s_2 = 0.2$mm;

The geometry is enclosed by a 40×10 mm shield.

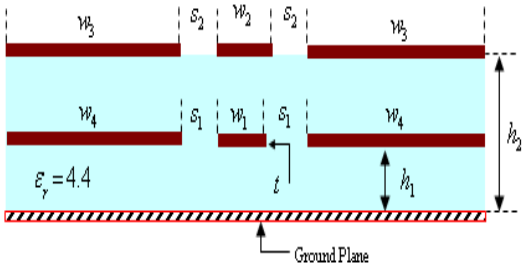


Fig. 2. Cross section of multilayer CPW broadside coupled-line balun.

From the model, we generate the finite element mesh plot as in Figure 3. Table I shows the statistical properties of the model mesh.

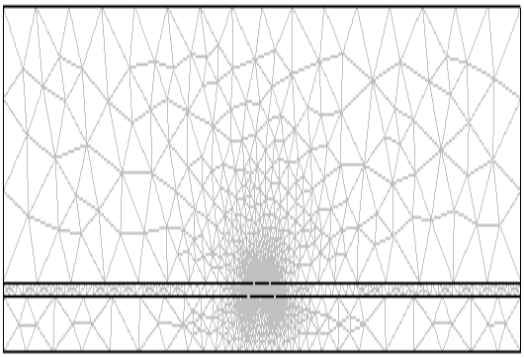


Fig. 3. Mesh plot of multilayer CPW broadside coupled-line balun.

Table I. Mesh statistics of the multilayer CPW broadside coupled-line balun

Items	Value
Number of degrees of freedom	12059
Total number of mesh points	2907
Total number of elements	5388
Triangular elements	5388
Quadrilateral elements	0
Boundary elements	428
Vertex elements	28

Figure 4 shows the 2D surface for electrical potential (V) distribution of the transmission lines, while the contour of electric potential (V) plot of the model is presented in Figure 5.

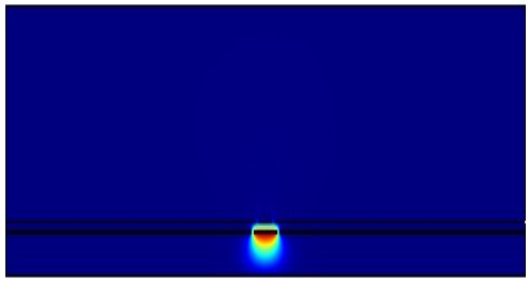


Fig. 4. 2D surface electrical potential distribution of the multilayer CPW broadside coupled-line balun.

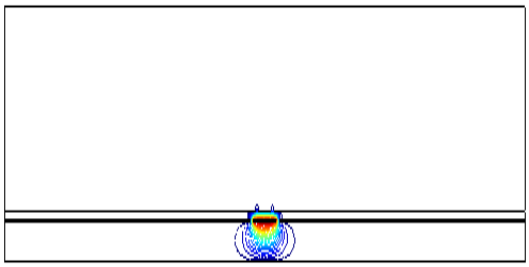


Fig. 5. Contour plot of the multilayer CPW broadside coupled-line balun.

Figure 6 shows the electric potential plot as a function of arc-length of the model. Fig. 7 shows the comparison analysis of potential distribution of the model with and without dielectric substrate. It observed that the peak value of electric potential approximately stays the same as the dielectric is placed in the substrate.

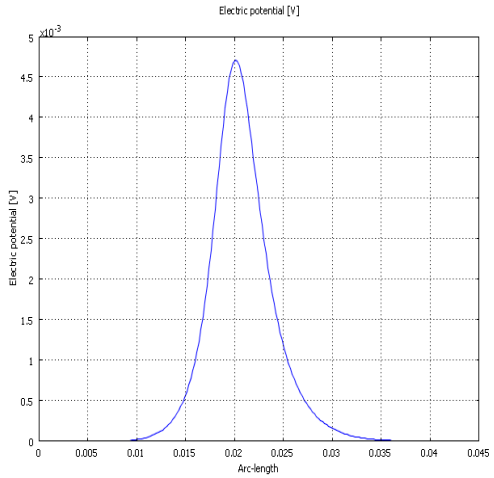


Fig. 6. Potential distribution of multilayer CPW broadside coupled-line balun from (x,y) = (0,0) to (x,y) = (40, 10) mm.

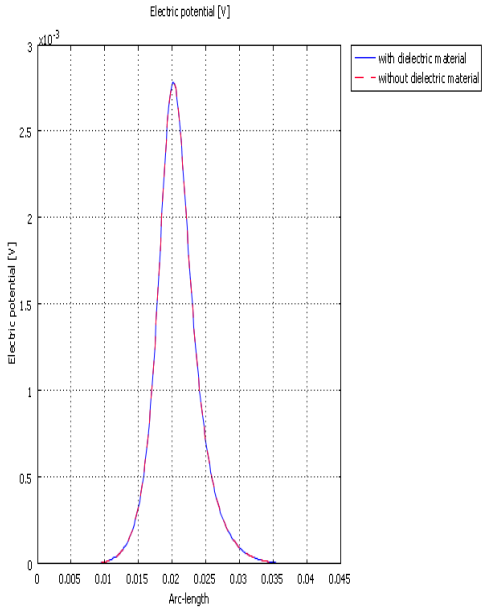


Fig. 7. Comparison analysis of potential distribution of the model with and without dielectric substrate.

The following electrical parameters, capacitance per unit length matrix ($[C]$ in pF/m) and inductance per unit length ($[L]$ μ H/m) are found as:

$$[C] = \begin{bmatrix} 325.9 & -115.5 \\ -115.5 & 159.5 \end{bmatrix} pF/m .$$

The inductance and capacitance per unit length of multiconductor transmission lines are related as follows:

$$[L] = \mu_o \varepsilon_o [C_o]^{-1}, \qquad (5)$$

where,

$[L]$ = Inductance matrix.

$[C_o]^{-1}$ = the inverse matrix of the capacitance of the multiconductor transmission line when all dielectric constants are set equal to 1.

μ_o = permeability of free space or vacuum.

ε_o = permittivity of free space or vacuum.

$$[L] = \begin{bmatrix} 0.1824 & 0.0944 \\ 0.0944 & 0.2661 \end{bmatrix} \mu H/m .$$

III. CONCLUTIONS

This paper has successfully demonstrated the use of the FEM method to solve open-region electrostatic problems involving 2-D model of multilayer Coplanar Waveguide (CPW) broadside coupled-line balun system. We computed the capacitance per-unit length and inductance per unit length matrices of the model. Also, we identified the quasi-static spectral for the potential distribution of the developed integrated circuit. The results obtained in this research are encouraging and motivating for further study.

REFERENCES

[1]. D. Pavlidis and H. L. Hartnagel, "The design and performance of three-line microstrip couplers," *IEEE Transactions on. Microwave Theory Techniques*, vol. 24, no. 10, pp. 631-640, 1976.

[2]. F. Mernyei, I Aoki, and H. Matsuura, "New filter element for MMICs: triple coupled CPW lines," *Electronics*

Letters, vol. 30, no. 25, pp. 2150-2151, 1994.

[3]. C. Nguyei and K. Chang, "On the analysis and design of spurline bandstop filters," *IEEE Transactions on. Microwave Theory Techniques*, vol. 33, no. 12, pp. 1416-1421, 1985.

[4]. C. Seguinot, E. Paleczny, F. Huret, J. F. Legier, and P. Kennis, "Experimental determination of the characteristic impedance matrix of multiconductor quasi-TEM lines," *Microwave Theory and Optical Technology Letters,* vol. 22, no. 6, pp. 429-431, 1999.

[5]. K. –K M. Cheng, "Characteristic parameters of symmetrical triple coupled CPW lines," *Electronics Letters*, vol. 33, no. 8, pp. 685-687, 1997.

[6]. R. Schwindt and C. Nguyen, "Spectral analysis of three symmetric coupled lines and application to a new bandpass filter," *IEEE Transactions on. Microwave Theory Techniques*, vol. 24, no. 10, pp. 631-640, 1976.

[7]. H. Ymeri, B. Nauwelaers, K. Maex, and D. De Roest, "A new approach for the calculation of line capacitances of two-layer IC interconnects," *Microwave Theory and Optical Technology Letters,* vol. 27, no. 5, pp. 297-302 ,2000.

[8]. Y. R. Cruten, G. Molinari, and G. Rubinacci (Eds.), *Industrial application of Electromagnetic Computer Codes*, Kluwer, Norwell, MA, p. 5 (1990).

[9]. N. Marchand, "Transmission line conversion transformers," *Electronics*, vol. 17, pp. 142-146, 1944.

SESSION

NOVEL SCIENTIFIC AND ENGINEERING ALGORITHMS AND APPLICATIONS

Chair(s)

TBA

Numerical Stabilisation of the Lattice Boltzmann method for higher Reynolds number fluid

N. Maquignon[1], **J. Duchateau**[1], **G. Roussel**[1], **F. Rousselle**[1], **Christophe Renaud**[1]
[1]LISIC - ULCO, 50 rue Ferdinand Buisson, 62228 Calais Cedex, France
INNOCOLD, 145 avenue Maurice Schumann - MREI 1, 59140 Dunkerque, France

Abstract— *In this article, schemes of numerical stabilisation for a thermal multiphase Lattice Boltzmann Method are presented. As the model used is fully described in [6], it has been found unworth to re-introduce it in this work. In this paper, we give techniques to widen its compatibility range. Application to real thermal flow using LBM is usually limited to low value for Reynolds and Péclet numbers if the simulation is to be completed with reasonable computational ressources. Furthermore, the Prandtl number has to be kept close to unity. Similar problems occur when components have different viscosities. A new method that uses a per-component grid with two seperate lattices for dynamics and thermal equations is introduced. Different grid resolutions should be used when components have important differences in viscosity, as well as when a component has higher Prandtl number. In order to keep reasonable need in computational resources, dynamic mesh refinement is used, and validated in the case of the increase of Reynolds number for isothermal case. The effect of the improved lattice on numerical stability and accuracy of the results are discussed. The decrease in need for computational resources is shown. A comparison between multiple relaxation scheme and single relaxation is shown, in terms of atteignable Reynolds number.*

Keywords: Lattice Boltzmann method, mesh refinement, phase transition, multi-component, heat transfer.

1. Introduction

Understanding the behavior of flows containing multiple components in different fluid states has become essential for improvement of many industrial processes and environmental modeling. The lattice Boltzmann method (LBM) is a great candidate for the simulation of such phenomenons, because its mesoscopic nature has the ability to incorporate multi-physics. The usual LBM for multiphase flow are often made consistent with thermodynamics and mechanical properties as surface tension are adjusted sometimes very close to theoritical values [8] [4] [5]. But applications to real cases is complicated because the range of compatibilty for physical properties is often too restrained. Furthermore, phase change is rarely evaluated for thermal cases, and heat transfer isn't coupled to mass transfer. In this work, we introduce a new lattice Boltzmann method that is able to simulate a thermal flow with multiple components, phase transition and heat transfer. We introduce static and dynamic mesh refinement to allow the increase of Reynolds, Péclet, and Prandtl numbers as well as viscosity ratio. These criteria are important for medium or large scale real simulations and determine the computational ressources needed. Coarse grids usually cannot simulate high Reynolds or Péclet numbers flows with reasonable spatial resolution without the use of LES or RANS filtering methods, as the Smagorinsky subgrid model [7]. Such filtering doesn't exist yet for multiphase flows. The subgrid can be used for the BGK part of the Boltzmann equation but instabilities coming from the forcing term aren't easily decreased. The present work shows that the use of multiphase model implies a cost in numerical stability that decreases compatibility. A solution would be to increase resolution but it indeed has a severe impact on computational time. In this work, an adaptive mesh allows to increase numerical stability while keeping reasonable computational needs.

As previously mentioned, compatibility range often restrains applications to a few cases, because of numerical instabilities. The different models should be applied on meshes of finer resolution. To the best of our knowledge, only the work of Fan & Yu [10] describes an adaptive mesh refinement for monocomponent isothermal case with two phases using traditional SC model. This model still needs an extension to real thermodynamics with temperature transport equation. Furthermore, it does not take into account the presence of multiple components.

2. Thermal multi-component lattice Boltzmann method with heat transfer and phase change

As described in [6], the new coupled model is of the following form :

$$f_{\sigma,i}(x + e_i\delta t, t + \delta t) - f_{\sigma,i}(x,t) = -\frac{f_{\sigma,i}(x,t) - f^{eq}_{\sigma,i}(x,t)}{\tau_{\sigma,f}} + \Delta f_{\sigma,i} \quad (1)$$

$$g_{\sigma,i}(x+e_i\delta t, t+\delta t) - g_{\sigma,i}(x,t) = -\frac{g_{\sigma,i}(x,t) - g^{eq}_{\sigma,i}(x,t)}{\tau_{\sigma,g}} + \omega_i \Phi_\sigma \delta t \quad (2)$$

A set of equation for every component σ is necessary. The density of component σ is :

$$\sum_i f_{\sigma i}(x,t) = \rho_\sigma(x,t) \quad (3)$$

Speed of component σ is :

$$\sum_i f_{\sigma i}(x,t) e_i = \rho_\sigma(x,t) u_\sigma(x,t) \quad (4)$$

The temperature of component σ is :

$$\sum_i g_{\sigma i}(x,t) = T_\sigma(x,t) \quad (5)$$

A composite temperature is necessary to model the whole fluid temperature :

$$T(x,t) = \frac{\sum_\sigma \frac{\rho_\sigma C p_\sigma T_\sigma}{\tau_{\sigma g}}}{\sum_\sigma \frac{\rho_\sigma C p_\sigma}{\tau_{\sigma g}}} \quad (6)$$

The associated algorithm and details are fully described in [6].

2.1 3D visualisation

We present here a result of a simulation that allows the separation of a fluid consisting of an assembly of two components. The simulation domain consists of a lightweight component and a heavy component. The domain initialization is done as in the algorithm presented above. The purpose of the simulation is to highlight the efficiency of the algorithm for processing multi-component simulations. In our case, we present a simulation with two components but the algorithm remains valid for a larger number of components.

We can see on figure 1 the initialization corresponding to the mixture with two components. The heavy component (blue) and the light component (yellow) will gradually separate throughout the simulation. Figure 2 allows us to observe this phenomenon of separation. We can see the simulation images after 1000, 2000, 3000 and 3500 iterations. It is clear that the separation is more obvious with a significant number of iterations.

The visualization of the simulation was performed using the VTK (Visualization Toolkit) library. VTK is an open-source, freely available software system for 3D computer graphics, image processing and visualization. VTK supports a wide variety of visualization algorithms including: scalar, vector, tensor, texture, and volumetric methods. In our case, we opted for a display of the scalar field density using a texture mapping method to visualize the global 3D domain.

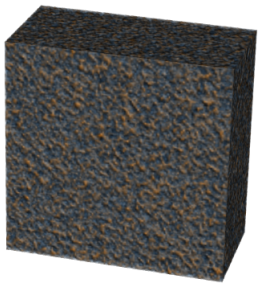


Fig. 1: Condensation of heavy component within light component - Initialisation of domain.

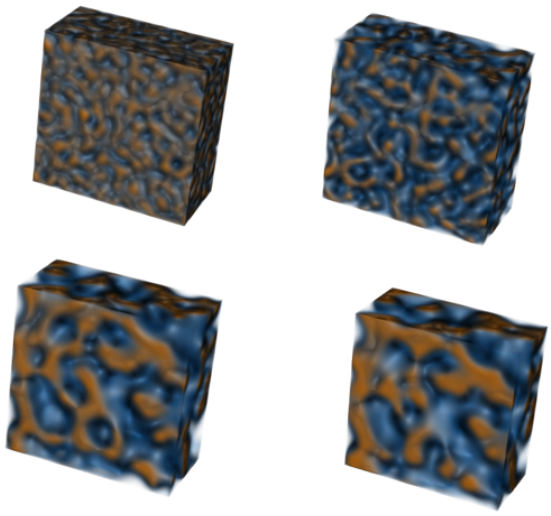


Fig. 2: Condensation of heavy component within light component.

3. Mesh refinement

3.1 Static mesh refinement

3.1.1 Increase of atteignable Reynolds or Péclet numbers using static mesh refinement

The model described in the previous sections needs improvement of numerical stability to be applied to real cases. Real flows can have high Reynolds or Péclet numbers, which determine the need of mesh refinement. Equation (8) and (7) show that for a given value of caracterictic speed U and length L, higher Reynolds or Péclet numbers require finer mesh size or smaller relaxation parameter. Assuming $\delta x = \delta t$:

$$\frac{UL}{Re} = \nu = \frac{\delta x^2}{3\delta t}(\tau_f - 1/2) = \frac{\delta x}{3}(\tau_f - 1/2) \quad (7)$$

$$\frac{UL}{Pe} = \alpha = \frac{\delta x^2}{3\delta t}(\tau_g - 1/2) = \frac{\delta x}{3}(\tau_g - 1/2) \qquad (8)$$

The relaxation parameters determine the stability of the numerical scheme. The model diverges more easily with small values of τ_g and τ_f. They cannot be chosen smaller than $1/2$ and solution becomes unstable when they get close to the limit value. A possibility to reach smaller ν or α with stable numerical scheme is to decrease spatial and time steps δx and δt. It indeed increases the need for computational ressources. Refinement of factor 2 needs four times more memory and is height times slower. A solution for memory and time saving is to refine mesh size locally where turbulence is expected. This is what next subsection is about.

3.1.2 Partial mesh refinement

In order to validate mesh refinement for density and increase of atteignable Reynolds number, a domain with two different grid sizes, as shown in figure 3, is defined and Poiseuille flow profile is recovered on each of the areas, as shown in figure 4, 5 and 6. A simulation of Karman instability is performed in figure 7, refined by a factor 2 around and after the cylinder where instabilities are supposed to appear. This configuration allows to increase Reynolds number by 35 %, in comparison with coarse grid simulation. This value has been established experimentally through the simulation of figure 7. In refined area, computational time is increased by a factor 8.

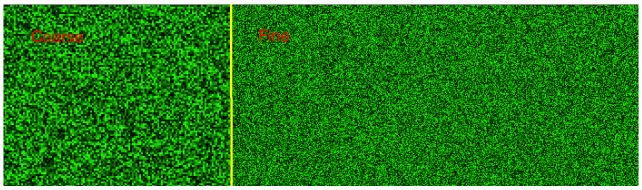


Fig. 3: Initialisation of a domain with two different resolutions.

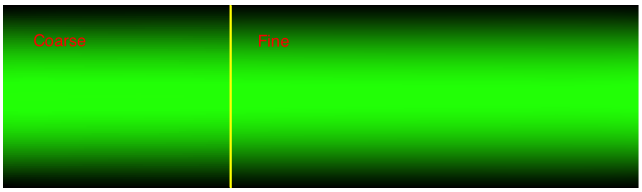


Fig. 4: Magnitude of velocity profile for Poiseuille flow simulation at Re = 200.

A possibility for computation time and memory saving would be to dynamically refine the grid, only at times and positions at which instability appears. The refinement would occur when one or more criteria would go above a threshold value. This will be introduced in the section about dynamic mesh refinement using a refined patch.

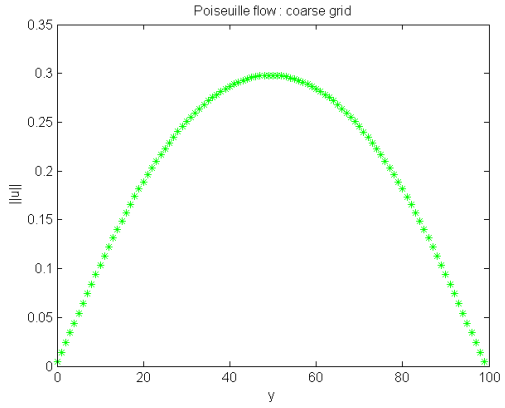


Fig. 5: Magnitude of velocity profile in coarse grid for Poiseuille flow simulation at Re = 200.

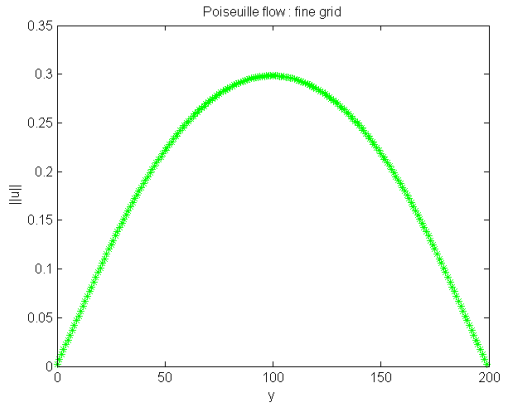


Fig. 6: Magnitude of velocity profile in fine grid for Poiseuille flow simulation at Re = 200.

3.1.3 Increase of atteignable Prandtl numbers using static mesh refinement

For a single component, the Prandtl number is the ratio of viscosity over thermal diffusivity. It also can been seen as the ratio of Péclet over Reynolds numbers, $Pr = \nu/\alpha = Pe/Re$. Usually, the use of same mesh size for density and temperature distribution functions restrains applications to species that have a Prandtl number close to unity. In order to allow simulations of $Pr << 1$ or $Pr >> 1$, different mesh sizes should be used for mass and heat transfer processes. Relaxation parameters have to be determined so that reference values for α_{ref} and ν_{ref} are conserved. The next formulas give relationship between lattice discrete properties :

$$\nu_{ref} = c_s^2(\tau_{fn} - 1/2)\delta t_f \qquad (9)$$

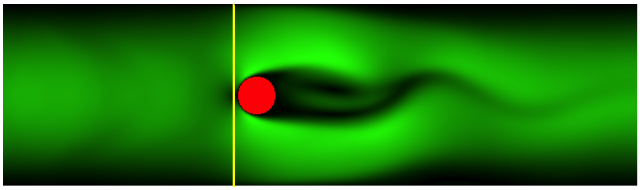


Fig. 7: Magnitude of velocity profile for Karmann instability simulation at Re = 200.

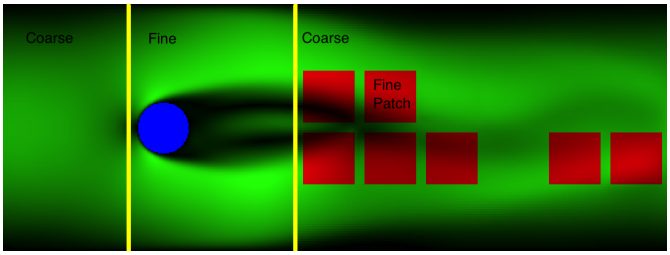


Fig. 8: Magnitude of velocity profile for Karmann instability simulation at Re = 195, using DMR.

$$\alpha_{ref} = c_s^2(\tau_{gn} - 1/2)\delta t_g \tag{10}$$

$\delta t = \delta x$ is better for stability, and the lattice speed of sound is kept as $cs = 1/\sqrt{3}$. A refinement factor can be defined as $n_{fg} = \delta t_f/\delta t_g = \delta x_f/\delta x_g$. If $Pr >> 1$, $n_{fg} >> 1$ and reciproquely $Pr << 1$ implies $n_{fg} << 1$. This new lattice layout means new rules have to be adopted for communication between density and temperature. If refinement factor is not unity, interpolations of some moments are necessary. Temperature has to be either averaged or interpolated when pressure is needed. Interpolation is performed with Taylor expansion. Speed is to be distributed equally or interpolated among cells of different resolution.

3.1.4 Increase of atteignable viscosity ratio using static mesh refinement

When two components have different viscosities, they cannot easily be simulated on the same grid resolution. The problem is the same with thermal diffusivity. Taking previous section into account, a per component mesh resolution for temperature and density should be determined. At communication steps between two components, the used macroscopic values have to be averaged or interpolated to a different resolution. Two components communicate at heat exchange step and by the external interaction force term. Those are the two macroscopic values that have to be converted to different resolutions.

3.2 Dynamic Mesh refinement (DMR)

3.2.1 DMR using a refinement patch

DMR is used to refine mesh locally when instability starts to develop. The refinement avoids instability to spread and is activated only on areas that require stabilisation. This step is performed with the use of a patch that translates distribution functions from coarse grid to a patch of finer resolution. The copy can be done with interpolation or with values equally distributed. After collison and advection are computed within the refined patch, values can be averaged back to coarse grid or kept on finer mesh. The rules for relaxation parameters are the same as for static mesh refinement.

In figure 8, Karman instability is simulated at $Re = 195$. Patch allows to increase maximum atteignable Reynolds number up to 31%. As at every time step values are copied from the patch to the coarse mesh, the only cost in memory is the size of the patch, which can be up to 24 times less than the cost of partial static refinement. The value of factor 24 has been established through the simulation of figure 8. The patch shouldn't be too small to keep stability. The computational time is also decreased by a factor depending on the number of patches used. The more the patches are, the slower the simulation is.

3.2.2 Criteria for activation of patch

The chosen criterium for activation of patch in figure 8 is the magnitude of non equilibrium viscous stress tensor, which is used in the Smagorinsky model to increase relaxation parameter. As it can be found in reference [7], using the Smagorinsky subgrid model consists in computing a correction step on relaxation parameter :

$$\tau^* = \tau + \tau_t \tag{11}$$

$$\tau_t = \frac{1}{2}\left(-\tau + \sqrt{\tau^2 + \frac{2C_{Smago}\delta_x^2}{\rho_0 c_s^4 \delta_t^2}\|\Pi_{\alpha\beta}^{neq}\|}\right) \tag{12}$$

$$\|\Pi_{\alpha\beta}^{neq}\| = \sum_{i=0}^{8} e_{i\alpha}e_{i\beta}(f_i - f_i^{eq}) \tag{13}$$

Parameter τ is corrected to τ^* by adding a vortex relaxation parameter τ_f. $\|\Pi_{\alpha\beta}^{neq}\|$ is the magnitude of non equilibrium viscous stress tensor. α and β are 2D space directions. ρ_0 is a density reference value. C_{Smago} is the Smagorinsky parameter, which has to be empirically determined. The Smagorinsky subgrid model has the advantage that it allows to simulate higher Reynolds number flows without increasing spatial resolution. The maximum attainable Reynolds number depends on the magnitude of C_{Smago}. Increasing this parameter means increasing maximum attainable Reynolds number. But the physical signification of C_{Smago} can remain quite unclear and some information loss can be suspected. Dynamic mesh refinement aims to increase Reynolds number without drastically increasing need in computational ressources, and without calculating correction

steps from equation (11) and (12), using a quite unclear C_{Smago} parameter.

DMR rather uses the non equilibrium viscous stress tensor $\|\Pi^{neq}_{\alpha\beta}\|$ as a criteria to activate refinement patch locally, as shown in figure 8. The magnitude is averaged on a sub-domain. If the obtained value is above a threshold value, a patch is activated on the sub-domain. For a thermal flow without source term the instability is of the same kind, because the numerical scheme is basically the same as LBGK model. Thus, a tensor of the same form of $\|\Pi^{neq}_{\alpha\beta}\|$ can be calculated and used to determine patched zones. Let us call this tensor $V^{neq}_{\alpha\beta}$:

$$\|V^{neq}_{\alpha\beta}\| = \sum_{i=0}^{8} e_{i\alpha}e_{i\beta}(g_i - g_i^{eq}) \tag{14}$$

For the case of a phase transition flow, the only criterium of viscous tensor norm is insufficient. The numerical instability does not come from the same phenomenon. The density ratio is usually the criterium responsible for divergence [9]. This means it has to be taken into account for refinement of multiphase flow. Thus, the criterium to be chosen is density gradient, which is significant at the interfaces between gas and liquid domains.

The calculation of this gradient can prevent high density ratios within the entire domain and thus easily refine the unstable places in order to gain stability. We present here a result of a simulation for evaluating the density gradient as a criterion for refinement. On images of figure 9, we take a benchmark of condensation for monocompnent fluid, on which we have added our criterion. An uncondensed domain is initiated and separation between liquid and gas phases occurs in isothermal condition. We can view the red locations where it is necessary to refine:

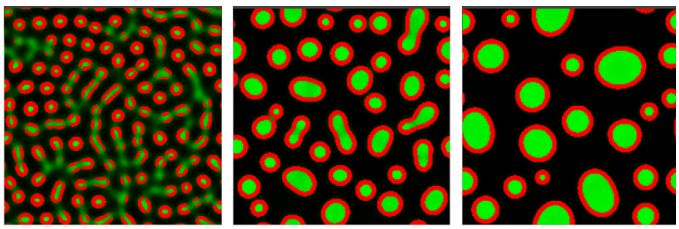


Fig. 9: Patched areas (red) where density gradient is significant.

We observe that only the areas where the density ratio is important are concerned. This criterion seems to be a good compromise to obtain a gain in stability for simulations of phase transition with large density ratios. It is important to remark that the purpose here is to show that the criterion of refinement with the density gradient is a relevant criterion, the threshold is currently set manually. It will be interesting to observe whether it is possible to automatically manage the threshold calculation according to the needs of the simulation.

3.2.3 Algorithm for DMR with patch

The whole coarse domain can be subdivided into sub-domains which can be of the size of the patch or larger. In figure 10, the sub-domains drawm in yellow are limited to regions of interest, after the cylinder, where instabilities are expected. Knowledge of the flow allows to decrease the number of regions of interest.

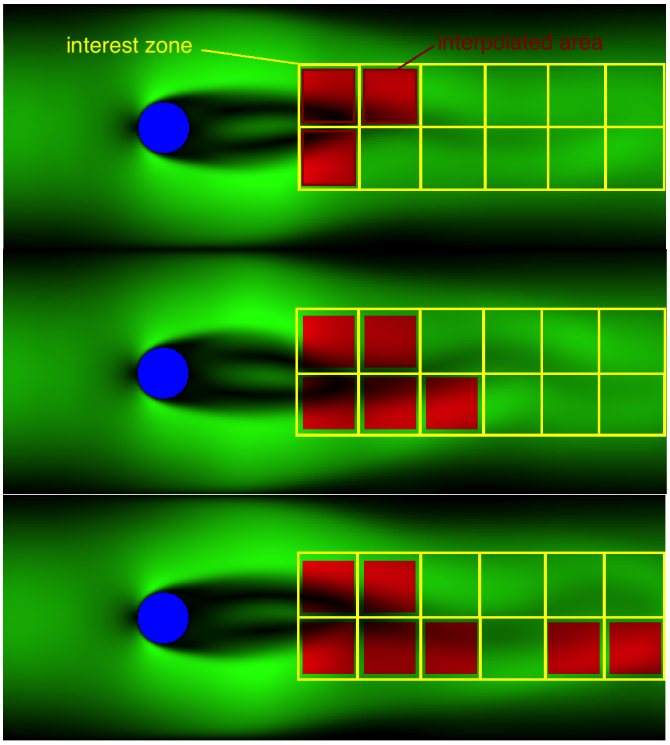


Fig. 10: Magnitude of velocity profile at different time steps for Karmann instability simulation with Re = 195, using DMR. The number of patched zone (red) increases with time.

The viscous tensor norm is calculated and averaged on a first interest zone from the twelve subdomains delimited in yellow. If the value is above a threshold, a patch is activated. At activation, a copy with or without interpolation from coarse grid to fine patch is computed. Then collison and advection steps are performed two times, if temporal resolution is refined with a factor 2. Notice that a factor two refined patch requires 4 times more elements in comparison to its corresponding coarse area. After those calculations, values are averaged and copied back to coarse grid. Then, the same work is performed on each of the remaining subdomains of interest.

Figure 10 shows increase of the number of patched zones, that gradually have viscous tensor norm above a defined

threshold value, at different time steps.

4. Multiple relaxation parameters parametrisation.

Another way to increase the atteignable Reynolds number is to use a multiple relaxation times (MRT) instead of a single relaxation time (SRT). This MRT scheme supposes a change from distribution functions space to the momentum space. The calculation is done with a transformation matrix and a collision matrix.

As previously reminded, the usual SRT LBM with force term has the following expression :

$$f_{\sigma,i}(x+e_i\delta t, t+\delta t) - f_{\sigma,i}(x,t) = -\frac{f_{\sigma,i}(x,t) - f^{eq}_{\sigma,i}(x,t)}{\tau_{\sigma,f}} + \Delta f_{\sigma,i} \tag{15}$$

A transformation matrix M allows the determination of corresponding per momentum equations :

$$Mf = m \tag{16}$$

Where $f = (f_0, ..., f_8)$ is the distribution function vector and $m = (m_0, ..., m_8)$ is the corresponding momentum vector. The transformation matrix M has the following expression :

$$M = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ -4 & -1 & -1 & -1 & -1 & 2 & 2 & 2 & 2 \\ 4 & -2 & -2 & -2 & -2 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & -1 & 0 & 1 & -1 & -1 & 1 \\ 0 & -2 & 0 & 2 & 0 & 1 & -1 & -1 & 1 \\ 0 & 0 & 1 & 0 & -1 & 1 & 1 & -1 & -1 \\ 0 & 0 & -2 & 0 & 2 & 1 & 1 & -1 & -1 \\ 0 & 1 & -1 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1 \end{bmatrix} \tag{17}$$

The new equation written in moment space and with multiple relaxation parameters is :

$$m_{\sigma,i}(x + e_i\delta t, t + \delta t) = m_{\sigma,i}(x,t) - s_i(m_{\sigma,i}(x,t) - m^{eq}_{\sigma,i}(x,t)) + \Delta m_{\sigma,i} \tag{18}$$

This new formula takes s_i as the relaxation parameter. It comes from the relaxation matrix which is given by :

$$S = \begin{bmatrix} s_0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & s_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & s_2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & s_3 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & s_4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & s_5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & s_6 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & s_7 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & s_8 \end{bmatrix} \tag{19}$$

This formulation allows to increase the atteignable Reynolds number, as shown in figures 11 and 12.

The usual limit for a non refined grid is $Re = 150$, but we came up to $Re = 1000$ with the MRT model. This shows the supperiority of the MRT scheme over SRT.

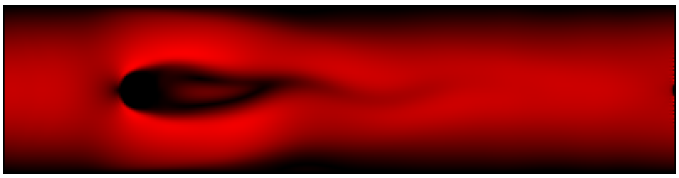


Fig. 11: SRT Karman instability at $Re = 150$.

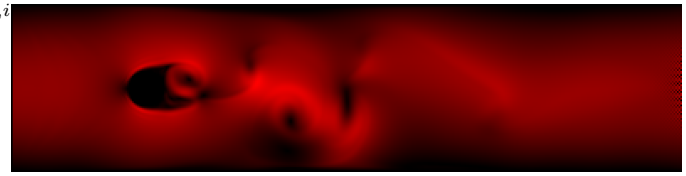


Fig. 12: MRT Karman instability at $Re = 1000$.

5. Conclusions

Static and dynamic mesh refinement have allowed to increase Reynolds number from up to 35 % and 31% for classic LBGK model. Criterium of viscous tensor norm shows to be efficient to activate patch and insure stability. Though stability is validated when the model does not diverge, and though we have computed Poiseuille flow, we believe further validation is necessary. Furthermore, the patch validity and tests should be extended to multiphase multicomponent flow. External and internal interaction force terms have to be established for multi resolution problems. Some work like reference [10] gives formula for multigrid resolution, but it still needs to be extended to multicomponent case and validated with the use of a patch. The criterium of density ratio for phase transition flow appears to be correct to determine the regions that should be patched.

The MRT [1] is another method of stabilisation which showed to be performant. The method allows to reach higher Reynolds numbers than with classic BGK collision term. GPU implementation that uses CUDA or OpenCL language allows parallel implementation of LBM [2] [3], and acceleration of computation time. Combination of the present work with MRT and GPU implementation can be a great improvement for the simulation of multicomponent flows in real cases.

References

[1] S. Succi A. Kuzmin, A.A. Mohamad. Multi relaxation time lattice Boltzmann model for multiphase flows. *International Journal of Modern Physics C*, 19 - 6:875–902, 2008.

[2] B. Tourancheau J.J. Roux C. Obrecht, F. Kuznik. Multi gpu implementation of a hybrid thermal lattice Boltzmann solver using the thelma framework. *Computers and Fluids*, 80:269–275, 2013.
[3] B. tourancheau J.J. Roux C. Obrecht, F. Kuznik. Multi gpu implementation of the lattice Boltzmann method. *Computers and Mathematics with Applications*, 80:269–275, 2013.
[4] J.G.Georgiadis R.O.Buckius D.J.Holdych, D.Rovas. An improved hydrodynamics formulation for multiphase flow lattice Boltzmann models. *International Journal of Modern Physics C*, 09(08):1393–1404, 1998.
[5] D.H. rothman M. Latva kokko. Diffusion properties of gradient based lattice Boltzmann models of immiscible fluids. *Physical review E 71*, (056702), 2005 May.
[6] G. Roussel N. Maquignon. A new lattice boltzmann model for thermal multi-component flow with heat transfer and phase change. *Computers & Mathematics with Applications*, under submission.
[7] S. Chen G.D. Doolen S. Hou, J. Sterling. A lattice Boltzmann subgrid model for high reynolds number flow. *Fields Insitute Communications*, 6, 1996.
[8] J.Y. Trepanier S. Leclaire, M. Reggio. Isotropic color gradient for simulating very high density ratios with a two-phase flow lattice Boltzmann model. *Computers & Fluids*, 48 (1):98–112, 2011.
[9] X. Shan. Analysis and reduction of the spurious current in a class of multiphase lattice Boltzmann models. *Phys. Rev. E*, 73 - 047701, 2006.
[10] L. Fan Z. Yu. An interaction potential based lattice Boltzmann method with adaptative mesh refinement (amr) for two-phase flow simulation. *Journal of Computational Physics*, pages 6456–6478, 2009.

Acknowledgment

This work has been made possible thanks to a collaboration between academic and industrial groups, gathered by the INNOCOLD association.

A New Algorithm for Multiple Key Linear Interpolation with the Formal Foundation

A. Tarek
Department of Science, Cecil College, North East, Maryland, United States of America

Abstract— *In this paper, a new algorithm for multiple key interpolation search is proposed, and the related formal, geometric foundations are presented. Multiple key interpolation search is an advanced key search strategy, which linearly interpolates a number of given keys from a sorted list, one after another onto a list, which is much larger in size, adjusting and optimizing the next key element's search space once a key is identified at a particular index location. The algorithm works through predictive forecasting while searching for a particular key index position. The algorithm may be considered with four different key and list order combinations. The index values at the end of the present search interval are subsequently adjusted to narrow down the search efforts through a linear interpolation. The proposed algorithm is considered with a Student Database System (SDS) in an academic institution.*

Keywords: Interpolation search, multiple keys, predictive forecasting, projected value, slope, sorted list

1. Introduction

In general, a search algorithm is not always required to compare only a given key, x with an array element. This means a modified search algorithm could compare between two array elements [4]. One significant improvement with the multiple key interpolation search over the previously proposed multiple key binary search algorithm [1] is the more precise guessing of the exact key location. With this improved search strategy, the key stays within the current interval of interest rather than blindly searching through the middle element at each step.

The proposed search algorithm deals with two different lists simultaneously. One is usually a much larger list, L_1 to which, the multiple key search algorithm is being applied to. The other list, L_2 is usually a much shorter list compared to L_1 holding the search keys. The algorithm requires both L_1 and L_2 to be sorted. In this paper, both L_1 and L_2 are assumed to be sorted in ascending order. However, it is also possible to consider a descending list of keys with an ascending list of elements with minor modifications in the proposed algorithm. Alternatively, an ascending list of keys may be searched for in a descending list of elements or a descending list of keys may be explored within a descending list of elements. The required modifications for the alternative combinations are also considered.

The proposed algorithm performs a straight line interpolation search with an intelligent prediction on the next index location to explore with the current key, k_i. Here, $i = 1, 2, \ldots, m$. If the elements in L_1 are uniformly distributed, then the slope of the interpolated straight line will remain constant throughout the entire search process. However, if L_1 does not contain uniformly distributed elements, the slope of the linearly interpolated straight line will vary from key to key as well as for each iteration in determining a particular key index position, t_i corresponding to the key, k_i. Here, $i = 1, 2, \ldots, m$. With each succeeding key in L_2, the search space is gradually narrowed down. The paper is organized as follows. Section 2 introduces the terms and notations used. Section 3 discusses the formal foundation of the proposed algorithm. Section 4 describes the algorithm. Initially, the 2 key version of the multiple key interpolation search algorithm is considered, which is followed by the more generalized m-key version. A part of this section analyzes the algorithm. The effect due to two uniformly distributed lists, L_1 and L_2 are also considered. Section 5 presents the analytical results relating to the proposed algorithm. Section 6 considers the geometric interpretation. Section 7 explores search optimization through variations of the proposed algorithm. Section 8 elaborates on an application of the algorithm.

2. Terms and Notation

n: Total number of elements in the given list.
m: Number of keys. Here, $m \geq 2$, and $n > m$.
L_1: List containing n given elements.
L_2: List containing m different keys.
k_i: The ith key in the list, L_2, $i = 1, 2, \ldots, m$.
r: Ratio between the number of list elements to the number of keys, $\frac{n}{m}$.
t_i: Index position of the ith key, k_i in list, L_1. Here $1 \leq t_i \leq n$. Also, $1 \leq i \leq m$.
$left$: Index position of the leftmost element in list, L_1.
$right$: Index position of the rightmost element in list, L_1.
l: Denotes the length of a list or a search space.

Definition 1: **Keygap, g:** The number of elements between two successive keys, k_i and k_{i+1} in L_1 is known as the *Keygap*. Here, $i = 1, 2, \ldots, (m-1)$. For instance, the number of elements in between the keys, k_j and k_{j+1} is, $t_{j+1} - t_j - 1$. Therefore, the *Keygap* between them is, $g = t_{j+1} - t_j - 1$ elements.

Definition 2: **Inter-Key Space Elements, I:** Elements in between all successive pairs of keys is collectively known as the Inter-Key Space Elements ($IKSE$). This is denoted by I. Therefore, $I = \sum_{j=1}^{m-1}(t_{j+1}-t_j-1)$. If a key is not identified inside L_1, then I measures the number of elements between the immediately preceding key and the next succeeding key that are existent in L_1. If none of the m keys exists within L_1, I = 0. If there is only 1 key, say the jth key, k_j that exists in L_1, then $I = (n-t_j)$. If only the first key, k_1 and the last key, k_m exist within L_1, then I = (t_m - $t_1 - 1$). If only the hth and the sth keys are in L_1 such that $1 \leq h < s \leq m$, then I = t_s - t_h - 1. Also, I for other key index combinations may be analyzed similarly.

3. Formal Foundation

With multiple key search strategy, instead of searching for a single element in each individual execution, all keys are simultaneously being searched for. This strategy significantly reduces the computational overhead due to individualized execution of the search algorithm with only one key. Multi-key linear interpolation provides with a very high yield for uniformly distributed list of elements. This yield may be significantly improved if the list of keys to be search for is also uniformly distributed. In that event, the search exploits the benefits due to two uniformly distributed lists. With the proposed search algorithm, there are two sets. One set is, S_1, the list of distinct elements where the proposed algorithm is being applied to. The other set, S_2 contains the distinct keys that are required to be mapped onto S_1. In general, $|S_1| = n$ and $|S_2| = m$, and $n >> m$. If f is the mapping function from S_2 to S_1, then $\{y = f(x) = x | y \in S_1 \ \& \ x \in S_2\}$. Assuming all keys may be mapped onto corresponding elements in S_1, the Predicate Logic Model for the mapping becomes, $\forall x \exists y (x \in S_2 \wedge y \in S_1 \wedge (y = x))$. Again, if $x = k_j$ and $y = Item_{t_j}$, then $\forall j (j \geq 1 \wedge j \leq n \wedge (x = k_j) \rightarrow (y = Item_{t_j}))$.

With distinct keys and list elements, the proposed algorithm essentially performs an one-to-one mapping from the S_2 to S_1. Since both keys and list elements are distinct, no two keys map onto the same list element. Due to the same reason, no key maps onto 2 or more different list elements. Hence, the problem of Collision due to Hashing does not arise with the proposed model. Following is the complete predicate logic model. $\forall x \forall y \forall z \forall k ((((z = f(x)) \wedge (y = f(x))) \rightarrow (z = y)) \wedge (((y = f(x)) \wedge (y = f(k))) \rightarrow (x = k))))$. The inverse mapping function, f^{-1} maps a list element onto a key to verify whether an element is a key or not, such that $x = f^{-1}(y) = y$. Hence, the multiple key functional mapping due to the proposed algorithm is both one-to-one and onto, and is essentially a **bijective mapping**.

Furthermore, assuming all keys from S_2 exist within the list, L_1, which is represented by S_1, S_2 becomes a pure subset of S_1. With the set theoretic notation, $S_2 \subset S_1$. In that event, S_2 becomes a member of the Power Set, Γ of S_1. Stated formally, $S_2 \in \Gamma(S_1)$. If some keys from S_2 are not present in L_1, $S_2 \not\subset S_1$. Therefore, there is a third set, S_3, which is a subset of both S_2 and S_1 that lies at the set intersection of S_2 and S_1. Therefore, $S_3 = S_2 \bigcap S_1$. Also, $(S_3 \subset S_2) \wedge (S_3 \subset S_1)$.

The Relation, R between the sets, S_2 and S_1 is an Equality Relation, which is a subset of the Cartesian Product, $S_2 \times S_1$. Hence, $S_2 R S_1$ is such that $\{(x, y) \in R | x \in S_2 \wedge y \in S_1 \wedge (y = x)\}$. Also, $R \subseteq S_2 \times S_1$. Here, R is a **binary relation** between the sets.

Initially, the proposed algorithm searches through the entire list, L_1 containing n different elements to identify the first key index position, k_1. As $|S_1| = n$, therefore, there are n possible one-to-one mappings from S_2 to S_1. Once k_1 is identified at location t_1, the $2nd$ key, k_2 is searched for only within the subset of S_1 containing $(n-t_1)$ elements. Hence, there are $(n-t_1)$ possible one-to-one mappings. Proceeding in this way, for the last key, k_m, there are $(n - t_{m-1})$ possible one-to-one mappings. Hence, the number of one-to-one mappings in identifying all key index positions is, $\Pi_{j=0}^{j=(m-1)}(n - t_j)$, where $t_0 = 0$.

4. The Proposed Search Algorithm

The 2-key version of the algorithm is considered first.

Algorithm MultiInterpol2key

Purpose: This algorithm performs 2-key interpolation search.

The supplied parameters are:

array arr[], index position of 1st element: $left$, position of the last element: $right$, smaller_key, and larger_key.

2-key interpolation search returns the array pos[] (in Java) with pos[0] and pos[1] corresponding to the smaller and the larger key index positions, respectively.

Require: smaller_key < larger_key

Ensure: Proper array index positions are identified.

int[] pos = new int[2] {The number of elements = 2 in array holding the keys.}

for j=0 to 1 **do**

$pos[j]$=−2 {Initially, all positional indices are set to an unrealistic value of −2.}

end for

int mid_high = $right$

while arr[$left$] < smaller_key and arr[mid_high]≥ smaller_key **do**

mid_low = $left$ + $\frac{((smaller\_key - arr[left]) * (mid\_high - left))}{(arr[mid\_high] - arr[left])}$

end while

if $arr[mid\_low]$ < smaller_key **then**

$left$ = (mid_low + 1)

else if $arr[mid\_low]$ > smaller_key **then**

mid_high = (mid_low - 1)

else if arr[mid_low] == smaller_key **then**

$pos[0]$ = mid_low

$left$ = (mid_low + 1)
mid_high = (mid_low + 1)
end if
if arr[$left$] == smaller_key **then**
$pos[0]$ = $left$
else if $pos[0] == -2$ **then**
$pos[0] = -1$
end if
while arr[mid_high] < larger_key and arr[$right$]≥ larger_key **do**
mid = mid_high + $\frac{((larger\_key - arr[mid\_high]) * (right - mid\_high))}{(arr[right] - arr[mid\_high])}$
if arr[mid] < larger_key **then**
$mid\_high = (mid + 1)$
else if arr[mid] > larger_key **then**
$right$ = ($mid - 1$)
else if arr[mid] == larger_key **then**
pos[1] = mid
$left$ = ($mid + 1$)
end if
end while
if arr[mid_high] == larger_key **then**
pos[1] = mid_high
else if pos[1] == -2 **then**
pos[1] = -1
end if
return pos

Following is the generalized m-key interpolation search algorithm with $m \geq 2$.

Algorithm MultiInterpol$m$key
Purpose: This algorithm performs m-key interpolation search.
The supplied parameters are: array arr[], position of the 1st element: $left$, position of the last element: $right$, and an array of m different keys, key[].
m-key interpolation search returns the array pos[] (in Java) with the index positions of m different keys.
Require: key[i] < key[$i + 1$] for $i = 0, 1, 2, \ldots (m - 2)$.
Ensure: Proper key index positions are identified.
int[] pos = new int[m] {Number of elements = m in array holding the keys.}
for j=0 to $m - 1$ **do**
$pos[j]$=-2 {Initially, all positional indices are set to an unrealistic value of -2.}
end for
$save\_value$ = $right$ {Save value of the supplied $right$ index required for each iteration.}
for j=0 to $m - 1$ **do**
{Restore the original value of $right$ for each iteration.}
$right$ = $save\_value$
{Indexing in Java, C++ and C starts at 0.} {Perform the m key interpolation search for each key, key[i], $i = 0, 1, 2, \ldots, (m - 1)$.}
while arr[$left$] < key[i] and arr[$right$]≥ key[i] **do**
int mid_pos = $left$ + $\frac{((key[i] - arr[left]) * (right - left))}{(arr[right] - arr[left])}$
if $arr[mid\_pos] < key[i]$ **then**
$left$ = (mid_pos + 1)
else if $arr[mid\_pos] > key[i]$ **then**
$right$ = (mid_pos - 1)
else if arr[mid_pos] == key[i] **then**
$pos[i] = mid\_pos$
$left$ = (mid_pos + 1) {Since this is a Multiple Key Interpolation Search ($MKIS$), left needs to be adjusted each time with each key.}
end if
end while
if arr[$left$] == key[i] **then**
$pos[i]$ = $left$
end if
if $pos[i] == -2$ **then**
$pos[i] = -1$
end if
if $pos[i] \neq -1$ or $pos[i] \neq -2$ **then**
$left = (pos[i] + 1)$
end if
end for
return pos {Return the Java array, pos[] containing the key index positions to the calling program.}

4.1 Algorithm Analysis

With multiple key interpolation search, if the first key (the smallest key) k_1 lies in between $left$ and $right$ index positions, the next iteration is taken to be about $\frac{(k_1 - arr[left])}{(arr[right] - arr[left])}$ of the way between $left$ and $right$ index positions towards $right$. This may be considered as an offset for the key being looked for. Here, $left$ is leftmost element and $right$ is rightmost element index in the list, L_1. The keys are numeric, and they increase in a roughly constant manner throughout the interval by assumption.

Once the key, k_1 is identified at t_1, the search for the next key, k_2 is restricted only within the range from t_1 through t_r. Here, t_r is the index position of the rightmost element, and $t_r = right$. Hence, k_2 is about $\frac{(k_2 - k_1)}{(arr[right] - k_1)}$ of the way between t_1 and $right$ towards $right$. Once k_2 is identified at index location, t_2, the search for k_3 is restricted only within the range of t_2 through $right$. Now, k_3 is about $\frac{(k_3 - k_2)}{(arr[right] - k_2)}$ of the way between k_2 and $right$ towards $arr[right]$. Proceeding this way, the search for k_m is restricted only within the range of t_{m-1} through t_r. Therefore, k_m is about $\frac{(k_m - k_{m-1})}{(arr[right] - k_{m-1})}$ of the way between the elements, $arr[t_{m-1}]$ and $arr[right]$ towards $arr[right]$. Thus, one step of the search reduces the uncertainty of locating a key element in a given list with size, n from n to $\sqrt{(n)}$. In contrast, 1 step of the binary search reduces the uncertainty of locating a key element in a given list with size n from n to $\frac{1}{2}n$. Hence, the Multiple

Key Interpolation Search algorithm as proposed in this paper is asymptotically superior to the Multiple Key Binary Search algorithm proposed in [1].

4.2 Advantages Rendered By Two Uniformly Distributed Lists

The multiple key interpolation search algorithm may exploit the potential advantage yielded through two uniformly distributed list of elements. Consider the following equation that describes the slope of the straight line between two adjacent keys. In the following equation, t_i denotes the index position of the ith key, k_i in list L_1, and a_i denotes the t_ith index element in L_1.

$$S_i = \left(\frac{t_i - t_{i-1}}{a_i - a_{i-1}}\right), \quad \text{Here } 1 \leq i \leq m \text{ and } a_i \in L_1 \quad (1)$$

In equation (1), S_i denotes the slope of the straight line in between the keys, k_i and k_{i-1}. Since $1 \leq i \leq m$, for the first key, k_1, $t_{i-1} = t_0 = 0$, which is the index position of the first element in L_1. If the keys in L_2 are equal value apart from each other, and if the elements in L_1 are uniformly distributed, then assuming all keys from L_2 exist within L_1, following results follow.

1) The index positional difference, $h = t_i - t_{i-1}$ is a constant for each i, $1 \leq i \leq m$, and $t_0 = 0$ (the 1st index in L_1).
2) The difference in values, $b = a_i - a_{i-1}$ between two successive L_1 elements corresponding to the two successive keys k_i and k_{i-1}, respectively, is also a constant for each i, $1 \leq i \leq m$.
3) The slope, $S_i = \frac{h}{b}$ is also a constant, s for all i, $1 \leq i \leq m$.
4) The total index positional distance between all successive pair of keys, $h_t = m \times h = t_m$.
5) If the list, L_1 is implemented as an array named *arr*, then $arr[t_i] = a_i = k_i$ for all i, $1 \leq i \leq m$.

Since the basic interpolation search works best with an uniform list, L_1, which is foundational to the Multiple Key Interpolation Search strategy, the proposed algorithm works best with two uniform lists, L_1 and L_2. Following shows how the proposed algorithm optimizes the search space with a uniform list of keys, L_2, which is also uniformly distributed over L_1. If the basic interpolation search is applied m different times to identify m key index positions on the list, L_1, then each application will search through the index positions 0 through (n-1). The search space length for each application is (n-1-0+1) = n. For m total applications, the total search space length is, mn. If the lists, L_1 and L_2 are organized in the same order (either in ascending or in descending order), then to search for k_1, the search space length is, $(n - 1 - 0 + 1) = n$. Search for the key, k_2 starts at index $(t_1 + 1)$. Hence, for k_2, the length of the search space is, $(n - 1 - t_1 - 1 + 1) = (n - t_1 - 1)$ (the rightmost index remains same at $n - 1$). For k_3, the length of the search space = $(n - t_2 - 1)$. Proceeding in this fashion, for the last key, k_m, the length of the search space is, $(n - t_{m-1} - 1)$. Hence, length of the overall search space is, $n + n - t_1 - 1 + n - t_2 - 1 + \ldots + n - t_{m-1} - 1 = (mn - \sum_{i=1}^{m-1} t_i - m + 1)$. If L_1 and L_2 are organized in the opposite order (either L_1 in ascending and L_2 in descending or vice-versa), then the search space length for k_1 is n, as before. The search space length for k_2 is, $(t_1 - 1 + 1) = t_1$. The search space length for k_3 is, t_2. Proceeding in this way, the search space length for k_m is, t_{m-1}. Therefore, the length of the total search space with m different keys = $(n + t_1 + t_2 + \ldots + t_{m-1}) = n + \sum_{i=0}^{m-1} t_i$ with $t_0 = 0$. Here, t_i is the index position of the ith key. As an instance, consider the following example.

Example 3 (Search Space Optimization): Consider a list, L_1 containing 10^6 elements, and a list, L_2 of keys with 10^3 elements. Assuming that the index positions due to the keys in L_2 are uniformly distributed over L_1. Then the first key will be identified at index position, $\frac{10^6}{10^3} - 1$. This is based on the assumption that the indexing in list, L_1 starts at 0. The 2nd key will be identified at index position, $2 \times 10^3 - 1$, and so on. If the basic interpolation search algorithm is applied m different times in searching for the m index positions, then the length of the overall search space is, $l_{basic} = m \times n = 10^3 \times 10^6 = 10^9$. If L_1 and L_2 are arranged in the same order, then the length of the overall search space, $l_{same} = mn - \sum_{i=1}^{m-1} t_i - m + 1 = 10^9 - (10^3 + 2 \times 10^3 + 3 \times 10^3 + \ldots + 10^3 \times 10^3) - 10^3 + 1 = 10^9 - 10^3 \times \frac{(10^3+1)(10^3)}{2} - 10^3 + 1 = 499499001 \approx \frac{1}{2} \times 10^9$. If L_1 and L_2 are in opposite order, then the length of the overall search space, $l_{opposite} = n + \sum_{i=0}^{m-1} t_i = 10^6 + (10^3 + 2 \times 10^3 + 3 \times 10^3 + \ldots (10^3 - 1) \times 10^3) = 10^6 + 10^3 \times 10^3 \times (10^3 - 1) \times \frac{1}{2} = 500500000 \approx \frac{1}{2} \times 10^9$. Now, $\frac{l_{same}}{l_{basic}} = \frac{499499001}{10^9} = 0.4995$. Hence, with the same arrangement of L_1 and L_2, the multiple key interpolation search effectively reduces the search space length by, $(1 - 0.4995) \times 100\% = 50.05\%$. Again, $\frac{l_{opposite}}{l_{basic}} = \frac{500500000}{10^9} = 0.5005$. Therefore, with opposite ordering of L_1 and L_2, the multiple key interpolation search effectively reduces the search space length by, $(1 - 0.5005) \times 100\% = 49.95\%$.

Hence, the search space length may effectively be reduced to about 50% with uniformly distributed lists, L_1 and L_2 leading to optimized search efforts through the proposed multiple key interpolation search algorithm.

5. Analytical Results

Theorem 4 (Average Distance Theorem): If all n elements in list, L_1 are uniformly distributed in ascending order, then the **average number** of elements encountered in identifying the key index positions is, $n_{avg} = n(1 - \frac{1}{2}(1 - \frac{1}{m}))$. Here, n = total number of elements in L_1, and m is the number of keys in list, L_2.

Proof: Suppose that L_1 is uniformly distributed over its n elements in ascending order. Since $r = \lfloor \frac{n}{m} \rfloor$, therefore, on

average, a key exists after each ascending r elements. The basic formulation used together with m-key interpolation search is,

$$mid\_pos = left + \frac{((k_i - arr[left])(right - left))}{(arr[right] - arr[left])} \quad (2)$$

For the 1st key with $left = 0$, $mid\_pos_1 = 0 + \frac{((k_i - arr[0])(n-1-0))}{(arr[n-1]-arr[0])}$. If the first key is identified at index 0, then $mid\_pos_1 = 0$ and $k_1 = arr[0]$. If the keys are uniformly spaced r elements apart in L_1, then for k_1, the number of intermediate elements encountered is, n. For k_2, the number of intermediate elements considered is, $(n - r)$. For k_3, the number of intermediate elements considered is, $(n - 2r)$. Proceeding this way, for k_m, the number of intermediate elements considered is, $(n - (m - 1)r)$. Therefore, the total number of intermediate elements encountered with m different keys, $n_{total} = n + (n - r) + (n - 2r) + \ldots + (n - (m - 1)r)$ = mn - $(r + 2r + 3r + \ldots + (m - 1)r)$ = $m(n - \frac{r(m-1)}{2})$ (as $(r + 2r + 3r + \ldots + (m - 1)r)$ = $\frac{r(m-1)(m-1+1)}{2}$ = $\frac{rm(m-1)}{2}$). Therefore, $n_{avg} = \frac{n_{total}}{m}$ = $(n - \frac{r(m-1)}{2})$. But r = $\frac{n}{m}$. Hence, n_{avg} = $(n - \frac{n(m-1)}{2m})$ = $n(1 - \frac{1}{2}(1 - \frac{1}{m}))$. ◇

Following result follows from above Theorem 4.

Corollary 5: The average number of elements n_{avg} encountered in identifying the m key index positions in an uniformly distributed list of elements is inversely proportional to the number of keys m.

Proof: From Theorem 4, n_{avg} = $n(1 - \frac{1}{2}(1 - \frac{1}{m}))$. As the number of keys, m increases, (1-$\frac{1}{m}$) also increases, and $(1 - \frac{1}{2}(1 - \frac{1}{m}))$ decreases. Given that the list, L_1 has a constant number of elements n, with the increasing value of m, n_{avg} decreases. Hence, $n_{avg} \propto \frac{1}{m}$. ◇

Example 6 (Average Distance Theorem): Consider a uniformly distributed list with $n = 100,000 = 10^5$ elements. If the number of keys, m = $1000 = 10^3$, then using the result from Theorem 4, n_{avg} = $50,050$ elements, which is about $\frac{1}{2}$ of n. For applying the basic interpolation search algorithm m different times with m different keys, the average number of elements encountered is, $m \times n$ = 10^8, which is about $2,000$ times more.

6. Geometric Interpretation

In performing linear interpolation, the proposed algorithm uses the following straight line equation.

$$mid_i = left + \frac{(k_i - arr[left]) \times (right - left)}{(arr[right] - arr[left])} \quad (3)$$

Here, mid_i is the linearly projected index for key, k_i, $1 \leq i \leq m$ from L_2. Also, $left$ is the leftmost index, and $right$ is the rightmost index in L_1. Equation (3) is in the following standard straight line equation form.

$$y = Mx + d \quad (4)$$

In equation (4), y is the Y coordinate value for a point on the straight line, which corresponds to the projected key index position, mid_i for k_i. Also, d is the intercept of the linearly interpolated straight line with the Y-axis, and M is the slope of the straight line, which may be expressed as follows:

$$M = tan\theta = \frac{(right - left)}{(arr[right] - arr[left])} \quad (5)$$

In the above equation, θ is the angle that the interpolated straight line makes with the X-axis. With the straight line model depicted as above, array elements are represented along the X-axis, and the positional index values are represented along the Y-axis. Also, d is the intercept with the Y-axis for X-axis value of 0. For simplicity, both L_1 and L_2 are considered in ascending order with $left$ = 0 and $right$ = (n-1) for a total of n elements. Hence, the interpolated straight line equation for key, k_1 is,

$$mid_1 = \frac{(k_1 - arr[0]) \times (n - 1)}{(arr[n - 1] - arr[0])} \quad (6)$$

The slope of the above interpolated straight line is,

$$M_1 = \frac{(n - 1)}{(arr[n - 1] - arr[0])} = tan\theta_1 \quad (7)$$

The $left$ index position is shifted towards $right$ with the search for each successive key in L_2. Once k_1 is identified at t_1, the search for k_2 starts at index $(t_1 + 1)$ with $left$ = $(t_1 + 1)$ and $right$ = $(n - 1)$ to optimize the search space for k_2. Hence, for k_2, the slope of the interpolated straight line is,

$$M_2 = \frac{(n - t_1 - 2)}{(arr[n - 1] - arr[t_1 + 1])} = tan\theta_2 \quad (8)$$

Proceeding in this way, for key, k_m, the slope of the interpolated straight line is,

$$M_m = \frac{(n - t_{m-1} - 2)}{(arr[n - 1] - arr[t_{m-1} + 1])} = tan\theta_m \quad (9)$$

In the expression for each slope, M_i, $1 \leq i \leq m$, the values of n and $arr[n - 1]$ are constant. Therefore, the positional index, t_i and the value of the array element, $arr[t_i + 1]$ will determine the slope, $tan\theta_i$ for each interpolated straight line segment, associated with each key, k_i, $i = 2, 3, \ldots, m$ except for the first key, k_1. If the elements in L_1 are uniformly distributed, the slope, $tan\theta_i$, $i = 1, 2, \ldots m$ will remain constant for all keys. However, if L_1 is not uniformly distributed, the slope will vary with the interpolated straight line for each key. Also, the intercept d being the index, $left$, is determined by $arr[left]$ or $arr[0]$ for the first key, and $arr[t_i + 1]$, $i = 1, 2, \ldots, (m - 1)$ for each successive key, k_{i+1}. For the first key, $left$ = 0, and the straight line passes through the origin. If L_1 is uniformly distributed, the ratio of $\frac{t_{i+1} - t_i}{k_{i+1} - k_i}$ will also remain relatively constant for each i, $i = 1, 2, 3, \ldots, (m - 1)$.

7. Search Variations

With different possible combinations of arrangements of elements in L_1 and also in L_2, four optimum search strategies are possible as variations to the proposed algorithm. These are precisely described below. Here, both L_1 and L_2 are sorted.

Combination 1**:** Consider when both L_1 and L_2 are sorted in ascending order. This combination is the most common, and is used throughout this paper. With this combination, once a key, k_i is identified at index position, t_i, the leftmost index is shifted right to $t_i + 1$. The rightmost index remains at its original position. Hence, both left and right index positions are gradually shifted towards each other in narrowing down the search space. However, the right index shift is only temporary, whereas the left index shift is not. Once a key position is exactly identified, before continuing search with the next key, the right index position is rolled back to its original location. Hence, the overall search space is optimized upon shifting the leftmost boundary towards the rightmost one.

Combination 2**:** Consider when L_1 is in ascending order and L_2 is in descending order. Therefore, the keys in L_2 are organized from higher to lower order. For this combination, the leftmost index position is kept fixed throughout the entire search. After determining each key index location, the rightmost index is shifted left immediately before the last key index position found. If the key, k_i is identified at t_i location, the search for k_{i+1} is restricted in between the original leftmost index and the index at $t_i - 1$. As $k_{i+1} < k_i$, and as L_1 is in ascending order; therefore, search for k_{i+1} is restricted to L_1 with elements smaller than k_i.

Combination 3**:** Here, both L_1 and L_2 are in descending order. This combination works pretty similarly to that in Combination 1. Here, the rightmost index is kept fixed, and the leftmost index is shifted towards the rightmost index. If k_i is identified at index t_i, then k_{i+1} is searched only within elements smaller than k_i in L_1 (as L_2 is in descending order, therefore, $k_{i+1} < k_i$). The list, L_1 is also in descending order with elements smaller than k_i are located to the right of t_i.

Combination 4**:** For this combination, L_1 is in descending order and L_2 is in ascending order. The search strategy with this combination is similar to that in Combination 2. Since L_2 is in increasing order, once a key, k_i is identified at t_i, the search for k_{i+1} is restricted to elements larger than k_i in L_1. As L_1 is in descending order, the elements larger than k_i are located to the left of t_i. Hence, the search for t_{i+1} is restricted between original $left$ and the index at $t_i - 1$.

With the proposed algorithm, the left index boundary linearly approaches towards the right index boundary. If the key exists within L_1, the index $left$ eventually coincides with the index $right$, and the interpolated straight line essentially converges to a single point at an index location, which is the index of the key currently being searched for.

8. Application

The proposed algorithm may be applied through look-up tables for performing efficient searches in large database applications. In general, the sizes of the physical databases are huge, and may not be loaded in its entirety inside the main memory. These databases are kept inside the hard disk, and randomly accessed based on the targeted record addresses. An access to a main database record inside the secondary memory is expensive compared to a typical instruction execution, because each comparison requires a hard disk access. Any changes to the database, for example, record edit, delete, etc. are required to be done on an individual record basis. This is usually performed through a temporary buffer inside the main memory. Hence, it is necessary to directly and randomly access the particular record locations inside the hard disk to be able to perform such operations with enough efficiency.

For faster efficient access to the database records, a smaller sequential access file, called an **Index File** is created. Once the database is active in operation, only the sequential index file is loaded onto computer's main memory, and search operations are always performed on the Sequential Index File to identify the exact record locations on the hard disk database for random access. In general, batch updates are performed on the random access databases on the hard disk. This means that there is a group of records available for update at one time, and the updates are performed on those records one after another inside the database file on the hard disk by identifying the exact record locations through the sequential index file loaded in the main memory. Due to human intervention after each record update, this process becomes slower. An alternative would be to use an efficient algorithm to identify the memory locations of all records to be updated at one single iteration, and update all identified records at the respective memory locations without human intervention. Since hard disk access is expensive, this will definitely improve the efficiency of batch update making the operation faster. The algorithm proposed in this paper may be used efficiently towards this objective.

Consider the Student Database of an academic institution. Generally, the academic institutions use student ID numbers to identify particular student record inside their hard disk databases. For instance, if an institution has $10,000$ students, and it uses exactly $1,000$ kilo bytes (KB) or 1 mega bytes (MB) of hard-disk space to store each individual student record, then the total database space required will be, $10,000$ MB or 10 GB. An index file for the student database operations may be created that has only one field, which is 10 bytes long to hold the Student ID, and another field, which is only 8 bytes long to hold the hard disk memory address corresponding to the Student ID. Altogether the index file will require $(10 + 8) \times 10,000$ bytes or 180 kilo bytes(KB) of memory space. Thus, the small index file may be easily loaded and operated from the computer's main

memory. Following are the typical operations that may be performed on the student database.

- Update Record: Student records may be updated from time-to-time.
- Clean-up Record: Student records may be deleted using a multiple key file supplied from outside. The supplied delete-key file may be organized in ascending or in descending order of Student IDs. Only the records, but not the assigned IDs will be deleted, such that these vacant IDs may be reassigned to incoming students.
- Adding New Students to Deleted Records: Student IDs inside L_2 supplied from outside containing the IDs of previously deleted records may be used to assign new IDs to the incoming students. The corresponding emptied records may be allocated to hold their information.

For an efficient search, the data is required to be sorted and fairly uniformly distributed. For instance, consider that the institution has assigned numbers 000019999 through 100009999 as student IDs to its $10,000$ students. Suppose that the students with ID numbers 003019999, 006259999 and 011279999 have left the institution during an academic year. Using the proposed algorithm for the deleted student with the key, 003019999:

$$recordPos_{003019999} = left_{index} + \frac{((studentId - arr[left_{index}]) * (right_{index} - left_{index}))}{(arr[right_{index}] - arr[left_{index}])}$$

As a result, $recordPos_{003019999}$ will be mapped onto: $00001 + \frac{(003019999-000019999)\times(10000-00001)}{(100009999-000019999)} = 00301$. If the address of the hard disk database file begins at 000000000000, and if each ASCII record is 1048576 bytes(1 mega bytes) long, then the record for the ID number 003019999 begins at: $(00301 - 00001) \times 1048576 = 314572800$. Therefore, the record beginning at 314572800 with 1048576 bytes will be wiped off. The $left_{index}$ will be shifted to $00301 + 1 = 00302$ for the next key in the list. The $recordPos_{006259999}$ will be calculated at: $00302 + \frac{(006259999-003029999)\times(10000-00302)}{(100009999-003029999)} = 00625$. As a result, the record beginning at $(00625 - 00001) \times 1048576 = 654311424$ will be deleted. Finally, the $recordPos_{011279999}$ will be calculated at: $00626 + \frac{(011279999-006269999)\times(10000-00626)}{(100009999-006269999)} = 01127$. The record beginning at $(01127-00001)\times1048576 = 1180696576$ will be deleted. Applying the proposed algorithm, the exact record location for each record to be deleted is identified through one interpolation. Using the Multi-Key Binary Search algorithm as proposed in [1], the first projected middle index will be computed at $int(\frac{(10000+00001)}{2}) = 05000$. The next index will be computed at: $int(\frac{(04999+00001)}{2}) = 02500$, and so forth. Proceeding this way, it takes 14 iterations to compute the index of the 1st student ID, and a few more for the 2nd and the 3rd student IDs. Hence, the proposed search algorithm is efficient in managing a common Student Database System at an academic institutions.

9. Conclusion

The paper proposes a new algorithm for linearly interpolated multiple key search strategy, and shows the effectiveness of the proposed algorithm. The paper also explores formal and geometric interpretations of the proposed algorithm. Examples are incorporated to illustrate the introduced concepts. Analysis shows that the overall search space may effectively be optimized up to 50% of the original size with a uniformly distributed list of keys and a uniform list of elements. The algorithm may be applied to database searches where the elements in the database are uniformly distributed based on a particular key value, and where the search being conducted is based on that particular key.

The essential difference between the algorithm proposed in this paper and the database hashing algorithms as discussed in [3] lies in the relative sizes of the lists, L_2 and L_1. With the algorithm as proposed in this paper, it essentially performs key to address mapping. The efficiency of the mapping is dependent upon the uniformity in distribution of elements in L_1 as well as that in L_2. In general, if the keys in L_2 are uniformly distributed, assuming all keys exist in L_1, the keys are also uniformly apart in L_1, provided that L_1 is also uniformly distributed. In general, the population of L_2 is much smaller compared to L_1. However, the population of keys for a hashed list is usually much greater than the available storage area for data (in this case, elements in L_1) [3]. As usual, there are many available keys from L_2, for each index location within L_1. As a result, more than one key map onto the same location in L_1. This phenomenon is known as a Collision. The keys from L_2 mapping onto the same index position in L_1 are known as Synonyms. Hence, with hashing, $|L_2| > |L_1|$, whereas for a Student Database System (SDS), $|L_2| \ll |L_1|$. With each distinct key in Multiple key Interpolation Search strategy, since $|L_2| \ll |L_1|$, a **collision** is far from the reality. This is an advantage of using the proposed algorithm.

In future, the algorithm will be implemented for different key and list order combinations as outlined in Section 7. The related algorithmic performances will be evaluated and compared with one another. Other performance issues will also be considered in depth.

References

[1] Ahmed Tarek, "Multi-key Binary Search and the Related Performance," in *Proc. MATH'08,* 2008, ISSN: 1790-5117, ISBN: 978–960–6766–47–3, p. 104.

[2] Donald E. Knuth, *The Art of Computer Programming – Volume* 3 *Sorting and Searching*, 2nd ed., New Jersey, USA: Addison-Wesley, 1998.

[3] Richard F. Gilberg and Behrouz A. Forouzan, *Data Structures–A Pseudocode Approach with C++*, California, USA: Brooks/Cole, Thomson Learning, 2001.

[4] Richard Neapolitan and Kumarss Naimipour, *Foundations of Algorithms Using Java Pseudocode,* Massachusetts, USA: Jones and Bartlett Publishers, 2004.

Application of Gain Scheduling Technique to a 6-Axis Articulated Robot using LabVIEW®

ManSu Kim[#,1], WonJee Chung[#,2], SeungWon Jeong[#,3]

[#]*School of Mechatronics, Changwon National University*
Changwon, 641-773, South Korea

[1]subek@naver.com

[2]wjchung@changwon.ac.kr

[3]dongnegosu@gmail.com

***Abstract*— Recently the high performance of industrial robots has been especially required according to the increase of demand and application range of industrial robots. Industrial robots performing repetitive motions do not have much trouble in the short term. Nonetheless, most robots are designed with a view to long-term operation, which is why it is very important to find optimal gain values for robots. In this paper, for accurate gain tuning of a (lab-manufactured) 6-axis articulated industrial robot (hereinafter called "RS2") with less noise, a program routine of DSA (Dynamic Signal Analyzer) for frequency response method will be programmed using LabVIEW®. Then robot transfer functions can be obtained experimentally using frequency response method with DSA program. Data resulted from the robot transfer functions are transformed into Bode plots, based on which an optimal gain tuning will be executed. Of course, gain tuning can enhance the response quality of output signal for a given input signal during the real-time control of robot. To cope with gain tuning for each work domain of the robot, an optimal gain value for each point will be found. Finally the tuned gains can be imported to each joint controller of RS2 in order to dynamically control the robot, which is just "gain scheduling." The domain-specific gain scheduling suggested by this paper can improve robots' tacking to input commands and thus the stability of kinematic parts.**

***Keywords*— Gain tuning, Gain scheduling, RS2, LabVIEW® DAQmx, Articulated robot, Dynamic Signal Analyzer, Frequency response, Gravity-against motion, Zero position**

I. INTRODUCTION

Traditionally robots have been used in various industry fields such as handling or welding mechanical parts. But, Recently the high performance of industrial robots has been especially required according to the increase of demand and application range of industrial robots. Specially, unlike Cartesian Robot and SCARA (Selective Compliance Assembly Robot Arm) which have wide application in assembling electronic parts, the dynamic performance of a 6-axes articulated robot is greatly changed according to the position and orientation of the robot. This means that the gain tuning of the robot's servo controllers should be tuned considering the dynamic characteristics of robot mechanism. It is well known that gain tuning can reduce the vibration phenomena of a robot so as to improve the performance of positional control.

As one of previous studies, Jung *et al.* [1] has presented "Application of SoildWorks® and LabVIEW®-based Simulation Technique to Gain Tuning of a 6-axis Articulated Robot." in CSC 2012. In this paper, gain tuning using LabVIEW® has been performed for a 6-axis articulated *(lab-manufactured)* robot (*called as 'RS2', see Fig. 1*). Especially simulation was conducted by interlocking SolidWorks® (6-axis robot modeling) with LabVIEW®, in order to verify the experimental results of gain tuning by being compared with the simulation results of gain tuning. In contrast to ref. [1], for accurate gain tuning of RS2 with less noise, we will utilize a program routine of DSA (Dynamic Signal Analyzer) [2] for frequency response method will be programmed using LabVIEW®. Then robot transfer functions can be obtained experimentally using frequency response method with DSA program. Data resulted from the robot transfer functions are transformed into Bode plots, based on which an optimal gain tuning will be executed. Also another contribution of this paper is that gain tuning can be performed according to the position of robot's end-effector in workspace, and then optimally tuned values of gain can be stored in an array form in the LabVIEW® program. Finally the tuned gains can be

imported to each joint controller of RS2 in order to dynamically control the robot, which is just "*gain scheduling*."

Fig. 1 6-axis articulated robot (RS2)

II. LabVIEW®-based Experimental Gain Tuning

A. *Gain Tuning of Zero Position*

As mentioned above, Fig. 1 illustrates the robot used in this paper, *i.e.*, RS2, which is a 6-axis articulated robot designed and developed in our laboratory. RS2 is a miniaturized version of a high-rigidity, high-torque and heavy-duty robot, which has a 600-kg load capacity and is 4 times the size of the established prototype of RS2. In this paper, to measure the frequency response of the 6-axis articulated robot, DSA (Dynamic Signal Analyzer) is implemented using LabVIEW® Sound and Vibration Toolkit. Also, the frequency response is measured to convert the robot's transfer function into a Bode Plot prior to the gain tuning. To receive the robot's frequency response, NI's LabVIEW® DAQmx is used.[3]. For the purpose of finding out *domain-specific* optimal gain values, we will carry out the gain tuning for each axis in the zero (or home) position of RS2 in each work domain of workspace.

For the higher control system of RS2, the Motion Controller of NI PXI-7350 equipment has been used with the universal control and measurement software of LabVIEW®. Figure 2 represents a program controlling the axis of RS2. Here a value of motor encoder is received as an output robot signal for the applied voltage of an input value so that the robot signal is plotted by LabVIEW®. Upon the execution of DSA program, a robot transfer function can be obtained as a plot as shown in Fig. 3.

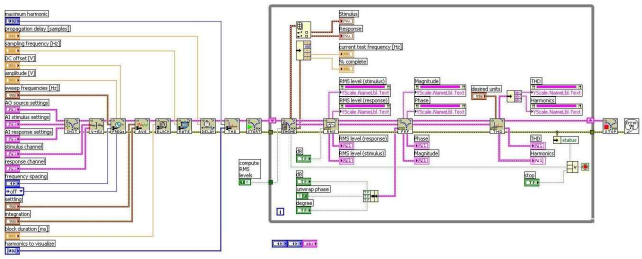


Fig. 2 DSA Programming by using LabVIEW®

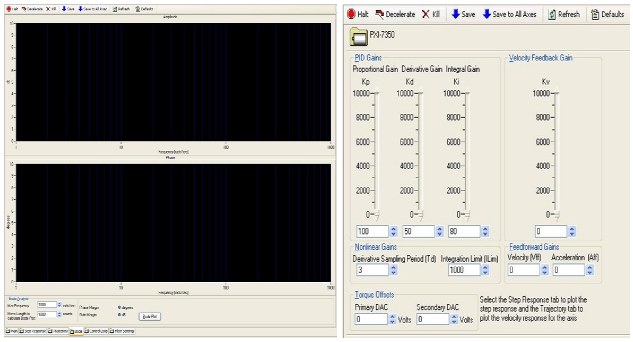
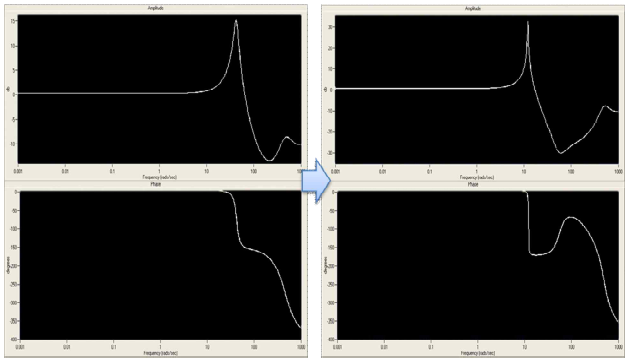


Fig. 3 Bode plot of DAQmx program

For the tuning of a gain value, the LabVIEW® DAQ (Data AcQuisition) equipment is connected with the 6-th (*i.e.*, the last) axis motor driver nearby the end-effector of RS2. First, an arbitrary value of proportional gain has been set for the motor driver. Then an appropriate value of the sine wave amplitude X is selected according to ref. [4]. At this time, an integration effect has been eliminated by setting the integration time constant at 1000.[5]. Finally frequency response test is conducted as follows: A sine wave of 0.5 V_{rms} (root mean square of voltage) from 2Hz through 500Hz is applied to the speed command pin of a servo driver as a source wave form from PXI-6733 of LabVIEW® DAQ; a Bode plot ($G_c(s)$) of a closed loop can be extracted using the programmed DSA; the closed loop transfer function $G_0(s)$ can be obtained by using the open loop transfer function $G_c(s)$. Then, the closed loop transfer function found as such was used for the gain tuning. For theories relevant to gain tuning, this paper has been based on Jung *et al.*[1]. Equation (1) denotes the relationship between the closed loop transfer function and the open loop transfer function. Figure 4 shows the closed and open loop bode plots for the 6-th axis.

$$G_0(s) = \frac{G_c(s)}{1 - G_c(s)} \tag{1}$$

Fig. 4 Bode plot of open loop & closed loop transfer function of the 6-th axis before tuning



In general, according to Nyquist stability [6], gain margin should be -6dB ~ -20dB, while phase margin should be larger than 45 degree. The extracted Bode plot of open loop leads to the gain margin of -14.4dB and the phase margin of -52.3 degree. Based on Eq. (2), we can obtain the new proportional gain of velocity control loop, K_v=132.

$$20\log x = (-6dB) - (\text{Gain Margin})$$
$$x = 10^{\frac{-6-(\text{Gain Margin})}{20}} \tag{2}$$
$$K_v' = xK_v$$

The integral gain (K_i) in the speed control mode is determined by the integral time constant (T_i). For the integral time constant (T_i), the tuning starts on the 6-th axis at the end. Finally, the K_i can be found by Eq. (3). The integral time constant, T_i, is the point equivalent to 10 times the applied frequency (Hz) of the phase margin, so that no phase variation occurs when the proportional gain value (K_v) of the speed loop on each axis of the robot is applied. For the proportional gain (K_p) of the position control loop, the tuning starts on the 6-th axis at the end. The frequency, f_c, is measured at the point of -3dB of the resonance point on the closed loop bode plot. For ζ, the K_p value was found by substituting the value 0.707, which was calculated with an experimental method for general industrial robots, in Eq. (4). Figure 5 shows the closed and open loop bode plots for each axis found with the aforementioned method. These were used to find K_v, K_i, K_p and T_i values as in Table I.

$$K_i = K_v / T_i \tag{3}$$

$$K_p = \frac{\pi f_c}{2\zeta^2} \tag{4}$$

$$f = \frac{\omega}{2\pi} = \frac{f_c}{2\zeta} \tag{5}$$

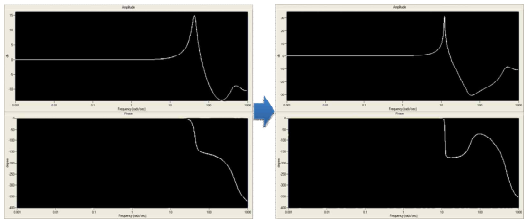
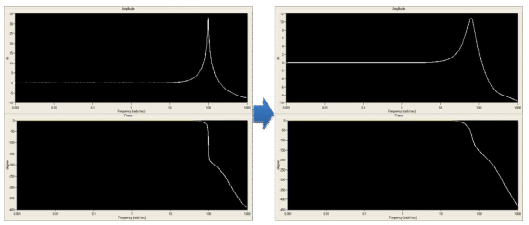
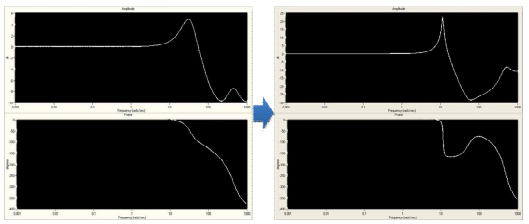
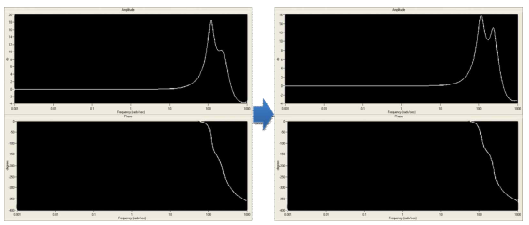
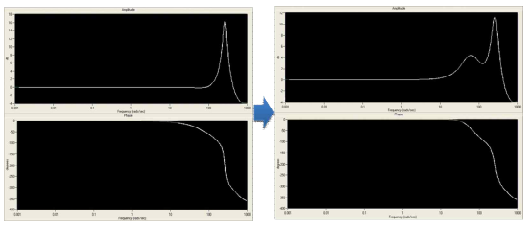
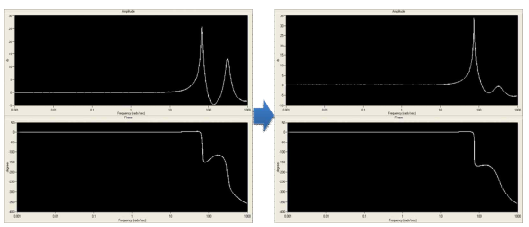
Axis	Bode plot
1st	
2nd	
3rd	
4th	
5th	
6th	

Fig. 5 Bode plot of open loop & closed loop transfer functions of the 6- axis before tuning

TABLE. II THE RESULT OF GAIN TUNING FOR EACH JOINT

Axis	State	1st	2nd	3rd	4th	5th	6th
K_v	before	50	50	50	50	50	50
	after	64	82	141	77	89	132
K_i	before	100	100	100	100	100	100
	after	256	713	1698	576	685	763
K_p	before	20	20	20	20	20	20
	after	125	273	379	236	242	182
T_i	after	0.252	0.115	0.083	0.133	0.130	0.172

B. *Response*

To verify the gain-tuning results, the frequency response technique was used as in Fig. 6. The frequency response technique is a practical and effective method for system analysis and design. The frequency response of system is defined as the normal response of system to trigonometric function input signals with varying frequencies such as $A\sin(\omega t)$. Figure 7 shows the stimulus command level, the response level at the optimized gains and the response level at the empirical gain settings simultaneously for each axis. As a result of Fig. 7, the response level at the optimized gains is much closer to the stimulus command level (red color line), compared with the response level at the empirical gain settings (blue color line). This means that the responses after tuning can be verified in line with the extent to which the output curve matches the sine wave input curve without disturbance.

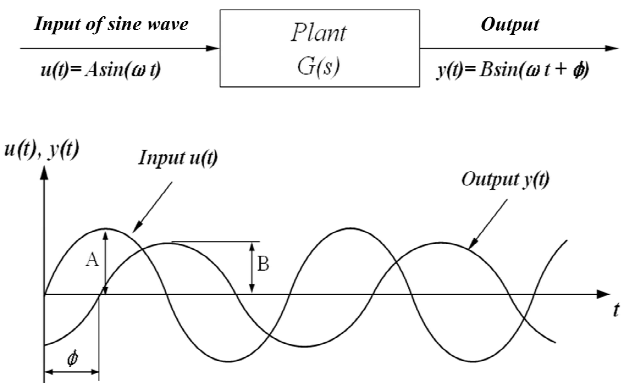


Fig. 6 Frequency response method

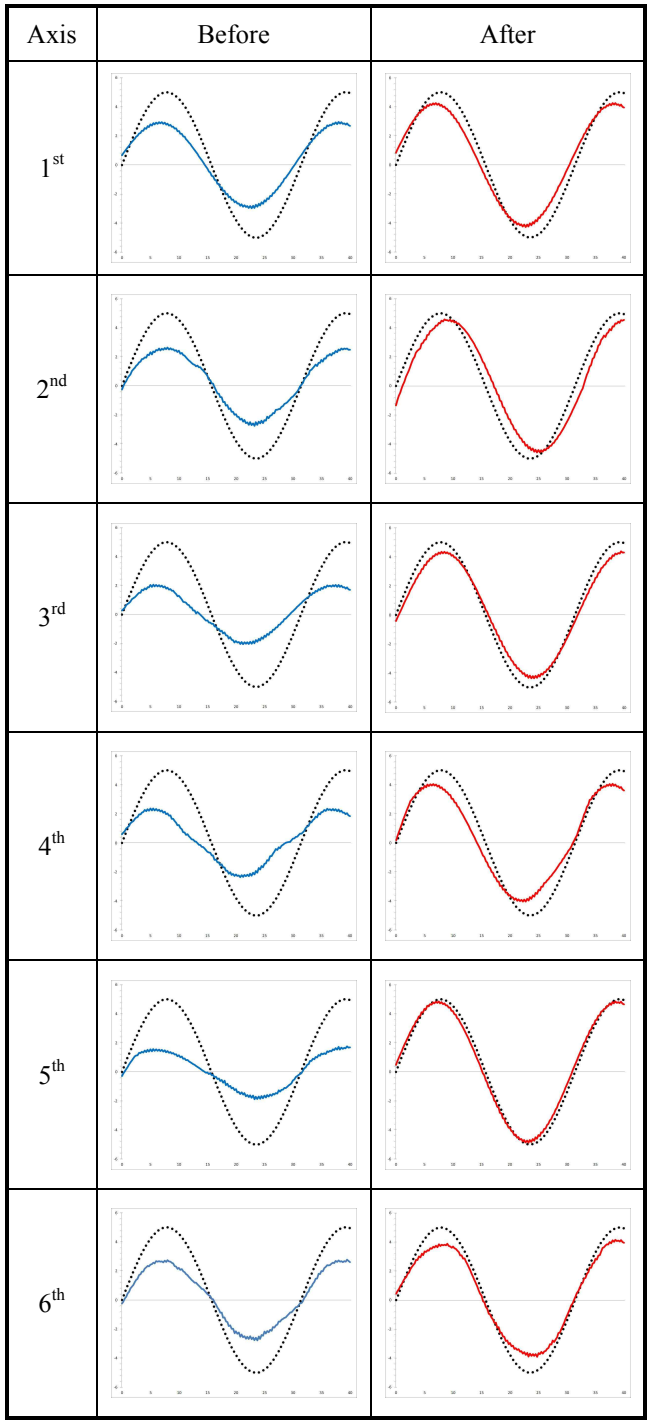


Fig. 7 Result of frequency response

III. GAIN SCHEDULING

So far, we have performed the gain tuning for the zero position prior to robot operation and verified the improvement of responsiveness. Now, this section deals with examining whether the gain values from the initial tuning carry out the

optimal motion in other movements and to elicit the optimal gain values for each domain. Then this leads to gain scheduling.

As shown in Fig. 8, the gain values initially found at A and B points where RS2 has moved in the direction of gravity in the zero position (called as "coordinated gravity-against motion") are applied to verify the responsiveness. *A* and *B* points are the positions corresponding to the second-axis joint rotating at an angle of 30° and 60° , respectively, in the zero position. Since mainly the second and third axes move for the operation of RS2 in the direction of gravity, we focus on the verification of responsiveness on those axes. [7]

Fig. 8 Coordinated gravity-against motion of RS2

As shown in Fig. 9, in the experiment, the response signals (blue solid lines) of gain tuning performed in the zero position did not affect the improvement of responsiveness following the command movements signal (black dotted line) at A and B points. This means that new gain values are needed after RS2's movements at other points (or positions in the direction of gravity) except the zero position. Thus we have performed new gain tuning for two work domains according to the robot's movements. The gain tuning has been done in the same way as in the aforementioned zero position. Only the actually moving second and third axes have gone through the gain tuning and the resultant gain values are compared as in Table IIIIV. The resultant tracking response (blue solid line) to the input command signal (black dotted line) in Fig. 10 is shown to be explicitly good, compared with Fig. 9. Based on the experimental findings above, this study has conducted a LabVIEW® programming to apply the new gain values to two motions as in Fig. 11. An optimal motion has been obtained by replacing the gain values in the designated position where the experiment was performed.

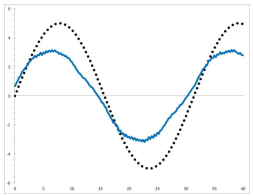
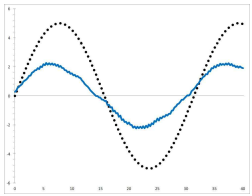
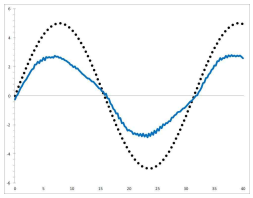
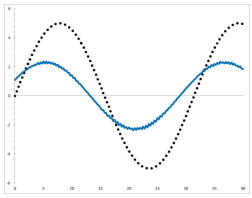
Axis	Position *A*	Position *B*
2nd		
3rd		

Fig. 9 Result of tracking (frequency) response level at zero position

TABLE VVI RESULT OF GAIN TUNING AT EACH POSITION

Axis	Zero position		A position		B position	
	2nd	3rd	2nd	3rd	2nd	3rd
K_v	82	141	89	223	69	182
K_i	713	1698	824	2468	589	1799
K_p	273	379	291	347	268	311
T_i	0.115	0.083	0.108	0.090	0.117	0.101

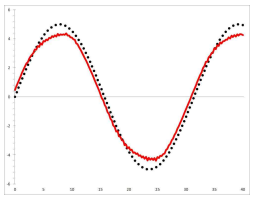
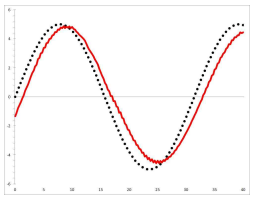
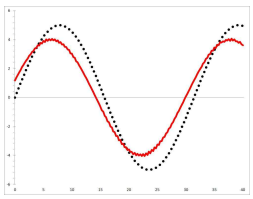
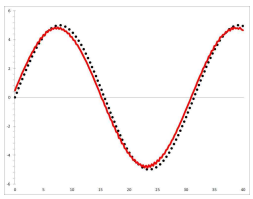
Axis	Position *A*	Position *B*
2nd		
3rd		

Fig. 10 Result of frequency response level for the 2nd and 3rd axes

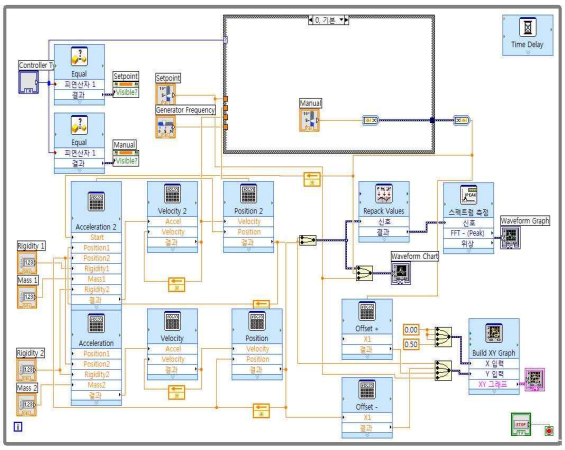


Fig. 11 LabVIEW® programming to apply the new gain values to two motions

IV. Conclusion

Industrial robots performing repetitive motions do not have much trouble in the short term. Nonetheless, most robots are designed with a view to long-term operation, which is why it is very important to find optimal gain values for robots. This paper aims at improving the responsiveness with gain tuning for each servo motor of the robot by implementing a Dynamic Signal Analyzer using LabVIEW® and measuring the frequency response. First of all, LabVIEW® was used for gain tuning in the zero position of the 6-axis vertical articulated (*lab-manufactured*) robot (*called as 'RS2'*). Then, the proportional gain (K_v) in the speed control loop and the proportional gain (K_p) of the position control loop were induced. Finally, the input trigonometric functions of varying frequencies were compared with the output values to verify the gain tuning. To sum up, the responsiveness following the gain tuning in the zero position was found to have improved. Also, with new gain tuning for each work domain of the robot, an optimal gain value for each point was found. The domain-specific gain scheduling suggested by this paper can improve robots' tacking to input commands and thus the stability of kinematic parts. The findings of this study suggest the viability of optimal control of robots via gain scheduling by applying the programming that finds gain values suitable for each moving point and then substitutes them sequentially in robots undertaking repetitive works.

V. Acknowledgement

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2011-0013902).

References

[1] Chang Doo Jung, Won Jee Chung, " Application of SoildWorks® and LabVIEW®-based Simulation Technique to Gain Tuning of a 6-axis Articulated Robot.", CSC12, 2012

[2] HEWLETT PACKARD, 1991, HP 35665A Dynamic Signal Analyzer Concepts Guide.

[3] National Instrments, 2012, NI-DAQmx Description, <http://www.ni.com/white-paper/3021/en/#toc1>

[4] Ogata, K., 1990, Modern Control Engineering, Prectice-Hall, Inc. pp. 448~467.

[5] Haugen, F., 2004, PID control of dynamic systems, Intl specialized book service inc. pp.342~349.

[6] Kuo, B. C., 1991, AUTOMATIC CONTROL SYSTEMS, Prentice-Hall, Inc. pp. 747~762.

[7] MITSUBISHI, General-Purpose Interface MR-J2S- A Servo Amplifier Instruction Manual.

Fast and Slow MHD Waves in the Turbulent Plasma

G. V. Jandieri
Department of Physics, Georgian Technical University, Tbilisi, Georgia

Abstract *- Statistical characteristics of the Alfvén waves propagating in the turbulent plasma flow and energy exchange between wave and nonstationary medium are obtained for the arbitrary correlation function of the density fluctuations in the ray (optics) approximation. Characteristic frequencies of the temporal pulsations of plasma irregularities leading to the broadening of the temporal spectrum of scattered Alfven wave in the plasma flow are calculated using the experimental data.*

Keywords: Alfven wave, plasma flow, statistical characteristics, broadening, irregularities.

1 Introduction

The wavy processes in the upper atmosphere have both, hydrodynamic and electromagnetic nature. In the first class of waves belong the acoustic (sonic), gravitational and MHD (Alfven and magnetoacoustic) waves, while the second class of waves contains planetary Rossby waves and magnetogradient waves [1]. General dispersion equation was derived for the magneto-acoustic, magneto-gravity and electromagnetic planetary waves in the ionospheric *E*- and *F*-regions [1,2]. The geomagnetic field generates small and medium-scale Alfvén waves with phase velocity depending on the orientation of the wave vector $\mathbf{k}$ with respect to the geomagnetic field $\mathbf{H_0}$. They are very slow ($10\div 50$ $\mathrm{m\cdot s^{-1}}$) and long-periodical ($1\div 2$ days) when the wave vector $\mathbf{k}$ is transversal to $\mathbf{H_0}$ and fast, when vectors $\mathbf{k}$ and $\mathbf{H_0}$ are parallel.

The ionospheric observations reveal the electromag-netic perturbations in the *E* –region known as the slow MHD waves [3,4]. These waves are insensitive to the spatial inhomogeneities of the Coriolis and Ampere forces and are propagated in the ionospheric medium more slowly than the ordinary MHD waves. In natural conditions, these perturbations are revealed as background oscillations [1]. According to the numerous observations [4] at the ionospheric moderate and high latitudes large-scale (up to 10^3 km), and long-period (with characteristic time scale of 0.5-2 hours) ionospheric wavy perturbations regularly exist which are propagating zonally over long distances (to ten thousand of kms) with the velocity more than 1 $\mathrm{km\cdot s^{-1}}$. The observed propagation velocity of wave can not be explained within the frames of hydrodynamic theory of ordinary acoustic gravity waves, since the maximum characteristic velocity of the letter, at the ionosphere altitudes does not exceed 700-800 $\mathrm{m\cdot s^{-1}}$. The velocities of the order of 1 $\mathrm{km\cdot s^{-1}}$ and more are arising when the influence of partial "freezing-in" of the geomagnetic field on the propagation of MHD waves in the ionosphere is taking into account. For the *E*-region the plasma component behaves like a passive impurity. The neutrals completely drag ions and the "ionospheric" friction between neutrals and ions can be neglect [1]. Therefore velocity of the neutral component $H_0/\sqrt{4\pi M N_n}$ is much lower than velocity of the plasma component $H_0/\sqrt{4\pi M N}$, where N_n and N denote concentrations of the neutral particles and charged particles of the ionospheric plasma, respectively. Below we consider slow, long period, large-scale MHD waves in *E*- layer of the ionosphere.

The features of low-frequency waves in homogeneous magnetized plasma are well studied [5], however little attention is devoted to the investigation of statistical characteristics of MHD waves in turbulent plasma flow observing in both cosmic and laboratory conditions. It was established that statistical moments of these waves substantially depend on a type of waves [6]. Therefore propagation of MHD waves in the turbulent plasma streams is of practical interest. Some peculiarities of statistical characteristics of MHD waves in randomly inhomogeneous plasma in the "freezing-in" turbulence approximation have been investigated [7].

The peculiarities of statistical characteristics of large-scale slow Alfvén waves propagating in weakly ionized plasma (the *E*-region of ionosphere) with randomly varying spatial-temporal parameters are considered in this paper. Important factor of the waves propagation in the nonstationary medium - the energy exchange between Alfven wave and turbulent plasma flow is analyzed analytically and numerically using experimental data. Characteristic frequencies of the temporal pulsations of plasma irregularities leading to the broadening of the temporal spectrum are calculated first time. Statistical parameters of scattered Alfven wave substantially depend on the ration of Alfven velocity and macroscopic velocity of a plasma flow.

2 Small oscillations of the Earth's Ionosphere

Linearized equation of motion of the MHD set of equations describing wavy processes in the ionosphere taking into account Hall's effect has the following form [1,8]

$$\frac{\partial \mathbf{V}}{\partial t} = \frac{1}{\rho c}[\mathbf{j}\cdot\mathbf{H}] - \frac{1}{\rho} grad\, P + \mathbf{g} + [\mathbf{V}\cdot 2\boldsymbol{\omega}_\mathbf{0}], \quad (1)$$

where P and $\rho = \rho_n + \rho_{pl} \approx \rho_n = M N_n$ are pressure and density of the neutral particles; M is mass of ions (molecules); $\boldsymbol{V}$ and $\mathbf{H}$ are vectors of the fluid velocity and magnetic field, respectively; $\mathbf{g}$ is the vector of gravitational acceleration, $\boldsymbol{\omega}_\mathbf{0}$ is the angular velocity of the Earth's rotation, $\mathbf{j}$ is the current density, c is the speed of light, $\mathbf{F_A} = [\mathbf{j}\cdot\mathbf{H}]/c\rho$ is the electromagnetic Ampere's force [1].

In numerous observations [9,10] the real atmosphere quickly restores the violation of both quasi-static (during several minutes) and quasi-geostrophic (roughly during one hour) states. Thus, for the synoptic processes (two weeks and more), it may be considered that the atmosphere always is in the quasi-static and quasi-geostrophic states [1]. It was shown [11] that neutral Rossby waves and acoustic-gravity waves are eliminated if the quasi-static and quasi-geostrophic conditions are fulfilled in the atmosphere. In this case, separating electromagnetic effects of slow MHD waves and neglecting all hydrodynamic forces Equation (1) reduces to: $\partial \mathbf{V}/\partial t = [\mathbf{j}\cdot\mathbf{H_0}]/\rho c$, $\mathbf{H} = \mathbf{H_0} + \mathbf{h} \approx \mathbf{H_0}$ [1]. In the upper layers of the *E*-region (at altitudes of 100-150 km, $\omega_i / \nu_{in} \sim 10^{-2} << 1$), fast waves with a wavelength of 2000 km are significantly damped due to the Pedersen conductivity. However, longer waves are damped weakly. Therefore, in this paper, in considering long-wavelength perturbations with $\lambda \sim 10^4$ km we neglect, for simplicity, the Pedersen conductivity in the Hall layer [12]. Restricting ourselves by moderate and high latitudes (geomagnetic field has only vertical component $\mathbf{H_0} = H_{0z}\mathbf{e_z}$), generalized Ohm's law for the *E*-region can be expressed in the following form [1]:

$$\frac{1}{c}[\mathbf{j}\cdot\mathbf{H}] = eN\left(\mathbf{E} + \frac{1}{c}[\mathbf{V}\cdot\mathbf{H}]\right). \quad (2)$$

Assuming the equality $[\mathbf{j}\cdot\mathbf{H_0}] = 0$ [4,1], we get:

$$\mathbf{E} = -\mathbf{w} - i\,g\,[\mathbf{w}\cdot\boldsymbol{\tau}], \quad \mathbf{j} = -\frac{c^2}{4\pi V_a^2}\frac{\partial \mathbf{w}}{\partial t}, \quad (3)$$

where $\mathbf{w} = [\mathbf{V}\cdot\mathbf{H_0}]/c$ is the dynamo field caused by the wind mechanism [1], $\boldsymbol{\tau} = \mathbf{H_0}/H_0$ is the unit vector along the strength of the geomagnetic field, $g = \omega/\Omega_i$ is the ratio of the wave frequency to the ion gyrofrequency, $\Omega_i = \eta\,\omega_t$ is modified by the ionization degree cyclotron frequency of ions (in the *E*-region Ω_i is of the order of $10^{-4} \div 10^{-5}$ s^{-1}) [1], $\omega_i = eH_{0z}/Mc$ is the cyclotron frequency of ions ($\omega_t \approx 10^2$ s^{-1}). In the neutral component of the ionosphere the velocities of the order of 1 km$\cdot$s^{-1} and they are insignificant for MHD waves in the plasma component ($\sim 10^3$ km$\cdot$s^{-1}). This is stipulated by the fact that in the ionosphere for the long-period processes the geomagnetic field is "frozen" into the plasma component and during the perturbations it passes its perturbation to the neutral component by collision processes. In the neutral part, it propagates with the Alfven velocity $V_a = H_0/\sqrt{4\pi\rho} = H_0/\sqrt{4\pi M N_n} = \sqrt{\eta}\,V_A$, where $V_A = H_0/\sqrt{4\pi M N}$ is the velocity of the MHD wave in the plasma component of ionosphere. In the ionospheric *E* (70-150 km) and *F* (150-600 km) regions the ionization degree $\eta = N/N_n$ is of the order of $10^{-8} \div 10^{-4}$. Therefore, the value of V_a is much less than V_A. Consequently we naturally come to the consideration of slow (in the electrodynamics sense) long-period MHD waves in the ionosphere [1].

Substituting equation (3) into Maxwell's equation $rot\, rot\, \mathbf{E} = -(4\pi/c^2)\,\partial \mathbf{j}/\partial t$ at $\mathbf{w} \sim \exp(i\,k_x x + i\,k_z z - i\,\omega t)$ we obtain the wave equation [1]:

$$\frac{\partial^2 \mathbf{w}}{\partial t^2} + V_a^2\, rot\, rot\, \mathbf{w} = i\, g\, V_a^2\, rot\, rot\, [\mathbf{w} \cdot \boldsymbol{\tau}]. \qquad (4)$$

The last term takes into account the Hall's effect. If $V_z = 0$, $V_x \neq 0$, $V_y \neq 0$, we obtain set of algebraic equations, and hence, the dispersion equation:

$$(\omega^2 - V_a^2 k^2)(\omega^2 - V_a^2 k_z^2) = g^2 V_a^4\, k^2 k_z^2, \qquad (5)$$

where: $k^2 = k_x^2 + k_z^2$. Equation (5) (in the electro-dynamics sense) describes very slow and long-period (from two days to two weeks and more) MHD waves in the ionospheric *E*-region. The first bracket describes propagation of transverse Alfvén's wave in the ionosphere. Compressibility and stratification of the ionosphere do not play any role in the Alfven's waves and, therefore, for these waves the transversality condition $[\mathbf{k} \cdot \mathbf{V}] = 0$ is always satisfied.

At large k_x, if $k_z^2 << k_x^2$ and $V_a k_z << \omega$, from Equation (5) we get [1]:

$$\omega^2 = V_a^2 k_x^2 \left(1 + \frac{V_a^2 k_z^2}{\Omega_i^2}\right). \qquad (6)$$

From Equation (6) it follows that in the *E*-region the characteristic horizontal wavelength $\lambda_0 = 2\pi V_a / \Omega_i$ exists, which determines the characteristic "length of dispersion" caused by the Hall's effect. If $V_a k_z << \Omega_i$ frequency of magneto-acoustic wave $\omega_M = V_a k_x$ increases linearly with k_x; if $\Omega_i << V_a k_x$ wave frequency is subject to the frequency of helicons ω_h:

$$\omega = \omega_h = k_x k_z \frac{V_a^2}{\Omega_i} = k_x k_z \frac{c\, H_{0z}}{4\pi N e}. \qquad (7)$$

In the ionospheric physics, they are known as "atmospheric whistlers". As a result, helicons in the *E*-region are the limiting case of magnetic sound. In helicons only electrons of the ionospheric plasma are oscillating together with the frozen in geomagnetic field lines. At small k_x, if $k_z^2 >> k_x^2$ and $\omega < \Omega_i$ from Equation (5) we obtain frequency of the Alfvén wave $\omega_A = V_a k_z$.

For the second root, taking into account Equation (5), at $k_z^2 << k_x^2$ and $\omega << V_a k_x$ we get:

$$\omega^2 = V_a^2 k_z^2 \frac{\Omega_i^2}{\Omega_i^2 + V_a^2 k_z^2}. \qquad (8)$$

At $\Omega_i >> V_a k_z$ expression (8) describes Alfven wave with dispersion $\omega = V_a k_z$. At big wavenumber k_z the wave frequency is subject to the characteristic frequency $\omega \to \Omega_i = \eta\, \omega_i$. Consequently, waves Ω_i in the ionosphere are the limiting case of the quasi-transversal very low-frequency Alfven waves [1].

If $k_z^2 >> k_x^2$ from Equation (5) we obtain new branch of the modified Alfven wave having frequency: $\omega_{A*} = V_a^2 k_z^2 / \Omega_i = V_a^2 k_z^2 / \eta\, \omega_i$ and the ordinary Alfvén wave: $\omega = V_a k_z$. For $k_x = 0$ from Equation (5) we obtain the dispersion relation for the Alfven-type waves propagating with phase velocity $V_{ph\,A}^2 = V_a^2 / (1 \pm \omega_h / \omega)$. For the frequencies $\omega >> V_a k_z$ and $k_z > k_x$ phase velocity of helicon is: $V_{ph} = c\, H_{0z} k_z / 4\pi e N$. In absence of dispersion (Hall effect), for magneto-acoustic and Alfven waves we obtain: $V_{ph\,M} = H_{0z} / \sqrt{4\pi M N_n} = V_a$ and $V_{ph\,A} = V_a \theta$, where $\theta = k_z / k_x$. Slow magneto-acoustic waves propagate with the velocity of Alfven waves in the direction perpendicular to the external magnetic field. Phase velocity of Alfvén waves depends on an angle $\theta << 1$. Using typical values of concentration of the neutral components N_n in the *E*-layer of ionosphere for large-scale magneto-acoustic waves we obtain $V_{ph\,M} = V_a = 1 \div 2$ km$\cdot$s^{-1}. For slow planetary waves with the period of $T_0 = 2$ days, horizontal wavelength $\lambda_x \approx 3000$ km, $k_x = 10^{-6}$ m^{-1} and the wave number $k_z = 2\pi / T_0 V_a$ we get $\sim 10^{-8}$ m^{-1}, $\theta \approx 10^{-2}$.

The phase velocity of large-scale, slow Alfvén waves is of the order of $V_{ph\,A} = 20$ m$\cdot$s^{-1}. For other periods T_0 and the horizontal wavelengths of planetary waves, the phase velocity of slow Alfvén type waves does not exceed typical values of the wind velocities in the *E*-layer of ionosphere (from several meters per sec up to $100 \div 300$ m$\cdot$s^{-1}). Alfvén type waves propagate transversally to the external magnetic field H_{0z} $(\theta << 1)$ and the total wave vector $\mathbf{k}$ is almost horizontal. If we take into account dispersion (Hall's effect), planetary waves are

circularly polarized. For typical values of the wind velocity V in the E-region of ionosphere using the formula $h \approx H_{0z} V / V_{phA}$ we can estimate perturbation of the geomagnetic field h varying within the range of 15-50 nT. Experimental results have confirmed existence of planetary waves with velocities of 20-100 $\mathrm{m \cdot s^{-1}}$) in E -layer of ionosphere [13, 14] in any season of year with wavenumber of 2-10 $\mathrm{m^{-1}}$. In contrast with the ordinary Rossby waves, they lead to the substantial distortion of the geomagnetic field (from several to several tens of nT) revealing electromagnetic character of these waves. Large-scale disturbances (with the velocity of 1-2 $\mathrm{km \cdot s^{-1}}$) can be identified with magneto-acoustic waves in the neutral component of the ionosphere, and the planetary waves with the velocity of the motion 20-100 $\mathrm{m \cdot s^{-1}}$ can be identified with slow Alfven waves.

Hence, in the E region of ionosphere are exist: magneto-acoustic wave with frequency $\omega_M = V_a k_x$, Alfven wave with frequencies ω_{A*} and $\omega_A = V_a k_z$, slow cyclotron wave of ions Ω_i and helicons with frequency ω_h.

Wave equation for the electric field $\mathbf{E}$ with the current density $\mathbf{j}$ (Equation (3)) has the following form: $rot\, rot\, \mathbf{E} - k_0^2 \mathbf{E} = -k_0^2 c^2 \mathbf{w} / V_a^2$. Multiplying Equation (3) scalar and vector on vector $\boldsymbol{\tau}$ we obtain

$$\mathbf{w} = \frac{1}{b}\left\{ \mathbf{E} - g^2 (\mathbf{E} \cdot \boldsymbol{\tau})\, \boldsymbol{\tau} - i g \left[\mathbf{E} \cdot \boldsymbol{\tau} \right] \right\}, \qquad (9)$$

where: $b = g^2 - 1$. Electric induction linearly connecting with the vector $\mathbf{w}$, $\mathbf{D} = -c^2 \mathbf{w} / V_a^2$, allow to calculate components of the permittivity tensor of magnetized plasma describing slow MHD waves in the coordinate system when Z-axis is directed along the line of forces of geomagnetic field $\mathbf{H_0}$, $E_z = 0$, which readily yield:

$$\varepsilon_{xx} = \varepsilon_{yy} = \frac{c^2}{V_a^2} \frac{1}{1-g^2}, \quad \varepsilon_{xy} = -\varepsilon_{yx} = i \frac{c^2}{V_a^2} \frac{g}{1-g^2},$$

$$\varepsilon_{xz} = \varepsilon_{yz} = \varepsilon_{zx} = \varepsilon_{zy} = 0, \quad \varepsilon_{zz} = \infty.$$

These expressions at $\eta = 1$ have been obtained in [15] on the bases of the equations of two-fluid hydrodynamics of cold plasma taking the ion inertia into account but neglect the electron inertia and particle collision.

On the other hand using the expression $\mathbf{D} = \varepsilon \mathbf{E} = 4\pi i \mathbf{j} / \omega$ [16] (here ε is scalar) and substituting Equation (3) we get:

$$\left(\varepsilon - \frac{c^2}{V_a^2} \right) \mathbf{w} = -i g \varepsilon [\mathbf{w} \cdot \boldsymbol{\tau}]. \qquad (10)$$

Multiplying this equation scalar and vector on vector $\boldsymbol{\tau}$ and taking into account $(\mathbf{w} \cdot \boldsymbol{\tau}) = 0$ we yield $\varepsilon = c^2 / V_a^2 (1 \pm g)$. Transversal wave propagates in medium with the velocity $\omega / k = c / N$, where $N = \sqrt{\varepsilon_\perp}$ is the refractive index. Hence phase velocity of transversal MHD wave is $V_{ph}^2 = V_a^2 (1 \pm g)$. This means that fast and slow Alfvén waves are circularly polarized due to Hall's effect.

3 Second order statistical moments of the Alfven wave in the turbulent plasma flow

Statistical characteristics of waves depend on both correlation features of the fluctuating parameters of a randomly inhomogeneous medium and the type of waves [6]. Frequency of the Alfven wave propagating along the external magnetic field is defined as $\omega = \pm V_a k_z$, upper and lower signs correspond to the "fast" and "slow" Alfvén's waves, respectively. Let small amplitude E_0 low frequency $\omega_0 << \omega_i$ monochromatic plane wave with the wave vector $\mathbf{k_0}$ generating in the $Z = 0$ plane propagates along the Z - axis. In the low pressure plasma Alfvén velocity exceeds thermal velocities of particles. Let's turbulent plasma stream with constant velocity $\mathbf{V_0}$ moves along the external magnetic field $\mathbf{B_0}$ locating in the XZ plane (principle plane) of the Cartesian coordinate system with the angle of inclination θ with respect to the Z-axis. In this case eikonal equation is $\omega - (\mathbf{k V}) = \pm V_a (\mathbf{k b})$; $\mathbf{b}$ - the unit vector along an external magnetic field; We suppose that in turbulent ionospheric plasma density fluctuations exceed velocity pulsations $V_1 / V_a << N_1 / N_n << 1$ ($V_1(\mathbf{r}, t)$ represents small turbulent pulsations of the macroscopic velocity). Frequency and wave number of the Alfvén wave in turbulent plasma with smooth spatial-temporal fluctuations of the neutral component density satisfy the conditions, i.e. $k_0 l >> 1$, $\omega_0 T >> 1$ and $\omega_0 l / V_0 >> 1$ (l and T are characteristic spatial-

temporal scales of irregularities) and therefore we can apply ray (–optics) approximation [17]. Neutral particles velocity and density are expressed as sum of the regular and fluctuating components which are slowly varying random functions of the spatial coordinates and time $\mathbf{V}(\mathbf{r},t)=\mathbf{V}_0+\mathbf{V}_1(\mathbf{r},t)$, $N_n(\mathbf{r},t)=N_0+N_1(\mathbf{r},t)$. Substituting the wavevector $\mathbf{k}(\mathbf{r},t)=-\nabla\varphi$ and the frequency $\omega(\mathbf{r},t)=\partial\varphi/\partial t$ in the dispersion equation of the Alfven wave, taking into account that the phase is sum of the regular $\varphi_0=\omega_0 t-k_0 z$ and fluctuating phases, $\varphi(\mathbf{r},t)=\varphi_0(\mathbf{r},t)+\varphi_1(\mathbf{r},t)$ ($\varphi_1<<\varphi_0$), we obtain stochastic eikonal equation for the phase fluctuation [7]:

$$\frac{\partial\varphi_1}{\partial t}+(\mathbf{V_{gr}}\cdot\nabla\varphi_1)=\mp\frac{1}{2}k_0 V_{a0}\cos\theta\,\frac{N_1}{N_0}, \quad (11)$$

where: $\mathbf{V_{gr}}=(V_0\pm V_{a0})\,\mathbf{b}=V_*\mathbf{b}$ is the group velocity of the Alfven wave, $V_{a0}=H_0/\sqrt{4\pi M N_{n0}}$. This equation easily solved using the method of characteristics:

$$\varphi_1(\mathbf{r},t)=p\int_0^L dz'\,N_1(x',y,z',t'), \quad (12)$$

where: $x'=x-(z-z')\,tg\theta$, $t'=t-\dfrac{z-z'}{V_*\cos\theta}$,

$p=\mp\dfrac{\omega_0 V_{a0}}{2N_0V_*^2\cos\theta}$ [11]. Correlation function of the phase fluctuations of scattered Alfvén wave in the turbulent plasma flow has the following form:

$$V_\varphi^{(A)}(\rho_x,\rho_y,L)=<\varphi_1(x+\rho_x,y+\rho_y,L)\,\varphi_1^*(x,y,L)>=$$
$$=2\pi p^2 L\int_{-\infty}^{\infty}dk_x\int_{-\infty}^{\infty}dk_y\int_{-\infty}^{\infty}d\omega$$
$$\cdot\,W_N\left[k_x,k_y,\frac{\omega}{V_*\cos\theta}-k_x\,tg\theta,\omega\right]\cdot$$
$$\cdot\exp(i\,k_x\rho_x+i\,k_y\rho_y), \quad (13)$$

the angle brackets indicate an ensemble average, the $*$ denotes a complex conjugate, L is a distance travelling by wave in turbulent plasma, $W_N(\mathbf{k},\omega)$ is arbitrary spectral function of plasma density fluctuations, ρ_x and ρ_y distances between observation points in the *XY* plane. Knowledge of the variance of the phase fluctuation allows estimate attenuation of the amplitude of an incident wave in turbulent plasma caused by energy transform from the mean field to the scattered one using the well-known formula [18]: $<E>=E_0\exp(-<\varphi_1^2>/2)$.

In contactless diagnostics of the nonstationary plasma the most important is the temporal spectrum of scattered waves. The variance of an instant frequency $<\omega_1^2>$ determines the broadening of the temporal power spectrum easily measuring by experiment. It can be obtained from Equation (13) multiplying integrand on the factor ω^2. Violation of coherence of a scattered field in medium with large-scale irregularities connecting with the phase fluctuations allows us to suppose that $<\omega_1^2>$ keeps the sense in the presence of diffraction too. Curvature of a constant phase surface in turbulent plasma is connected with the fluctuations of the unit vector **s** perpendicular to the wave front: $<s_{1x}^2>=<(\partial\varphi_1/\partial x)^2>/k_0^2$. Both statistical characteristics $<s_{1x}^2>$ and $<s_{1y}^2>$ determine the angle-of-arrival of scattered waves in the *XY* plane.

The obtained statistical characteristics of slow Alfvén wave are valid for arbitrary correlation function of the density fluctuations of the neutral component taking into account: anisotropy factor of irregularities, the angle of inclination of prolate irregularities with respect to the external magnetic field, the angle between wave vector of an incident wave and the external magnetic field, regular velocity and characteristic spatial-temporal scales of the density fluctuations characterizing turbulent plasma flow.

The most important problem of waves propagation in a nonstationary medium is the problem of energy exchange between wave and medium. Specific features arise at propagation of the Alfven waves in plasma flow with chaotically varing parameters. Neglecting dissipation processes in the geometrical optics approximation amplitude E satisfies transport equation [17]:

$$\frac{\partial}{\partial t}(\eta E^2)+div(\mathbf{V_{gr}}\cdot\eta E^2)=-\frac{\partial\varepsilon}{\partial t}E^2, \quad (14)$$

where: $\eta=\dfrac{1}{\omega}\dfrac{\partial}{\partial\omega}(\omega^2\varepsilon_{xx})$ is the coefficient between the energy density and E^2 [17], $\varepsilon_{xx}=\dfrac{c^2}{V_a^2}\dfrac{1}{1-g^2}$. For low-frequency MHD wave we obtain:

$$\eta = \frac{c^2}{V_a^2}\frac{2}{(1-g^2)^2},\ \frac{\partial \varepsilon}{\partial t} = \frac{c^2}{V_{a0}^2(1-g^2)}\frac{1}{N_0}\frac{\partial N_1}{\partial t}. \quad (15)$$

On the other hand the scattered field is connected with the log-amplitude by the relation $\chi = \ln(E/E_0)$ [18, 19]. It can be shown that the mean square of log-amplitude wave fluctuations $<\chi_1^2>$ for slow Alfven wave is:

$$<\chi_1^2> = \frac{<\omega_1^2>}{4\omega_0^2}\frac{V_*^2}{V_{a0}^2}. \quad (16)$$

Investigate energy exchange between the Alfven waves and nonstationary plasma flow. The mean energy flux density is $<\mathbf{S}> = <\eta E^2 \mathbf{V_{gr}}>$. Growth and decrease of the energy flow in the turbulent plasma means the energy transfer from medium to the wave and vice versa. For Z component we have $<S_z> = S_{z0} + <S_z>_2$. Using Equation (14) for the second term we obtain

$$<S_z>_2 = \pm\frac{E_0^2}{N_0\omega_0}\frac{V_*}{V_{a0}}\frac{c^2 p L}{V_{a0}^2(1-g^2)}\int_{-\infty}^{\infty} d\rho_z$$

$$\frac{\partial^2}{\partial\tau^2}W_N(\rho_x, 0, \rho_z, \tau)\ , \quad (17)$$

where integral should be taken along characteristics: $\rho_x = \rho_z\, tg\theta$, $\tau = \rho_z / V_* \cos\theta$. This expression is connected with the broadening of the temporal spectrum and finally we obtain

$$<S_z> = 2E_0^2\frac{c^2}{V_{a0}^2}\frac{V_*\cos\theta}{(1-g^2)^2}\left[1+(1-g^2)\frac{V_*^2}{V_{a0}^2}\frac{<\omega_1^2>}{\omega_0^2}\right] \quad (18)$$

From this formula follows that growth or decrease of the energy flow of the Alfven wave in the turbulent plasma substantially depend on: the group velocity of this wave V_*, the degree of ionization of the ionosphere η and frequency of an incident wave frequency ω_0. If $V_0 < V_{a0}$ and $g^2 < 1$, energy of the fast Alfven wave growth and decreases for the slow one; at $g^2 > 1$ - vice versa. If $V_0 > V_{a0}$, regular velocity and nonstationarity of plasma has identical influence on both fast and slow Alfven waves leading to the growth of the energy flow at $g^2 < 1$, and substantially decreases at $g^2 > 1$. The relationship $<S_x> = tg\,\theta <S_y>$ is a consequence of anisotropy of the task.

It should be noted that if the condition: $|V_0 - V_{a0}| << V_{a0}$ for Alfvén wave is fulfilled, the wavelength becomes very small and application of the geometrical optics approximation to the slow large-scale MHD waves is violated. All above derived formulae are valid for the angles θ not close neither zero nor $\pi/2$, because at small angle θ phase velocities of the Alfvén and magnetoacoustic waves approximately coincide and strong linear interaction takes place [17] which we did not take into account. The second-order statistical moments calculating in the ray (-optics) approximation not include diffraction effects.

4 Numerical calculations

Observations of ionospheric irregularities detecting by radio wave sounding of the lower *E*-region (altitudes near 100 km) have shown [20] that the speeds and the horizontal spatial scales of the dominant irregularities are in the range 30 and $160\,\mathrm{m\cdot s^{-1}}$ and 10 and 75 km, respectively; with the corresponding average values being near 80 $\mathrm{m\cdot s^{-1}}$ and 30 km. The mean drift speed in the *E*-region of ionosphere is of an order 100-150 $\mathrm{m\cdot s^{-1}}$ depending on geomagnetic activity. Below 110 km drift velocity coincides with the wind speed; above 130 km ionized component drifts towards the direction of an external magnetic field. In the plane perpendicular to a geomagnetic lines of force drift speed by an order of magnitude is less than a speed of the wind. Large-scale anisotropic irregularities have been observed in the *E*-region of ionosphere. Horizontal spatial scale of these irregularities is about 150-200 km. They generated due to wavy movements of an internal waves. Inhomogeneous structure of the ionosphere is investigated using the space diversity techniques. Observations have shown that anisotropic factor of irregularities at $\chi < 5$ is not connected with the geomagnetic field [21], but substantial elongation $\chi \geq 10$ is defined by it. Velocities of irregularities movement are in the range of $40 \div 160\ \mathrm{m\cdot s^{-1}}$; the most probable drift speed is $\sim 100\ \mathrm{m\cdot s^{-1}}$ that is an agreement with other experimental data. The variance of concentration $\sigma_N^2 = <N_1^2>/N_0^2$ has been measured using pulse and radio-astronomical methods.

Observations of the *E*-region have shown that characteristic linear scale of irregularities is about 1-2 km and $\sigma_N^2 \sim 10^{-4}$ [22].

Analytical and numerical calculations are carried out for anisotropic Gaussian correlation function of density of the neutral components having in the principle plane following form [22]:

$$\tilde{V}_n(k_x,k_y,k_z) = \sigma_n^2 \frac{l_\perp^2 l_\parallel}{8\pi^{3/2}} \cdot$$

$$\cdot \exp\left(-\frac{k_x^2 l_\perp^2}{4} - p_1 \frac{k_y^2 l_\parallel^2}{4} - p_2 \frac{k_z^2 l_\parallel^2}{4} - p_3 k_y k_z l_\parallel^2 \right), \quad (19)$$

where: $p_1 = (\sin^2\gamma_0 + \chi^2\cos^2\gamma_0)^{-1}\left[1+(1-\chi^2)^2 \cdot \right.$

$\left. \cdot \sin^2\gamma_0 \cos^2\gamma_0 / \chi^2 \right]$, $p_2 = (\sin^2\gamma_0 + \chi^2\cos^2\gamma_0)/\chi^2$,

$p_3 = (1-\chi^2)\sin\gamma_0\cos\gamma_0/2\chi^2$, σ_n^2 is variance of electron density fluctuations. This function contains anisotropy factor of irregularities $\chi = l_\parallel / l_\perp$ (ratio of longitudinal and transverse linear scales of plasma irregularities) and inclination angle γ_0 of prolate irregularities with respect to the external magnetic field.

Temporal pulsations of plasma density fluctuations leading to the broadening of the temporal spectrum are analyzed first time. Figure 1 depicts the dependence of the normalized variance of the frequency fluctuations characterizing broadening of the temporal power spectrum versus non-dimensional parameter $M_A = l_\parallel / V_* T$ containing all characteristic spatial-temporal scales of the turbulent plasma flow. Using experimental data: $\theta = 30^0$, $\chi = 3$, $\gamma_0 = 15^0$, $V_{a0} = 50\ \mathrm{m\cdot s^{-1}}$, $V_0 = 100\ \mathrm{m\cdot s^{-1}}$, $l_\parallel = 30\ \mathrm{km}$ numerical calculations show that at: $X = Y = 0.01$ (observation points are spaced apart at distances $\rho_x = \rho_y = 300\ \mathrm{m}$), maximum of the temporal spectrum of scattered Alfven wave in the turbulent plasma flow is at the frequency $\nu_{\max} = 6\ \mathrm{mHz}$ and the frequency band of a half width of the temporal spectrum is equal $\Delta\nu = 20\ \mathrm{mHz}$. Increasing distance between observation points, $X = Y = 0.08$ ($\rho_x = \rho_y = 2.4\ \mathrm{km}$), maximum and half width of the temporal spectrum are equal to: $\nu_{\max} = 4\ \mathrm{mHz}$, $\Delta\nu = 10\ \mathrm{mHz}$, respectively. At $X = Y = 0.1$ ($\rho_x = \rho_y = 3\ \mathrm{km}$), $\nu_{\max} = 3\ \mathrm{mHz}$, $\Delta\nu = 6\ \mathrm{mHz}$.

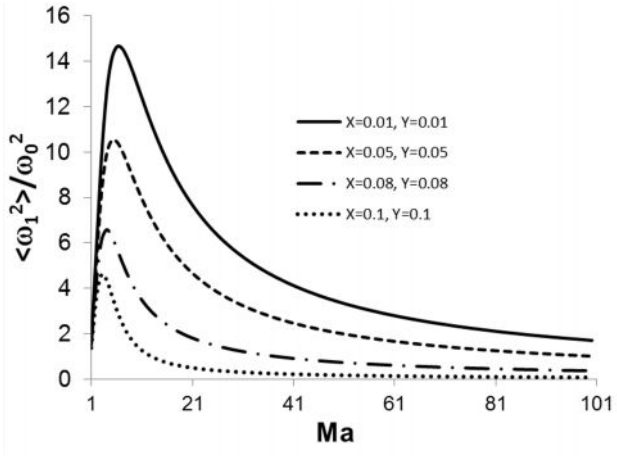


Fig. 1 illustrates broadening of the temporal power spectrum of scattered Alfven wave in the turbulent plasma flow versus non-dimensional spatial-temporal parameter M_A characterizing nonstationary plasma for the anisotropic Gaussian spectrum and different locations of the observation points.

These problems have a direct relation to the problem of generation, detection and propagation of VLF radiation having great geophysical applications. The obtained results may be applied also at investigation of solar and galactic plasma.

5 Conclusion

It was established that in the weakly ionized *E* region of ionosphere exist: magneto-acoustic wave with frequency $\omega_M = V_a k_x$, Alfvén wave with frequencies ω_{A*} and $\omega_A = V_a k_z$, slow cyclotron wave of ions Ω_i and helicons with frequency ω_h.

Slow Alfvén waves with the phase velocity of the order of $20\ \mathrm{m\cdot s^{-1}}$ propagating transverse to the external magnetic field lead to the substantial distortion of the geomagnetic field within the range of 15-50 nT. Experimental observations confirmed the existence of planetary waves with velocities of 20-100 $\mathrm{m\cdot s^{-1}}$ in the *E* -layer of ionosphere in any season of year with wavenumber of 2-10 $\mathrm{m^{-1}}$. Planetary waves with the velocity of the motion 20-100 $\mathrm{m\cdot s^{-1}}$ can be identified with slow Alfvén waves; fast and slow Alfvén waves are circularly polarized due to Hall's effect.

Correlation functions of the phase fluctuations of scattered Alfven wave, broadening of the temporal power spectrum and the mean square of log-amplitude fluctuations describing angle-of-arrival of scattered Alfvén wave in a turbulent plasma flow have been obtained in the ray (optics) approximation for arbitrary correlation function of the density fluctuations of the neutral component taking into account: anisotropy factor of irregularities, the angle of inclination of prolate irregularities with respect to the external magnetic field, the angle between wave vector of an incident wave and the external magnetic field, regular velocity and characteristic spatial-temporal scales of the density fluctuations characterizing turbulent plasma stream. Knowledge of the variance of the phase fluctuation allows estimating attenuation of the amplitude of an incident wave in turbulent plasma caused by energy transformation from the mean field to the scattered one.

Energy exchange between Alfven wave and turbulent plasma flow is investigated analytically and numerically using experimental data. Characteristic frequencies of the temporal pulsations of plasma irregularities leading to the broadening of the temporal spectrum are calculated first time. Statistical parameters of scattered Alfven wave substantially depend on the ration of Alfven velocity and macroscopic velocity of a plasma flow.

6 References

[1] Khantadze A.G., Jandieri G.V., Ishimaru A., Diasamidze Zh.M. Electromagnetic oscillations of the Earth's upper atmosphere (review), Annales Geophysicae, vol. 28, pp. 1387-1399, 2010.

[2] Jandieri G.V., Ishimaru A., Gavrilenko V.G., Surmava A.A., Gvelesiani A.I., On the features of magnetogradient planetary waves in the approximation of the spherical symmetry of the ionosphere, The Open Atmospheric Science Journal, vol. 5, pp. 33-42, 2011.

[3] Kamide Y, Baumjohann W., Magnetosphere-ionosphere coupling, Berlin: Springer, 1993.

[4] Sorokin V.M., Fedorovich G.V., Physics of slow MHD waves in the ionospheric plasma, Nauka, Moscow, 1982 (in Russian).

[5] Aleksandrov A.F., Bogdankevich L.S., Rukhadze A.A.. Electrodynamics of Plasma, Moscow: Higher Educational Institution, 1988 (in Russian).

[6] Kravtsov Yu.A., Ostrovsky L.A., Stepanov N.S. Geometrical optics of inhomogeneous and nonstationary dispersive media, Proc. IEEE Vol. 62, # 11, pp. 1492-1510, 1974.

[7] Jandieri G.V., Gavrilenko V.G. Semerikov A.A. To the theory of magnetohydrodynamic waves propagation in turbulent plasma stream, Plasma Physics Report, vol. 11, # 10, pp. 1193-1198, 1985.

[8] Khantadze A.G., On the dynamics of conducting atmosphere, Nauka, Tbilisi, 267 p., 1973 (in Russian).

[9] Kelley M.C. The Earth's ionosphere, Academic, San Diego, Calif., 1989.

[10] Alperovich L.S., Fedorov E.N., Hydromagnetic waves in the magnetosphere and the ionosphere, Springer, 425 p., 2007.

[11] Monin A.S., Obukhov A.M., Small oscillations of the atmosphere and adaptation of meteorological fields, Izv.,Acad. Nauk Geol., vol. 11, pp. 1360-1373, 1958.

[12] Jandieri G.V., Khantadze A.G., Aburjania G.D., Mechanism for the generation of a vortex electric field in the ionospheric E-region, Plasma Physics Report, vol. 30, # 1, pp. 83-90, 2004.

[13] Khantadze A.G., Sharadze Z.S. "Ionospheric effects of planetary waves". In Wave Disturbances in Atmosphere. Alma Ata: Nauka, pp. 143-158, 1980.

[14] Khantadze A.G., Sharadze Z.S., Kobaladze Z.L. About planetary Alfven types waves in ionosphere. In Research of Dynamical Process in Upper Atmosphere. Moscow, Nauka, pp. 110-116, 1988.

[15] Dmitrenko I.S., Mazur V.A. On waveguide propagation on Alfven waves at the plasmapause, Planetary and Space Scinece, vol. 33, # 5, pp. 471-477, 1985.

[16] Kadomtsev B.B., Collective phenomena in plasma, Pergamon Press, New York, 1982, 69.

[17] Kravtsov Yu.A., Orlov Yu.I, Geometrical optics of inhomogeneous media, Moscow, Nauka, 1980 (in Russian).

[18] Rytov S. M., Kravtsov Yu. A., Tatarskii V. I., Principles of Statistical Radiophysics. vol.4. Waves Propagation Through Random Media. Berlin, New York, Springer, 1989.

[19] Ishimaru, A. Wave Propagation and Scattering in Random Media, Vol. 2, Multiple Scattering,

Turbulence, Rough Surfaces and Remote Sensing, IEEE Press, Piscataway, New Jersey, USA, 1997.

[20] Vincent R.A., Ionospheric irregularities in the E-region, Journal of Atmospheric and Solar Terrestrial Physics, vol. 34, pp. 1881-1898, 1972.

[21] Kokurin Yu.L. Shape and movements of small irregularities in the ionosphere, In the Special Issue "Drift and irregularities in the ionosphere, # 1, pp. 62-71, 1959.

[22] Jandieri G.V, Ishimaru A., Jandieri V.G., Khantadze V.G., Diasamidze Zh.M., "Model computations of angular power spectra for anisotropic absorptive turbulent magnetized plasma," Progress In Electromagnetics Research, PIER, vol. 70, pp. 307-328, 2007.

Building Fuzzy Inference System in the Political Domain

Dr. Sameera Alshayji, Nasser Al-Sabah, and Abdulla Al-Sabah
Political and Economic Affairs Department, Amiri Diwan, Seif Palace, Kuwait

Abstract - *The world's increasing interconnectedness and the recent increase in the number of notable regional and international events pose ever-greater challenges for political decision-making. This is especially true when considering whether to strengthen bilateral economic relationships between nations, as such critical decisions are influenced by certain factors and variables that are based on scattered, heterogeneous, and vague information. A common language is thus needed to describe variables that require human interpretation. Applying a fuzzy ontology method is one of the possible solutions that address integration of information and lack of clarity of concept. Fuzzy logic is based on natural language and is tolerant of imprecise data. Furthermore, a Fuzzy Inference System's (FIS) greatest strength lies in its ability to handle imprecise data. This research focuses on developing a fuzzy inference in the political domain. In addition, this paper will highlight the sense of thinking in some political centers in the region.*

Keywords: Fuzzy logic, FIS, Ontology, Fuzzy-logic-based ontology, Political centers

1 Introduction

1.1 Overview

Considerable knowledge has been generated, organized, and digitized in various governmental sectors, but the political field still needs to be more organized for decision-makers. Most political terms are language-based and need to be interpreted. For example, existing relationships between countries can be described from a variety of perspectives, such as historical, respectful, and neighboring. A conscientious decision-maker who takes responsibility for promoting and strengthening bilateral economic relationships needs access to well-structured information that is relevant to his/her decisions.

1.2 Current challenges

Unfortunately, in reality, the basic concept of this information is a linguistic variable, that is, a variable with values in words rather than numbers, including the political and investment domains. This makes it extremely difficult for the decision-maker to understand the concepts that exist in these domains. For example, Alshayji et al. [5] identified some concepts that influence decisions to strengthen economic relationships with other countries, such as the agreements concept [4], the nuclear affairs concept, the peace in the Middle East concept, also presented by Alshayji et al. [7]. The decision-maker considering whether to strengthen economic relationships requires structured information. Examples of information that may be assessed in the decision-making process include competency questions: Is country x good or weak in terms of political stability? What type of bilateral relations does my country have with country x? The answers may involve a description such as good, weak, high, medium, and so on. In addition, most states go through several fluctuations of the variable of political stability. In this situation, the political decision-maker could describe political stability in several phases as "very good" at a specific time, "good" in another time, or "weak" at the current time.

1.3 Problem formulation

A serious problem that the decision-maker faces is the difficulty of building an efficient political decision support system (DSS) with heterogeneous and vague information in the political and investment domains, especially the decision to strengthen bilateral economic relationships with friendly nations. Typically, these critical decisions are influenced by certain factors and variables that are based on heterogeneous and vague information that exists in different domains. Most of the political decision maker's documents use linguistic variables whose values are words rather than numbers and therefore are closer to human intuition. A natural language is needed to describe such information, which requires more human knowledge for interpretation.

1.4 Political centers

Political and diplomatic research centers (also known as think tanks) play a very large role in driving and shaping a country's domestic and international policy issues. Alshayji et al. [7] have highlighted the need to establish a center for the political decision-makers in the government. As such, many governments around the world depend on such think tanks to provide analysis and recommendations that help policymakers make domestic and foreign policy decisions. For the most part, think tanks in the Arab world are perceived as being repressed or carefully chosen by authoritarian regimes to push their agenda. Furthermore, the general idea is that they are seen as being politically neutral [16]. In the Gulf region of the Middle East, such policy-oriented research institutions are finding themselves under pressure and are being heavily scrutinized by the Gulf Cooperation Council (GCC)

governments due to the range of sensitive topics such think tanks are debating over. The uprisings and revolts brought about by the Arab Spring have led to much dialogue and discourse for change among the citizens of the Arab world. As think tanks also serve as a system for listening to the voices and ideas of the citizens, the GCC countries were not comfortable with the sensitive topics being discussed and examined at the research institutions [14].

Kuwait currently has two research institutes, the Diplomatic Center for Strategic Studies and the Kuwait Center for Strategic and Future Studies. The latter is affiliated with Kuwait University, which is a government-owned university. According to their website, its main objective is "to spread awareness and to encourage communication on critical issues, in addition to organizing discussion and debates." Despite its affiliation with Kuwait University, it does not play any role in providing recommendations or analysis to the Kuwaiti government. The other institute mentioned, the Diplomatic Center for Strategic Studies, was formed by former diplomat, Abdullah Bishara. This institute also does not play any significant role in helping the Kuwaiti government in domestic or foreign policy decision making either.

According to Think Tank Watch, Qatar is the least restrictive Gulf state in terms of its treatment of academic and media debates; and it is the home to 10 think tanks. Qatar has opened their doors to prominent multinational think tanks, and has set up The Brookings Doha Center, The RAND-Qatar Policy Institute and The Center for International and Regional Studies at Georgetown University's Qatar branch, among others [3]. The political research centers mentioned play a major role in providing Qatar's government with research and analysis, and are directly involved in the policymaking process.

On the contrary, other Gulf states are not on the same page as Qatar with regard to think tanks. Since the Arab Spring, the United Arab Emirates (UAE) has exercised more control over public debates, and therefore The Dubai School of Government has had a difficult time operating its research center [21]. It was set up in 2005, in a partnership with Harvard University. However, Dubai's government has recently reduced funding to the center due to the sensitive and controversial political topics that were being discussed. Furthermore, the dean and other researchers at the think tank have resigned from their posts due to this censorship being imposed. Even though the 2014 Press Freedom Index (published by Reporters Without Borders) ranks Qatar far behind Kuwait and the UAE in terms of the freedom that the citizens, journalists and news organization enjoy in each country, it seems that Qatar is the only GCC state that has active and legitimate political research centers. During a 2011 conference organized by Oman, titled "First GCC Think Tank Development Dialogue", the small percentage of think tanks in the region when compared to think tank figures around the world was pointed out. Emphasis was placed on the urgent need to increase the role of think tanks in the region. As Windecker [22] has argued that, "Scientific research and decision-making are still worlds apart in the region" and that the time has come to "bridge the gap through a more active role of think tanks." There is a need for political research centers in the GCC region to be more advanced and brought to the awareness of the public.

1.5 Proposed solutions

A popular way to handle scattered data is to construct the so-called fuzzy ontology as presented by Inyaem et al. [12]. The fuzzy membership value μ is used for the relationship between the objects in question, where $0<\mu<1$ and μ corresponds to fuzzy membership relationships such as "low," "medium," or high" for each object. The purpose of fuzzy control is to influence the behavior of a system by changing the inputs to that system according to the rule or set of rules under which that system operates. The purpose of applying fuzzy systems is to enable one to weigh the consequences (rule conclusions) of certain choices based on vague information.

1.6 Contribution knowledge

The fuzzy inference system contributes to understanding the context and perspectives that are important to understanding the impact of political variables on strengthening bilateral economic relationships. The proposed technique efficiently utilizes algorithms to access, integrate, and manage the contributed information at the international level. Using object paradigm ontology and Protégé-OWL methods to contribute to understanding the domain as well as the relation between objects, the technique also contributed significantly to simplifying the concept by extracting the main variables that affect the decision process [5]. These methods facilitate implementation. In addition, they enhance the clarity of the natural concepts and encourage us to shed light on other, more difficult domains like parliament. Utilizing fuzzy logic contributed to the understanding of linguistic and imprecise data. The utilization of the fuzzy cognitive mapping (FCM) scheme provides insight into and better understanding of the interdependencies variables (vague data). FIS is a high-level technical computing language and interactive environment for algorithm development, data visualization, data analysis, and numeric computing. Its contribution lies in the secret of the calculations that automate dealing with imprecise language and vague information.

2 Methodology

2.1 Introduction

Ontology facilitates the communication between the user and the system, and the success of the information systems is based on integration of information. Different methodological approaches for building ontology have been proposed in the literature [9], [10], [11], [17].

2.2 Proposed ontology

Two approaches are described in this paper. The first is adopted from the ontology modeling approach of Noy and McGuinness [17] and Fernandez-Lopez [11]. The second approach is adopted from Inyaem et al. [12] The main framework is to complete the construction of fuzzy ontology for a specific domain through the following steps: 1) input unstructured data; 2) specify the definition of related concepts in the domain and their relationships; 3) clarify the generation of domain ontology; 4) extend the domain ontology to fuzzy ontology; and 5) apply the fuzzy ontology to the specific domain. Figure 1 depicts the complete process of the construction of fuzzy ontology [12].

We will use the same developed model of fuzzy ontology for several reasons: 1) the authors used this model in the terrorism domain, which is considered an integral part of the political domain because terrorism undermines political stability; it is a part of political variables such as "stability" and "terrorism" and 2) the author used linguistic variables and ambiguous concepts that are roughly equivalent to vague variables used in the political domain.

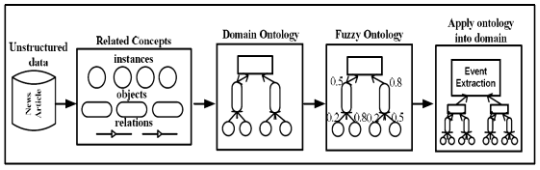


Figure 1: Process of Construction of Fuzzy Ontology for the Specific Domain (Inyaem et al., 2010)

However, more sub-steps (processes) will be added within the main steps used by the Inyaem model [12]. The five new processes (sub-steps) are as following: 1) construct object paradigm (OP) ontology; 2) apply ontology language OWL-editor from the World Wide Web Consortium (W3C); 3) construct fuzzy cognitive map theory (FCM); 4) apply fuzzy causal algebra method; and 5) apply fuzzy inference system (FIS). See figure 2.

2.3 Mechanism for using new sub-steps of fuzzy construction

Alshayji et al. [6] used an object paradigm (OP) ontology to identify important concepts and capture a high level for ontology conceptualization of knowledge to facilitate the work of decision processes [4, 5, 6, 8]. More details of OP were presented by Alasswad et al. [2]. Accordingly, this paper presents the concept of the loan by using an OP ontology, the OWL editing tools ontology, and then proceed to integrating fuzzy logic with ontology. Alshayji et al. [8] used OWL to present the concept in the political domain [4, 5]. More justification for using Protégé was presented by Alshayji et al. [5], Islam et al. [13], and Noy & Guinness [17]. On the other hand, the third and fourth processes, which involve Fuzzy Cognitive Mapping (FCM) and causal algebra, are especially applicable in the soft knowledge domains (e.g., political science, military science, international relations, and political elections at government levels). Alshayji et al. [7] demonstrated the causal inter-relationship between certain variables in the domain, such as "stability" and "terrorism," and processes of fuzzy ontology construction presented in investment domains and the agreement ontology in political domains, respectively [4, 5, 6, 8].

2.4 Justification for using new sub-steps of fuzzy construction

In this regard, and coinciding with the previously mentioned process, the new sub-steps are added for several key reasons: 1) to accelerate the application process for the construction of fuzzy ontology; 2) to simplify the extraction of the most variables that in some way affect the political decision-making process. Political decision-makers would thus be aided by a system that would allow them to formulate constructive rule conclusions by dealing with vague variables as described and drawing rule conclusions in the form of an IF-THEN: an if-antecedent (input) and then-consequent (output). Because of this situation, and along with FCM and causal algebra propagation, the fifth process includes displaying what is going on in the political mind in the form of a calculation through the use of fuzzy sets and linguistic models consisting of assets of IF-THEN fuzzy rules. Fuzzy systems enable one to weigh the consequences (rule conclusions) of certain choices based on vague information. Rule conclusions follow from rules composed of two parts: the "if" (input) and the "then" (output). Fuzzy logic toolbox graphical user interface (GUI) tools enable us to build a fuzzy inference system (FIS) to aid in decision-making processes. For the purposes of the ontology, we refer the readers to Alshayji et al. [8].

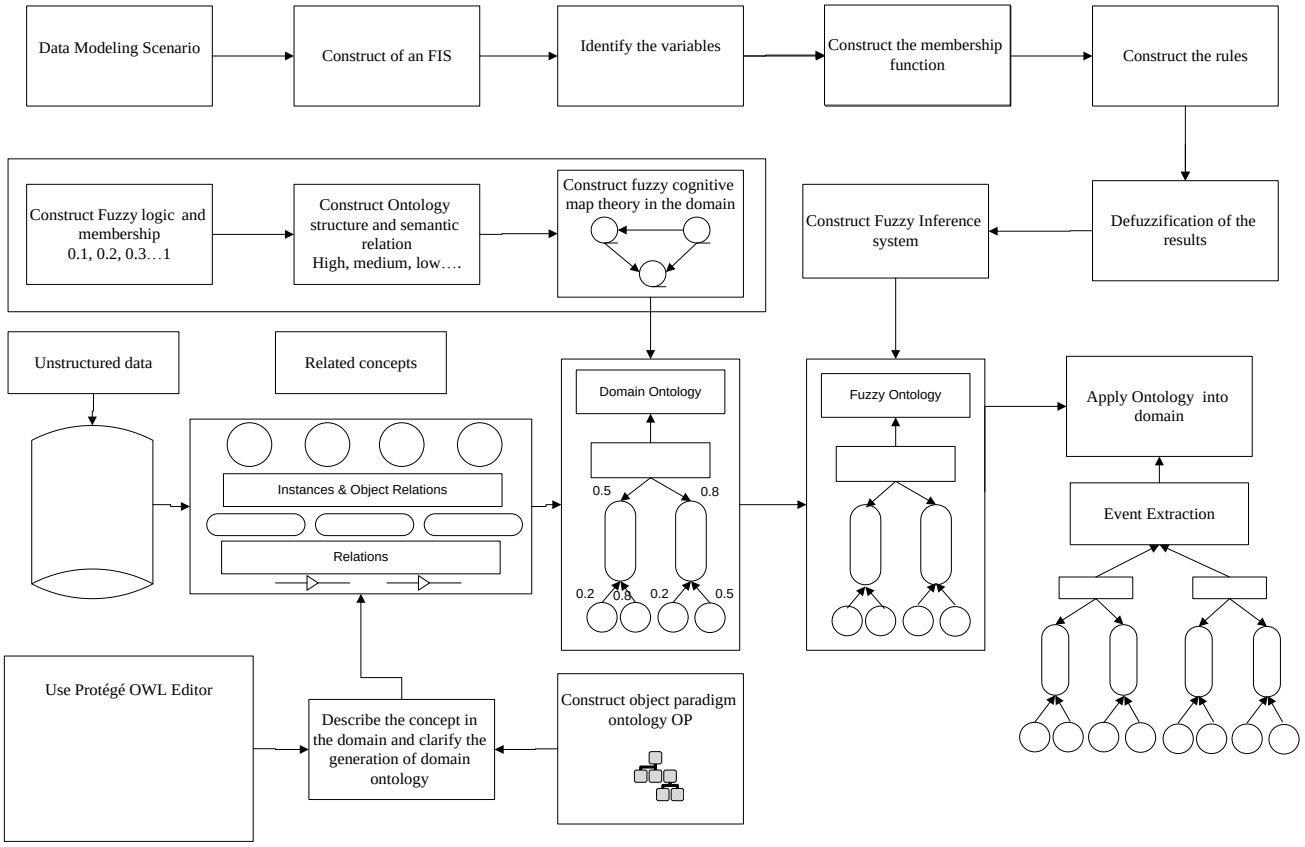


Figure 2: Process of Construction of Fuzzy Ontology with inference system for the Specific Domain

3 Specifying the definition of related concepts

The proper ontology concepts have been identified by Alshayji et al. [5], who used the loan as case study: To capture all concepts of a loan, the "Loan" class is linked to the "Date" class through the "hasDate" tuple type. In addition, to capture the number of the loan, the "Loan" class is linked to the "LoanNumber" class. To be more semantically precise, the engineering process links link with all concepts that related to loans.

3.1 Using OWL ontology

Also, using the OWL editing tool allows the user to integrate and describe the concept using linguistic variables. Figure 3 presents OWL ontology and Figure 4 presents object paradigm ontology.

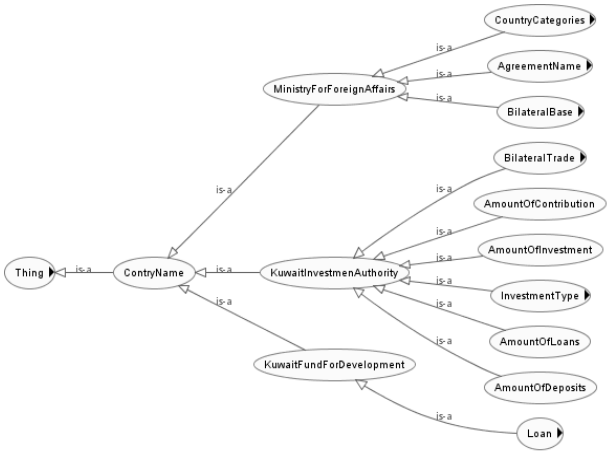


Figure 3: Using OWL to describe the concepts

Now at this stage we have 1) input unstructured data; 2) specified the definition of related concepts in the domain and their relationships; and 3) clarified the generation of domain ontology. In the next section, we will extend the domain ontology to fuzzy ontology.

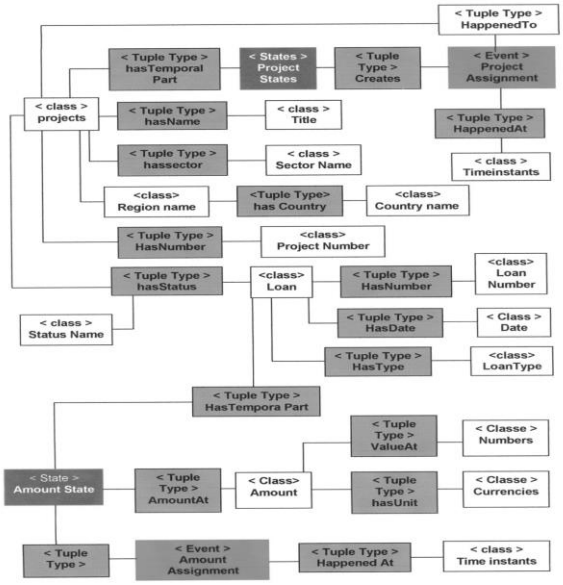


Figure 4: The OP loan ontology

4 Extending ontology to fuzzy ontology

At this point, it is important to understand and specify the classes in loan domain. Class specification includes attributes, and each attribute has a meaning. To generate fuzzy ontology, the "Loan Number," for example, has different attributes such as "Loan Name," "Loan Date," and so on, as presented in section 3.

4.1 Fuzzy Set and Membership

In this section, we will integrate fuzzy logic in our ontology. Fuzzy logic presented by Abulaish & Dey [1] and also Alshayji et al. [5] presented different properties. More fuzzy concepts in the same domain have been presented [5]. Integrating information with rich concepts undoubtedly helps political decision-makers make appropriate and correct decisions. Answering whether to "prevent" or "reduce" the bilateral economic relationships requires also considering the concept of "investment indicator."

4.2 Fuzzy Cognitive Map Theory

FCM is a fuzzy-graph structure for representing causal reasoning with a fuzzy relationship to a causal concept [7]. Justification for its use is described in subsections 1.6 and 2.3; more justification can found in the literature [18], [7]. Signed fuzzy non-hierarchic digraphs and metrics can be used for further computations, and causal conceptual centrality in cognitive maps can be defined with an adjacency matrix [7], [15].

4.3 Use of Fuzzy Casual Algebra

This work seeks to clarify the relationships between concepts and to elucidate the positive or negative effects on each concept while clarifying knowledge of the relationships. Furthermore, an FCM structure allows systematic causal propagation, and arrows sequentially contribute to the convenient identification of the causes, effects, and affected factors [15]. Figure 5 has seven variables that describe the impact of some conditions on strengthening bilateral economic relationships and causal variables. For example, (C1→C2, C1) are said to impact C4. This is apparent because C1 is the causal variable, whereas C4 is the effect variable. Suppose that the causal values are given by p {none ≤ some ≤ much ≤ a lot}. The FCM appears below in figure 5.

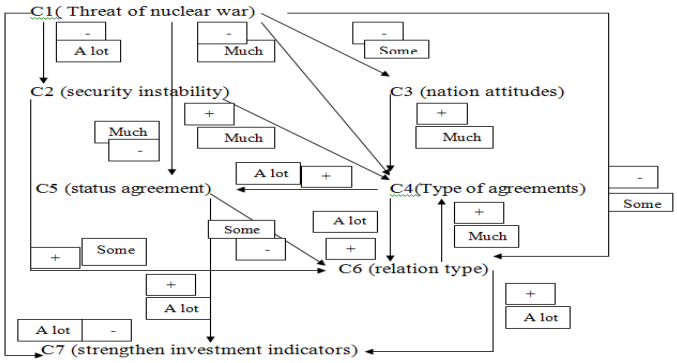


Figure 5: A fuzzy cognitive map on the impact of strengthening economic bilateral relationship

In Figure 5, phrases such as "much" and "a lot" denote the causal relationship between concepts. The causal paths from C1 to C7 are nine, the direct effect is (1,7), so the eight indirect effects of C1 to C7 are: (1,2,4,5,6,7), (1,2,4,6,7), (1,2,6,7), (1,4,5,6,7), (1,3,4,6,7), (1,3,4,5,6,7), (1,4,6,7), and (1,6,7). The first and the second indirect effects of C1 on C7 can be described as:

I1 (C1,C7) = min {e12,e 24,e 45,e 56,e 67} = min {a lot, much, a lot, some, a lot} = some

I2 (C1,C7) = min{e12, e24,e46,e67} = min {a lot, much, a lot, a lot} = much

Thus, the total effect of C1 on C7 is T(C1,C7) = max { I1(C1,C7), I2(C1,C7), I3(C1,C7), I4(C1,C7), I5(C1,C7), I6(C1,C7), I7(C1,C7), I8(C1,C7)} = max {some, much, some, some, some, some, much, some}= much.

Therefore, C1 impacts much causality on C7. Now that the fuzzy conceptual Ci has been computed, the advantage is that the causal quality is established.

5 Inference system in the political domain

Incorporating the concept of specific domain, this step applies a method that can deal with dismantling each variable to several parameters. Decision-makers would be aided by a system that would allow them to formulate constructive rule

conclusions by dealing with several parameters (membership) for each variable. The advantage of GUI tools in Matlab is the capability of building a productive graphical fuzzy inference system (FIS). There are five primary GUI tools for building a fuzzy inference systems: 1) the FIS editor; 2) the membership function editor (MFE), which allows users to define and shape the membership function associated with the input and output variables of the FIS; 3) the rule editor, for editing the list of rules that define the behavior of the system (IF-THEN); 4) the rule viewer, for diagnosing the behavior of specific rules and viewing the details (it is a technical computing environment); and 5) the surface viewer, which generates a 3D surface from two input variables and displays their dependencies; see Figure 6.

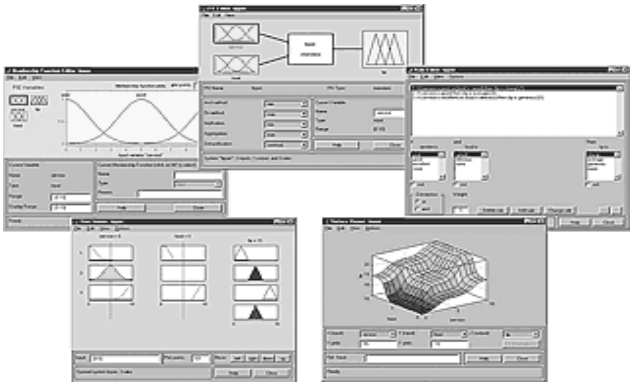


Figure 6: The five primary editing

These GUI tools are dynamically linked and can interact and exchange information.

5.1 Data and modeling scenario

First, we need to collect all input/output data in a form that can be used by inference. For example, input 1 is "security stability." In this step we need to add the parameters for "security stability" input, so we need to define all inputs and all values for those inputs. The second step is the membership function editor. In the third step, which involves the rule editor, we need to construct the rules, for example, we construct the first two rules as follows:

if (SecurityStability is poor) and (PoliticalStability is poor), then (investment is never) (1).

if (SecurityStability is good) and (PoliticalStability is medium) and (ThreatOfTerrorism is medium), then (investment is cut)

These rules are in verbose. The result is an extremely compressed version of the rules in a matrix where the number of rows is the number of rules and the number of columns is the number of variables, as follows:

1 1 0 0 0 0 0 0 0 0 0 0 3, 1 (1): 1

2 2 2 0 0 0 0 0 0 0 0 0 3, 3 (1): 1

Using such functions in the political domain provides the opportunity to choose a membership value with infinite accuracy, or at least to explain the strength and weaknesses of an expression. Reading across the first row, a literal interpretation of rule 1 is "input 1 is MF1" (the first value for the membership function associated with input 1). This means that from the first input (Security) we select {poor}, the value for the membership function associated with input 1; {poor, good, very good}. Continuing across, MF1 from input 2 was selected, and so on. Obviously, the functionality of this system does not depend on how well the operator named the variables and membership functions and does not even bother with variable names. The next step is to use the rule viewer to display a roadmap of the whole fuzzy inference process. The decision will depend on the input values for the system. The defuzzified output is displayed as a bold vertical line on plot. The resultant plot is shown also, see figure 7. The defuzzified output value is represented by the thick line passing through the aggregation fuzzy set. The fourth step is using rule viewer, in order to display a roadmap of the whole fuzzy inference process. This is the implication process in action. The aggregation occurs down in the fifth column. The defuzzified output value is represented by the thick line passing through the aggregation fuzzy set.

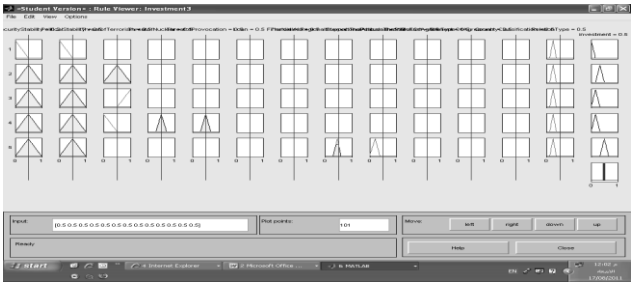


Figure 7: Fuzzy inference diagram containing calculation

The fifth step is using a surface viewer that generates a 3D surface from two input variables and displays their dependencies.

6 Conclusion and Future Work

By handling uncertainty of information and knowledge, fuzzy ontology expands the application domain of ontology. It is useful to share and reuse fuzzy knowledge in collaborative design. In this paper, we have proposed that the fuzzy domain ontology model has stronger abilities for expressing uncertain linguistic variables. Our further research lies in the automatic generation of fuzzy ontology from more fuzzy systems. The idea of applying the work of the political centers to support top decision-makers may not be new, but it would be a new development within the Arab region. We consider political centers in other countries in order to forecast the obstacles that

may face our country, in addition to understanding the differences between them. This will not only help us strengthen the political centers of the future, but also encourage us to consider carefully what kind of political center suits our region. Our analysis of the area's coordinates will inform the appropriate selection and form for the political decision-maker. This center requires technologically advanced specifications that support the government's highest authority decision-makers, helping them to navigate properly through the urgent events that affect the region. In further research we will explain the need for such political centers and detail both their intended functions and the challenges that hinder and cripple their work.

7 References

[1] Abulaish, M. and Dey, L. "Interoperability among distributed overlapping ontologies: A fuzzy ontology framework". Proceedings of the 2006 IEEE/IWC/ACM International Conference on Web Intelligence, 2006.

[2] Al Asswad, M. M., Al-Debei, M. M., de Cesare, S. and Lycett, M. "Conceptual modeling and the quality of ontologies: A comparison between object-role modeling and the object paradigm". Proc. 18th European Conf. Information Systems, Pretoria, 2010.

[3] Al-Ibrahim, H. "In Qatar, do think tanks matter?". Brookings Doha Center, Doha, Qatar, 2011.

[4] Alshayji, S., El Kadhi, N. and Wang, Z. "Building Fuzzy-Logic for Political decision-maker". D 20, E 814, Naun.org/International journal on Semantic Web, Romania, 2011a.

[5] Alshayji, S., El Kadhi, N. and Wang, Z. "Building ontology for political domain". 2011 International. Conference on Semantic Web and Web Services. 'SWWS'11'. Inspec/IET/The Institute for Engineering & Technology; DBLP/CS Bibliography; CNRS, INIST databases, Las Vegas, USA, 2011b.

[6] Alshayji, S., ElKadhi, N. and Wang Z. "Fuzzy-based ontology intelligent DSS to strengthen government bilateral economic relations". Kcess'11 (Second Kuwait Conference on e-System and e-Services), April 9, ACM 978-1-4503-0793-2/11/04, 2011c.

[7] Alshayji, S., ElKadhi, N. and Wang, Z. "Fuzzy Cognitive Map Theory for the political Domain". Federated Conference on Computer Science and Information System, The Fed CSIS'2011 September 19-21, Szczecin, Poland, IEEE Digital library CEP1185N-ART, 2011d.

[8] Alshayji, S., El Kadhi, N. and Wang Z. "On fuzzy-logic-based ontology decision support system for government sector". 12th WSEAS International Conference on Fuzzy Systems, Brasov, 34, 2011e.

[9] Beck, H. and Pinto, H. S. "Overview of approach, methodologies, standards, and tools for ontologies". Agricultural Ontology Service (UN FAO), 2003.

[10] Calero, C., Ruiz, F., and Piattini M. "Ontologies for software engineering and software technology". Springer-Verlag, Berlin, Heidelberg, New York, USA, 2006.

[11] Fernandez-Lopez, M. "Overview of methodologies for building ontologies". Journal Data & Knowledge Engineering, 46, 2003.

[12] Inyaem, U., Meesad, P. and Haruechaiyasak C. "Dat Tran: Construction of fuzzy ontology-based terrorism event extraction". Third International Conference on Knowledge Discovery and Data Mining, IEEE. DOI 10.1109/WKDD.113, 2010.

[13] Islam N., Abbasi A. Z. and Zubair A. "Semantic Web: Choosing the right methodologies: Tools and Standards". IEEE, 2010.

[14] Kerr, S. "Gulf think-tanks feel heat in political debate". Financial Times, 2011.

[15] Kosko, B. "Fuzzy cognitive maps". London: Academic Press Inc., 65-75, 1986.

[16] Morillas, P. March. "The role of think tanks in Arab transitions". IEMed, 2013.

[17] Noy, N. and McGuinness, D. "Ontology development 101: A guide to creating your first ontology", 2001.

[18] Stanford Knowledge Systems Laboratory Technical Report KSL-01-05, Stanford Medical Informatics Technical Report SMI-0880.

[19] Sharif, A.M. and Irani Z. "Knowledge dependencies in fuzzy information systems evaluation", 2005.

[20] Proceeding of the Eleventh Americas Conference on Information Systems, Omaha, NE, USA.

[21] UPI. "Gulf states tighten grip on think tanks". United Press International, 2011.

[22] Windecker, G. "The role of think tanks in the gulf region: potential, challenges, and benefits". Konrad-Adenauer-Stiftung, Berlin, Germany, 2011.

DSP Filter Calculation for a Digital Doppler Velocity Flow Meter

Daniel L. Garcia
Student at SPSU
1100 S. Marietta Parkway
Marietta, GA 30060-2896
678-925-9672
dgarcia3@spsu.edu

Daren R. Wilcox
Southern Polytechnic State University
1100 South Marietta Parkway
Marietta, Georgia 30060-2896, USA
678-915-7269
dwilcox@spsu.edu

Introduction

The goal of this article is to provide proof of concept that digital signal processing has progressed to the point of equivalence to traditional analog Doppler flow meter designs. This article will show the results of the test.

Categories and Subject Descriptor
Proof of Concept for a Digital Ultrasonic Doppler Liquid Velocity Flow Meter

General Terms
Digital Ultrasonic Doppler Liquid Velocity Flow Meter

Keywords
Doppler Liquid Flow Meter, Doppler Affect

Differences between a Digital Doppler Meter and Analog Doppler Meter

A digital Doppler velocity meter differs from an analog meter only in the methods used to determine the Doppler shift discrepancy. While an analog meter uses continuous time components, numerous analog filters, analog sampling holders, analog comparators, analog rate multipliers with scaling, etc. to ultimately convert the signal into either a voltage or current equivalent to the Doppler shift value, a digital Doppler would only require a few low order low pass analog filters to smooth out the signal and eliminate all unrealistic frequencies before entering the analog-to-digital (A/D) converter to digitize the values. The rest of the calculations/components will be done in the digital-domain. The limit of the filters which can be implemented in the digital-domain is only restricted by the frequency of the chip and memory availability. Ergo, a digital Doppler could potentially reproduce the analog filters and even improve on the filter orders with no increase in physical components.

Known Limitations for Doppler Flow Meters

Doppler velocity flow meters do not function in all locations. Some basic prerequisites are required in order for the technology to work regardless of analog or digital architecture. Some of the limitations are:

- Variations of greater than or less than approximately two percent difference in density than the liquids are required. The following are examples of acceptable variations :
  - bubbles
  - rocks
  - dirt
- The approximate minimal size of the variations within the liquid needs to be at least 100 microns.
- A Doppler calculates the average rates of the variations of density moving through the pipe to determine the speed of the liquids. This means, there needs to be a regularity in variations of density equivalent to 100 parts per million (ppm) evenly distributed throughout the sampled liquids.
- The variations of density need to be homogeneously distributed with at least five percent variations within the liquid.

The Doppler affect is not an exact science. The angle in which the wave strikes the variation of density will affect the shift. The position of the wave when the object interacts with it will also play a role in the shift. A Doppler system determines the average speed of the variation of density which means the liquid could be moving a little faster or slower around the objects. A close approximation, typically within two percent of the instantaneous rate, could be determined.

Determining the Basic Requirements

Digital Filters

Because of the low frequency sampling rate, a 10^{th}order high pass digital filter and a 10^{th} order low pass digital filter was used. More testing is recommended if a design is pursued for not only the low pass and the high pass digital filters, but also for a dynamic notch filter in order to eliminate the power frequency noise that occurred during the testing. The current low pass filter coefficients used in the z-domain calculated are 1.0e-008 * 0.0018 0.0182 0.0819 0.2184 0.3822 0.4586 0.3822 0.2184 0.0819 0.0182 0.0018 for the numerator and 1.0000 * -8.8616 35.3965 -83.9180 130.7642 -139.9304 104.1352-53.2146 17.8698 -3.5607 0.3197 for the denominator. For the high pass filters, the numerator are 0.9934 -8.9406 35.7625 -83.4458 125.1687 -125.1687, 83.4458 -35.7625, 8.9406, -0.9934 and 1.0000, -8.9868 35.8942 -83.6299 125.2605 -125.0763 83.2617 -35.6311, 8.8947, -0.9868 for the denominator.

Analog-to-Digital Converter Speed

The analog to digital (A/D) converter is the backbone of the digital Doppler design. Rather than using numerous analog filters, analog direct current (DC) detectors, voltage controllers, etc., samples of the signal would be temporarily logged through an A/D converter. The samples are then stored until enough are collected to be able to accurately convert the signal into the frequency-domain through the use of a modification to Gauss' algorithm called "Fast Fourier Transform (FFT)."

The algorithm (which is a set of predetermined manipulations to the signal values) converts the samples to the frequency spectrum with the amplitude at said frequency. The frequency spectrum is the quantitative amplitude at a given frequency within the samples collected.

By determining the frequency of the signal at the relative highest average amplitude, then dividing that frequency by the specified

conversion ratios (located under the filter specifications in the results section), the speed of the liquids could be approximated. A collection of 1024 samples per conversion to the frequency spectrum will give greater than 0.1Hz accuracy for a full spectrum of 2.5 kHz. With the use of an AM style multiplexer, a standard audio A/D microphone digitizer is all that is necessary to use with the stated filter coefficients.

Implementation of the Code

In the code the following steps were taken.

1. The recorded data was converted back into voltage values using the wavread() file.
2. All constant data arrays and data points were created in order to make quick changes to any values.
3. The filter coefficients were then created using the 'butter' and 'buttord' commands.
4. All filters were then graphed to separate plots.
5. The impulse responses of all the created filters and the raw collected data were converted to the frequency-domain.
6. The impulse responses were then convoluted with the recorded signal.
7. A plot of those results was graphed.

A sample coding used can be found in the appendix.

Unforeseen Issues

The recording had a great deal more noise than expected, and the results did not yield the peak amplitude value that was envisioned. A modification to determining the average peak values of all the relative maximum amplitudes of approximately 90% of the entire sampling will have to be written after filtering the signal.

Another unforeseen issue was that the low frequency (below about 35Hz) had a great deal of noise. A high pass filter was then added to ensure the elimination of unreal low end values. The current testing was through an AM style multiplier circuit. This is a valid system but as a result the low end values (under 0.5fps) will be lost. This will be a limit to how low of flow this unit can read. Modification to the multiplier circuit could be done to try to decrease the readable values on the low end, but more testing will be required. This testing was outside the objective of this article.

The calculation as to the value of fps to Hz ratio was expected to be linear. The realization that the signal was not linear was unforeseen.

Results

Micro Controller Minimal Recommendation

In theory, a 2MHz micro controller could do the core calculations. The core required process is as followed:

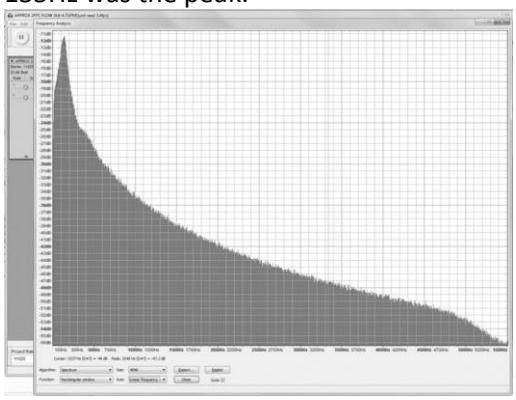
- Sample the return multiplex signal 1024 times
- Convert that sample to the frequency-domain using FFT conversions
- Find the relative average peak amplitude
- Divide said value by the previously specified frequency to fps ratio
- Finally, output that value to an LCD screen or signal port

In order to allow for modification and future improvements, a faster chip is suggested to be used if a prototype is attempted. A minimal of 20-100 times faster chip is recommend (40MHz-200MHz). (One sample for every approximate 2MHz processing speed with no additions to the code is speculated to be the ratio.)

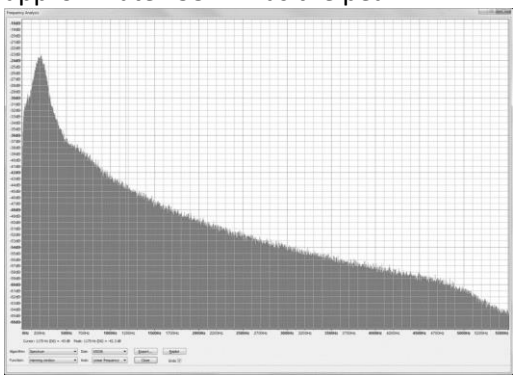
Plots and Graphs

The following will be screen shots of the collected frequency-domain results from Test 1:

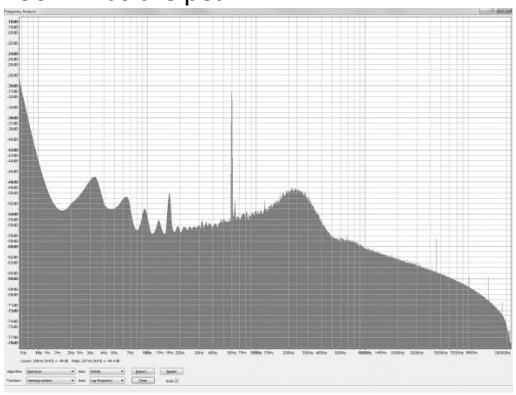
- For approximately 1.9 fps an approximate 135Hz was the peak.



- For approximately 2.666 fps an approximate 195Hz was the peak.



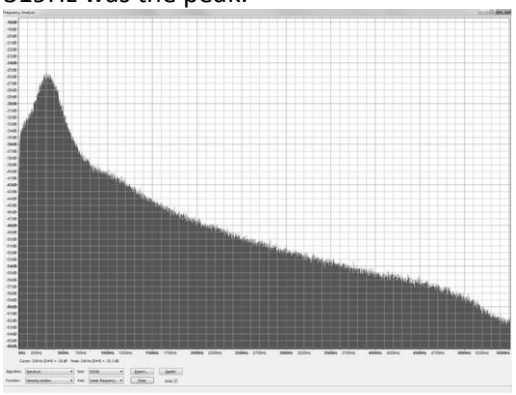
- For approximately 3.5 fps an approximate 256Hz was the peak.



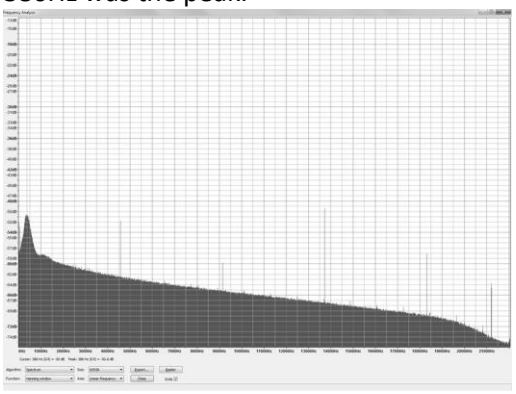
(Logarithmic Scaling)

A 60 Hz power signal is also present in this one because no notch filter was applied to that frequency.

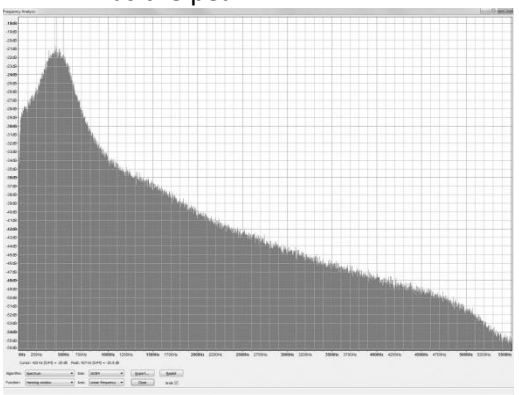
- For approximately 4.6 fps an approximate 319Hz was the peak.



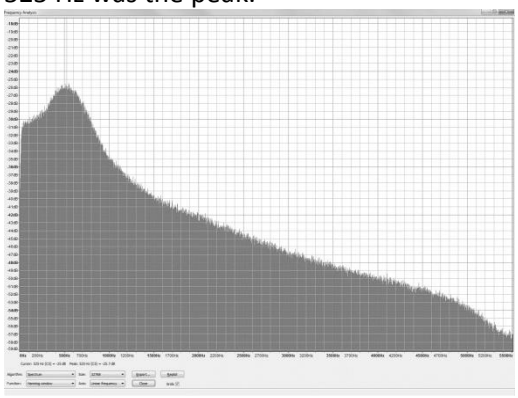
- For Approximately 5.5 fps an approximate 386Hz was the peak.



- For approximately 6.2 fps an approximate 427 Hz was the peak.



- For approximately 7.7 fps an approximate 525 Hz was the peak.



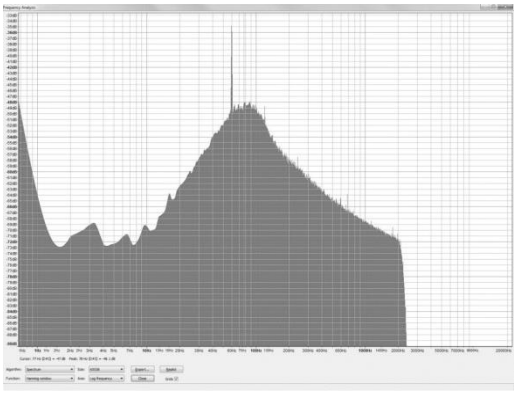
The complete results of test 1 are as followed:

Test 1

Reference		Results		
gpm	fps	Hz	Calculation	Error
2.4	0.891	68	0.95	6.622%
4.9	1.819	135	1.899	4.398%
7.2	2.673	195	2.758	3.180%
9.6	3.564	256	3.642	2.189%
12.4	4.603	319	4.564	-0.847%
14.3	5.309	368	5.291	-0.339%
15	5.568	386	5.56	-0.144%
16.7	6.199	427	6.175	-0.387%
18.8	6.979	478	6.946	-0.473%
20.75	7.703	525	7.664	-0.506%

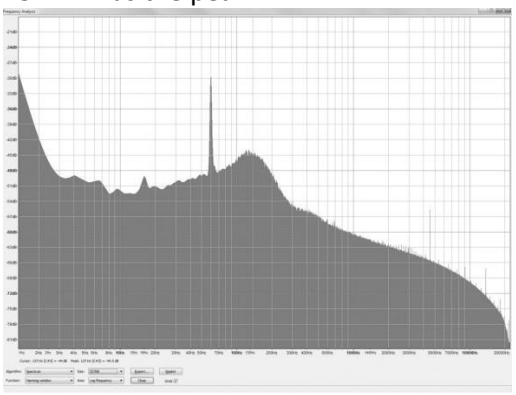
Test 2 was done with the same hardware but a different transducer. The goal was to determine if the transducer placed a role in the fps to Hz ratio. The following will be screen shots of the collected frequency-domain results from Test 2:

- For approximately 0.95fps an approximate 78 Hz was the peak.



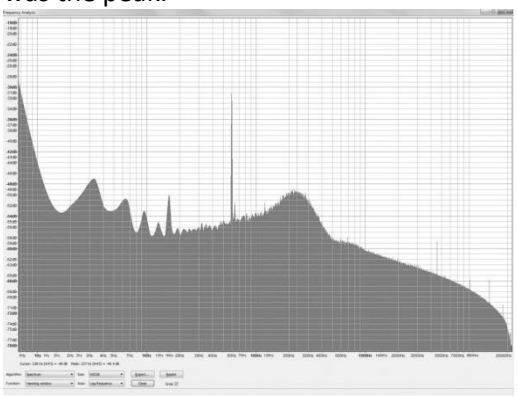
A 60Hz power peak is also present. A notch filter at 60Hz needs to be applied.

- For approximately 1.95 fps an approximate 137 Hz was the peak.



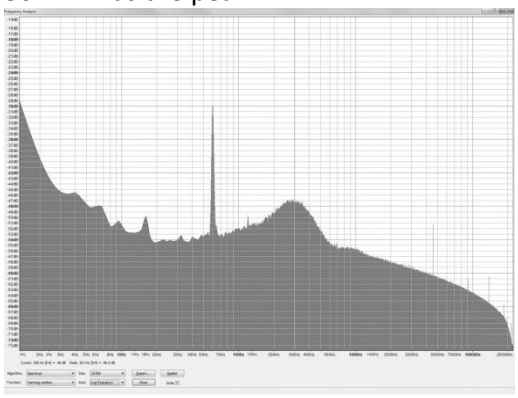
Again, a notch filter at 60 Hz needs to be applied to eliminate the power peak.

- For approximately 3.4 fps an approx. 237 Hz was the peak.



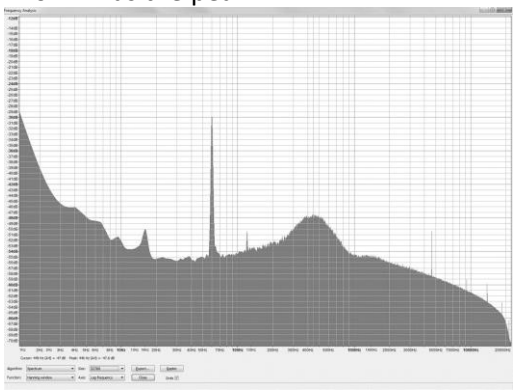
Again, a notch filter at 60 Hz should be applied to eliminate the power noise peak.

- For approximately 4.4 fps an approximate 301 Hz was the peak.



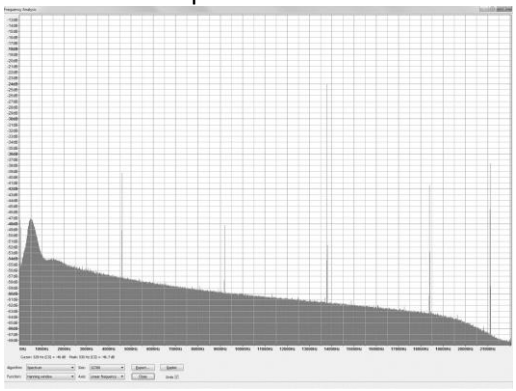
Again, a notch filter at 60 Hz needs to be applied to eliminate the power peak.

- For approximately 6.7 fps an approximate 446 Hz was the peak.



Again, a notch filter at 60 Hz needs to be applied to eliminate the power peak.

- For approximately 8.9 fps an approximate 530 Hz was the peak.



Again, a notch filter at 60 Hz needs to be applied to eliminate the power peak.

The complete results of test 2 are as followed:

Test 2

Reference		Results		
gpm	fps	Hz	Calculation	Error
2.6	0.965	78	1.091	13.057%
5.25	1.949	137	1.927	-1.129%
9.25	3.434	237	3.366	-1.980%
11.95	4.436	301	4.3	-3.066%
18	6.682	446	6.461	-3.307%
20.85	7.74	530	7.741	0.013%

Doppler Effect to Feet per Second (fps) Variations Speculated Formula

One of the most important questions that need to be answered is if the Doppler shift in frequency to speed of liquids linear throughout the examined spectrum, or is it similar to the linear aspect of a parabola? After testing, it appears to be almost linear around 70Hz. On the lower speeds, it was about 67.96Hz at a velocity of 0.947 feet per second (fps) which is about 71.75 Hz per fps. At the maximum speed available with the testing flow loop, it was approximately 524.1Hz at a velocity of 7.684fps which is 68.2Hz per fps. A mid-flow test was also done which yielded a return frequency of 318.1Hz for a liquid speed of 4.566fps which is 69.666Hz per fps. A nonlinear pattern appears to be present in the lower velocities. More testing is required to grasp the full formula, but at first glance it appears to be something like $\frac{frequency}{72-\frac{frequency}{150}} = fps$.

Conclusion

It is the opinion of this engineer regarding the premise that a digital Doppler is possible. The basic digital filters and speeds of the micro controller are the minimal speculated values for general unmodified code (with more research, it could be possible to decrease the number of cycles needed, but the advantage of

decreasing the number of cycles would not warrant the investment at this time).

The approximate Hz to fps value is 70Hz. The ability to capture the return signal and preform the necessary calculations to determine the fps can be done in near real time.

MATLab Code Sample

```
NameOfFile=['APPROX 2FPS.WAV'];
[y,fs,nBits]=wavread(NameOfFile);

Lfft=1024;

Nth=10;
wl=2500/fs;
wh=30/fs;
wb = 60/fs;
bw = wb/fs;
bn=0:1/fs:pi-(1/fs);

%create Butterworth low pass filter
[numl,denl]=butter(Nth,wl,'low');
[irl, time]=impz(numl,denl,Lfft);

%create Butterworth highpass filter
[numh,denh]=butter(n,wn,'high');
[irh, time]=impz(numh,denh,Lfft);

%create a notch filter
[numb,denb]= iirnotch(wb,bw,25);
[irn, time]=impz(numb,denb,Lfft);

%convoluting filter w signal in
%frequency-domain
Y=abs(fft(y,Lfft));
IRl=fft(irl,Lfft);
IRh=fft(irh,Lfft);
IRn=fft(irn,Lfft);

YC=Y.*IRh;
YC=IRn.*YC;
YC=IRl.*YC;
```

SESSION

SHORT RESEARCH PAPERS

Chair(s)

TBA

Testing of Cellular automata clustering algorithm for Forward EM Calorimeter.

Baba V.K.S. Potukuchi[1],ChamanLal[2]

1DepartmentofPhysics&Electronics, UniversityofJammu, JAMMU – TAWI, J&Kstate, INDIA.
2GovernmentCollegeforWomen, Parade, JAMMU – TAWI, J&Kstate, INDIA.

ABSTRACT:We have implemented Forward Calorimeter (FoCAL) geometry using GEANT code in AliRoot Frame work[1] . As designed FoCAL contains 4 super modules and each containing 30 unit modules . We have also implemented Cellular Automata clustering algorithm for reconstruction of clusters from Hits seen by our FoCAL detector . We have tested efficiency & reliability of this algorithm. We have developed a code in C++ object oriented language. Various adjacent active cells are included in the Clusters of more energy and the cluster centroids are determined. Using Invariant mass analysis code, we have reconstructed π^0s from the Gammas found in FoCAL .

1. Introduction

From the Monte Carlo simulation results , it is clear that π0 production rate increases at LHC energies . During any high energy heavy ion collisions the π0 particles are produced in quite high amounts (about 85%), therefore, it is easier to study physics based on π0 production events. Also from the simulation results it is found that π0 production is much enhanced for low P_T at LHC energies. This means that π0 production will be more in forward direction at LHC energies. It also suggests that the ratio of π0 production at low P_T to that at high P_T is greater than unity at LHC energies. The π0 production study has one important advantage that it can be used to subtract background events when high energy π0s faking γs. In the following

sections we will discuss the design and simulation study of the proposed FoCAL detector in ALICE , which is used to study the π0 production in forward direction. In order to extract π0s from FoCAL detector , we have employed cellular automata clustering algorithm method to reconstruct π0s . Using this algorithm we have made clusters by clubbing the hits (as per the procedure discussed in the following sections) seen by the FoCAL detector. By regrouping these small clusters further cluster centroids and their total energies are estimated for each photon . Those obtained Photons are used for making invariant mass plot of π0s .

2. Forward Calorimeter (FoCAL)

Calorimeters are most useful detectors which allow us to explore new physics in the energy range from eV to more than 10^{20} eV. Till now there is no calorimeter in Alice in the

forward direction. The proposal is to put a calorimeter (ie) FoCAL(Forward Calorimeter) in Alice at 2.4 ≤ η ≤ 4.0. The ALICE experiment at the LHC is dedicated for the heavy-ion collisions to exploit the unique physics potential of high energy heavy-ion collisions at LHC energies. Forward Calorimeter (FoCAL) is a sampling EMCAL. It is one of the upgrade plans for the ALICE. FoCAL intends to locate at z =366 cm from Interaction Point. FoCAL detector will do electromagnetic measurements such as pi0 and photons, jet measurements and their correlations with respect to the central rapidity region. FoCAL also allows us to study the parton distributions and high gluon density (Color Glass Condensate) in proton and nuclei at small-x region, which is the key to understand the early thermalization of the hot and dense medium.

The FoCAL design consists of the following parts:
• 40 layers of (W Scintillator)
• Super Modules
• Unit Module

The FoCAL is sampling type calorimeter in which tungsten follows Polystyrene scintillator for
all the 40 layers. The tungsten is used as the converter material because of space restriction in
the forward region. The Polystrene scintillator gives the better position resolution.
Each Polystyrene pad detector is 0.3 mm thick, also tungsten (W) absorber is 0.3mm
thick. It has 21 X_0 depth and 3 longitudnal segments. The total radiation length of 21 X_0 of
the modules provide the full energy absorption of the elctromagnetic particles upto very high

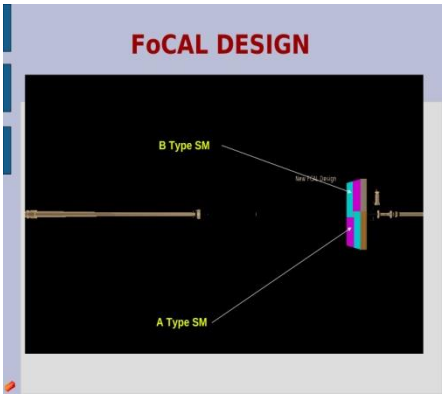


Figure 1: (a) An array of 72 × 60 cells in A-type super module.

energies. Each layer consists of two types of super modules A − type and B − type. Each type of
super module has unit modules distributed in a matrix of specific rows and columns.
A − type super module has 30 unit modules distributed in 6 Col × 5 Row while B − type super module has 30 unit module distribute in 5 Col × 6 Row as shown in Fig 1(a).

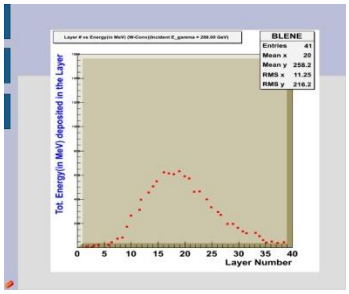


Figure 2: Energy Deposition in various Layers of FoCAL

To test the FoCAL Geometry , we have put a 200 GeV Photon on to FoCAL detector, to see the response. The EM shower has expanded into the detector to Various layers. We have summed up the total Energy deposited in each FoCAL Layer and plotted. In the fig 2.0, one can see that the energy distribution has a nice peak at 18th layer and Energy deposition almost runs upto 36th Layer. It shows that the FoCAL is capable of measuring Photons upto 200 GeV and above.

3. Clustering

In complex events and within particles, multiple particles will deposit energy in the same calorimeter cells and showers will overlap. Fine calorimeter segmentation and good clustering

are essential to resolve such showers. Additionally, an intelligent cluster splitting and merging

strategy is needed. The basic unit for clustering is a cell. The Cell contains an index by which it is referenced. For instance, a two-dimensional cell Cell2D could contain an Index 2D composed of integer indices i and j to indicate row and column. The Cell also contains a value for relationship to the other cells. Given an Index, a Neighborhood is responsible for returning a list of neighboring Indices. Since the dimensionality of the problem is encapsulated within the Index, the clustering algorithm can be written very simply and its extension to higher dimensions is automatic. When developing calorimeter designs with varying segmentation in the transverse(r-φ or r-z) and longitudinal directions (layer depth) it will be essential to be able to easily investigate these different clustering elements. We discuss below the ' Cellular Automata ' algorithm , which we employed for clustering our FoCAL hits.

4. Cellular automata

The Clustering method called Cellular Automata is also used to find clusters in EMCAL [2]. A cellular automaton is an array of simple individual processing cells [3]. The input of each cell is the set of information from the other neighbouring cells. The corresponding output is uniquely determined with a set of fixed rules acting on the inputs. A cellular automaton evolves iteratively at each step, each examines its inputs, decides on the basis of a transition rule whether or not to change its state, and sends its new output value to the inputs of its neighbours. At the next step, these new inputs are examined and the cells evolve simultaneously.

5. Cellular Automaton Evolution Rules

• A cellular automata evolves iteratively: at each step , each cell examines its inputs, decides on the basis of a transition rule whether or not to change the state.
• Each cell is conencted to a limited number of neighbouring cells (3 × 3)
• The change in that cell occurs, once it sees all of its neighbours.

6. Cellular Automaton Transition Rules

The first step is to identify local maxima, i.e. cells with larger energy deposit than their neighbours. These cells are identified with a tag. The second step is to treat them as a virus that is going to propagate to the other cells. Propagation rules are as follows:

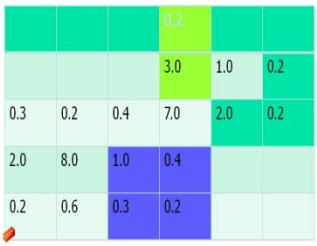


Figure 3: Initial state of the cellular Automation

• A given Cell is only sensitive to its eight neighbours in a 3×3 cell matrix. The cell is treated as a virus, if its value is more than its each neighbour.
• At a given step a cell will take the value of its highest energy neighbour.
• A cell already contaminated in an earlier stage by a virus is immunized against any other virus. This is restriction to the RULE−2.

This simple set of rules allows us to find clusters very efficeintly in the E142/E143 calorimeter as illustrated in the following example. Using a test sample of pion and electron showers from a Monte−Carlo simulation based on Geant 3.16, we generated pseudo−events in the calorimeter. Considering a pseudo−event where two electrons hit the calorimeter

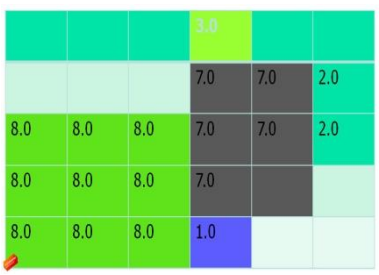


Figure 4: Stage 1 of cellular automaton.

giving two overlapping clusters, the initial state of the active part of the cellular automaton is shown in Fig 3 . where each box represents a CA cell. The initial value is the energy deposit in the it cell. Only hit cells have been represented, as the other cells do not evolve. According to Rule−1, the cells containing 7.0 and 8.0 GeV are viruses because they are local energy maxima. At the next step, the CA reachees the state 2, as shown in Fig. 4 . Due to what we call contamination, two groups of cells are appearing; however a steady state has not been reached yet. So the CA evolves further to state 3, shown in Fig 5. Rule−3 stops the CA evolution at this stage, avoiding that the cluster tagged by the 8.0 GeV cell absorbs the cluster tagged by the 7.0 GeV cell. Once the clusters are identified, the energy of each particle is obtained by adding the true energy deposit in each block of the cluster.

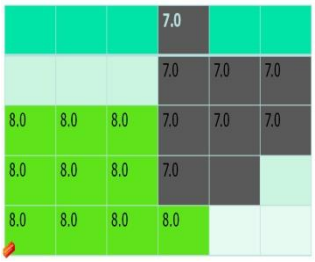


Figure 5: State 3 of cellular automaton.

7. Clustering implementation in FoCAL

The Cellular Automata algorithm is implemented in FoCAL detector , for making clusters from the hits seen in individual cells of FoCAL . The cellular automata not only gave us better cluster number estimate but also produced clusters whose energies are more close to the energy of the particle . The Figure 6 shows the XY distribution of FoCAL for different layer 2 for incident gamma at 15 GeV. As can be observed , the clusters obtained for starting layer 2 is found to be near to the actual number of Gammas falling on the FoCAL . In further layers of the detector cluster numbers increased a little more. Our motive is to identify individual photons using our clustering algorithm and to do invariant mass analysis later. M.C simulations of π0 production cross-section during quark-gluon interaction in QCD at LHC, RHIC shows that π0 production is much enhanced for low P_T at LHC energies. This means that π0 production will be more in forward direction at LHC energies , that's the reason we implemented FoCAL in ALICE. It also suggests that the ratio of π0 production at low P_T to that at high P_T is greater than unity at LHC energies.

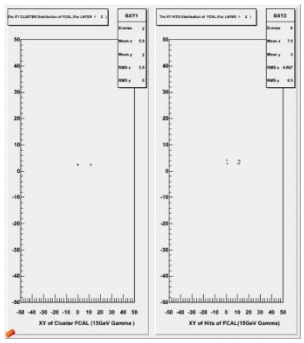


Figure 6: XY distribution of Hits and Clusters at 15GeV gamma for layer No. 2 of FoCAL.

8. Reconstructed π^0 mass distribution .

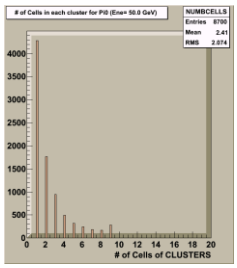


Figure 7: No. cells in Clusters produced by single π^0 at 50 GeV

To reconstruct π^0, the first step is to consider cluster strategies. It was observed that for each layer the number of clusters in a layer increases with increasing incident energy. It is observed that No. of cells increases in each cluster produced by the incident particle increases for the studies of incident Gamma energies of 1, 5 and 100 GeV . Combinatorial background increases if an incorrect number of clusters are summed over, because the clusters summed over should have a strict dependence on energy. We have studied and observed that maximum number of cell hits as seen in FoCAL for various energies of various particles. It is observed that for 1 GeV gamma, we have less number of clusters .

We have studied # of cells for one cluster for different incident energies for (e.g.)1GeV, 5 GeV, 50GeV π^0s . The cluster number also increases and shows increase in splitting in the clusters . From figure 7, we can see that for energy above 50 GeV of π0, the number of clusters with No. of cells as many as 10 and above. This plot shows that still a large number of clusters are splitting and need to be taken care in the clustering alogorithm. π^0's are identified via the $\pi^0 \rightarrow \gamma\gamma$, channel with a branching ratio 98.8%.

To get the total energy for incident gammas, we have defined a cone around the cluster Of maximum energy . We took all clusters falling in that cone to get the total clusters and by using the fit parameter we got the energy of gammas in GeVs. For different cone radii, we have studied the efficiency of gammas detected in FoCAL. The total energy within the cone is calculated to be $E^J = \Sigma_i E_i$. We have studied different cone radii, $0.2 \leq R \leq 2.0$ and tested efficiency of the π0 detection. The uncorrelated photon pairs, which were not originating from a parent π0, produce a combinatorial background. These backgrounds were separated via event mixing technique .

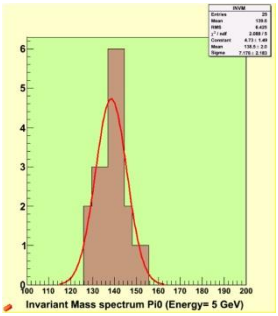


Figure 8: π^0 Invariant Mass.

We have tested the clustering algorithm for a small statistics and generated an invariant mass spectrum using the combinatorial method. The plot is shown in figure 8. We have also fitted this histogram with a gaussian. From the fit of gaussian, the mean is obtained to be 138.5 ± 2.0 with a sigma 7.176 ± 2.183. As the statistic is low, the sigma looks to be some what more. We require to test our clustering algorithm with very large statistics to see a better mean and sigma. Also we need to test our clustering algorithm for better estimation of the π^0 peak.

9 Conclusions

The geometrical implementation of the new design of the FoCAL in ALICE framework was done successfully. A full simulation study of the basic characteristics of the FoCAL such as deposited energy distributions, linearity response and relative energy resolutions have been performed. The clustering algorithm Cellular automata gives good results but we need to futher tune it to take care of splitting clusters. We have also used Cone algorithm to sum up energies of various photons coming from π^0. Some small discripancies are found in energy estimations of gammas and the Cone algorithm requires some more refinement. One of the gamma (gamma2) verses Cone radius looks to be quite satisfactory and does not show any dependency on the Cone radius. π^0 reconstruction looks to be good and generation of larger statistics will solve these specified problems and make way for further progress to get better results.

References

[1] http://aliweb.cern.ch/Offline/AliRoot/Installation.html .
[2] V. Breton et al, Application of neural networks and cellular automata to interpretation of calorimeter data, NIM A362 (1995) 478-486
[3] B. Denby, LAL Orsay internal report 87.56 (1987)

Automated Electoral System in a Developing Country Democracy

Okonigene Robert[1], Ojieabu Clement[2], Samuel John[3], and Agbator Austin[4]

[1,2,4]Ambrose Alli University, Ekpoma, Edo State, Nigeria
[3]Covenant University, Otta, Ogun State Nigeria

Abstract - *In this paper we critically examined the feasibility, acceptability, conformity and the economic advantages in the application of our developed automated electoral system in line with democratic principles in a developing country like Nigeria. This automated system has already been presented in an earlier publication. In this system electoral irregularities have been reduced to the barest minimum. It was estimated that to fully implement this automated electoral system will cost USD $416m. Due to the huge cost involved in real live test a simulated automated system analyses were carried out. A total files size of 16 terabyte data, 140,000 laptops and accessories, 200 million people, 36 states, 6 geopolitical regions, 774 Local Government Area (LGA) were considered. The results showed its suitability for Nigeria electoral process. Hence, elimination of electoral irregularities and conformity with electoral laws were achieved. The automated electoral system guaranteed transparency, secrecy, free and fair electoral process. With this automated process the citizens are guaranteed full enfranchisement. Due to the flexibility of the system more than 91% of the eligible registered voters are expected to vote. However, more security, both software and hardware were embedded in the system that reduces the insider threat to a probability of $0.1x10^{-5}$. The protection of the secrecy of each voter is assured. The application of this automated electoral process is in line with democratic principles and values.*

Keywords: Automated electoral process, democratic principles, electoral irregularities, electoral laws, Local Government Area, enfranchise

1 Introduction

The rigorous development and design of the automated electoral system was not discussed in this paper. Also the simulated electoral network and the test results were not presented. However, we took into cognizance the analyzed results and relate these results to democratic principles in a developing country like Nigeria. Democracy is the rule of the people to form a government in which all eligible citizens participate equally in the proposals, developments and creation of laws [1-5]. Hence, democracy is a system of government in which a country's political leaders are chosen by the people in regular, free, and fair elections. In a democracy, the citizens decide who to choose between aspirants of different parties who want to be elected representatives [6-8].

Nigeria practices representative democracy, in which, were it not for electoral fraud, the representatives are elected by the people to act in the people's interest. However, these representatives exercise their own judgement over several issues without regards to the voters [9-12].

Presidential Democracy is a system where the public elects the president through free and fair elections. The president serves as both the head of state and head of government controlling most of the executive powers. The president has direct control over the cabinet, specifically appointing the cabinet members [13].

The citizens have the right to criticize and replace their elected leaders and representatives who fail in their performance of duties [14–16]. For peaceful society the decision of government must be based on the will of the people. Therefore elected representatives at the national and local levels must listen to the people and be responsive to their needs. Generally in a democratic system of governance the representatives are elected by the counting of secret votes during an election.

The basic principles of democracy guarantee Citizen's right, Federalism, due process, citizen's right to participate, executive power, separation of power and judicial independence [17]. In Democracy the rules of law is paramount and ensure the protection of the rights of the citizens. In a society where there is the rule of law order is maintain and this help to limit the power of government. In countries that practice true democracy all citizens are equal under the law; no one is discriminated against on the basis of race, religion, ethnic group, or gender. All citizens are entitle to fair hearing and may not be denied their freedom. No one is above the law. No one is guilty until proven guilty by a competent court of law. In a democracy, a person

accused of a crime has the right to know the charges against him, to remain silent, to have legal representation, to participate in his defense, and to question witnesses for the prosecution. No person who is acquitted of a crime may be tried again on that charge.

The declared results must be seen to be free and fair in order to be accepted and avoid violence. In Nigeria Section 77(2) of the 1999 constitution specifies those who are eligible to vote [18].

In today's world Electronic and Internet voting has become slowly, but increasingly, widespread in some democratic Nations [19]. Due to the scale of electoral irregularities associated to elections conducted in Nigeria and the uncertainty of the results declared, the electoral process are often challenged in electoral petition tribunals. We developed the automated electoral system to eliminate these electoral irregularities [20]. Here we examined the acceptability of this automated system in a democratic developing society.

There are many criticisms against the use of electronic and internet voting. These schools of thought have argued that results of electronic elections are not verifiable nor are they able to guarantee fairness of elections.

It was also argued that in more than two centuries of existence there are no serious trouble recorded in western democracy arising from the use of ballot papers and to date most democracies of the world use ballot papers to elect their representatives and subsequently the formation of Governments. Thus there is nothing wrong with the use of ballot papers for voting. In a true democratic electoral process it is paramount to ascertain absolute vote secrecy. Over three decades ago only a few countries practice democracy. As at today democracy has rapidly expanded throughout most countries of the world in which people choose their leaders in free and fair, multiparty elections. So many others including Nigeria are struggling to achieve true democracy. Deliberate efforts are made by these governments to ensure that people of every religious faith, such as Islam, Christianity, Buddhism, Hinduism, Judaism, and others do aspire to live in free and democratic societies.

2 The Automated system and democratic principles

This automated system considered the secrecy of each voters' vis-à-vis the question as to whether from the stored results the vote of a voter can be linked back to the voter. To resolve this puzzle the system after verifying and authenticating an eligible voter it then open up a subroutine for voting which can identify the voter's vote. During legal proceedings to verify a vote this link can become handy. The idea behind the development of this subroutine is equivalent to forensic identification of fingerprints on ballot papers used for election. The stored data uniquely identify each voter and this information can only be made available after fulfilling the appropriate legal demand and proceedings.

Technically the vote is considered to be anonymous but uniquely linked to an individual eligible voter before a competent court of law. This solution is however adequate to the extent of a voter vote secrecy and its verifiability.

According to Tom Stoppard "It's not the voting that's democracy; it's the counting". Also Joseph Stalin was quoted as saying "those who cast the votes decide nothing, those who count the votes decide everything". The insider threat is by far the greatest challenge to any kind of electoral process adopted. This automated electoral system has reduced this threat to the barest minimum. The fact here is that the system accepts only certified eligible voters to vote and it is the individual vote that counts.

As long as the verification process, with the current state of technology, can be observed to be transparent, free and fair, then one of the fundamental aspects of democracy which is the right to participate is fulfilled.

This system was designed to accommodate all age groups including infants. But only those within the constitutional age of 18 years and above are eligible to vote. The data-capturing or Registration proposed for this automated system is a continuous process. In developing this automated electoral system we considered some of the basic fundamental principles of democracy into consideration. The practice of democracy requires that every citizen has certain basic rights that the state cannot take away from them. The right in this case is to be registered and franchise to vote as shown in Figure 1.

Everyone has the right to assemble and to protest against observed uncertainty in declared results. It is expected that citizens should exercise these rights peacefully in line with appropriate law and also respect for the rights of others. Therefore this automated electoral system does guarantee that each vote categorically represents its elector's will. This electoral system also ensures that more citizens can participate in all elections. Participation is one of the key role of citizens in a democracy.

The Nigerian Independent National Electoral Commission (INEC) limits the number of voters for each Unit between 500 to 700 voters [19]. And also voters are required and restricted to vote in the ward where they registered. Many voters are confronted with these constrains during election when they travel out of their state, LGA, or ward. Many also are not within the vicinity of their voting centers during election because they travelled out of the country.

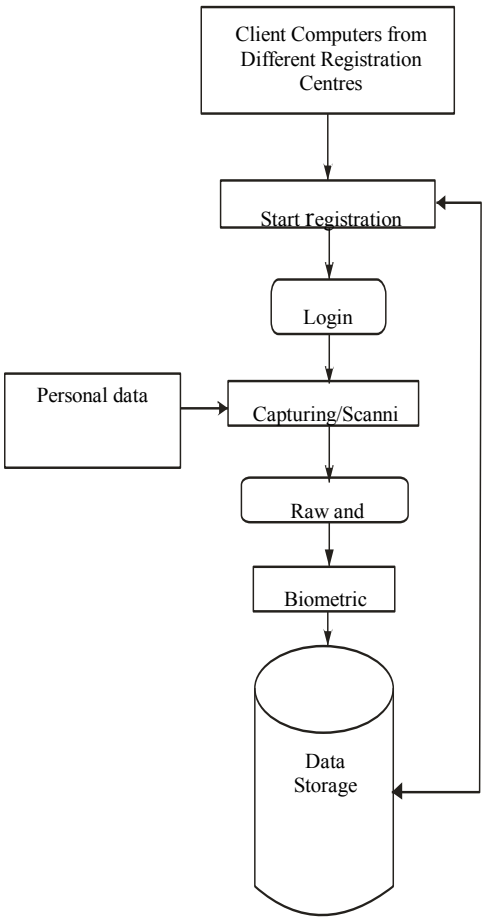


Figure 1. Registration of voters and data acquisition

This automated system allows these citizens to fully participate in the electoral voting process without hindrance. This electoral system increase the chances for these categories of citizens to vote wherever the maybe during election. For these reasons it is expected that the number of citizens that would participate in any election would increase tremendously.

In Nigeria the electoral process begins with the registration of political parties. The center point of democracy is in the election. Any process that is adopted for an election must critically be seen to be transparent. Winston Churchill once describe democracy as the worst form of government , except for all those other forms that have been tried from time to time. Most elections in Nigeria are often characterized by fraud.

The automated system flexibility involves a series of checks and balances between the parties involved, the state, the civil society, the judiciary and the free media. It does not end with voting during election period alone. This advantage of the automated electoral system is also an integral part of democracy. The automated system serves as a 100% neutral administrator for a free and fair election and also which treats all political parties and candidates equally.

The system guarantees that all parties and candidates have their right to campaign freely and to present their proposals to the voters. During voting the voters are giving conducive condition to vote in secret, free of threats from anyone. Independent observers are able to observe the voting and the vote results to ensure that the process is free of any electoral irregularities.

3 Conclusions

It is proper to encourage Government to endorse the use of electronic voting. This automated system was developed to be compatible with Democracy. In Nigeria those who are in government have the economic power and are more interested in falsifying electoral results thereby violating the secrecy of votes to preserve the power. Elections are a crucial element of democracy, and the current use of ballot box in Nigeria elections supports illegality by dirty leaders and those fraudulent people that count the votes. The consequences of their illicit action have stired up racial, religious and sectarian divisions. The automated electoral system guarantees the accuracy of electoral results and the secrecy of votes. The final results are a true reflection of the will of the people.

4 References

[1] Wilson, N. G. "*Encyclopedia of ancient Greece"*. New York: Routledge. p. 511, 2006.

[2] Roger Scruton "A Point of View: Is democracy overrated?". BBC News, (2013-08-09).

[3] "Parliamentary sovereignty". UK Parliament. Retrieved 18 August 2013.

[4] Larry Jay Diamond, Marc F. Plattner. "Electoral systems and democracy" p.168. Johns Hopkins University Press, 2006.

[5] "Citizen or Subject?". The National Archives. Retrieved 2013-11-17.

[6] Tocqueville, Alexis de. "Democracy in America". USA: Barnes & Noble. pp. 11, 18-19, 2003.

[7] A. Barak, *The Judge in a Democracy*, Princeton University Press, p. 40, 2006.

[8] U. K. Preuss, "Perspectives of Democracy and the Rule of Law." *Journal of Law and Society*, 18:3. pp. 353–364.

[9] Köchler, Hans. *The Crisis of Representative Democracy*. Frankfurt/M., Bern, New York, 1987.

[10] Urbinati, Nadia. *Representative Democracy: Principles and Genealogy*. (October 1, 2008).

[11] Fenichel Pitkin, Hanna. "Representation and Democracy: Uneasy Alliance". *Scandinavian Political Studies* **27** (3): 335–342, (September 2004).

[12] Keen, Benjamin, A History of Latin America. Boston: Houghton Mifflin, 1980.

[13] O'Neil, Patrick H. "Essentials of Comparative Politics". 3rd ed. New York: W. W. Norton &, 2010. Print

[14] Vincent Golay and Mix et Remix. "*Swiss political institutions"*, Éditions loisirs et pédagogie, 2008.

[15] Garret, Elizabeth, "The Promise and Perils of Hybrid Democracy". The Henry Lecture, University of Oklahoma Law School. Retrieved 2012-08-07. (October 13, 2005).

[16] Harald Wydra. "Communism and the Emergence of Democracy", Cambridge: Cambridge University Press, pp.22-27, 2007.

[17] http://www.stanford.edu/~ldiamond/iraq/Democracy Education0204.htm

[18] 1999 Constitution of the Federal Republic of Nigeria.

[19] Norbert Kersting, **"**Electronic Voting and Democracy: A Comparative Analysis", Palgrave Macmillan, (February 10, 2005).

[20] Okonigene, R.E. and Ojieabu, C.E., Developed Automated Electoral System Algorithm using Biometric Data to Eliminate Electoral Irregularities in Nigeria, International Journal of Computer Applications (0975 – 8887), Volume 14 – No. 6, Page 27.

SESSION

POSTERS

Chair(s)

TBA

Numerical Analyses of the Expansive Flow of a Thrust-Vectoring Nozzle

Tsung Leo Jiang, Yan-Yi Liu and Hsiang-Yu Huang
Department of Aeronautics and Astronautics, National Cheng Kung University, Tainan, Taiwan, ROC

Abstract - *In the present study, the expansive flow of a thrust-vectoring nozzle is analyzed and its thrust performance is evaluated numerically. The dynamic-grid simulation mechanism has been developed successfully, and has been employed to carry out the flow analyses of a thrust-vectoring nozzle. The numerical results obtained from the present study show that the swinging or turning of the flaps makes the averaged Mach-number drop, reducing its thrust. The flow near the upper flap has been predicted to first accelerate to a supersonic one and then reduce to a subsonic one at the nozzle exit as the flaps swing upward. Similarly, the flow near the left flap first accelerates to a supersonic one and then reduces to a subsonic one at the nozzle exit as the flaps swing left.*

Keywords: Thrust-vectoring nozzle, Numerical Simulation, Expansive Flow

1 Introduction

The thrust of traditional aircrafts is along the flight direction, since the direction of the exhaust flow is fixed and opposite to the flight direction. In the modern development of propulsion technology, the expansion nozzle of the aircraft engine has been designed to have the capability of changing the direction of the exhaust flow, making the change of thrust direction possible [1]. The expansion nozzles with the capability of changing the thrust direction are known as thrust-vectoring nozzles. With a thrust-vectoring nozzle installed, the aircraft would have better manipulations under low-speed flight [2]. The reversing thrust of a thrust-vectoring nozzle can even be used as a brake. As a result, the new-generation fighters, such as F-22 and F-35 of USA as well as Su-30 and Su-37 of Russia, all employ the technology of the thrust-vectoring nozzle to improve their combat advantage.

In the present study, the software FLUENT adopting the SST-k-ω turbulence model and the dynamic moving-grid system [3] is employed for the simulation of the expansive flow of a thrust-vectoring nozzle, as shown in Fig. 1. For the configuration considered, the flaps are able to swing up or down in the *x* direction. The flaps can also turn left or right in the *y* direction. The boundary conditions are shown in Table 1.

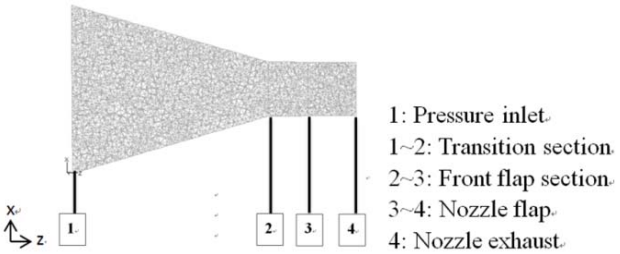


Fig. 1 Side view of a thrust-vectoring nozzle

Table 1 Boundary conditions

$A_1(m^2)$	9.409×10^{-3}
P_1(psi)	23.521
T_1(K)	1070
$A_4(m^2)$	4.096×10^{-3}
P_4(psi)	14.7 psi

2 Results and Discussion

The pressure contour with the flaps swinging up by 30° is shown in Fig. 2. The local expansion yields a low-pressure zone near the upper flap, and the pressure is even lower than the exit pressure. On the other hand, the local compressing causes a high pressure zone near the lower flap. The Mach-number contour with the flaps swinging up by 30° is shown in Fig. 3. The maximum Mach number is predicted to be 1.44 exhibited in the region near the upper flap. As a contrast, the Mach number is around 0.5 in the region near the lower flap. The average exhaust Mach number for this case is 0.7849 which is lower than that without swinging, which is predicted to be 0.8461.

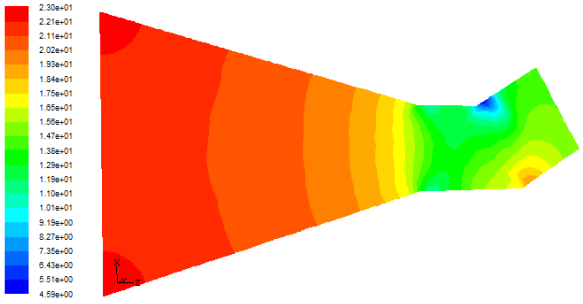


Fig.2 Pressure contour with the flaps swinging up by 30°

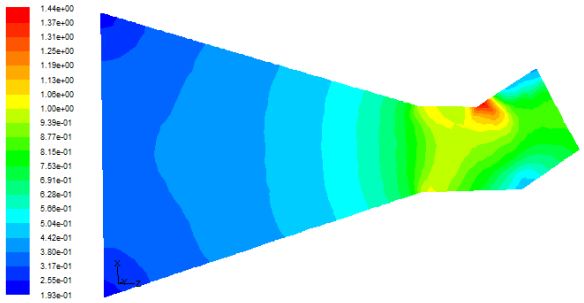


Fig.3 Mach-number contour with the flaps swinging up by 30°

The pressure contour with the flaps turning left by 15° is shown in Fig. 4. A local low-pressure zone is predicted near the left flap, while a local high-pressure zone is predicted near the right flap. The Mach number contour with the flaps turning left by 15° is shown in Fig. 5. The maximum Mach number is 1.36 which is predicted in the region near the left flap. The average exhaust Mach number for this case is 0.823 which is lower than that without turning, which is predicted to be 0.8461.

The thrust predicted at various swinging angles is depicted in Fig. 6. The thrust generally decreases with increasing swinging angles. The maximum thrust is 34.79 kN, while it is 16.76 kN with the flaps swinging up by 30°. The thrust predicted at various turning angles is shown in Fig. 7. The thrust reaches the maximum with the flaps turning left by 1.5°, and it is 37.86 kN. With the turning angle further increased, the thrust decreases with the increasing turning angle. The thrust is reduced to 33.65 kN with the flaps turning left by 15°.

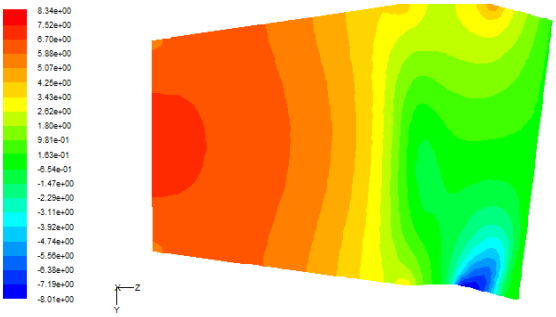


Fig.4 Pressure contour with the flaps turning left by 15°

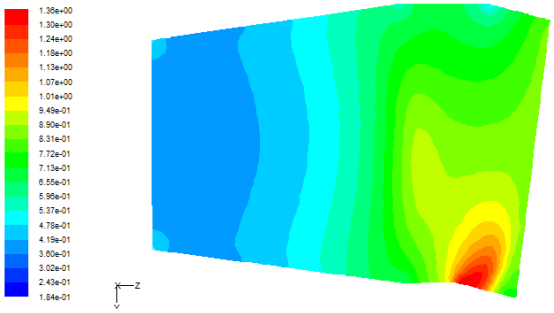


Fig.5 Mach-number contour with the flaps turning left by 15°

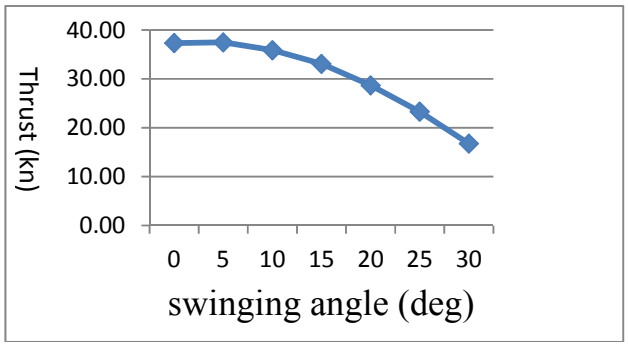


Fig.6 The thrust predicted at various swinging angles

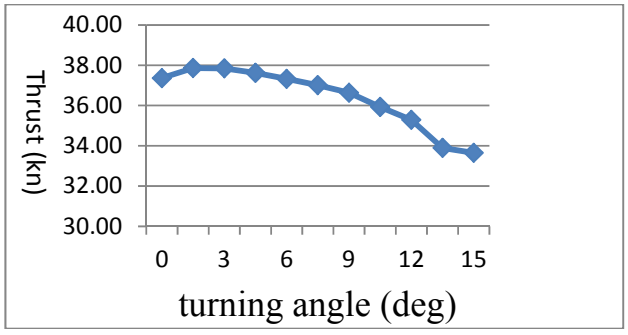


Fig.7 The thrust predicted at various turning angles

3 Conclusions

The dynamic moving-grid simulation mechanism has been developed successfully, and has been employed to carry out the analyses of the expansive flow of a thrust-vectoring nozzle. The numerical results show that the swinging or turning of the flaps makes the averaged exhaust Mach-number drop, and reduces the thrust. The flow near the upper flap has been predicted to first accelerate to a supersonic one and then reduce to a subsonic one at the nozzle exit as the flaps swing upward. Similarly, the flow near the left flap first accelerates to a supersonic one and then reduces to a subsonic one at the nozzle exit as the flaps swing left.

4 References

[1] C. W. Alcorn, M. A. Croom, M. S. Francis, and H. Ross, "The X-31 Aircraft: Advances in Aircraft Agility and Performance," Prog. Aerosp. Sci., Vol. 32, No. 4, pp. 377-413, 1996.

[2] A. J. Steer, "Low Speed Control of a Second Generation Supersonic Transport Aircraft Using Integrated Thrust Vectoring," Aeronaut. J., Vol. 104, No. 1035, pp. 237-245, May 2000.

[3] FLUENT 6.3, User Guide, FLUENT Incorporated, 2006.

Graphical User Interface for Arm Strength Training Machine

Mu-Song Chen[1], Tze-Yee Ho[2], Chih-Hao Chiang[2], Wei-Chang Hung[2]
[1] Dept. of Electrical Engineering, Da-Yeh University, ChangHua, Taiwan, R.O.C
[2] Dept. of Electrical Engineering, Feng Chia University, Taichung, Taiwan, R.O.C

Abstract - The realization of interactive communication between Arm Strength Training Machine (ASTM) and the users can make the training exercise and rehabilitation therapy become more convenient and realistic. In this article, we present a user-friendly Graphical User Interface (GUI) interface for an ASTM. The complete system consists of hardware implementation and software design, including a UART communication, an ADC converter, a microcontroller, and a permanent magnet synchronous motor (PMSM) drive for simulating the weight stack. The GUI interface is written in C language. Furthermore, the firmware of motor drive is designed based on the MPLAB development tool. To verify the feasibility and usefulness of the proposed method, the system is tested with different settings of desired torque and speed control to demonstrate its efficiency and convenience.

Keywords: Arm Strength Training Machine, Graphical User Interface

1 Introduction

Several conventional exercise apparatus, e.g. arm strength training machine, are usually coupled with a stack of iron weights through a series of pulleys and levels to hand grips [1-2]. Certainly, the user requires assistance with the experienced trainer or qualified rehabilitation therapist. However, the overall process always involves many time consuming stages to add or remove weights from the stack for efficient training and testing [3-4]. To resolve these problems, a permanent magnet synchronous motor is used to generate the opposition force for the user. As a result, a user-friendly interface for the arm strength training machine is realized to monitor the complete exercise cycle in real time situation. The framework of the system consists of a UART communication, ADC converter, personal computer, and a PMSM motor drive which is basically realized by a microcontroller for simulating the weight stack. The system software for human interface is developed under personal computer and written in C language. The rest of the paper is organized as follows. In section 2, the hardware configurations and the corresponding software descriptions of the arm strength training machine are discussed. Experimental results are demonstrated in section 3 and a conclusion is given in section 4.

2 Hardware Configurations and Software Descriptions

In the following, the complete framework of the proposed system is discussed in terms of hardware configurations and software descriptions, respectively.

A. Hardware Configurations

The hardware configurations mainly comprise two parts, as shown in Fig. 1. The first one is the ASTM which simulates the weight stack by using the PMSM motor drive. The second part is the human interface which provides an interface between the user and the ASTM. In Fig. 1, the dsPIC 30F4011 controller is the core part of the system. It is a 16-bit CPU with the capability of digital signal processing. The user can designate the desired instructions or commands, such as the desired speed or torque, to the ASTM by using the UART communication interface. The commands are then processed by the dsPIC digital signal controller. The dsPIC digital signal controllers provide designers with an easy upgrade path from 8-bit PIC microcontrollers and a cost effective option to 32-bit MCUs. Combined with hardware and free software, these 16-bit products are ideal for designs including high efficiency motor control. Thus, the dsPIC controller then generates the proper sinusoidal pulse width modulation (SPWM) signal to control the output of inverter in such a way to obtain the adequate motor torque according to the input command. Consequently, the speed, current, and encoder signals, are sensed and processed further. Therefore, the applied force and speed rate that the user currently exerts can be displayed on the human interface in real time condition.

Moreover, the built-in PWM module, addressable encoder interface module, and input capture module, can make the design become more efficient. The motor currents are also sensed through the current detection circuit. The magnet pole and rotor position are detected by the Hall effect sensor and the encoder.

B. Software Descriptions

The system software for human interface, including the serial communication protocol, is developed in C language. The firmware of motor drive is written and tested, based on the MPLAB development tool. The MPLAB Starter Kit for

dsPIC Digital Signal Controllers is a complete hardware and software tool suite for exploring applications based upon Microchip's dsPIC DSCs. The block diagram of the developed software is illustrated in Fig. 2.

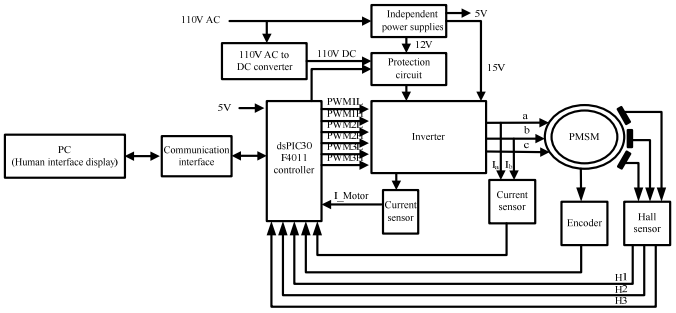


Fig. 1. The hardware configurations of the ASTM.

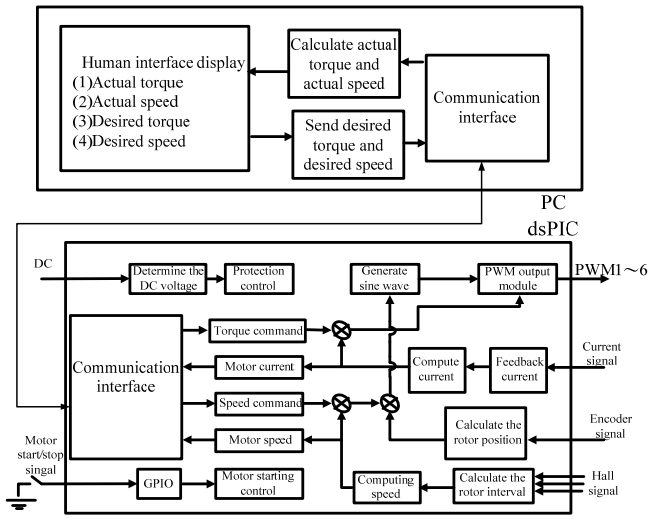


Fig. 2. The software descriptions of the ASTM.

The GUI interface is also designed to facilitate the operational commands to ASTM. Thus, the communication between them is very critical. In this paper, the actual torque and motor speed are received by the UART controller through the communication interface. Then, the SPWM technique is applied to drive the thee-phase inverter. Since the resolution of encoder is 2,500 pulses per revolution, the value of the counter in the microcontroller will be 5,000 counts. Because the sinusoidal waveform is symmetric for 0^o to 180^o, only the sine values of 0^o to 90^o are created which covers the 312 counts of encoder for a 8-pole rotor.

3 Experimental Results

The GUI interface for the arm strength training machine is tested by giving desired commands to verify its functions and correctness. In this experiment, the desired torque is 10 kg-cm and the desired speed control is 10 cm/sec. During operations of manipulating the ASTM, the actual torque and actual speed of the user exercising are correctly demonstrated in the GUI, as shown in Fig. 3.

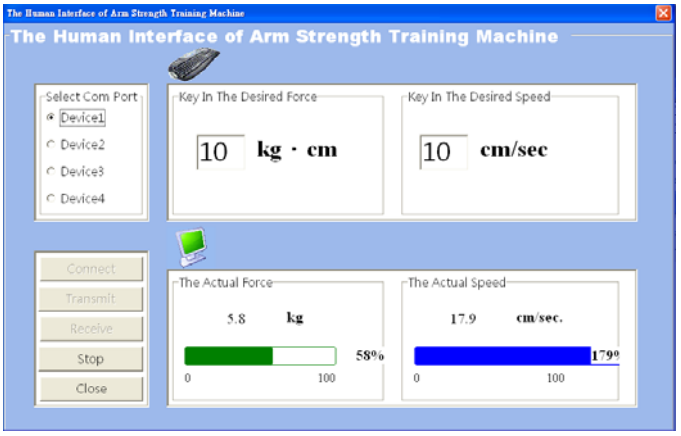


Fig. 3. The actual values displayed in human interface, Torque:10 kg-cm, Speed:10 cm/s.

4 Conclusions

In this paper, the implementation of a GUI interface for the arm strength training machine, based on the dsPIC controller, is realized and demonstrated. Without assistance of professional trainer or qualified rehabilitation therapist, the proposed system provides a user-friendly interface and can also facilitate the exercise for the users.

Acknowledgment

This research was partially supported by the National Science Council under Contract No. NSC 102-2623-E-035-002-D, NSC 101-2218-E-035-009, and NSC 102-2221-E-212-013.

5 References

[1] Frnado Bugallo, "Weight lifting apparatus having increased force on the return stroke," U.S. Patent No. 4563003 1983.

[2] Neiger et al, "Constant force exercise device," U.S. Patent No. 4678184, 1987.

[3] Prud'Hon, "Apparatus for training, investigation and re-education in particular for the neuro-muscular function," U.S Patent No. 4979733, 1990.

[4] Willim H. Englehardt, "Motor control for an exercise simulating weight stack-life fitness," U.S. Patent No. 5020794, 1991.

$\mathscr{H}_\infty$ Sampled-Data Control of LPV systems with time-varying delay*

H.Y. Jung, Ju H. Park and S.M. Lee

Abstract— **This paper considers the problem of sampled-data control for continuous linear parameter varying (LPV) systems. It is assumed that the sampling periods are arbitrarily varying but bounded. Based on a time-varying delay system transformed from the sampled-data control system, some less conservative results are obtained compared with the existing results. The proposed method for the designed gain matrix should guarantee asymptotic stability and a specified level of performance on the closed-loop hybrid system. The criteria for the existence of the controllers are derived in terms of LMI (Linear Matrix Inequality). Numerical examples are presented to demonstrate the effectiveness and the improvement of the proposed method.**

I. INTRODUCTION

Linear parameter-varying (LPV) systems have received considerable attention due to the fact that LPV models are useful to describe the dynamics of linear systems affected by time-varying parameters as well as to represent nonlinear systems in terms of a family of linear models. Also, time delays often appear in many physical, biological and engineering systems, and the existence of time delay may cause instability. Therefore, Stability analysis and control of delayed LPV systems have been received considerable attention by man researchers [1-4]. The main focus of the present work is to examine the sampled-data control design for linear parameter varying (LPV) systems. Because of the rapid growth of the digital hardware technologies, the sampled-data control method, whose the control signals are kept constant during the sampling period and are allowed to change only at the sampling instant, has been more important than other control approaches. The proposed design method guarantees asymptotic stability and optimized energy-to-energy gain of the closed-loop system from disturbance input to the system output. The criteria for the existence of the controllers are derived in terms of LMI (Linear Matrix Inequality). Compared with the method in [7,8], less conservative result is obtained by using new Lyapunov-Krosvoskii functional was utilized and reciprocally convex approach [9] is employed, which can derive a less conservative result than [7,8]. By means of numerical simulations, it is shown that the proposed results are effective and can significantly improve the existing ones.

II. PROBLEM STATEMENT AND PRELIMINARIES

Consider the following state-space representation for a linear parameter varying (LPV) system with time-varying delay

$$\begin{aligned}\dot{x}(t) &= A(\rho(t))x(t)+A_h(\rho(t))x(t-d(t))\\ &\quad +B_1(\rho(t))u(t)+B_2(\rho(t))w(t),\\ z(t) &= C(\rho(t))x(t)++C_h(\rho(t))x(t-d(t))\\ &\quad +D_1(\rho(t))u(t)+D_2(p(t))w(t), \qquad (1)\\ u(t) &= Kx(t_k), \qquad (2)\end{aligned}$$

where $x(t)\in\mathscr{R}^n$ is the state vector, $z(t)\in\mathscr{R}^n$ is the vector of controlled outputs, $w(t)\in\mathscr{R}^{n_w}$ is exogenous disturbance vector containing both process and measurement noise with finite energy and $u(t)\in\mathscr{R}^{n_u}$ is the control input vector. The system matrices $A(\cdot),B_1(\cdot),B_2(\cdot),C_1(\cdot),D_{11}(\cdot),D_{22}(\cdot)$ are real continuous functions of a time varying parameter vector $\rho(t)\in\mathscr{F}^v_{\mathscr{P}}$ and of appropriate trajectories defined as

$$\begin{aligned}\mathscr{F}^v_{\mathscr{P}} = \{\rho : \rho(t)\in C(\mathscr{R},\mathscr{R}_s) : \rho(t)\in\mathscr{F}, |\dot{p}(t)|\le v_i\\ i=1,2,\ldots,s,\forall t\in\mathscr{R}_+\}, \qquad (3)\end{aligned}$$

where $C(\mathscr{R},\mathscr{R}_s)$ is the set of continuous-time functions from $\mathscr{R}$ to$\mathscr{R}^s$, and $\{v_i\}_{i=1}^s$ are nonnegative numbers. The constraints in (3) imply that the parameter trajectories and their variations are bounded. The time delay is satisfied $0\le d(t)\le d,\dot{d}\le\mu$.

In this paper, the control signal is assumed to be generated by using a zero-order-hold (ZOH) function with a sequence of hold times $0\le t_0<t_1<\cdots<t_k\cdots<\lim_{k\to\infty}t_k=+\infty$.

$$t_{k+1}-t_k\le h.$$

The purpose of this paper is to design a proper sampled-data controller (2) such that the following condition hold.

1) The system (1) with $w(t)=0$ is asymptotically stable.

2) For some positive scalar γ, the following condition hold

$$||T_{wz}||_{i,2}<\gamma,$$

where $||T_{wz}||_{i,2}=\sup_{\rho\in\rho(t)\in\mathscr{F}^v_{\mathscr{P}}}\sup_{w(t)\in\mathscr{L}_2-0}\frac{||z||_{\mathscr{L}_2}}{||w||_{\mathscr{L}_2}}$

III. MAIN RESULTS

In this section, we present the stability and $\mathscr{H}_\infty$ norm performance analysis conditions for delayed LPV systems by deriving a set of linear matrix inequality conditions.

Theorem 1. For given $\gamma>0,h>0$, the LPV system (1) is asymptotically stable for all $0<h(t)\le h$ and satisfies $||z||_{\mathscr{L}_2}\le\gamma||w||_{\mathscr{L}_2}$, if there exist positive constant matrix

S.M. Lee is with the School of Eletronics Engineering, Daegu University, Kyongsan, Republic of Korea moony@daegu.ac.kr

H.Y. Jung and Ju.H. Park are with the Department of Electrical Engineering, Yeungnam University, Dae-Dong, Kyongsan 712-749, Korea hoyoul@yu.ac.kr, jessie@ynu.ac.kr

$\bar{P}, \bar{Q}, \bar{R}$, symmetric G and any matrix T_1, M satisfying the following LMIs

$$\begin{bmatrix} \hat{\Sigma}_{11} & \bar{\Sigma}_{12} & \bar{\Sigma}_{13} & \bar{\Sigma}_{16} & B_2 & GC_1^T \\ * & \hat{\Sigma}_{22} & \bar{\Sigma}_{23} & \bar{\Sigma}_{26} & \lambda_1 B_2 & 0 \\ * & * & -\bar{Q}-\bar{R} & -\lambda_2 G & \lambda_2 B_2 & M^T D_1^T \\ * & * & * & \Sigma_{44} & \lambda_3 B_2 & 0 \\ * & * & * & * & -\gamma I & D_2^T \\ * & * & * & * & * & -\gamma I \end{bmatrix} < 0, \quad (4)$$

$$\begin{bmatrix} R & T_1 \\ * & R \end{bmatrix} \geq 0, \quad (5)$$

where

$$\begin{aligned} \hat{\Sigma}_{11} &= \bar{Q} - \tilde{R} + AG + GA^T \\ \Sigma_{12} &= R - T_1 + G^{-1}B_1 K + \lambda_1 A^T G^{-1}, \\ \Sigma_{13} &= T_1 + \lambda_2 A^T G^{-1}, \\ \Sigma_{16} &= P - G^{-1} + \lambda_3 A^T G^{-1}, \\ \Sigma_{23} &= R - T_1 + \lambda_2 K^T B_1^T G^{-1}, \Sigma_{24} = \lambda_1 G^{-1} A_h, \\ \Sigma_{26} &= -\lambda_1 G^{-1} + \lambda_3 K^T B_1^T G^{-1}, \\ \hat{\Sigma}_{44} &= -h^2 \bar{R} - 2\lambda_3 G, \end{aligned}$$

Further, the sampled-data controller gain matrix in (**??**) is given by $K = MG^{-1}$.

Proof. The detailed proof is omitted.

IV. NUMERICAL EXAMPLE

Example 1 This example is motivated by the control of chattering during the milling process [14]. The dynamical equations of the system is

$$\begin{aligned} \ddot{x}_1 &= \frac{1}{m}\left[-k_1 - k sin(\phi) sin(\phi+\beta)\right] x_1 \\ &+ \frac{k_1}{m_1} x_1 \left(t - \frac{\pi}{w}\right) + \frac{k_1}{m_1} x_2 + \frac{k_1}{m_1} w(t) \\ \ddot{x}_2 &= \frac{k_1}{m_2} x_1 - \frac{k_1 + k_2}{m_2} x_2 - \frac{c}{m_2}\dot{x}_2 + \frac{k_1}{m_2} u, \end{aligned} \quad (6)$$

with $m_1 = 1, k_1 = 10, k_2 = 20, c = 0.5, \beta = 70^0$. It is noted that $sin(\phi+\beta)sin(\phi) = 0.1710 - 0.5cos(2\phi+\beta)$. We define the scheduling parameter vector as $\rho(t) = [\rho_1(t), \rho_2(t)]^T$ with $\rho_1(t) = cos(2\phi(t) + \beta(t))$ and $\rho_2(t) = w(t)$. The rotation speed w is assumed to vary between 200 rpm and 2000 rpm, and the maximum variation rate is 1000 rpm/sec. The parameter space associated with the LPV parameters is as follows

$$\begin{aligned} &\rho_1(t) \in [-1, 1]. \ \ |\dot{\rho}| \leq 418.9(rad/sec), \\ &\rho_2(t) \in [20.94, 209.4](rad/sec). \ \ |\dot{\rho}| = 104.7(rad/sec^2). \end{aligned}$$

For this example, we assume that $d = 0.15, h = 0.1, \lambda_1 = 1.4, \lambda_2 = 0.5$ and $\lambda_3 = 0.2$. Solving the LMIs in Theorem 1, we obtain an H_∞ performance bound is $\gamma = 1.9$, which is less than the derived results $\gamma = 2.4$ in [14]. The corresponding controller gain matrix is

$$K = \begin{bmatrix} 16.9370 & -2.3223 & 7.6668 & -15.4072 \end{bmatrix}$$

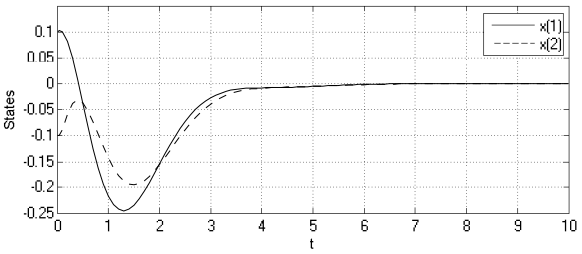


Fig. 1. State response of the delayed LPV system in Example 1.

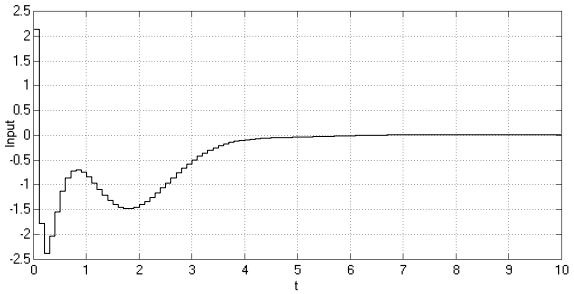


Fig. 2. Input of the delayed LPV system in Example 1.

With the above controller gain, taking $w(t) = \sin te^{-10t}$, Fig 1 and Fig. 2 shows the simulation results.

V. CONCLUSIONS

This paper proposed sampled-data control design for a continuous-time system. By using a reciprocally convex approach, less conservative results than existing methods in the literature for control of LPV time delay systems are obtained. The proposed method guarantees asymptotic stability and optimized energy-to-energy gain of the closed-loop system from disturbance input to the system output. Finally, using two examples, we demonstrate the effectivenes of the proposed method.

REFERENCES

[1] F. Wu and K.M. Grigoriadis, LPV Systems with parameter-varying time delays: analysis and control, Automatica, 37(2): 221-229, 2001.
[2] C. Briat, O. Sename, and J.F. Lafay. Parameter dependent statefeedback control of LPV time-delay systems with time-varying delays using a projection approach. IFAC World Congress, Seoul, 2008.
[3] S.M. Lee, J.H. Park, D.H. Ji and S.C. Won, Robust model predictive control for LPV systems using relaxation matrices, IET Control Theory Appl., 1(6): 1567.1573-2007.
[4] S.M. Lee, S.C. Jeong, D.H. Ji,S.C. Won, Robust model predictive control for LPV systems with delayed state using relaxation matrices, Proc. American Control Conference, 2011.
[5] A. Ramezaninfar, J. Mohammadpour, and K. Grigoriadis. Sampled-data control of LPV systems using input delay approach. Proc. IEEE Conference on Decision and Control, to appear, 2012.
[6] A. Ramezanifar, J. Mohammadpour, K.M. Grigoriadis, Sampled-data Control of Linear Parameter Varying Time-delay Systems Using State Feedback, Proc. American Control Conference, 2013.
[7] P. Park, J. W. Ko, and C. Jeong, Reciprocally convex approach to stability of systems with time-varying delays, Automatica, 47(1): 235-238, 2011.

A Dynamical System Approach to Compute the Evolution of Daily Commuting Traffics Interacted with Travel Time Information[1]

M.-C. Hwang[2], H.-J. Cho[3], and Y.-H. Huang[3]
[2]Microelectronics and Information Systems Research Center, National Chiao Tung University, Hsinchu, Taiwan, Republic of China
[3]Department of Transportation Technology and Management, National Chiao Tung University, Hsinchu, Taiwan, Republic of China

Abstract - *In this paper, a day-to-day flow evolution model is developed for a vehicular network equipped with Intelligent Transportation Systems (ITS) providing road users with pre-trip travel time information. Daily route swapping process of commuting traffic is considered and formulated as a dynamical system. It means that a single user will change to a less costly path tomorrow compared to his travel time over today's route and travel time of other routes detected today by ITS. The result tells that the proposed model has the ability to provide the evolution of traffic states that helps to design control strategies and to mitigate daily congestion not just based only on equilibrium state. A simple network is provided to illustrate the volume-shifting among competing paths and the asymptotic behavior of system variables.*

Keywords: Dynamical System, Lyapunov Stability, Day-to-day Traffic Assignment, Intelligent Transportation Systems (ITS)

1 Introduction

This paper attempts to formulate how traffic information distributed by ITS operations influences the temporal trajectory of network flows in a theoretical viewpoint. Based on similar behavioral assumptions of minimal-travel-time seeking (MinTTS) and ordinal perception of travel-time-updating process (OPTTU)[1-4], the interactions between road users and ITS operator are decomposed into four parts: the travel time induced path flow dynamics (PFD_{TT}), the demand induced path flow dynamics (PFD_D), the travel time induced demand dynamics (DD_{TT}), and the predicted travel time dynamics for an origin-destination (O-D) pair ($PTTD_{OD}$). PFD_{TT} describes the collective results of user's daily travel decision by transfer to less congested path with path travel time information provided from ITS services. The other three components, $PTTD_{OD}$, DD_{TT}, and PFD_D, are concentrated on the evolutionary behavior of system variables predicted by ITS operators to act as a benchmark in guiding whole systems towards an expected and preferable status. For paper space saving, detailed literature review and stability analysis [5] are omitted in this version of paper.

2 Modeling network dynamics

The basic stimulus-response structure incorporates three main implications in this study. They are: all responses and stimuli discussed hereinafter are macroscopic and divided into two clusters, users and ITS operator; response $\dot{\boldsymbol{R}}_i^t$ is defined as the time change rate of system variable with one day lag and denoted as a function of stimulus $\boldsymbol{St}_i^t$ and sensitivity $\boldsymbol{Se}_i^t$; and stimulus is specified as the difference between experienced (or observed) status $\tilde{\boldsymbol{x}}_i^t$ and expected (or predicted) status $\bar{\boldsymbol{x}}_i^t$ of a system variable *i* at day *t*.

$$\dot{\boldsymbol{R}}_i^t \equiv \boldsymbol{F}_i(\boldsymbol{Se}_i^t, \boldsymbol{St}_i^t), \quad (1)$$

$$\boldsymbol{St}_i^t \equiv \tilde{\boldsymbol{x}}_i^t - \bar{\boldsymbol{x}}_i^t, \quad (2)$$

2.1 User dynamics

User dynamics defined in this paper is the travel time induced path flow dynamics (PFD_{TT}). By similar proposition of [4], the intention of developing PFD_{TT} is extremely straightforward that if users wish to improve travel time tomorrow, they might select a faster path than today. The stimulus defined in the following formulation is the collective effects of travel time difference [6] at day *t* between two statuses of each user *i* selecting path *p*, the "experienced" status $\boldsymbol{c}_{i,\boldsymbol{p}}^t$ and the "expected" status $\breve{\boldsymbol{c}}_{\boldsymbol{w}}^t$ (minimal travel time) for O-D pair *w*.

$$\dot{\boldsymbol{h}}_{\boldsymbol{p},\boldsymbol{PFD}_{TT}}^t \equiv -\left(\alpha_{\boldsymbol{p}}^t\right)\left(\sum_{i=1}^{h_p^t}\left(\boldsymbol{c}_{i,\boldsymbol{p}}^t - \breve{\boldsymbol{c}}_{\boldsymbol{w}}^t\right)\right) = -\alpha_{\boldsymbol{p}}^t \boldsymbol{h}_{\boldsymbol{p}}^t\left(\boldsymbol{c}_{\boldsymbol{p}}^t - \breve{\boldsymbol{c}}_{\boldsymbol{w}}^t\right) \quad (3)$$

With full information of travel time, Equation (4) is the result of ordinal perception process which determines the target

[1] Corresponding author : M.-C. Hwang, E-mail : mch0619@yahoo.com. The authors would like to thank the Ministry of Science and Technology of Taiwan for the financial support of this study.(NSC-102-2221-E-009-111)

routes of flow shifting, from path p to path q if $\boldsymbol{c}_q^t < \boldsymbol{c}_p^t$. It is assumed that the amount of shift is weighted by travel time ($\omega_{p\to q}^t$) to reflect the congestion effect.

$$\dot{\boldsymbol{h}}_{p\to q}^t \equiv -\dot{\boldsymbol{h}}_{p,PFD_{TT}}^t \omega_{p\to q}^t \quad (4)$$

2.2 ITS operator dynamics

With similar adaptive and learning process presented previously, ITS operators utilize the detected information to compare and update three system variables. The first one of ITS operator dynamics is demand induced path flow dynamics, PFD_D. It comes from the difference between "observed" path flow $\boldsymbol{h}_w^t$ and "expected/predicted" demand $\boldsymbol{D}_w^t$ for an O-D pair w weighted with sensitivity $\alpha_{d_w}^t$ and travel time $\gamma_{p,w}^t$.

$$\dot{\boldsymbol{h}}_{p,PFD_D}^t \equiv \alpha_{d_w}^t (\boldsymbol{D}_w^t - \boldsymbol{h}_w^t)\gamma_{p,w}^t \quad (5)$$

The second ITS operator dynamics is the predicted travel time dynamics for an O-D pair ($PTTD_{OD}$). The difference between "observed" path flow and "expected/predicted" demand for an O-D pair w generates a simultaneous effect on $PTTD_{OD}$ to reflect an expected increase of $\breve{\boldsymbol{c}}_w^t$ if path flow is predicted to increase, i.e. if $\boldsymbol{D}_w^t > \boldsymbol{h}_w^t$.

$$\dot{\boldsymbol{c}}_w^t \equiv \beta_w (\boldsymbol{D}_w^t - \boldsymbol{h}_w^t) \quad (6)$$

The last ITS operator dynamics is the travel time induced demand dynamics (DD_{TT}). Demand of an O-D pair is allowed to be a little disturbed purely by travel time fluctuations. It is formulated as the response due to the difference between the minimal path travel time (an "observed" status) and the predicted travel time (an "expected" status) of an O-D pair.

$$\dot{\boldsymbol{D}}_w^t \equiv \alpha_w (\breve{\boldsymbol{c}}_w^t - \boldsymbol{c}_w^t) \quad (7)$$

3 Numerical results

A five links network with strict monotone cost function with respect to link flow is used to evaluate the proposed model numerically. There is one O-D pair connected by three paths in this example network. Figure 1 shows the effects of path flow dynamics (path flow switching) that comes from the net volumes simultaneously generated by PFD_{TT} and PFD_D. And the trajectory seems to be convergent as time evolves.

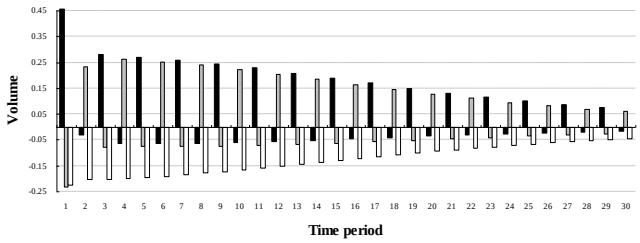


Figure 2. Path flow switching within the first 30 time periods

Table 1 shows the selected results of system variables at the initial time, at the 200th time step, and at the steady state respectively. The three path travel times and the predicted travel time of O-D pair are all the same which reveals the user equilibrium [7]. The predicted demand for an O-D pair is also equal to the sum of path flows at steady state.

Table 1. Selected results of system variables

	Flow			Travel time		
	Initial	State at t=200	Steady state	Initial	State at t=200	Steady state
Path 1	30	40.55	40.55	91.74	93.74	93.74
Path 2	45	41.12	41.11	96.35	93.74	93.74
Path 3	10	4.96	4.96	94.60	93.75	93.74
Link 1	40	45.51	45.55	41.25	42.09	42.09
Link 2	45	41.12	41.11	63.00	62.09	62.09
Link 3	10	4.96	4.95	20.00	20.00	20.00
Link 4	30	40.55	40.55	50.49	51.65	51.65
Link 5	55	46.08	46.07	33.35	31.65	31.65
Predicted travel time of O-D pair w				92.00	93.75	93.74
Demand of O-D pair w				90.00	86.63	86.61
Sum of path flows				85.00	86.63	86.61

4 Conclusion

The proposed nonlinear system considering both user and operator dynamics is organized as four parts: travel time induced path flow dynamics (PFD_{TT}), predicted travel time dynamics for an O-D pair ($PTTD_{OD}$), travel time induced demand dynamics (DD_{TT}), and demand induced path flow dynamics (PFD_D). A simple numerical example is provided to demonstrate the shifting effect among path flow. By collecting path-flow shifting information, traffic control strategies can then be simulated in transition state but equilibrium state which usually appeared in historical studies. That is essential to make the evaluation of the effectiveness and efficiency of ITS-related deployments more reasonable

5 References

[1] Cho, H.J., and Hwang, M.C. "Day-to-day Vehicular Flow Dynamics in Intelligent Transportation Network"; Mathematical and Computer Modelling, 41, 501-522, 2005a.

[2] Cho, H.J., and Hwang, M.C. "A Stimulus-Response Model of Day-to-day Network Dynamics"; IEEE Transactions on Intelligent Transportation Systems, 6, 17-25, 2005b.

[3] Friesz, T.L., Bernstein,D., Mehta, N.J., Tobin, R.L., and Ganjalizadeh, S. "Day-to-day Dynamic Network Disequilibria and Idealized Traveler Information Systems"; Operations Research, 42, 1120-1136, 1994.

[4] Smith, M.J. "The Existence, Uniqueness, and Stability of Traffic Equilibria"; Transportation Research Part B, 13, 295-304, 1979.

[5] Alligood, K.T., Sauser, T.D. and Yorke J.A. "Chaos: An Introduction to Dynamical Systems". Springer-Verlag, 1997.

[6] Mahmassani, H.S., Chang, G.L., and Herman, R. "Individual Decisions and Collective Effects in Simulated Traffic System"; Transportation Science, 20, 258-271, 1986.

[7] Wardrop, J.G. "Some Theoretical Aspects of Road Traffic Research", Proceedings of the Institute of Civil Engineers Part II, 325-378, 1952.

SESSION

LATE BREAKING PAPERS AND POSITION PAPERS: SCIENTIFIC COMPUTING

Chair(s)

TBA

Parallel Scaling Performance and Higher-Order Methods

A. Jared Buckley[1] **and B. Gaurav Khanna**[1]
[1]Physics Department, University of Massachusetts Dartmouth, Dartmouth, MA, USA

Abstract— *There is considerable current interest in higher-order methods and also large-scale parallel computing in nearly all areas of science and engineering. In this work, we take a number of basic finite-difference stencils that compute a numerical derivative to different orders of accuracy and carefully study the scaling performance of each, on a parallel computer cluster. We conclude that if one has a code that exhibits a high order of convergence, then there is likely to be no significant gain through cluster parallelism in the context of total execution or "wall clock" time. Conversely, for a low order code that exhibits good parallel scaling, there is insignificant gain through the implementation of a higher-order convergent algorithm.*

Keywords: higher-order, finite-difference, parallel, scaling

1. Introduction

In recent decades there has been a tremendous rise in numerical computer simulations, in nearly every area of science and engineering. This is largely due to the development of (Beowulf) cluster parallel computing that involves connecting together "off-the-shelf" computing units (for example, commodity desktop or laptop computer processors) into a configuration that would achieve the same level of performance, or even outperform, traditional supercomputers at a fraction of the cost [1]. The main reason behind the significant cost benefit of cluster computing is that it is entirely based on mass-produced, consumer hardware. Computational science has benefited and expanded tremendously in the last decade due to the rapid improvements in processor performance (*Moore's Law*) and major price drops due to mass production and associated market forces.

In addition to the strong increase in the interest in parallel cluster computing, there is also a rising trend in developing numerical algorithms that converge faster than the common second-order accurate schemes [2]. Some examples of such higher-order convergent methods are – higher-order finite-differencing, spectral collocation method, radial basis function method, finite-element and others [3], [4], [5], [6], [7], [8], [9], [10], [11]. In this work, we restrict ourselves to higher-order finite-difference schemes, however, we anticipate that our findings are generic enough that they would apply to any higher-order method.

The main goal of this work is to clearly demonstrate a form of "trade-off" between parallel computing and higher-order methods. This trade-off stems from the detailed parallel scaling behavior of the various higher-order schemes under specific conditions. In particular, our main assumption in this work is that the physical or engineering problem to be solved numerically has a known degree of tolerance or error acceptable for the solution, and is a given fixed quantity. We interpret this error to be the scale of the "discretization" or truncation error arising from the numerical scheme, which is, of course, a significant simplification – however, one that is reasonable for a wide class of problems. In other words, *we study the scaling performance of different order finite-difference methods given a fixed level of the discretization error*.

The outcome of our study suggests several very significant conclusions: (a) If one has a parallel code that scales well and exhibits second-order convergence, there is insignificant gain to be expected from a higher-order method implementation, assuming one has a large enough computational resource available, and the major consideration is the total execution time; (b) if one has a serial code exhibiting higher-order convergence (say, higher than fourth-order) then there is no significant gain from a parallel algorithm in a similar context; and (c) depending upon the acceptable error level, there is likely an optimal approach i.e. a combination of parallelism and method-order that would be ideal for the problem.

This article is organized as follows: In Section 2, we present a simple parallel scaling model that predicts the outcome of our planned study, based on simple heuristic reasoning. In Section 3, we detail the method of our study and present explicit mathematical expressions and our approach towards cluster parallelism. In Section 4, we show and discuss our results, and we end with some conclusive remarks in Section 5.

2. Simple Scaling Model

In this section, we present a simple model that predicts the parallel scaling behavior for a finite-difference method of any order. The model will help explain our findings and may provide some predictive value for other codes beyond the simple sample code we consider in this work.

Let us say that one is interested in performing a simple one-dimensional (1D) numerical derivative using a second-order finite-difference stencil. An important parameter that must be chosen is the grid resolution, that is typically set by the grid size N. The number of numerical calculations necessary to perform the derivative computation on the entire grid would then be on the order of N (more accurately, it would

be closer to $2N$ calculations – but let us ignore the constant pre-factors for this discussion). Now, let's assume that one attempts the same computation on a large parallel cluster with n processors using a standard domain-decomposition approach. Each processor would then perform N/n computations and would have to communicate two values (the boundaries of its subgrid) to neighboring processors. The parallel scaling behavior is largely determined by the ratio of the time-scale associated with this communication, to the time-scale of the actual numerical calculations on the subgrid. For good scaling, the entire computation should be heavily dominated by the calculations being performed by the processors and not by the communication. Therefore, as N increases, better scaling behavior is expected here.

Now, let us consider the same in the context of a higher-order finite-difference stencil of order p in 1D. For the *same* level of error as the second-order case above, one would only need a grid size on the scale of $N^{2/p}$. Thus, in a parallel computation environment, with n processors, each subgrid would be of size $\frac{N^{2/p}}{n}$ and the number of computations performed by each processor would be on the scale of $\frac{p}{n}N^{2/p}$. The number of values to communicate from one processor to another would increase to p. This is explained in detail in the next section. Thus, the ratio of computation to communication would behave as $N^{2/p}$ which drops dramatically as one increases the method order p, due to the power of $2/p$ in the expression[1]. This implies that *the scaling of higher-order methods, in general, is expected to be worse compared to the second-order case, in the context of a fixed discretization error.*

The question that we will address in this work, is whether the improved scaling of lower order methods is *enough* to give them an advantage in the context of the most important aspect of a numerical computation – the *total execution or "wall clock" time*. In fact, given our model above, we can make an estimate of how this could occur. Let us assume that for a given higher-order method, say, fourth-order and a problem size of interest, the parallel scaling performance is such that one is actually better off simply utilizing a serial code. The execution time would then be on the scale of $\sqrt{N}$. On the other hand, assuming that the second-order code has better parallel scaling, as would be expected, one would estimate the wall clock time to be on the scale of N/n. Thus, if one has computational resources with $n \sim \sqrt{N}$ at one's disposal, then at least in terms of total execution time, the two methods will complete the computation on the same time-scale. For the case of order p, this would change to $n \sim \frac{N^{1-2/p}}{p}$. This is the main point that we explicitly investigate in the following sections using finite-difference schemes of order 2, 4, 6 and 8.

[1]Since such communication is typically *latency* bound, as opposed to *bandwidth* bound, the p-dependence of this ratio is better estimated to be $pN^{2/p}$.

3. Methodology

In this section, we describe in detail the method of study adopted in this work. We begin with a discussion of higher-order finite-difference stencils, followed by our results from correlating the error with grid size N and end with a description of our parallel code implementation.

For the finite-difference calculations, we used a cosine function on a 1D domain from 0 to 12π:

$$f(x) = \cos x, \;\; x \in [0, 12\pi)$$

Now, for a numerical implementation, x is discretized simply as:

$$x_i = ih$$

where

$$h = \frac{12\pi}{N}$$

and i is an index that labels an arbitrary grid point on the domain.

To calculate the derivative of a function at grid point i using the finite-difference schemes, it is necessary to use the function values at neighboring grid points. As the order of the scheme increases, more grid points are needed for the calculation.

In the context of this work, we focus our attention on the first derivative of $f(x)$. At grid point i, using the various different order central finite-difference schemes, the derivative is given as [2]:
Order 2:

$$\frac{1}{h}(\frac{-1}{2}f_{i-1} + \frac{1}{2}f_{i+1})$$

Order 4:

$$\frac{1}{h}(\frac{1}{12}f_{i-2} + \frac{-2}{3}f_{i-1} + \frac{2}{3}f_{i+1} + \frac{-1}{12}f_{i+2})$$

Order 6:

$$\frac{1}{h}(\frac{-1}{60}f_{i-3} + \frac{3}{20}f_{i-2} + \frac{-3}{4}f_{i-1} + \frac{3}{4}f_{i+1} + \frac{-3}{20}f_{i+2} + \frac{1}{60}f_{i+3})$$

Order 8:

$$\frac{1}{h}(\frac{1}{280}f_{i-4} + \frac{-4}{105}f_{i-3} + \frac{1}{5}f_{i-2} + \frac{-4}{5}f_{i-1} + \frac{4}{5}f_{i+1} + \frac{-1}{5}f_{i+2} + \frac{4}{105}f_{i+3} + \frac{-1}{280}f_{i+4})$$

where the notation, $f_i = f(x_i)$.

It is clear from the above expressions that finite-difference schemes at higher orders produce an increasingly wider stencil. These wide stencils become important in the context of parallel computing as the passing of messages increases significantly with stencil size. As an example, a grid point

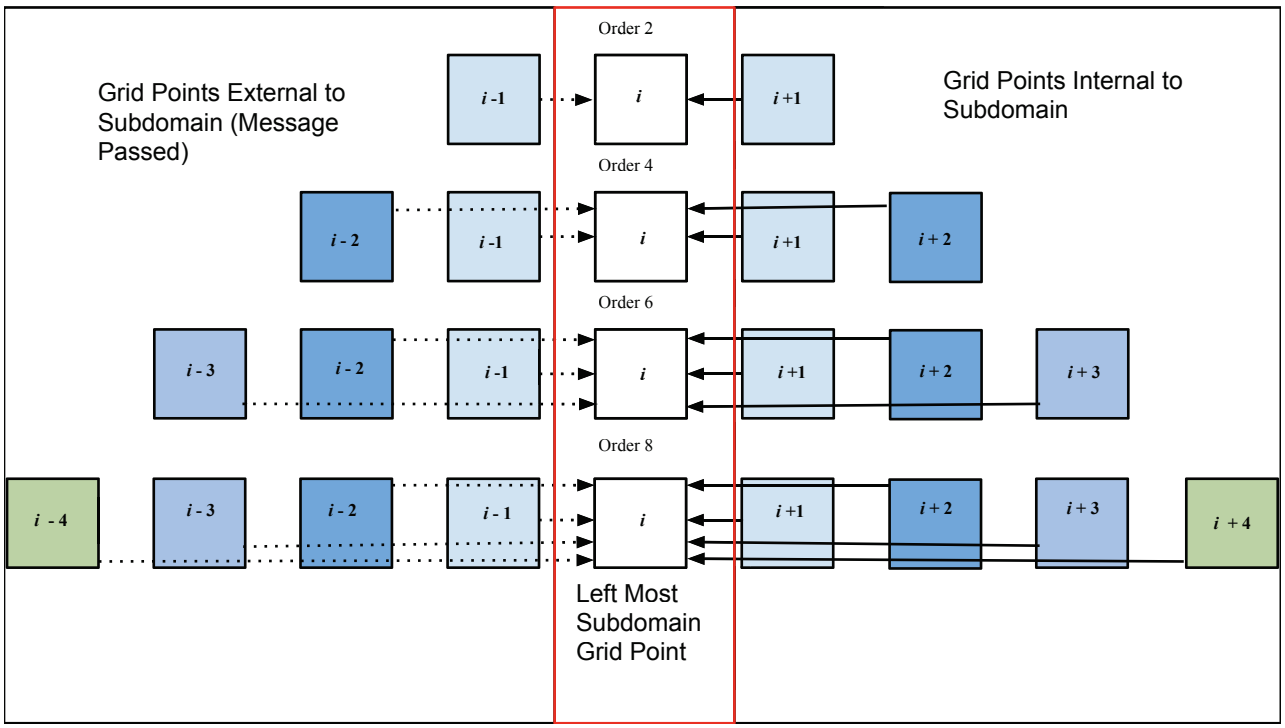


Fig. 1: The stencil structure for various finite-difference orders on a parallel computer cluster. The grid point labelled with index i is meant to reside on the left-edge of a subdomain. The grid points further on the left, must be communicated over the network (dotted-lines) while the ones on the right do not (solid-lines). Note that for higher-order stencils information from additional grid points have to be communicated across the network.

calculation at the edge of a process' subdomain would need 1 grid point value passed from outside the subdomain for order 2, while order 8 would require 4 grid point values to be passed. It is this communication overhead that influences the parallel scaling of the finite-difference schemes as described in Section 2. We developed computational routines, written in the C programming language, to better understand the behavior of the parallel scaling over a network.

Because we are concerned with the scaling at a fixed level of error, it was first necessary to correlate N with a given error level. This was achieved using an iterative algorithm that searched for a given error value for each finite-difference order being investigated. The algorithm calculated the first derivative of the cosine function in two ways: using the math library sine function and using the finite-difference formula at a value of N. The error was calculated at each point in the domain, and the maximum error on the domain was compared to the given error value. If the calculated error was less than the given error, the value of N was recorded; otherwise, N was incremented up by one and the process was repeated. The correlation of N with the error for each investigated finite-difference order is given in Tab. 1 and Fig. 2. The values of N fit the expected patterns extremely well.

With the correlation of N and error known, we developed a parallel message-passing (MPI) [12] routine to study the scaling behavior of the finite-difference schemes at fixed error values. The MPI routine divided the domain into even subdomains, with each subdomain associated with an MPI rank. Separate routines were developed for finite-difference orders 2, 4, 6, and 8. Each routine contained the appropriate finite-difference stencil and a modified MPI communication setup to allow for the transfer of the appropriate grid point values. MPI calls for higher-order cases were set to have higher buffer sizes to accommodate the increased need for grid points external to an MPI rank subdomain. Before any finite-difference calculations were performed, MPI ranks communicated in order to transfer their respective subdomain edges to the nearest logical MPI rank. We used MPI blocking calls for all communication. To prevent blocking calls from locking up the routine, communication was broken into two steps. The left edge of the MPI rank subdomains was sent to the nearest rank on the left (rank 0 excluded), then the right edge of the MPI rank subdomains was sent to the nearest rank on the right (maximum rank excluded). This approach is depicted graphically in Fig. 1. Once communication was completed, each rank independently computed the finite-difference first derivative of their respective subdomain. This process was repeated several times to allow for measurable execution wall clock times.

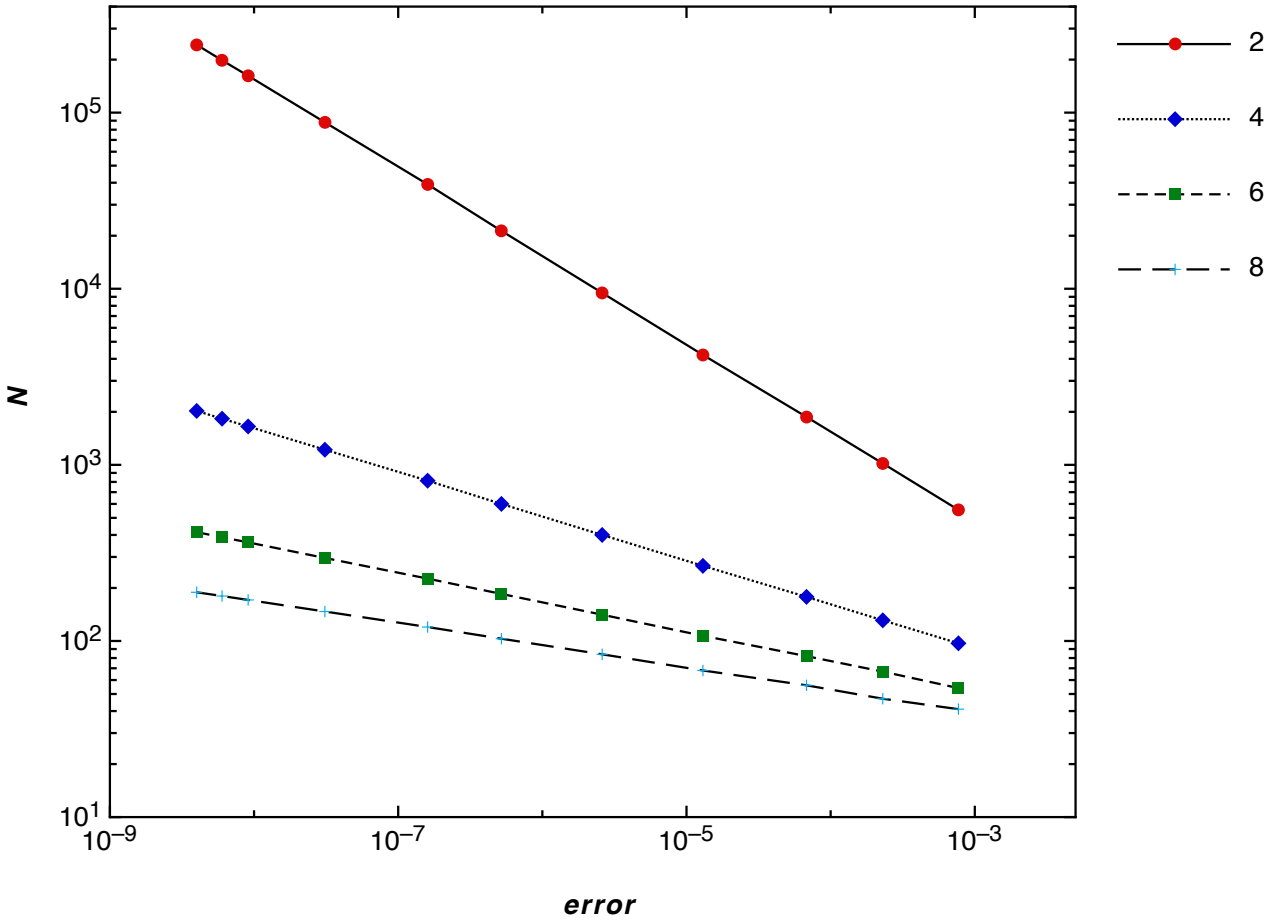


Fig. 2: The correlation of N with the error for various finite-difference orders. The powers of N computed from this data are: -1.998, -3.955, -5.942 and -7.876 respectively.

Our test cluster was Air Force Research Lab's CONDOR supercomputer. This system is a heterogeneous supercomputer comprised of commercial-off-the-shelf commodity components with 500 TFLOPS of processing power, and is a"green" supercomputer, designed to consume significantly less energy than comparable supercomputers [13]. All computations were performed using quadruple-precision floating-point numerical accuracy, as is often necessary in the context of higher-order methods, in order to reduce the roundoff errors to acceptable levels.

Table 1: Correlation of N and Error

Error	2	4	6	8
7.7×10^{-4}	555	97	54	41
2.3×10^{-4}	1019	131	67	47
6.8×10^{-5}	1871	178	82	56
1.3×10^{-5}	4210	267	107	68
2.6×10^{-6}	9473	400	141	84
5.2×10^{-7}	21313	600	185	103
1.6×10^{-7}	39153	813	226	120
3.1×10^{-8}	88094	1219	296	147
9.1×10^{-9}	161843	1652	363	171
6.0×10^{-9}	198222	1829	388	180
4.0×10^{-9}	242766	2024	415	189

4. Results

In this section we present the results obtained due to the approach and methodology as detailed in the previous sections.

In Fig. 3 we depict the speedup as a function of the number of processors n, for a grid size $N = 39,153$ or error-level 1.5×10^{-7}. The speedup is defined relative to the second-order code running on a single processor. We show the same for all the method orders considered in this work, namely 2, 4, 6 and 8. The expectations, as presented in Section 2, are clearly borne out in our speedup data. The second-order method, scales the best with n, and the higher-order cases exhibit much poorer scaling. In fact, the sixth and eight-order cases do not scale at all! One is clearly better off running those in serial mode.

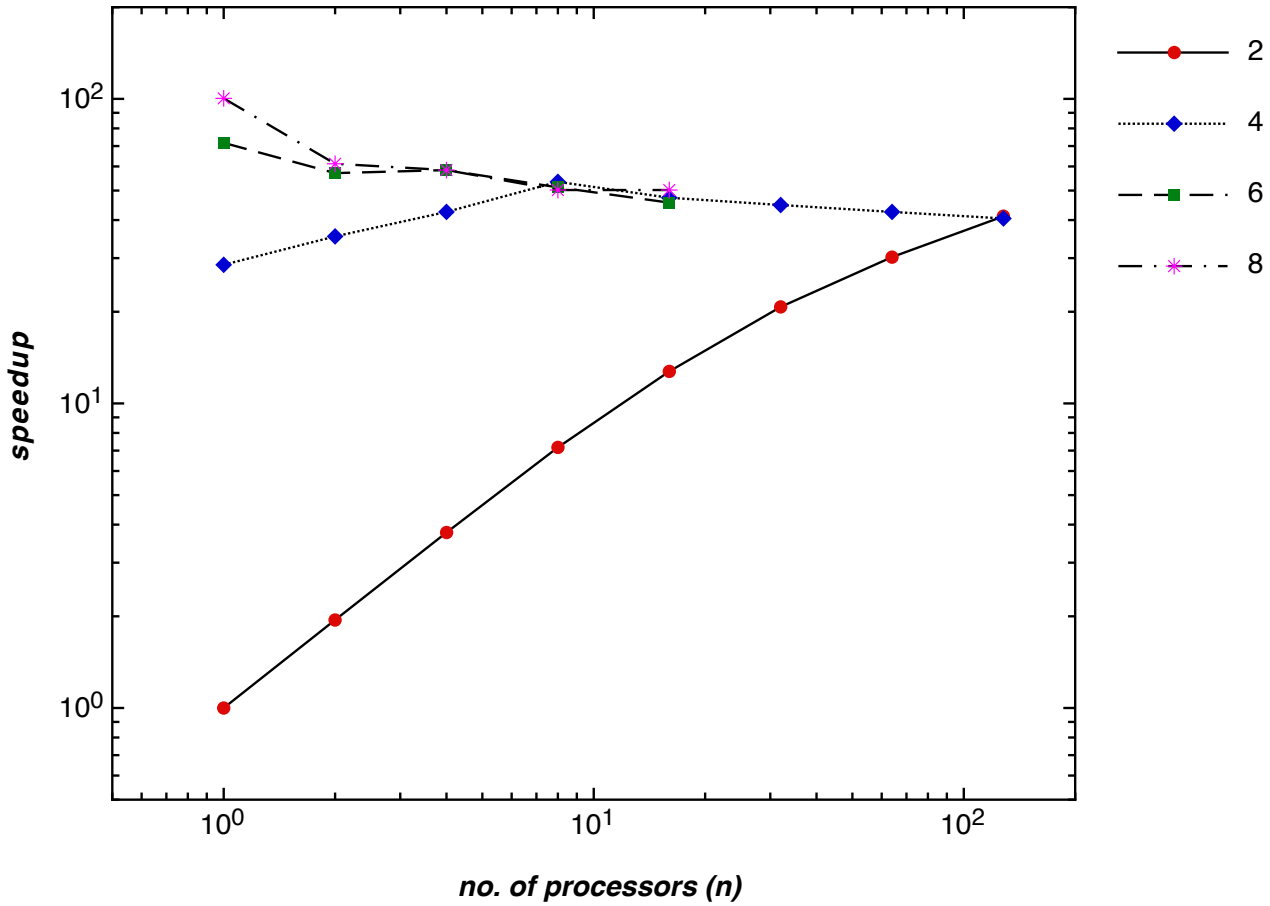


Fig. 3: Speedup as a function of the number of processors for a fixed level of error, for various finite-difference orders. Speedup is defined relative to the second-order code running on a single processor. It is clear that the second-order method scales very well comparatively.

It is also interesting to note the value of n i.e. the number of processors when the different methods begin to compare well in performance. A quick look at Fig. 3 suggests that for the second-order case this occurs in the ballpark of $n = 100$. This value agrees quite well with our estimate from Section 2, wherein we argued that this should happen at $n \sim \frac{N^{1-2/p}}{p} \approx 50^2$. Thus, at least for the case under consideration, it *only takes a hundred processors for the second-order method to achieve comparable performance to the higher-order methods*. This is fairly modest from the perspective of most modern clusters, even those of relatively small size.

In Fig. 4 we show the speedup as a function of the error level for multiple order methods. The *speedup is defined relative to the second-order code running on a single processor, and it is obtained by choosing the value of n for the least wall clock time*. At low accuracy (right side of the graph) one can see that the higher-order methods (4, 6, 8) deliver a significant benefit over the second-order method. However, for high accuracy (left side of the plot) they all deliver *comparable* performance. As argued in the previous section, this is due to the fact that the second-order method exhibits much better parallel scaling behavior throughout (because of the much larger grid sizes required for the same level of discretization error). Note that the second-order method graph starts with a single processor on the extreme-right to nearly 500 on the extreme-left. The eighth-order method is on a single processor throughout (because it actually performs worse with multiple processors, as expected). The sixth and fourth order cases are on a single processor as one goes from right to left decreasing the error level, however, at some point they begin to scale better (due to the rapid increase in N) and that is why one sees the slope "kink" in their speedup graphs. Note that since that improved scaling appears to match the consistent scaling exhibited by the second-order method throughout, we expect

[2]We choose $p = 4$ because the fourth-order method's speedup compares well to the second-order case here.

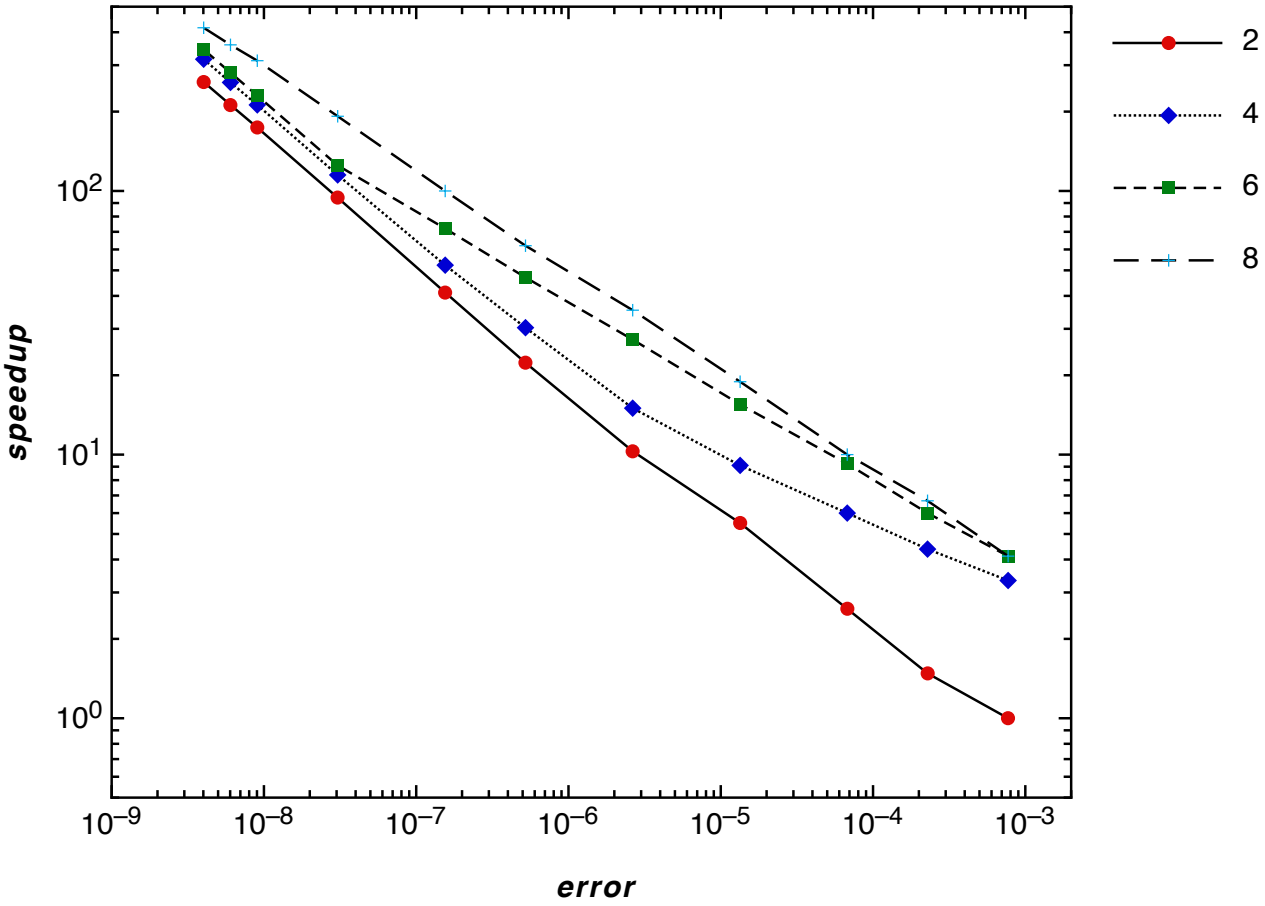


Fig. 4: Best performance (speedup) for various finite-difference orders on a parallel computer cluster. Speedup is defined relative to the second-order code running on a single processor. It is clear that all orders perform comparably for low error levels i.e. on the left side of the graph.

that our main results will hold for error levels even lower than the ones we have tested.

In Fig. 5 we show the same speedup as a function of the method order for multiple error levels. The main point to note again is that *all methods begin to perform comparably for high accuracy computations.*

5. Conclusions

In this work, we have demonstrated that different order finite-difference methods exhibit different scaling behavior on a parallel computer cluster. In general, given a fixed level of accuracy, lower-order methods scale better due to the fact that they require higher resolution and therefore, larger grid sizes. Using a basic example, we have been able to show that *the gain in performance from improved scaling of the second-order method is just enough to have its overall performance match that of a higher-order method.* Of course, the second-order method uses significantly higher computational resources to achieve the same outcome.

Conversely, we have been able to show that a *higher-order method, say, eighth-order, converges so fast that such a method simply does not require any parallel resources at all.* In fact, parallel scaling performance of a higher-order method may be such that one may obtain performance degradation instead of an expected speedup. While we have made our arguments and claims using a simple 1D derivative finite-difference stencil, we expect our main outcomes to hold more generally, including even in 2D and 3D.

To conclude, with the interest of minimizing total execution time and given a sufficiently large computational resource, a "brute force" approach with a lower-order method is likely to perform comparably to a more advanced highly convergent, higher-order method. Since many "real world" science and engineering research codes are written using second-order accurate algorithms, and it is often very challenging to develop algorithms that converge faster, an investment in parallel code development may prove to be quite worthwhile. On the other hand, if one already has a

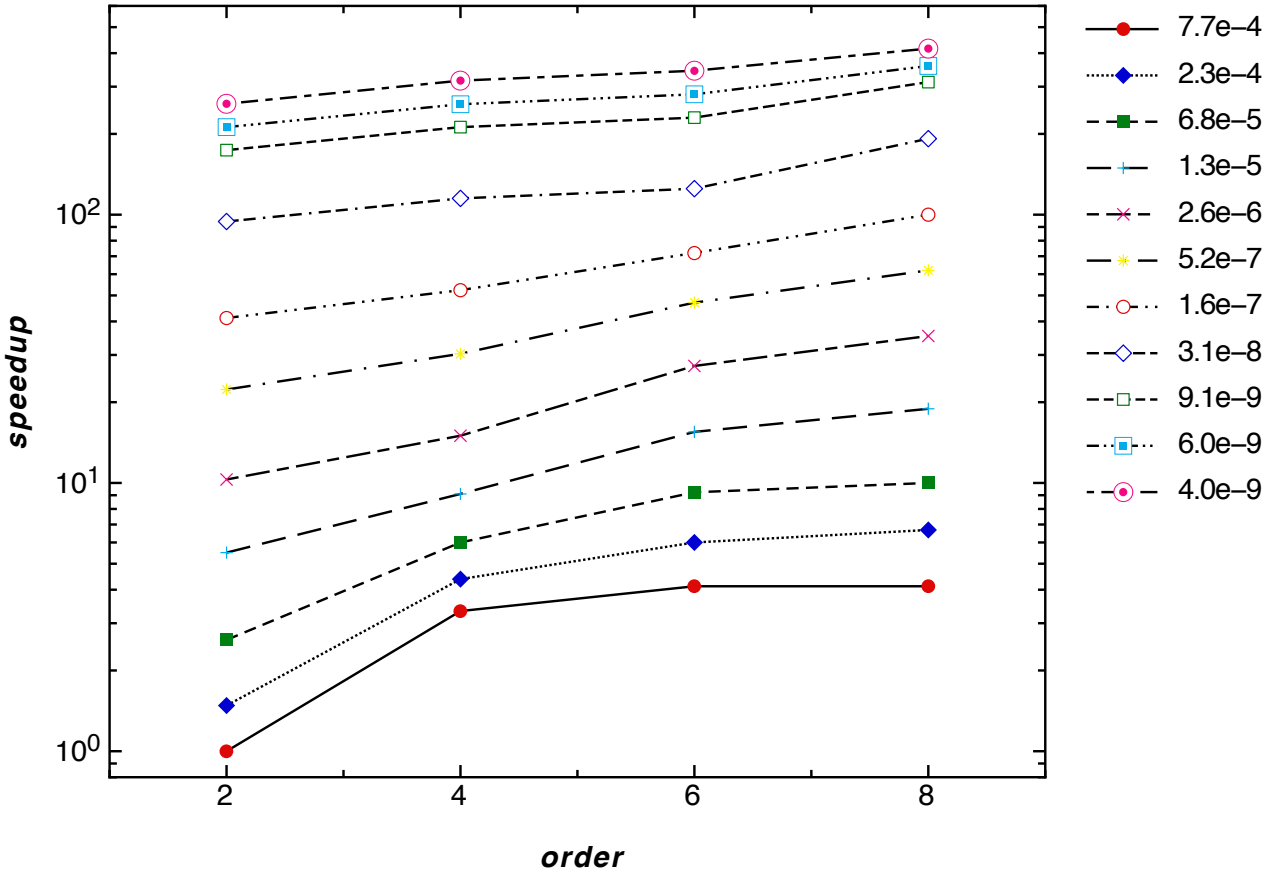


Fig. 5: Same as Fig. 3. Best performance (speedup) for various finite-difference orders on a parallel computer cluster. Speedup is defined relative to the second-order code running on a single processor. It is clear that all orders perform comparably for low error levels.

fast converging serial code then it may be much better to invest in making modest improvements to the convergence rate as opposed to a fully parallel implementation.

6. Acknowledgements

We would like to thank Jay Wang, Alfa Heryudono and Glenn Volkema for feedback on an early draft of this article. G.K. acknowledges research support from NSF Grants No. PHY-1016906, No. CNS-0959382, No. PHY-1135664, and No. PHY-1303724, and from the U.S. Air Force Grant No. FA9550-10-1-0354 and No. 10-RI-CRADA-09.

References

[1] The Top 500 List: http://top500.org/
[2] B. Gustafsson, "High Order Difference Methods for Time Dependent PDE", Springer Series in Computational Mathematics, Volume 38 (2008).
[3] J.S Hesthaven,T. Warburton, " Nodal High-Order Methods on Unstructured Grids: I. Time-Domain Solution of Maxwell's Equations", Journal of Computational Physics, Volume 181, Pages 186-221 (2002).
[4] M. O. Deville, P. F. Fischer, E. H. Mund, "High-Order Methods for Incompressible Fluid Flow", Cambridge University Press (2002).
[5] G. Karniadakis, "High-order splitting methods for the incompressible Navier-Stokes equations", Journal of Computational Physics, Volume 97, Pages 414-443 (1991).
[6] D. Xiu, J. S. Hesthaven, "High-Order Collocation Methods for Differential Equations with Random Inputs", SIAM J. Sci. Comput., 27(3), 1118-1139 (2005).
[7] C-W. Shu, "High-order Finite Difference and Finite Volume WENO Schemes and Discontinuous Galerkin Methods for CFD", International Journal of Computational Fluid Dynamics, Volume 17, Issue 2, Pages 107-118 (2003).
[8] J.T. Beale, " High order accurate vortex methods with explicit velocity kernels", Journal of Computational Physics, Volume 58, Pages 188-208 (1985).
[9] M. R. Visbal, D. V. Gaitonde, "High-Order-Accurate Methods for Complex Unsteady Subsonic Flows", AIAA Journal, Volume 37, No. 10, pp. 1231-1239 (1999).
[10] J. Shen, T. Tang, "Spectral and High-Order Methods with Applications", Mathematics Monograph Series 3 Science Press (2006).
[11] J. Hesthaven, D. Gottlieb, S. Gottlieb, "Spectral Methods for Time-Dependent Problems", Cambridge Monographs on Applied and Computational Mathematics (2007).
[12] OpenMPI website: http://openmpi.org/
[13] http://www.afmc.af.mil/news/story.asp?id=123232827

Unsolved Problems In Computational Science: I

Shanzhen Gao, Keh-Hsun Chen
Department of Computer Science, College of Computing and Informatics
University of North Carolina at Charlotte, Charlotte, NC 28223, USA
Email: sgao3@uncc.edu, chen@uncc.edu

***Abstract*—We will talk about some interesting open problems in computational science. Most of them are new. These problems are related to number theory, geometry theory, combinatorics, graph theory, linear algebra and group theory. They are easy to state and understand although they are very difficult to be solved by researchers in mathematics or computer science. It seems to us that it is very challenging to find suitable mathematical methods or efficient algorithms to deal with them.**

Keywords: Computational number theory, computational geometry, formula, integer sequence, algorithm

I. Introduction

The development of computational science continues in a rapid rhythm, some open problems are made clear and simultaneously new open problems to be solved come out. To solve open problems is a good way to deepen the study in computational science. In the following, we will present some (new) unsolved problems in the fields of: computational number theory, computational geometry, integer matrix enumeration, determinant of matrix, partition, Riordan group, lattice path, tennis ball problem, random k-tupe, cycle in graph, identity. We hope that the readers can solve some of these problems.

II. A Computational Number Theory Problem

Let p be a prime number. We discussed the sets:[1]
$\{\binom{pi}{i+1} \bmod p, i = 1, 2, 3, ...\}$, $\{\binom{pi-1}{i-2} \bmod p$,
$i = 1, 2, 3, ...\}$,and $\{\binom{pi-1}{i} \bmod p, i = 1, 2, 3, ...\}$,
and the following conjecture was given:
Let $p \neq 5$ be a prime, then
$\{2, 4, 6, .., p-3\} \subset \{\binom{p-1}{a_1}\binom{a_1-1}{a_2}\binom{a_2}{a_3}\binom{a_3}{a_4}...\binom{a_{n-1}}{a_n} \bmod p :$ a_i is a positive integer,
$p - 1 \geq a_1,\ a_1 - 1 > a_2 > a_3 > a_4 > ... > a_n\}$.
We can prove that this conjecture is true for $p > 10^{30}$.

We can verify this conjecture is true for some small p, for instance $p < 10^7$.

Problem 1. *How to prove or disprove this conjecture.*

III. A Computational Geometry Problem

A Heron triangle is one with integer sides and integer area. It is clear that the angles of a Heron triangle all have rational sines and cosines. Such are called Heron angles. Conversely, if the angles of a triangle are all Heron angles, then an appropriate magnification yields a Heron triangle.

We study sequences whose consecutive terms determine Heron triangles. We construct arbitrarily long increasing sequences in which every three consecutive terms are the sides of a Heron triangle. By a modified Heron sequence we mean an increasing sequence (u_n) of positive integers such that every three consecutive terms u_{n-2}, u_{n-1} and u_n determine a Heron triangle with sides

$$a_n = u_{n-2} + u_{n-1}, \quad b_n = u_{n-2} + u_n, \qquad c_n = u_{n-1} + u_n,$$

and integer area. While it is not a priori obvious that a Heron sequence may be infinite, we show that, apart from a few exceptions, an infinite modified Heron sequence always results if one begins with two distinct positive integers.

The problem whether there exists an infinite increasing sequence in which every three consecutive terms are the sides of a Heron triangle is still open.[2]

Theorem 2. *Given an integer $n \geq 3$, there is a sequence*

$$a_1,\ a_2,\ \ldots, a_n$$

every three consecutive terms of which are the sides of a Heron triangle.

IV. Enumeration of Integer Matrices

Many famous people have studied the enumeration of integer matrices which are equivalent to some kind of bipartite graphs. We will discuss a few such problems in the following.

A. (0,1)-Matrices with Row Sum s and Column Sum t

Problem 3. *Let m, n, s, t be positive integers such that $sm = tn$. Let $f(m, n, s, t)$ be the number of $m \times n$ matrices over $\{0, 1\}$ with each row summing to s and each column summing to t. How to compute $f(m, n, s, t)$?*

1) Introduction: Equivalently, $f(m, n, s, t)$ is the number of semiregular bipartite graphs with m vertices of degree s and n vertices of degree t. This problem has been the subject of considerable study, and it is unlikely that a simple formula exists. The asymptotic value of $f(m, n, s, t)$ has been much studied but the results are incomplete. Historically, the first significant result was that of Read [3], who obtained the asymptotic behavior for $s = t = 3$. McKay and Wang (2003) solved the sparse case $\lambda(1 - \lambda) = o((mn)^{-1/2})$ using combinatorial methods [4]. Canfield and McKay used analytic methods to solve the problem for two additional ranges. In one range the matrix is relatively square and the density is not too close to 0 or 1. In the other range, the matrix is far from square and the density is arbitrary. Interestingly, the asymptotic value of $f(m, n, s, t)$ can be expressed by the same formula in all cases where it is known. Based on computation of the exact

values for all m; n 30, they got the conjecture that the same formula holds whenever $m+n \to \infty$ regardless of the density (they defined the density $\lambda = s/m = t/m$). [5]

The number $f(m,n,s,t)$ in question can be related in various ways to the representation theory of the symmetric group or of the complex general linear group, but this does not make their computation any easier. The case $s = t = 2$ is solved by Anand, Dumir, and Gupta [6] . A formula for the case $s = t = 3$ appears in L. Comtet's Advanced Combinatorics (1974) [7], without proof. There are more such closed formulas of $f(m,n,s,t)$ in [8], [9], [10].

In some row, let $x_{i_1}x_{i_2}\cdots x_{i_k}$denote the $i_1 - th$ column, the $i_2 - th$ column, $\cdots$, the $i_k - th$ column entries are 1 in some row and other entries are all 0, where $i_1, i_2, \cdots, i_k \in \{1, 2, \cdots, n\}$.

Example: Let $m = n = 4, s = t = 3$, then $x_1x_2x_3|x_1x_2x_4|x_1x_3x_4|x_2x_3x_4$ denotes the matrix as follows:

$$\begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \end{bmatrix}$$

Obviously, $f(m,n,s,t)$ equals the coefficient of $x_1^t x_2^t \cdots x_n^t$ in the symmetric polynomial

$$\left(\sum_{i_1 < i_2 < \cdots < i_s} x_{i_1} x_{i_2} \cdots x_{i_s} \right)^m$$

where $i_1, i_2, \cdots, i_s \in \{1, 2, \cdots, n\}$, and the sum is over all the possible of $s-$ combinations from $\{1, 2, \cdots, n\}$ with $i_1 < i_2 < \cdots < i_s$. It is easy to get,

$$f(m,n,s,t) = f(n,m,t,s)$$
$$f(m,n,s,t) = f(n,m,n-s,m-t)$$
$$f(m,n,1,t) = \frac{m!}{(t!)^n}$$
$$f(m,n,s,1) = \frac{n!}{(s!)^m}$$

2) Some Formulas:

$$f(n,n,2,2) = 4^{-n} \sum_{i=0}^{n} \frac{(-2)^i (n!)^2 (2n-2i)!}{i!((n-i)!)^2}$$

Example:

We get a sequence for $f(n,n,2,2)$:

0, 1, 6, 90, 2040, 67950, 3110940, 187530840, 14398171200, 1371785398200, ...

$f(m,n,2,3) = 2^{-m} \sum_{i=0}^{n} \frac{(-1)^i m!n!(2m-2i)!}{i!(m-i)!(n-i)!6^{n-i}}$

Example:

$f(9,6,2,3) = 90\,291\,600$

$f(300,200,2,3) = 5.341\,4 \times 10^{1161}$

$f(m,n,4,2) = 24^{-m} \sum_{\alpha=0}^{m} \sum_{\beta=0}^{m-\alpha} \frac{(-1)^{(m-\alpha-\beta)} 3^\alpha 6^{(m-\alpha-\beta)} m!n!(4\beta+2(m-\alpha-\beta))!}{\alpha!\beta!(m-\alpha-\beta)!(2\beta+(m-\alpha-\beta))!2^{2\beta+(m-\alpha-\beta)}}$

Example:

$f(10,20,4,2) = 2532\,230\,252\,503\,738\,514\,963\,235\,000$

$f(50,100,4,2) = 1.334\,5 \times 10^{275}$

$f(m,n,4,3) = 24^{-m} \sum_{\beta=0}^{\min\{\lfloor n/2 \rfloor, m\}} \sum_{\gamma=0}^{\min\{m-\beta, n-2\beta\}} \sum_{\delta=0}^{\min\{m-\gamma-\beta, n-2\beta-\gamma\}} \frac{(-1)^\gamma 3^\beta 6^\gamma 8^\delta m!n!(4m-4\beta-2\gamma-3\delta)!}{(m-\beta-\gamma-\delta)!\beta!\gamma!\delta!(n-2\beta-\gamma-\delta)!6^{(n-2\beta-\gamma-\delta)}}$

Example:

$f(9,12,4,3) = 2407\,147\,216\,735\,338\,000$

$f(30,40,4,3) = 9.010\,0 \times 10^{124}$

3) Algorithm Description For f(m, n, s, t): The algorithm used to verify the equations presented counts all possible matrices, but does not construct them.

It is a bit involved, so it is best described with an example.

Suppose we wanted to compute the number of 4x6 matrices over nonnegative integers with row sum 12 and column 8. We first create a list of all nonincreasing partitions of 12: 12, 11 1, 10 2, 10 1 1, 9 3, etc., and store this in memory. We make sure that each partition stored is not of length greater than the number of columns of the matrix. We then create a state vector of length 6 filled with 8s:

#(8 8 8 8 8 8)

This state vector symbolizes the sum of integers we must place in each column, and each time the state changes, it is sorted in nondecreasing order.

An additional vector, called the cap vector, is created when we deal with a new state. It records the length of the contiguous blocks of numbers found in the state. Here, it is

#(6).

Next, we iterate over each of the (valid) partitions of 12 that we could possibly use for the choice of the first row of the matrix. Here, our first partition is 8 4. We then create a partition block (pb) vector, which is exactly a "cap vector" of the partition, instead of the state. Here, it is

#(1 1).

Finally, we create all the assignment vectors that are valid for this partition and this cap vector. An assignment vector dictates where the indicated element of the partition will be placed in the row. Assignment vectors always have the same length as the partition we are planning to use. The entries of the assignment vector refer to the (zero-based) indices of the cap vector. Since the cap vector in this case only has one index (namely, 0) and both 8 and 4 can be elements in the matrix row, we assign 8 and 4 to the 0th index:

#(0 0)

In other words, both the 8 and the 4 will appear in block 0 of the state. Now, there are (6 take 1)*(5 take 1) ways of placing the 8 and 4, so we note that when we drop the state vector. We pretend that the first row of the matrix will be (8 4 0 0 0 0), and so, dropping the state vector, the remaining three rows must sum to

#(0 4 8 8 8 8)

and we record that the number of ways of obtaining a matrix of state #(8 8 8 8 8 8) is 30 times the number of ways we can obtain a matrix of state #(0 4 8 8 8 8).

Of course, we must add to our count the other ways to assign the 8 and 4. Since there are no other ways, no more assignment vectors can be constructed. We then add to our count the ways in which we can use the partition 8 3 1 (with all applicable assignment vectors), and then 8 2 2 (with all applicable assignment vectors), and so forth.

To get a better feel for how the assignment vectors are created, let's say that, in the middle of our counting, we achieve the state

#(1 1 4 6 6 6)

with two rows left to fill. Our cap vector is then

#(2 1 3)

and suppose we are considering the partition 4 4 3 1. Its pb is #(2 1 1). Since the cap vector has length 3, the indices for it are 0, 1, and 2, so the entries of each assignment vector can be comprised only of 0, 1, and/or 2.

To create the first assignment vector, we note that the first element of the partition, 4, cannot be placed in block 0 of the state (the block of two 1s), since 4 > 1. A single 4 can be placed in block 1 of the state (the block consisting of the single 4), so the first 4 in the partition can be assigned to block 1:

#(1 ? ? ?)

But block 1 is only length 1 (as noted by the cap vector's entry of 1 at index 1), so no more 4s can go in that block. The second 4 in the partition can also be placed in block 2 of the state (the block of three 6s), since 4 <= 6. Thus, our assignment vector changes to

#(1 2 ? ?).

Next in the partition, we have a 3, which is also greater than 1, so it too cannot go into block 0. Block 1 has already been taken by the 4. Hence the only remaining place for it is in block 2:

#(1 2 2 ?)

Finally, the last element of the partition is a 1, which can go anywhere in the state. We begin by assigning it to block 0, giving the resulting assignment vector as

#(1 2 2 0).

How many ways could these assignments be carried out? The first 4 has only one way. The second 4 and the 3 are both in block 2, but they are different numbers, so they can be inserted in (3 take 1)*(2 take 1) ways. Finally, the 1 has (2 take 1) ways to be inserted into block 0. Hence we multiply to get 12 ways for this assignment vector, and dropping the state, we get #(0 1 0 2 3 6). Sorting it, it becomes #(0 0 1 2 3 6), which we will process after we deal with the remaining assignment vectors possible for 4 4 3 1.

To get the next assignment vector, we note that we can keep everything the same, but the 1 in the partition can be put in block 2. This gives

#(1 2 2 2)

and to compute the number of ways, we have 1*(3 take 1)*(2 take 1)*(1 take 1) = 6.

To get the next assignment vector, we note we've exhausted all possibilities for #(1 2 ? ?), so we then find the 'next' way to assign the two 4s in the partition. The only remaining option is to put them both in block 2, so we start with

#(2 2 ? ?).

Now, the 3 can go in block 1 and the 1 can go in block 0, giving

#(2 2 1 0)

and total number of ways (3 take 2)*(1 take 1)*(2 take 1) = 6.

Now, we think of a "block" of the assignment vector as the entries that correspond to an equal number in the partition; here, the first two entries correspond to the partition entry 4, so they form a block. The pb tells us the length of each block of the assignment vector. For example, recall that here, pb is #(2 1 1), so each assignment vector corresponding to this partition has three blocks, the first of which has length two, and the remaining two have length one. We construct assignment vectors that are nondecreasing in each block, though we can have a decrease when we move to a new block from an old one. The remaining three assignment vectors and the number of ways to make the assignment are then

#(2 2 1 2) with ways (3 take 2)*(1 take 1)*(1 take 1) = 6
#(2 2 2 0) with ways (3 take 2)*(1 take 1)*(2 take 1) = 12
#(2 2 2 1) with ways (3 take 2)*(1 take 1)*(1 take 1) = 6.

Let's consider a larger example. Suppose the state was

#(0 1 1 1 1 2 2 2 3 3 3 3 4 5 5)

with row sum 18. This state will produce a cap vector of #(4 3 4 1 2) (since zeroes in the state are ignored). Let's suppose we were considering the partition

3 3 3 2 2 2 1 1 1,

which gives a pb of #(3 3 3). There are 433 total assignment vectors for this partition. The first one we could construct is

#(2 2 2 1 1 1 0 0 0) with ways (4 take 3)*(3 take 3)*(4 take 3) = 16,

an intermediate one we could construct is

#(2 3 4 1 1 2 0 1 2) with ways (4 take 1)*(1 take 1)*(2 take 1) for placing the three 3s
(3 take 2)(3 take 1) for placing the three 2s
(4 take 1)(1 take 1)*(2 take 1) for placing the three 1s (total 576),

and the last one we could construct is

#(3 4 4 2 2 2 1 1 2) with ways (1 take 1)*(2 take 2) for placing the three 3s
*(4 take 3) for placing the three 2s
(3 take 2)(1 take 1) for placing the three 1s (total 12).

Notice that each block of each assignment vector has its entries in nondecreasing order, but often there is a decrease when we move from block to block. Since the state vectors are nondecreasing, this is to be expected.

In general, for each state vector that is achieved, this algorithm will iterate over all assignment vectors for each valid partition, multiplying cofactors and adding the results. When fitting the last row, though, the calculation is surprisingly easy: continuing the example we had above, if we examine the state #(0 0 1 2 3 6), we see that there is only one possible partition

of 12 that fits it (namely 6 3 2 1) and there is only one way to fit it in. Hence, there is only one way to achieve this state. The situation is the same for every state with one row left to be filled.

For further speedup, a fast storage object must be used, so that if a given state is seen again, we can recall from memory how many partially-filled matrices can produce it. This speedup is necessary, for without it, the algorithm will take too long. Other approaches are certainly possible.

We have a algorithm as described in the above, and our program can compute $f(m,n,s,t)$ for $m,n \leq 20$. It is very hard to obtain a formula when $s,t \geq 4$.

B. No 1 Stands on the Main Diagonal

Problem 4. *Let $f_s(n)$ be the number of $(0,1)$ - matrices of size $n \times n$ such that each row has exactly s 1's and each column has exactly s 1's and with the restriction that no 1 stands on the main diagonal. How to compute $f_s(n)$?*

1) Reformulations:

a) Reformulation One: There are $s \times n$ balls with s balls labelled A_i, $i = 1,2,...,n$. Distribute these $s \times n$ balls into n distinct boxes numbered $1,2,\cdots,n$, such that each box contains s different balls, and the i-th box i does not contain A_i, $i = 1,2,...,n$. How many distributions are there?

b) Reformulation Two: There are $s \times n$ letters, each letter A_i appears exactly s times($i = 1,2,...,n$). Let these $s \times n$ letters be arranged in a row according to:

(from the left to the right on the row, we define the first location, the second location, ... , the $s \times$ n -th location)

There is only one letter in each location.

There are no two equal letters A_i $(i = 1,2,\ldots,n)$ in any two of the following s locations: the $(sk+1)$-th location, $(sk+2)$-th location, ….....,the $(sk+s)$-th location $(k = 0,1,2,\ldots,s-1)$.

If A_{i_1}, A_{i_2}, ... , A_{i_s} are in the following s locations respectively: the $(sk+1)$-th location, $(sk+2)$-th location, …..... ,the $(sk+s)$-th location $(k = 0,1,2,\ldots,s-1)$, then $i_1 < i_2 < ... < i_s$.

A_i $(i = 1,2,\ldots,n)$ is not in any of the $(sk+1)$-th location, $(sk+2)$-th location, …..... ,the $(sk+s)$-th location $(k = 0,1,2,\ldots,s-1)$.

How many arrangements are there?

c) Reformulation Three: $f(n)$ is equal to the number of labeled $s-$regular bipartite simple graphs on $2n$ vertices with the vertex set $V = V_1 \cup V_2$, $V_1 \cap V_2 = \emptyset$, $V_1 = \{u_1,u_2,..,u_n\}$, $V_2 = \{v_1,v_2,..,v_n\}$, and no edge between u_i and v_i, $i = 1,2,..n$.

2) Formulas:

$$f_1(n) = \sum_{k=0}^{n} (-1)^k \frac{n!}{k!}$$

Example:

We get a sequence from $f_1(n)$:

0, 1, 2, 9, 44, 265, 1854, 14833, 133496, 1334961,...

$f_1(n)$ also counts the number of derangements of permutation of n elements with no fixed points.

$$f_2(n) = \sum_{k=0}^{n} \sum_{s=0}^{k} \sum_{j=0}^{n-k} \frac{(-1)^{k+j-s} n!(n-k)!(2n-k-2j-s)!}{s!(k-s)!((n-k-j)!)^2 j! 2^{2n-2k-j}}$$

Example:

We get a sequence 0, 0, 1, 9, 216, 7570, 357435, 22040361,...

V. Determinant of Matrix

Problem 5. *Let $A_{n\times n} = (a_{i,j})_{n\times n}$ be a matrix with $a_{i,j} = 1$ if $i \neq j$ and $i \neq j+1$;and $a_{i,i} = 1+a_i$, $a_{i+1,i} = 1+b_i$.How to compute the determinant of $A_{n\times n}$, $\det A_{n\times n}$? And how to represent it?*

Example:

$$\det \begin{bmatrix} 1+a_1 & 1 \\ 1+b_1 & 1+a_2 \end{bmatrix} = a_1 + a_2 - b_1 + a_1a_2$$

$$\det \begin{bmatrix} 1+a_1 & 1 & 1 \\ 1+b_1 & 1+a_2 & 1 \\ 1 & 1+b_2 & 1+a_3 \end{bmatrix} = a_1a_2 + a_1a_3 - a_1b_2 + a_2a_3 - a_3b_1 + b_1b_2 + a_1a_2a_3$$

$$\det \begin{bmatrix} 1+a_1 & 1 & 1 & 1 \\ 1+b_1 & 1+a_2 & 1 & 1 \\ 1 & 1+b_2 & 1+a_3 & 1 \\ 1 & 1 & 1+b_3 & 1+a_4 \end{bmatrix} = a_1a_2a_3 + a_1a_2a_4 - a_1a_2b_3 + a_1a_3a_4 - a_1a_4b_2 + a_2a_3a_4 + a_1b_2b_3 - a_3b_1a_4 + b_1a_4b_2 - b_1b_2b_3 + a_1a_2a_3a_4$$

$$\det \begin{bmatrix} 1+a_1 & 1 & 1 & 1 & 1 \\ 1+b_1 & 1+a_2 & 1 & 1 & 1 \\ 1 & 1+b_2 & 1+a_3 & 1 & 1 \\ 1 & 1 & 1+b_3 & 1+a_4 & 1 \\ 1 & 1 & 1 & 1+b_4 & 1+a_5 \end{bmatrix} = a_1a_2a_3a_4 + a_1a_2a_3a_5 - a_1a_2a_3b_4 + a_1a_2a_4a_5 - a_1a_2a_5b_3 + a_1a_3a_4a_5 + a_1a_2b_3b_4 - a_1a_4b_2a_5 + a_2a_3a_4a_5 + a_1b_2a_5b_3 - a_3b_1a_4a_5 - a_1b_2b_3b_4 + b_1a_4b_2a_5 - b_1b_2a_5b_3 + b_1b_2b_3b_4 + a_1a_2a_3a_4a_5$$

Problem 6. *Let $A_{n\times n} = (a_{i,j})_{n\times n}$ be a matrix with $a_{i,j} = 1$ if $i \neq j$,$i \neq j+1$ and $i+1 \neq j$;and $a_{i,i} = 1+a_i$, $a_{i+1,i} = 1+b_i$ and $a_{i,i+1} = 1+c_i$.Compute the determinant of $A_{n\times n}$, $\det A_{n\times n}$. And how to represent it?*

Example:

$$\det \begin{bmatrix} 1+a_1 & 1+c_1 \\ 1+b_1 & 1+a_2 \end{bmatrix} = a_1 + a_2 - b_1 - c_1 + a_1a_2 - b_1c_1$$

$$\det \begin{bmatrix} 1+a_1 & 1+c_1 & 1 \\ 1+b_1 & 1+a_2 & 1+c_2 \\ 1 & 1+b_2 & 1+a_3 \end{bmatrix} = a_1a_2 + a_1a_3 - a_1b_2 + a_2a_3 - a_3b_1 - a_1c_2 + b_1b_2 - a_3c_1 - b_1c_1 - b_2c_2 + c_1c_2 + a_1a_2a_3 - a_1b_2c_2 - a_3b_1c_1$$

$$\det \begin{bmatrix} 1+a_1 & 1+c_1 & 1 & 1 \\ 1+b_1 & 1+a_2 & 1+c_2 & 1 \\ 1 & 1+b_2 & 1+a_3 & 1+c_3 \\ 1 & 1 & 1+b_3 & 1+a_4 \end{bmatrix}$$

$= c_1c_2a_4 - a_1a_2c_3b_3 - a_1b_2c_2a_4 + b_1c_1c_3b_3 - c_1c_2c_3 - a_1c_3b_3 - a_1a_2c_3 - a_1c_2a_4 - c_1a_3a_4 + c_1c_3b_3 - b_2c_2a_4 - a_2c_3b_3 + a_4b_2b_1 - a_4a_3b_1 - a_1b_2c_2 + a_1c_2c_3 - b_1c_1a_4 - b_1c_1a_3 + b_1c_1b_3 + b_1c_1c_3 - b_1c_1a_3a_4 + b_1c_3b_3 - a_4b_2a_1 + a_1a_2a_4 + a_1a_3a_4 + a_1b_2b_3 - b_1b_2b_3 + a_1a_2a_3 - a_1a_2b_3 + a_1a_2a_3a_4 + a_2a_3a_4$

$$\det\begin{bmatrix} 1+a_1 & 1+c_1 & 1 & 1 & 1 \\ 1+b_1 & 1+a_2 & 1+c_2 & 1 & 1 \\ 1 & 1+b_2 & 1+a_3 & 1+c_3 & 1 \\ 1 & 1 & 1+b_3 & 1+a_4 & 1+c_4 \\ 1 & 1 & 1 & 1+b_4 & 1+a_5 \end{bmatrix}$$

$= -a_1a_2c_3b_3 - a_1b_2c_2a_4 + b_1c_1c_3b_3 + b_4b_2b_3b_1 - b_1c_1a_3a_4 - a_2a_3c_4b_4 - a_2b_3c_3a_5 - b_2c_2a_4a_5 + b_2c_2c_4b_4 - a_5b_2b_3b_1 + c_1c_2c_3c_4 + a_5b_2b_3a_1 - c_1c_2c_4b_4 - c_1c_2c_3a_5 + c_1c_2a_4a_5 - c_1a_3a_4a_5 + c_1b_3c_3a_5 + c_1a_3c_4b_4 - b_4b_2b_3a_1 + b_1c_1a_3c_4 - b_1b_2c_4b_4 + b_1c_1b_3c_3a_5 + b_1c_1a_3c_4b_4 - b_1c_1a_3a_5 + b_1c_1a_3b_4 - b_1c_1b_4b_3 - b_1c_1c_3c_4 + b_1c_1a_5b_3 - b_1c_1a_4a_5 + b_1c_1c_4b_4 - b_1c_1a_3a_4a_5 + b_1a_3c_4b_4 + b_1b_3c_3a_5 + b_1c_1c_3a_5 - a_1a_5b_3a_2 + a_1a_5a_3a_2 - a_1a_3c_4b_4 - a_1b_3c_3a_5 + a_1b_2c_4b_4 + a_1c_2c_3a_5 - a_1c_2a_4a_5 + a_1c_2c_4b_4 - a_1a_2b_3c_3a_5 - a_1b_2c_2a_4a_5 + a_1b_2c_2c_4b_4 - a_1a_2a_3c_4b_4 - a_1a_2c_3a_5 + a_1a_2c_3c_4 + a_1b_2c_2b_4 + a_1b_2c_2c_4 - a_1b_4a_3a_2 + a_1b_4b_3a_2 + a_1a_2a_4a_5 - a_1a_2c_4b_4 - a_1a_2a_3c_4 - a_1c_2c_3c_4 - a_1b_2c_2a_5 - b_2a_5a_4a_1 + a_5a_1a_3a_4 + a_5a_2a_3a_4 + b_2a_5a_4b_1 - a_5a_4a_3b_1 + a_1a_2a_3a_4 + a_1a_2a_3a_4a_5$

VI. Partition Problem

$s_1 < s_2 < ... < s_n$

$s_i \in [1, m+n]$ (i.e. $s_i \in \{1, 2, 3, ..., m+n\}$)

$\omega = \sum_{i=1}^{n} s_i$

$s(n, m, w) = \#$ of ways to sum n ordered members of $[1, m+n]$ to give ω

$= \#$ of partitions of ω into n distinct parts, each part$\leq m+n$

$$s(1, m, w) = \begin{matrix} 1, if\ 1 \leq \omega \leq m+1 \\ 0, else \end{matrix}$$

$$s(n, 0, w) = \begin{matrix} 1, if\ \omega = \binom{n+1}{2} \\ 0, else \end{matrix}$$

$$s(n, 1, w) = \begin{matrix} 1, if\ \binom{n+1}{2} \leq \omega \leq \binom{n+2}{2} - 1 \\ 0, else \end{matrix}$$

$s(n, m, w) = s(n-1, m, w-(m+n)) + s(n, m-1, w)$

$s(2, 2, 4) = s(2-1, 2, 4-(2+2)) + s(2, 2-1, 4) = 1$

$s(2, 2, 5) = s(1, 2, 5-(4)) + s(2, 2-1, 5) = 2$

$s(n, m, w) = s(n-1, m, w-n) + s(n, m-1, w-n)$

$s(2, 2, 4) = s(1, 2, 2) + s(2, 1, 4-2) = 1$

$s(2, 2, 5) = s(1, 2, 3) + s(2, 1, 5-2) = 2$

$s(n, 2, w) = s(n-1, 2, w-n) + s(n, 1, w-n)$

$= s(n-2, 2, w-n-1) + s(n-1, 1, w-n-1) + s(n, 1, w-n)$

$= s(n-3, 2, w-n-2) + s(n-2, 1, w-n-2) + s(n-1, 1, w-n-1) + s(n, 1, w-n)$

$s(2, m, w) = s(1, m, w-2) + s(2, m-1, w-2)$

$= s(n-1, 2, w-n) + s(n, 1, w-n)$

We want to know something more about $s(n, m, w)$.

VII. Riordan Group/Matrix

Consider an infinite matrix [11], $U = (m_{ij})_{,i,j \geq 0}$, with entries in $\mathbb{C}$ ($\mathbb{C}$ the complex numbers).

Let $C_i(x) = \sum_{n \geq 0}^{\infty} m_{n,i} x^n$ be the generating function of the $i-$th column of U. We assume that

$$C_i(x) = g(x)[f(x)]^i$$

where

$$g(x) = 1 + g_1 x + g_2 x^2 + g_3 x^3 + ...$$
$$f(x) = x + f_1 x + f_2 x^2 + f_3 x^3 + ...$$

In this case we write $U = (g(x), f(x))$ and say that $(g(x), f(x))$ is a Riordan matrix.

Example:

$$P = \left(\frac{1}{1-z}, \frac{z}{1-z}\right) = \begin{matrix} 1 & 0 & & & & & . \\ 1 & 1 & 0 & & & & . \\ 1 & 2 & 1 & 0 & & & . \\ 1 & 3 & 3 & 1 & 0 & & . \\ 1 & 4 & 6 & 4 & 1 & 0 & . \\ . & . & . & . & . & . & . \end{matrix}$$

$$\frac{1}{1-z} = 1 + z + z^2 + z^3 + z^4 + z^5 + z^6 + z^7 + ...$$

$$\frac{1}{1-z}\frac{z}{1-z} = z + 2z^2 + 3z^3 + 4z^4 + 5z^5 + ...$$

$$\frac{1}{1-z}\left(\frac{z}{1-z}\right)^2 = z^2 + 3z^3 + 6z^4 + 10z^5 + ...$$

$$\frac{1}{1-z}\left(\frac{z}{1-z}\right)^3 = z^3 + 4z^4 + 10z^5 + 20z^6 + ...$$

Now multiply $U = (g(x), f(x))$ on the right by a column vector

$(a_0, a_1, a_2, ...)^T$

and note that the resulting column vector

$(b_0, b_1, b_2, ...)^T$

has the generating function

$B(x) = a_0 C_0(x) + a_1 C_1(x) + a_2 C_2(x) + ...$

$= a_0 g(x) + a_1 g(x) f(x) + a_2 g(x)[f(x)]^2 + ...$

$= g(x)[a_0 + a_1 f(x) + a_2 [f(x)]^2 + ...]$

$= g(x) A(f(x))$.

Another way to denote this is by

$(g(x), f(x)) * A(x) = g(x) A(f(x)) = B(x)$.

What happens when two Riordan matrices are multiplied?

The typical column of

$(h(x), l(x))$ is $h(x)[l(x))]^i$.

$(g(x), f(x)) * (h(x), l(x)) = (g(x) h(f(x)), l(f(x)))$.

This is a group multiplication with identity $I = (1, x)$ and group inverse

$(g(x), f(x))^{-1} := \left(\frac{1}{g(\bar{f}(x))}, \bar{f}(x)\right)$,

where $\bar{f}$ is the compositional inverse of f,

i.e. $f(\bar{f}(x)) = \bar{f}(f(x)) = x$.

The existence of a unique compositional inverse in $\mathbb{C}[[x]]$ is guaranteed by
$f(x) = x + f_1x + f_2x^2 + f_3x^3 + ...$
Then we get the Riordan group.
The *Riordan group* denoted $(\mathcal{R}, *)$. An element $R \in \mathcal{R}$ is an infinite lower triangular array whose k-th column has generating function $g(z)f^k(z)$.

To see that the members of the group with the form $(xf\prime(x)/f(x), f(x))$ belong to a subgroup denoted by H, note the following:

(i) The identity
$(1, x) = (x(x)\prime/x, x)$
(ii) The product
$(xf\prime(x)/f(x), f(x)) * (xh\prime(x)/h(x), h(x))$
$= (\frac{xf\prime(x)}{f(x)} \frac{f(x)h\prime(f(x))}{h\prime(f(x))}, h(f(x)))$
$= (\frac{x((h(f(x)))\prime}{h(f(x))}, h(f(x))) \in H.$
(iii) The inverse of $(xf\prime(x)/f(x), f(x))$ is
$(\frac{1}{\bar{f}(x)\frac{f\prime(\bar{f}(x))}{f(\bar{f}(x))}}, \bar{f}(x)) \in H$

Problems for further considering on Riordan group: the subgroup properties, the normal subgroup properties.

VIII. A Lattice Path Problem

$\uparrow^{\geq k}$: k or more than k consecutive $\uparrow$ steps
$\uparrow^{=k}$: k consecutive $\uparrow$ steps
avoiding $\uparrow^{\geq k}$: no k or more than k consecutive $\uparrow$ steps

Problem 7. *How to find the number, say $f(m, n)$, of lattice path from $(0, 0)$ to (m, n) avoiding $\uparrow^{\geq 3}$, $\longrightarrow^{\geq 3}$, and weakly above $y = x$?*

You can see some numbers of this problem:

$n = 12$						1
$n = 11$						6
$n = 10$					1	20
$n = 9$					5	44
$n = 8$				1	14	67
$n = 7$				4	25	70
$n = 6$			1	9	29	45
$n = 5$			3	12	20	17
$n = 4$		1	5	9	8	
$n = 3$		2	4	4		
$n = 2$	1	2	2			
$n = 1$	1	1				
$n = 0$						
	$m = 0$	$m = 1$	$m = 2$	$m = 3$	$m = 4$	$m = 5$

For $m + 2 \leq n$,

$$f(m, n) = f(m-1, n) + f(m-1, n-1) + f(m-1, n-2) - f(m-3, n) - f(m-3, n-1) - f(m-3, n-2).$$

For $m + 1 = n$,

$$f(n, n+1) = f(n-1, n+1) + f(n-1, n) - f(n-3, n+1) - f(n-3, n).$$

For $m = n$,

$$f(n, n) = f(n-1, n) - f(n-3, n).$$

Example:

$$\begin{aligned} f(9, 12) &= f(8, 12) + f(8, 11) + f(8, 10) \\ &\quad - f(6, 12) - f(6, 11) - f(6, 10) \\ &= 1023 + 1156 + 956 - 27 - 70 - 130 \\ &= 2098 \end{aligned}$$

$$\begin{aligned} f(6, 7) &= f(5, 7) + f(5, 6) - f(3, 7) - f(3, 6) \\ &= 70 + 45 - 4 - 9 = 103 \end{aligned}$$

$$\begin{aligned} f(5, 5) &= f(4, 5) - f(2, 5) \\ &= 20 - 3 = 17 \end{aligned}$$

(1) We can say the numbers on the first line with slope 2 are:

$$1, 1, 1, 1, 1, 1, 1, 1, ...$$

(2) The numbers on the second line with slope 2 are:

$$1, 2, 3, 4, 5, 6, 7, ..$$

(3) The numbers on the third line with slope 2 are:

$$0, 2, 5, 9, 14, 20, 27, 35, 44, 54, ...$$

$$\binom{n+1}{2} - 1, \text{ for } n \geq 2$$

(4) The numbers on the fourth line with slope 2 are:

$$0, 1, 4, 12, 25, 44, 70, 104, 147, 200...$$

$$f(n) = \frac{1}{6}n(n+5)(n-2), \text{ for } n \geq 3$$

$f(n+3)$ is the number in the sequence [12] which counts other things:

Some kind of partitions by P. Erdos, R. K. Guy and J. W. Moon [13].

Rooted trees by J. Riordan [14].

(5) The numbers on the fifth line with slope 2 are:

$$0, 0, 2, 9, 29, 67, 130, 226, 364, 554, 807, ...$$

$$\frac{1}{24}(n-2)(n^3 + 8n^2 - 9n - 48), \text{ for } n \geq 4$$

(6) The numbers on the sixth line with slope 2 are:

$$0, 0, 0, 4, 20, 70, 176, 370, 693, ...$$

$$\frac{1}{120}(n-2)\left(n^4 + 12n^3 - 21n^2 - 232n + 360\right), \text{ for } n \geq 5$$

(7) The numbers on the seventh line with slope 2 are:

$$0, 0, 0, 0, 8, 45, 168, 454, 1023, 2045, 3751, ...$$

$$\frac{1}{720}n\left(n^5 + 15n^4 - 65n^3 - 675n^2 + 3304n - 3300\right), \text{ for } n \geq 6$$

IX. A Tennis Ball Problem

The tennis ball problem can be stated as follows: Given integers r, s, n, with $0 < r < s$, label sn balls $1, 2, \ldots, sn$. Place the first s balls, labeled $1, 2, \ldots, s$ into a bin, then remove r balls from the bin. Repeat this process n times, each time inserting the next s balls in sequence and removing r balls. The question we seek to answer is, "How many different sets of rn balls lie outside the bin after n turns?"

The tennis ball problem can be viewed as a lattice path enumeration. Consider lattice walks in the plane with East $\langle 1, 0\rangle$ and North $\langle 0, 1\rangle$ steps. We count the number of paths from $(0, 1)$ to $((s-r)n+1, rn+1)$ that stay weakly above the boundary $(E^{s-r}N^r)^n$. It is easy to see that, for each $1 \leq i \leq n$, at least ri of the first si steps must be N steps, and for $i = n$ this is an equality. We associate with each of the N steps one of the labeled balls, namely the ball with the label matching the number, in sequence, of the N step as a member of the path to $((s-r)n+1, rn+1)$. Note that this method of counting does not take into consideration the order in which the balls are removed, only the set of labels.

Example 8. $r = 2,\ s = 4,\ n = 3$.
Case 1: Insert balls 1-4, and remove balls 1 and 2. Then insert balls 5-8 and remove balls 3 and 5. Then insert balls 9-12 and remove balls 7 and 9. The set of balls outside the bin is $\{1, 2, 3, 5, 7, 9\}$*, corresponding to the lattice walk* $NNNENENENEEE$.
Case 2: Insert balls 1-4, and remove balls 1 and 3. Then insert balls 5-8 and remove balls 5 and 7. Then insert balls 9-12 and remove balls 2 and 9. The set of balls outside the bin is $\{1, 2, 3, 5, 7, 9\}$*, corresponding to the lattice walk* $NNNENENENEEE$.
Although the balls removed on each turn differ in the two cases, both the sets of labels and lattice walks were the same. We see, then, that this method of counting avoids redundancy.

X. More Problems

A. Random k-tuple

Let $(I_1; I_2;\ \ldots\ ; I_k)$ be a random $k-$tuple of subintervals of the discrete interval $[1; n]$, and L_n the random variable that measures the size of their insersection. How to derive the exact and asymptotic distribution of L_n under the assumption of equally likely drawn $k-$tuples? The enumeration of such $k-$tuples and refinements of the given statistic lead to interesting relations to other topics, like octahedral numbers and bipartite graphs.

B. Graphs

Problem 9. *For any 3-connected simple cubic graph G, let X be any vertices set (of G) with three vertices. Determine the maximum number of cycles which contain X.*

Problem 10. *For any 3-connected simple graph G, let X be any vertices set (of G) with three vertices. Determine the maximum number of cycles which contain X.*

Problem 11. *For any 3-connected simple cubic graph G, let Y be any edges set (of G) with three edges and G-Y is connected. Determine the maximum number of cycles which contain Y.*

C. Identity Problems

Problem 12. *Prove or disprove*

$$\sum_{j=1}^{l} \frac{(w_j-1)^{l-1}}{\prod_{i\neq j}(w_j-w_i)} = 1.$$

Problem 13. *Prove or disprove*

$$\sum_{j=0}^{n} \binom{n}{j}\binom{j/2}{i} = 2^{n-2i}[\tfrac{n}{i}\binom{n-i-1}{i-1}].$$

D. Recursion Problem

$d_n = ((2n-1)x-1)d_{n-1} - (n-1)^2x^2d_{n-2}$ for $n \geq 1$,and $d_0 = 1, d_n = 0$ for $n < 0$.

You can see $d_n = \sum_{i=0}^{n}(-1)^{n-i}\binom{n}{i}^2 i!x^i$.

How to get a nice formula for d_n?

E.

References

[1] S.Gao, C. Caliskan, S. Sullivan,Some Sets Based on Lucas's Theorem and a Recent Paper of George Andrews, Congr. Numer., 182 (2006), 183-191

[2] P. Yiu, K. R. S. Sastry), S. Gao, Heron Sequences and Their Modifications, Combinatorial Number Theory (Walter de Gruyter) (2009), 199-204

[3] R.C. Read, Some enumeration problems in graph theory, Doctoral Thesis, University of London, (1958).

[4] B. D. McKay and X. Wang, Asymptotic enumeration of 0-1 matrices with equal row sums and equal column sums, Linear Alg. Appl., 373 (2003) 273-288.

[5] E. Rodney Canfield and Brendan D. McKay, Asymptotic enumeration of dense $0-1$ matrices with equal row sums and equal column sums. Electron. J. Combin. 12 (2005), Research Paper 29, 31 pp.

[6] Anand, Dumir, and Gupta in Duke Math J., 33 (1966) 757-769.

[7] L. Comtet, Advanced Combinatorics (page 236),Kluwer Academic Publishers,1974(page 236)..

[8] Zhonghua Tan and Shanzhen Gao,$(0,1)-$Matrices with Constant Row and Column Sums, *Congressus Numerantium* **177**(2005)3-13.

[9] Zhonghua Tan,Shanzhen Gao and Heinrich Niederhausen, Enumeration of $(0,1)-$matrices with constant row and column sums. Appl. Math. J. Chinese Univ. Ser. B 21 (2006), no. 4, 479–486.

[10] Tan Zhonghua, Shanzhen Gao and Kenneth Matheis,Some Formulas Of (0,1)-Matrices,*Congressus Numerantium* **182**(2006)53-63.

[11] L.W. Shapiro, S. Getu, W.-J. Woan and L. Woodson, The Riordan group, Discrete Appl. Math. **34** (1991), 229–239.

[12] The Online Encyclopedia of Integer Sequences (2014), published electronically at http://oeis.org

[13] P. Erdos, R. K. Guy and J. W. Moon, On refining partitions, J. London Math. Soc., **9** (1975), 565-570.

[14] John Riordan, An Introduction to Combinatorial Analysis (originally published: New York: John Wiley 1958), Dover Publications (2002).

Development of Fault Confirmation Module of Input Parameters in RLW Process by Using BPNN

Kezia A. Kurniadi[1], Kwangyeol Ryu[1], Duckyoung Kim[2], and Ho-yeun Ryu[3]
[1]Department of Industrial Engineering, Pusan National University, Busan, South Korea
[2]School of Design and Human Engineering, UNIST, Ulsan, South Korea
[3]Dongnam Regional Division, Korea Institute of Industrial Technology, Jinju, South Korea

Abstract - *Remote Laser Welding (RLW) been considered as a new and promising green technology for sheet metal assembly in automotive industry because of its benefits on several concerns. However, the recent RLW systems are limited in their applicability due to lack of systematic control methodologies. In order to overcome this problem, this study aims to develop a control module to obtain good quality joints of RLW by using Back Propagation Neural Network (BPNN). A certain parameters are used in this study, such as laser power, welding speed, part-to-part gap, plasma, temperature, and reflection. The output of this study only includes 1 variable, which is the results of visual inspection by human, ranging to 2 options (good and bad). The proposed module will provide a systematic control of RLW joints and facilitate acceptable process control during the production, and faults detection to reduce defectives.*

Keywords: Remote Laser Welding, Eco-Automotive Factories, Parameter Adjustments, Artificial Neural Network

1 Introduction

RLW is emerging as a promising and powerful laser welding technology in joining process of manufacturing industries [1]. RLW has been proved to have several benefits over another conventional technology [2] such as reduced processing time (50-75%), increased floor factory footprint (50%), reduced environmental impact/energy use (60%), and a flexible base for product changeover. The advantages of RLW provide exceptional opportunities for rapid and significant improvements in the automotive manufacturing sector with the potential to ensure its competitive edge in the world markets while leading the way on the environmental concerns [3].

Generally, the quality of a weld joint is directly influenced by the welding input parameters during the welding process [4]. Therefore welding can be considered as a multi-input multi-output process. Traditionally, it has been necessary to determine input parameters for every new welded product to obtain a welded joint with the required specifications. To do so, a time-consuming trial and error is usually required with input parameters chosen by the skill of the engineer or machine operator.

Unfortunately, a common problem that has faced the manufacturer is to control input parameters in order to obtain a good welded joint with a certain required quality. At present, there is lack of systematic methodologies and control system for the efficient application of RLW in automotive manufacturing processes [5]. This study of RLW process aims to address these obstacles for the successful implementation of RLW. Therefore, a control system is required to develop a control module to obtain good quality joints of RLW by using two stages of Artificial Neural Network (ANN) model. A certain combination of parameters is used as an input for the network to recognize the fault patterns and gives an estimated fault type as an output. These settings facilitate the development of an intelligent prediction module during the production in order to compensate for defective processes

This study makes two distinct contributions. Firstly, this study introduces a systematic methodologies and control system for the efficient application of RLW which is one of the latest technologies in automotive manufacturing processes. We formally define a set of input and output in RLW process in order to develop the control module. Second, as a main purpose of this study, we propose a fault confirmation module of RLW by using ANN. This approach is supported with the experimental study and analysis for better consideration of the approach. Moreover, this module will provide a significant support to help the operator/user to adjust and observe the behavior of RLW processes.

2 Technical background of RLW system

Laser welding can be regarded as a special way of applying the heat required for melting the materials meant to be joined—this can be done for a continuous seam as well as for a single spot [6]. Traditionally, laser welding was performed by robots equipped with laser-focusing

heads for close processing [7]. Using a laser beam to this end has a number of advantages over more conventional forms of welding like Resistive Spot Welding (RSW). First, laser welding eliminates the need of direct tool contact with the workpiece that implies large obstacle-free tool and robot sweep volumes for conventional technologies. Laser welding requires only a narrow straight line of sight free of obstruction, allowing welding in tight corners many conventional tools would not reach. Second, RLW is performed from a distance, usually with a scanner that uses mirrors and lenses to set the orientation and focal length of the beam. These components can act much faster (i.e., have a larger control bandwidth due to their small inertia) which can speed up the entire process (both tracing the welding locations as well as repositioning between seams). Also, welding can take place with all robot joints and scanner elements continuously moving, resulting in smaller control transients, implying energy savings and allowing faster operation.

The costs of an RLW cell are one more application constraint of a different kind. Switching to this new technology is only justified if expected advantages like reduction of cycle time or quality improvement balance out the costs in the given manufacturing context. The RLW system is illustrated as shown in Fig. 1.

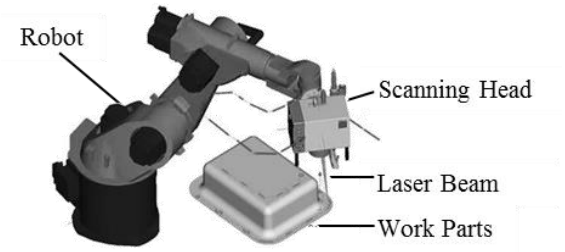


Figure 1. Body of RLW system

2.1 RLW system and development

Laser welding relies on a finely focused beam to achieve high penetration and low distortion [8]. The only exception would be if the seam to be welded was difficult to track or of a variable gap, in which case a wider beam would be easier and more reliable to use [9]. In this case, however, once the beam is defocused the competition from plasma processes should be then considered [10].

RLW takes advantages of three main characteristics of laser welding: non-contact, single side joining technology, and high power beam capable of creating a joint in fractions of a second [11]. These advantages have the potential to provide tremendous benefits on several fronts such as faster processing speed, reduced floor space, lower investment and operating costs, reduced tooling requirements, and significant reduction in energy usage and reduced environmental impact of vehicles [12].

In RLW systems, the laser beam is focused over the workpiece from a distance of about 0.5 m or more [13]. A combination of mirrors and mechanical movement of the laser delivery mechanism results in very fast beam pointing. In fact, weld-to-weld repositioning may be less than 50 milliseconds. This is more efficient than traditional spot welding or more recent laser welding involving just robot motion because the seconds needed to move the robot from one weld to another are now eliminated [14]. The quality of RLW output can be assessed visually which consists of three steps [15] such as quality control before welding, quality control after welding, and quality reliability test. Quality control before welding is done by conducting a dimension control process. The next visual inspection is quality control after welding, done by observing the weld bead conditions and counting the weld points. The last quality control process is the periodic reliability test on welding quality, which is carried out by doing tensile test and microscopic test [16]. Fig. 2 shows a fishbone diagram representing several factors affecting the quality of joint from RLW process [17].

2.2 Parameter database

The parameter database stores the experimental data for the confirmation estimation. Fundamental data of the estimation is the performance and structure described in the equipment specification. Confirmation data are obtained by experiments with various setups. Especially, the steady-state of all equipment before the welding process is measured as the basic consumed energy.

The data schema of parameter database is shown in Figure 3, adapted from Um and Stroud [18]. Each experiment has one Test table. Major measurement factors are energy of the robot and laser source. To record experimental conditions the welding part, its process, and equipment specification are included in the database. WeldingpartMat data are material type and reflection rate of the laser. Process data consist of position and orientation of the laser scanner, stitch, and welding joint. Equipment information is laser source type and robot model in order to find the specification.

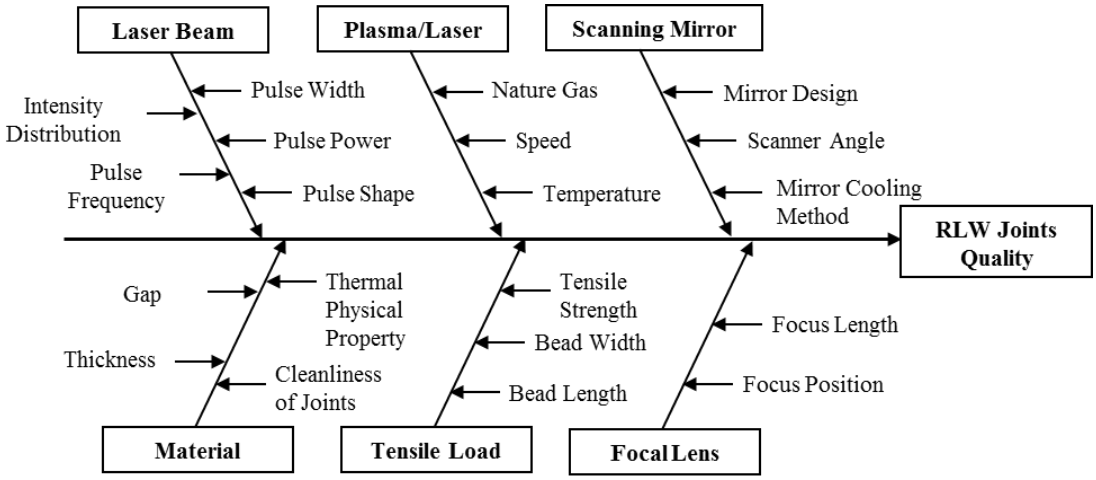


Figure 2. Fishbone diagram of RLW joints quality [17]

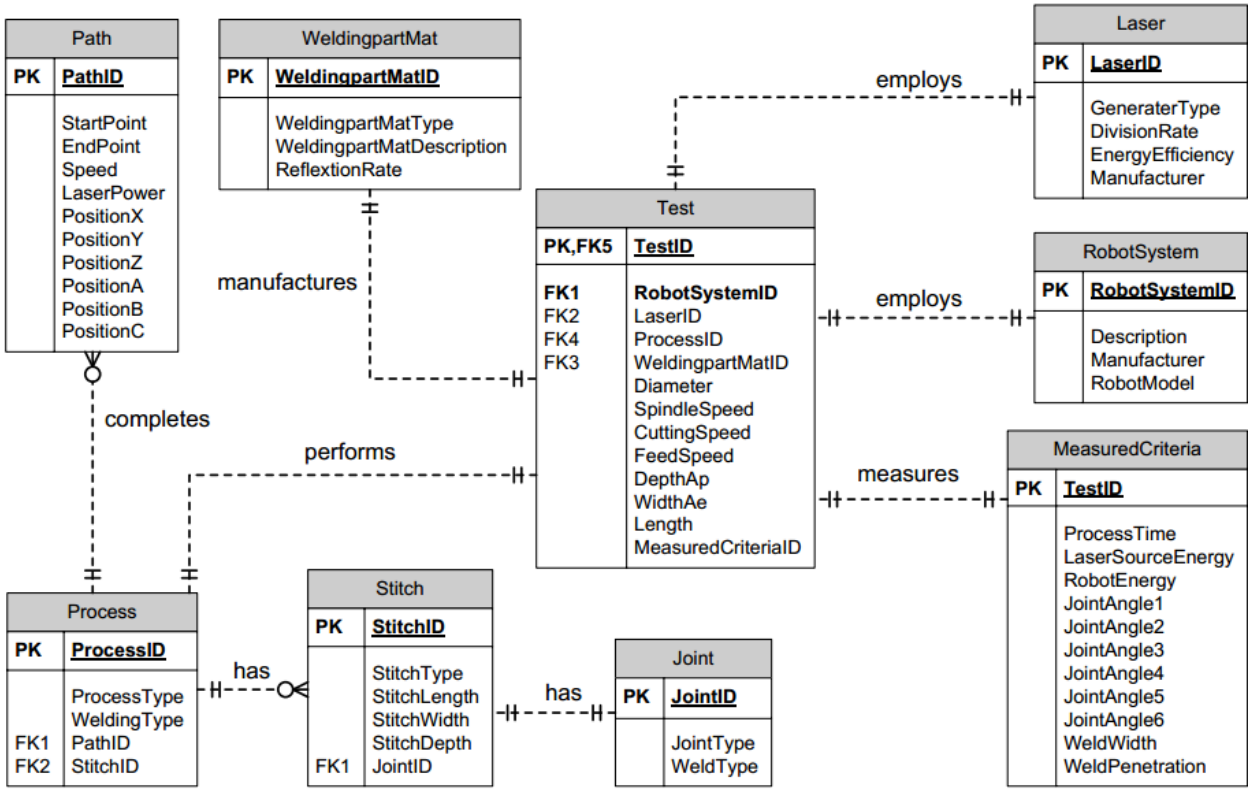


Figure 3. Data schema of parameter database [18]

2.3 Welding defects

Besides avoiding welding defects through process understanding, detection is important in industrial production. It can be distinguished between pre-, in- and post-process inspection, and between on- and off-line. Off-line post inspection is often expensive. Today in-process monitoring is provided by photodiodes or cameras, but owing to the lack of understanding it is limited to empirical correlations between the appearance of a defect and signal changes.

Despite large research efforts, understanding and detection of laser welding defects is still very limited and unsatisfactory, hindering industrial implementations. Further research will be needed to fully control this critical welding process and in turn to guarantee reliable production and safe product function. The detection or

suppression of laser welding defects is essential for successful welding in various kinds of applications. Most welding defects have to be avoided for mechanical reasons, as they cause fracture and thus (often catastrophic) failure of the product in service, i.e., under load conditions. A range of different welding defects can be distinguished as can be seen in Fig. 4.

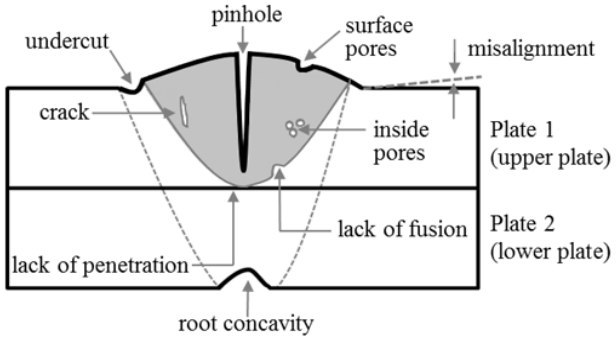


Figure 4. Type of defects

3 Related works

3.1 Fault detection techniques

In order to detect any faults within the welding process, various methods can be applied to define the desired output variables through developing mathematical models to specify the relationship between the input parameters and output variables. In the last two decades, design of experiment (DoE) technique has been used to carry out such detection. Evolutionary algorithms and computational network have also grown rapidly and been adapted for many applications in different areas [19].

Concerning Neural Network (NN), it is noted that NN performs better than the other techniques such as DoE when the case shows a non-linear behavior [19]. This technique can build an efficient model using a small number of experiments; however, the technique accuracy would be better when a larger number of experiments is used to develop a model. On the other hand, the NN model [20] itself provides little information about the design factors and their contribution to the response if further analysis has not been done.

3.2 Artificial neural networks

ANN was implemented to perform the experimental analysis using its generalization capabilities [21]. This is a powerful technique to evaluate complex interactions among process parameters and defects without referring to a particular mathematical model. It can functionally adapt its response, learning from experimental patterns [22]. The training set was obtained from the data acquisition of defect measurements. The ANN uses process parameters as inputs and gives the defect measurement values (response) as outputs[19]. ANN has been successfully applied to diverse areas such as speech synthesis and pattern recognition [23].

As recent reviews of ANN applications in manufacturing, Vitek et al. [24] have developed a model to predict the weld pool shape parameters (penetration, width, width at half-penetration and cross-section area) in pulsed Nd-YAG laser welds of Al-alloy 5754. Chan et al. [25] have proposed a model to predict the bead-on-part weld geometry (bead width, height, penetration and bag length at 22.5°) in gas metal arc welding (GMAW) of low alloy steel with C25 shielding gas. The process parameters were current, voltage, wire travel speed and workpiece thickness by using back propagation neural network (BPNN). Juang et al. [26] have used both BPNN and learning vector quantization of ANN networks to predict the laser welding parameters for butt joints. Nagesh and Datta [27] applied the BPNN to predict the bead geometry and penetration in shielded metal-arc welding without considering the structure of the ANN. They claimed that the ANN appears to constitute a workable model to predict the bead geometry and penetration under a given set of welding conditions. Gao and Zhang developed a prediction model of weld width by establishing BPNN and radial based function (RBF) neural network [28].

However, there are few literatures which discussed the modeling process of laser welding using the ANN possibly because many of the process parameters affecting welding quality are unknown. It shows that the design parameters of the ANN (the number of hidden layers and the number of nodes in a layer) can be chosen from an error analysis, and the developed ANN model can predict the bead geometry with reasonably high accuracy. In the current research, an attempt has been made to develop an ANN model in order to predict the depth of penetration, bead width and tensile strength as a function of key output parameters in the laser welding process, and to provide a basis for a computer-based control system in the future [29]. The results obtained from this ANN with several different configurations are then compared to find the one that yields the best performance based on given criteria.

4 Development of fault confirmation module of input parameters in ANN

The development of RLW control system begins with determining inputs and responses of RLW process. Therefore, the fault/defect rules are to be developed and modeled by using ANN. Based on the fault shown previously in Fig. 4, there are several input parameters and responses that can be achieved to proceed on the development of fault detection module. This experiment used Lens F160 and fiber wire with diameter of 200 micro-meters. Table 1 shows the list of input parameters and output variable used in this study.

For data acquisition, the database of raw data is extracted from the output of three different sensors for checking temperature, back reflection and plasma. The extraction also describes the data signal/trend consisting of three variables, plasma, temperature, and reflection. Therefore, as can be seen in Table 1, plasma, temperature, and reflection have no units since they are in a form of signal/normalized numbers.

Table 1. Input and output parameters

Input Parameters (6)	**Value Range**
Laser Power (kW)	2
Welding Speed (mm/min)	2100
Gap (mm)	0.00, 0.14, 0.35
Plasma	0 - 18
Temperature	0 - 27
Reflection	0 - 4
Output (1)	**Value Range**
Visual Inspection Result	1 (Good), 2 (Bad)

Training data for NN consists of 60 datasets, while validation data consists of 30 datasets. The experiment for each specimen (specimen 452-512) took 1 minute. Each specimen number consists of 5,000 experimental data including the value of plasma, temperature and back reflection for every 0.0002 minute. A welding specimen is detailed as shown in Fig. 5.

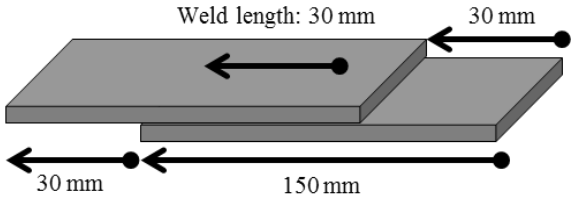


Figure 5. Weld specimen

In order to show a better data trend, firstly, the data were separated according to the gap value, and then classified based on the visual inspection result ("good" or "bad") and main input parameters (plasma and temperature). Figs. 6 –11 show the data trend. Out of training data sets (60 specimens × 5,000 data), one of the abnormalities can be shown in Fig. 5 where the graph shows that the data are not always uniform, but there are some outlier data. Table 2 shows the amount of abnormal data and specimens having abnormality. For example, specimens welded with gap of 0.00 mm show a certain signal which then classified into "good" and "bad" category. Among them, there are 6 good specimens and 15 bad specimens. As shown in Fig. 6 and Fig. 7, there are curve that shows an abnormality, those curves do not follow the data trend. After a closer observation, the abnormal curves in "good" category are specimen 476 and 489.

Table 2. Summary of training dataset

Gap (mm)	**Number of training data set**	
	Good (abnormal)	**Bad**
0.00	6 (2 → specimen 476, 489)	15
0.14	19 (2 → specimen 490, 505)	1
0.35	9 (1 → specimen 482)	10
Total	34 (5)	26

The performance function of NN used to solve the case is the Mean Square Error (MSE) minimization of the network errors on the training set. The sequence of steps based on the back propagation technique is as follows:

- A training sample is presented to the neural network.
- The network's output is compared with the desired output from that sample. The error in each output neuron is calculated.
- For each neuron, each output is calculated, and a scaling factor, how much lower or higher the output must be adjusted to match the desired output.
- The weight of the neurons is adjusted to lower the local error.

In this study, the networks have been trained by using transfer function such as Logsid, Tansig, and Purelin. Reflecting on the error value produced by those three, it turned out that using Tansig shows the smallest error. Therefore, Tansig is used for this problem, which is also known as a transfer function used for pattern recognition or similar study cases.

The type of NN is BPNN. The data has been learned by randomly selected, not sequentially. The learning rate is 0.2, momentum is 0.9, and number of epochs is 10,000. The convergence criterion employed in the network training is the Root Mean Square Error (RMSE). After some experiments to decide the number of hidden nodes in the hidden layer, we found that the structure 6-3-6-1 was selected to obtain a better performance. Note that the numbers in the structure indicate the number of nodes in each layer. In other words, the ANN architecture consists of 1 input layer with 6 input nodes, 2 hidden layers with 3 and 6 hidden nodes in first and second hidden layers respectively, and 1 output layer with just 1 output node, as shown in Fig. 12, while Fig. 13 shows the training state in Matlab 2009.

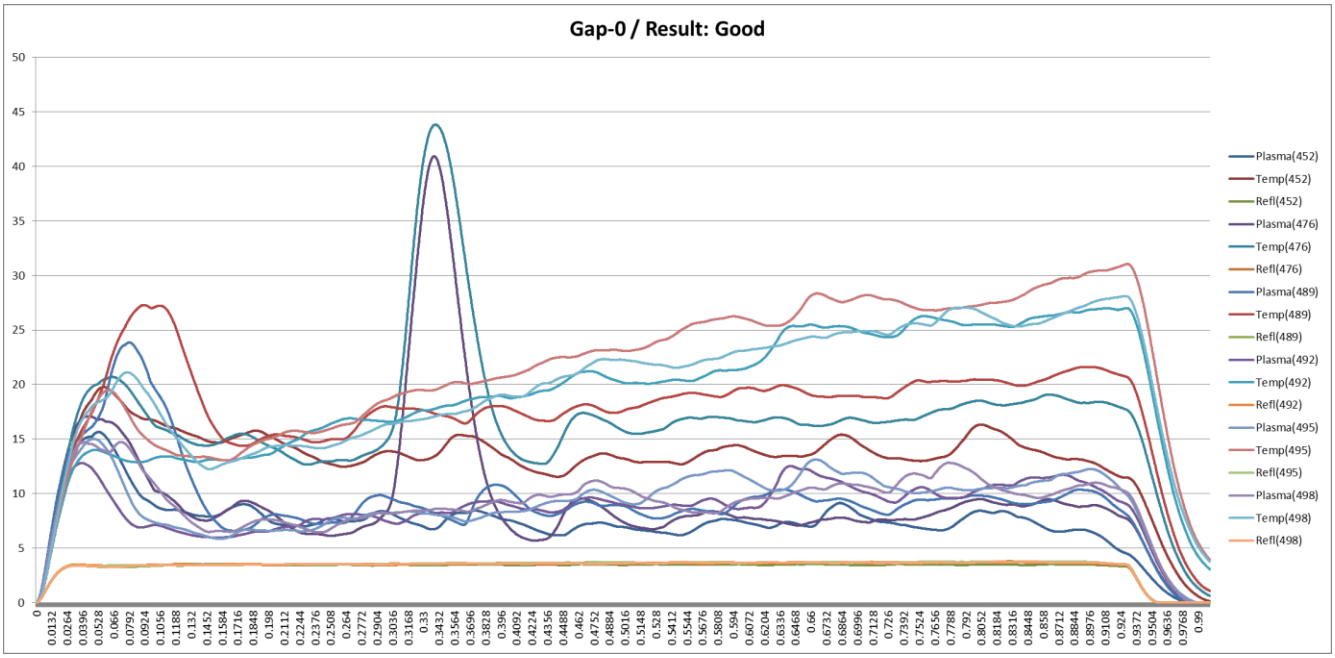


Figure 6. Gap = 0.00 mm (good)

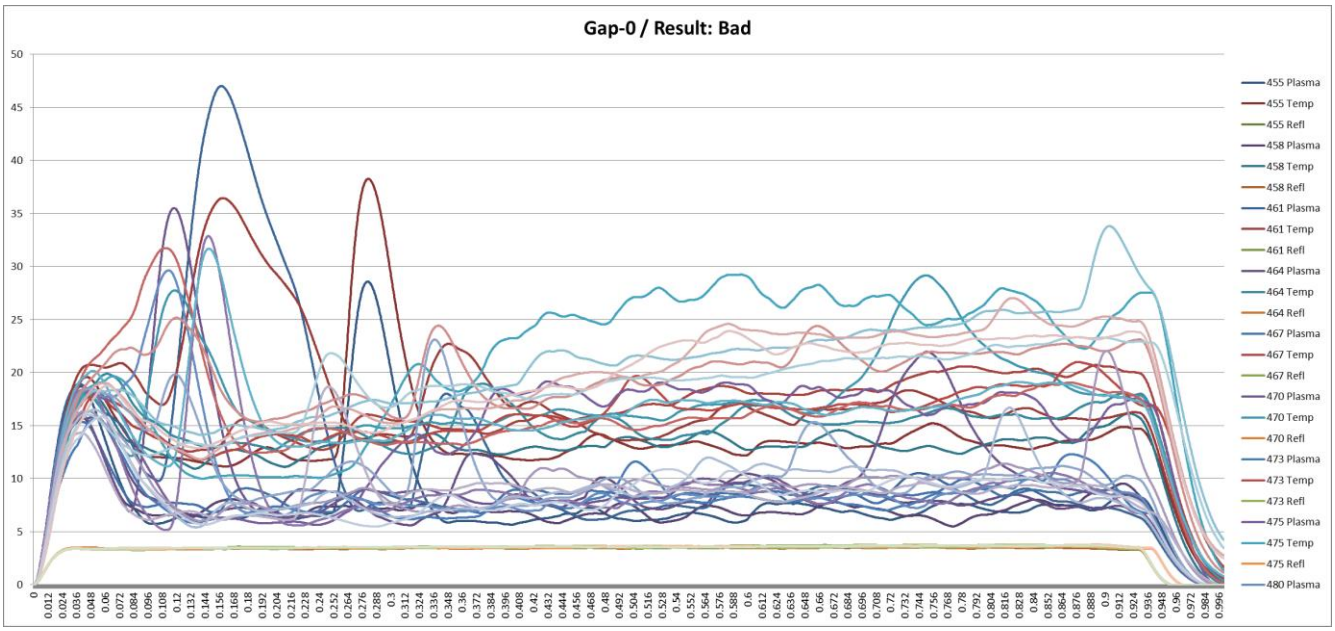


Figure 7. Gap = 0.00 mm (bad)

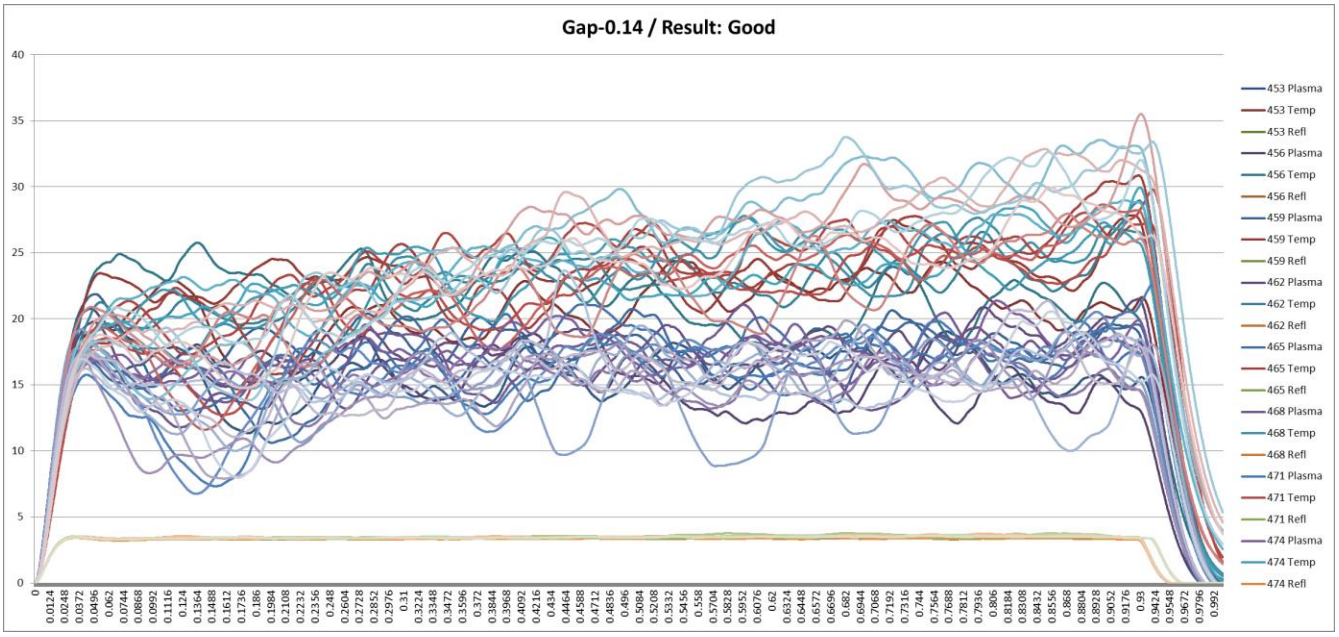


Figure 8. Gap = 0.14 mm (good)

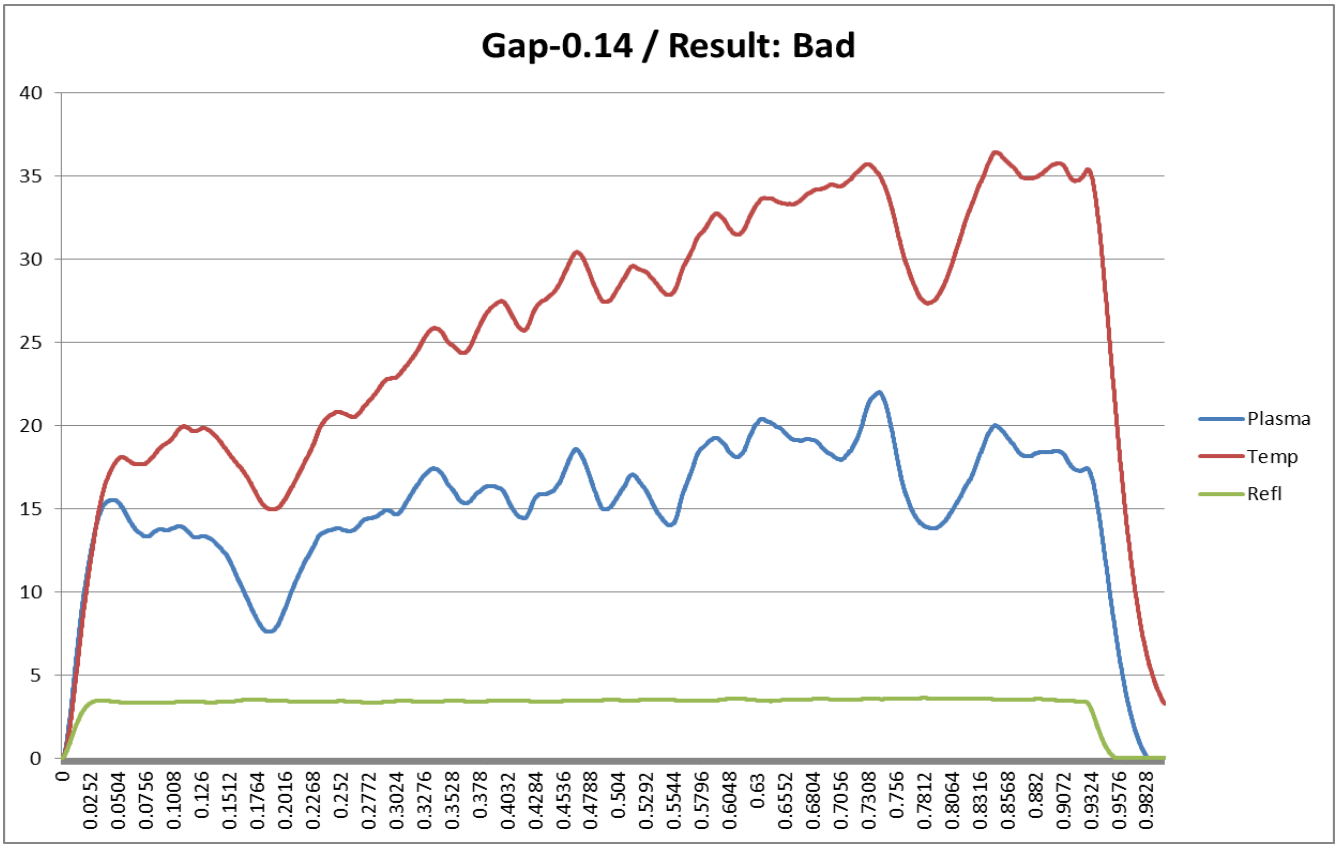


Figure 9. Gap = 0.14 mm (bad)

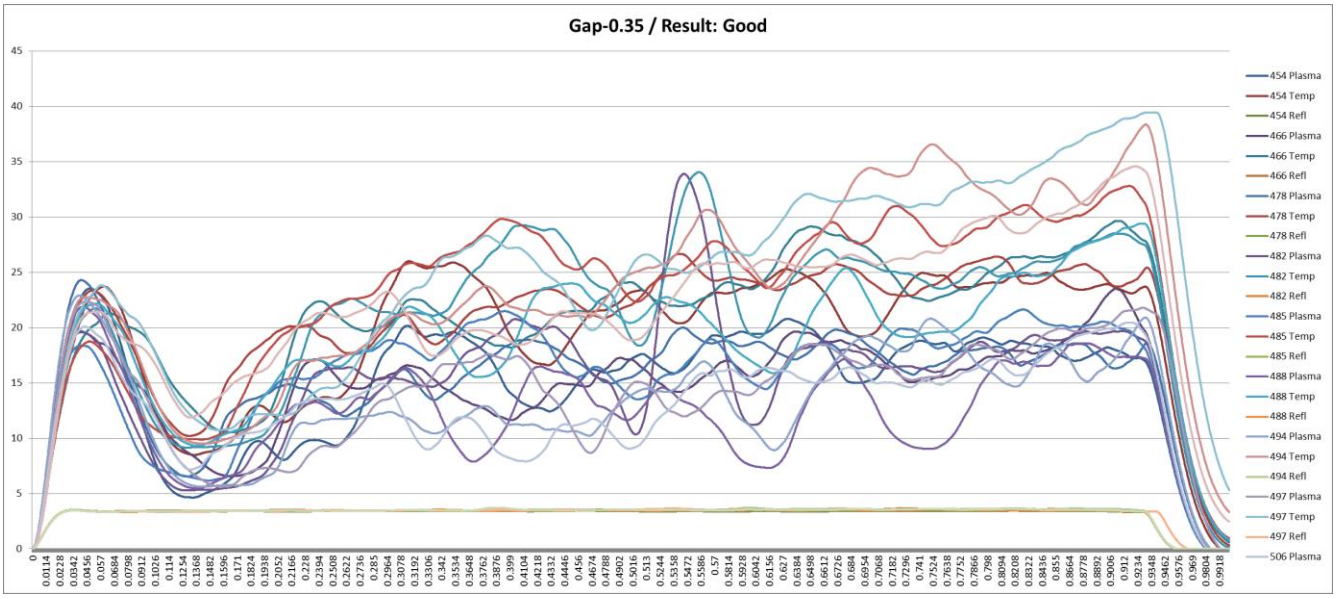


Figure 10. Gap = 0.35 mm (good)

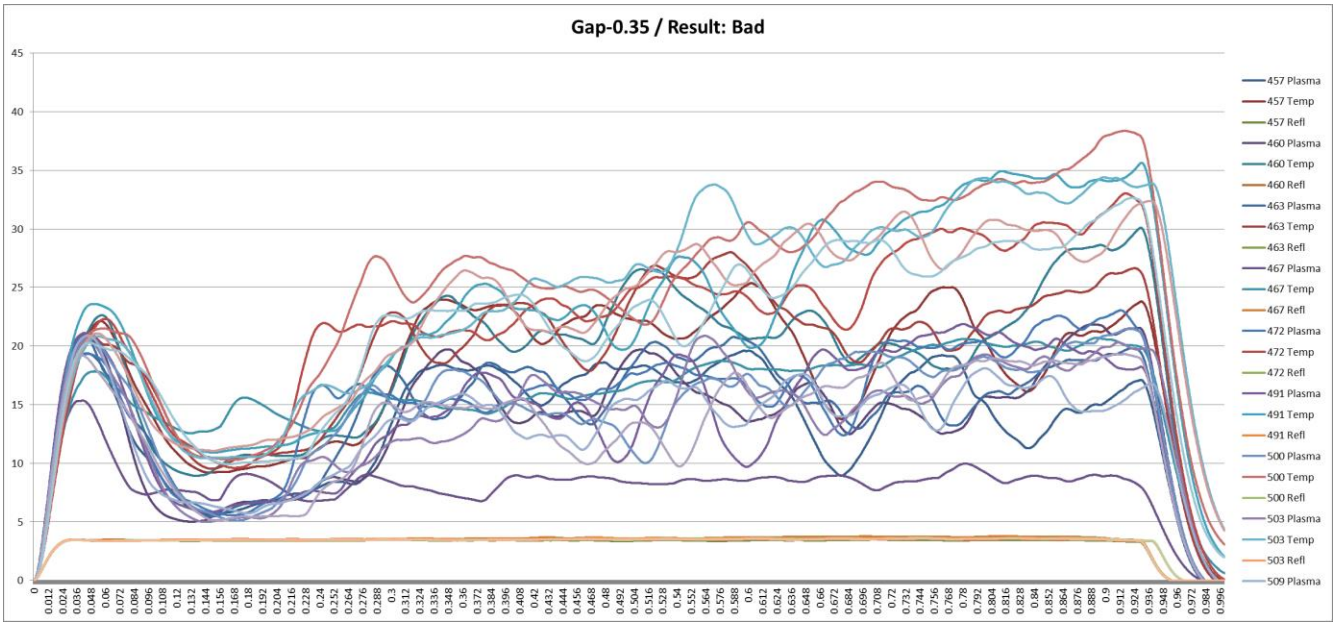


Figure 11. Gap = 0.35 mm (bad)

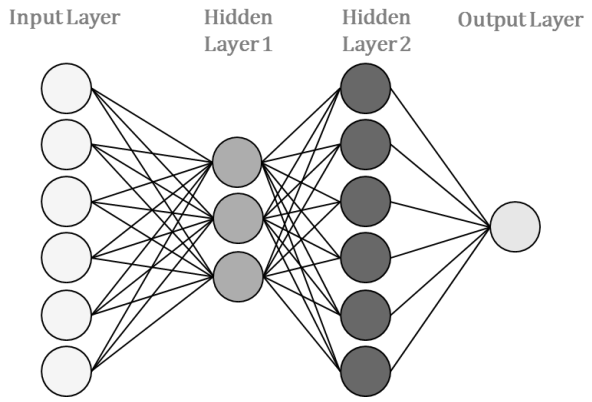


Figure 12. ANN architecture

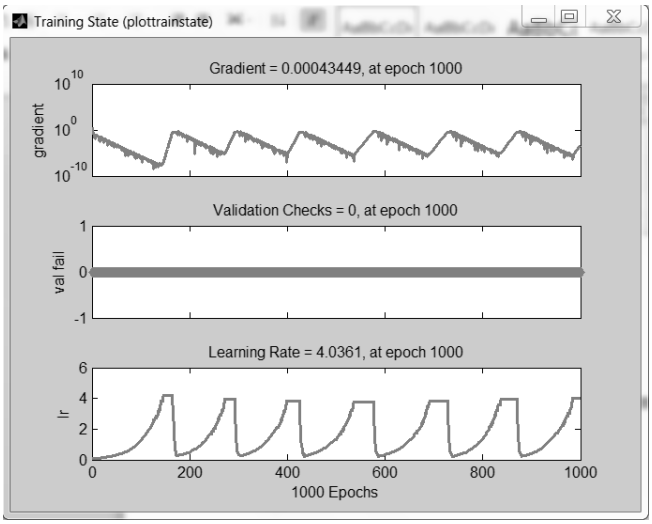


Figure 13. Training state of ANN model in Matlab 2009

The training result showed that the model matched perfectly. The validation testing is used to determine the performances of the ANN. The validation is not used to update the network. As the ANN "learns" the error over the validation set decreases. The trend will eventually stop or even reverse: this phenomena is called "overfitting", which means the NN is now chasing the training set, "memorizing" it at the cost of the ability to generalize. The purpose of the validation is to determine when to stop the training before the NN starts overfitting. Finally for external testing, to leave you a set of data untouched by the training procedure. With the same ANN used for training, that ANN model was then tested by using validation dataset. However, NN Validation Result for Specimen 565-571 shows abnormalities, therefore we need to re-train the ANN in order to fit the value of those abnormal data. In order to re-train the ANN, the training data needs to be modified. The abnormal specimens are modified by changing their category from “good” to “bad”. Table 3 shows the result from the modification of the training data. After changing their category, by applying the same NN architecture, the NN model is re-trained re-validated. Table 4 shows the new NN Result from the validation data. The effort of re-training the abnormal data makes no effects. As shown in Table 3 and Table 4, the value of NN result is even more different than before and the total error rate is getting bigger, as shown in Table 5. The error values shown in Table 5 are calculated from the difference of error value between the old and new (modified) NN result from each; training and validation data.

Table 3. Modified training data (abnormal data only)

Gap (mm)	Specimen Number	Initial Visual Inspection		Modified Visual Inspection	
		Initial Result (Good)	NN	Modified Result (Bad)	NN
0.00	476	1	1	2	4
0.00	489	1	1	2	1
0.14	505	1	1	2	1
0.14	490	1	1	2	1
0.35	482	1	1	2	1

Table 4. New result of validation data

Gap (mm)	Specimen Number	Visual Inspection	NN Result	
			Old	New
0.40	565	2	1	3
0.45	566	2	1	0
0.00	567	2	1	1
0.01	568	2	1	1
0.03	569	2	1	1
0.05	570	2	1	1
0.11	571	2	1	1

Table 5. Total error rate comparison (absolute value)

	Training	Validation
Initial	0	7
Modified	6	8

5 Conclusion

This study proposes BPNN to discover the relationship between welding signal and quality of RLW products. RLW quality is classified into two categories, “good” and “bad”. All experiments are conducted under MATLAB 2009 software with real experiment data and visual inspection (with human error). The great number of experiment data is shown in graphs to clearly picture the signal trend of whole welding process for one minute each specimen.

The result of visual inspection featuring human error is somehow producing few abnormalities which can be

clearly seen in the signals. Those abnormalities are the obstacles of producing perfect result of proposed NN. In order to fix those problems, the experiment (training) data needs to be modified by changing the category, from "good" to "bad", or the opposite. After changing it, then the NN should be re-trained and re-validated. Then the error (RMSE) is to be compared between initial and modified NN result.

The proposed NN architecture can predict the value of responses derived from the complex process with multiple responses. These proposed settings facilitate the users in achieving acceptable process control during the production. Not only can the prediction results be very useful in selecting suitable welding parameters, they can also help in avoiding inappropriate welding design.

Acknowledgements

This research was supported by the project, 'Remote Laser Welding Process Control for Eco-Automotive Factories,' funded by Ministry of Trade, Industry & Energy(MOTIE), Republic of Korea.

6 References

[1] Macken, J., "Remote Laser Welding," Proceedings of the International Body Engineering Conference, Cambridge, Massachusetts, pp. 11–15, 1996.

[2] Ostendorf, A., "Laser Remote Welding-from Development to Application," European Automotive Laser Application, Bad Nauheim, Frankfurt, pp. 196–230, 2005.

[3] Emmelmann, C., "Laser Remote Welding-Status and Potential for Innovations in Industrial Production," Proceedings of the 3rd International WLT Conference on Lasers in Manufacturing, Stuttgart, Munich, pp. 1–6, 2005.

[4] Menin, R., "Remote Laser Welding. The COMAU Solution," Proceedings of the 10th Annual Automotive Laser Applications Workshop, Dearborn, pp. 101–116, 2005.

[5] Menin, R., "The COMAU Standard 3D Remote Solution," European Automotive Laser Application, Bad Nauheim, Frankfurt, pp. 331–349, 2005.

[6] Erdõs, G., Kemény, Z., Kovács, A., Váncza, J., "Planning of Remote Laser Welding Processes," Proceedings of the 46th CIRP Conference on Manufacturing Systems, Vol. 7, pp. 222–227, 2013.

[7] Colombo, D., Colosimo, B. M., Previtali, B., "Comparison of Methods for Data Analysis in The Remote Monitoring of Remote Laser Welding," Journals of Optics and Lasers Engineering, Vol. 51, pp. 34–46, 2013.

[8] Kim, C. H., Kim, J. H., Lim, H. S., Kim, J. H., "Investigation of Laser Remote Welding Using Disc Laser," Journal of Materials Processing Technology, Vol. 201, pp. 521–525, 2008.

[9] Connor, L. P., "Welding Handbook-Welding Processes," American Welding Society, 8th ed., 1991.

[10] Benyounis, K. Y., Olabi, A. G., "Optimization of Different Welding Processes using Statistical And Numerical Approaches-A Reference Guide," Advances in Engineering Software, Vol. 39, pp. 483–496, 2008.

[11] Murugan, N., Gunaraj, V., "Prediction and Control of Weld Bead Geometry and Shape Relationships in Submerged Arc Welding of Pipes," Journals of Materials Processing Technology, Vol. 168, pp. 478–487, 2005.

[12] Yang, L. J., Bibby, M. J., Chandel, R. S., "Linear Regression Equations for Modeling The Submerged-Arc Welding Process," Journal of Material Processing Technology, Vol. 39, pp. 33–42, 1993.

[13] Kim, I. S., Son, J. S., Kim, I. G., Kim, O. S., "A Study on Relationship Between Process Variable and Bead Penetration for Robotic CO_2 Arc Welding," Journal of Material Processing Technology, Vol. 136, pp. 139–145, 2003.

[14] Andersen, K., Cook, G., Karsai, G., Ramaswamy, K., "Artificial Neural Network Applied to Arc Welding Process Modelling and Control," IEEE Transactions on Industry Applications, Vol. 26, pp. 824–830, 1990.

[15] Cook, G., Barnett, R. J., Andersen, K., Strauss, A. M., "Weld Modeling and Control Using Artificial Neural Networks," IEEE Transactions on Industry Applications, Vol. 31, pp. 1484–1491, 1995.

[16] Juang, S. C., Tarng, Y. S., Lii, H. R., "A Comparison Between The Backpropagation and Counter-Propagation Networks in The Modeling of The TIG Welding Process," Journal of Material Processing Technology, Vol. 75, pp. 54–62, 1998.

[17] Kurniadi, K. A., Ryu, K., Kim, D. Y., "Real-Time Adjustment and Fault Detection of Remote Laser Welding Parameters by Using ANN," International Journal of Precision Engineering and Manufacturing (Special Issue: ISGMA 2013), Vol. 15, No. 6, pp. 979–987, 2014.

[18] Um, J. Y., Stroud, I. A., "Total Energy Estimation Model For Remote Laser Welding Process," Proceedings of the 46th CIRP Conference on Manufacturing Systems, Vol. 7, pp. 658 – 663, 2013.

[19] Vitek, J. M., Iskander, Y. S., Oblow, E. M., "Neural Network Modeling of Pulsed-Laser Weld Pool Shapes in Aluminum Alloy Welds," Proceedings of the 5th International Conference On Trends In Welding Research, Pine Mountain, GA, pp. 442–448, 1998.

[20] Chan, B., Pacey, J., Bibby, M., "Modeling Gas Metal Arc Weld Geometry using Artificial Neural Network Technology," Journal of Canadian Metallurgical Quarterly, Vol. 38, pp. 43–51, 1999.

[21] Juang, S. C., Mau, T., Leu, S., "Prediction of Laser Butt Joint Welding Parameters using Back Propagation and Learning Vector Quantization Networks," Journal of Material Processing Technology, Vol. 99, pp. 207–218, 2000.

[22] Nagesh, S., Datta, G. L., "Prediction of Weld Bead Geometry and Penetration in Shielded Metal-Arc Welding using Artificial Neural Networks," Journal of Material Processing Technology, Vol. 123, pp. 303–312, 2002.

[23] Loredo, A., Martin, B., Andrzejewski, H., Grevey, D., "Numerical Support for Laser Welding of Zinc-Coated Sheets Process Development," Journal of Applied Surface Science, Vol. 195, pp. 297–303, 2002.

[24] Martin, B., Loredo, A., Grevey, D., Vannes, A. B., "Numerical Investigation of Laser Beam Shaping for Heat Transfer Control in Laser Processing," Journal of Lasers in Engineering, Vol. 12, pp. 247–269, 2002.

[25] Hepworth, J. K., "Finite Element Calculation of Residual Stresses in Welds," Proceedings of the International Conference on Numerical Methods for Non-Linear Problems, pp. 51–60, 1980.

[26] Tekriwal, P., Mazumder, J., "Transient and Residual Thermal Strain-Stress Analysis of GMAW," Journal of Engineering Materials and Technology, Vol. 113, pp. 336–343, 1991.

[27] Lindgren, L. E., "Finite Element Modeling and Simulation of Welding Part 3: Efficiency and Integration," Journal of Thermal Stresses, Vol. 24, pp. 305–334, 2001.

[28] Gao, X. D., Zhang Y. X., "Prediction Model of Weld Width during High-Power Disk Laser Welding of 304 Austenitic Stainless Steel," International Journal of Precision Engineering and Manufacturing, Vol. 15, No. 3, pp. 399–405, 2014.

[29] Canas, J., Picon, R., Pariis, F., Blazquez, A., Marin, J. C., "A Simplified Numerical Analysis of Residual Stresses in Aluminum Welded Plates," Computers & Structures, Vol. 58, pp. 56–69, 1996.

Evolutionary change in the way of information access. Searching versus browsing information in ECM systems.

L. Osuszek[1], **S. Stanek**[2]
[1]Wydział Zarządzania, GWSH, Katowice, Poland

Abstract *This paper renders evolution of typical information structures in Enterprise Content Management (ECM) systems, as well as the change in user's routines and way of working with data. Early implementation of ECM systems dictated some patterns for working with unstructuralized information. Participants of ECM and BPM systems were browsing the documents hierarchy to discover necessary data and enable better decisions faster.*

As the ECM evolves – approach to work changes. Result of such operation was analyzed for optimization in category of time & resources consumption. Change in ECM user's behavior - results in tangible savings and economical benefits. This article presents differences between browsing and searching for information in modern Enterprise Content Management Systems.

Keywords: knowledge workers, information structures, hierarchy, searching, browsing documents, unstructuralized information,

1 Introduction

For each company and organization content management, business processes and the related decisions are the key element, which provides the momentum for their operations. The management of documents and information within process paths has a major impact on the speed, flexibility and quality of decision-making processes. This is why the acceleration and optimization of processes is decisive for the success of any organization.

Processes involve information, people and systems. The maximum efficiency is possible only if all of these elements interoperate in an automated environment. Note also that optimized processes enable a faster response to the changing market situation and to new customers' demands while guaranteeing compliance with applicable regulations. In short, better organization of information and processes contribute towards continuous improvement of the efficiency of company's operations, and therefore, allow gaining competitive advantage in the industry.

Nowadays amount of produced information is overwhelming and proper information management is the key. According to the published researches:

- Servers in companies worldwide in 2008 processed 9.57 zettabytes of information. That amounts to:
  - 3.0 terabytes of information per worker per year, or 12 gigabytes per worker per day (based on the ILO and CIA Facebook's estimate of 3.18 billion people in the world labor force in 2008).
  - 63 terabytes of information per company per year (based on Dun & Bradstreet's 151 million world businesses registered with D&B'S D-U- N-S system in 2008)

Facebook grew 228% year-over-year, from 20 million unique visitors in February 2008 to 65.7 million visitors in February 2009. Projecting these trends forward, researchers expects a 650% growth in enterprise data over the next five years, with more than three-quarters of that growth in unstructured data.

Typical proportion for structured and unstructured data is classical Pareto 20/80. This paper is focusing on managing the unstructured information in ECM platforms. It shows the changing trend of typical system user's behavior of working and managing the information.

To illustrate the actual state of the art and to further describe how and when users decide to search versus browse I will use research of Teevan, Alvarado, Ackerman and Karger (2004) report a modified diary study of motivated information seeking across email, files and the Web. They conducted burst interviews with each of the participants at two unspecified interruption points per day for 5 consecutive days. The "interviews" were simple 5 minute debriefs in which the researchers asked the participant to describe what s/he had recently "looked at" or "looked for" in their email, files or on the Web. The burst interviews were supplemented

by direct observation and longer semi-structured interviews to explore their information patterns.

Given the participants' advanced computer experience and familiarity with complex information spaces and sophisticated search tools, Teevan and colleagues were surprised to find that their participants used key-word searches in only 39% of their searches – despite the fact that they almost always knew key details of the information they needed up front.

Based on their findings, Teevan and team describe two strategies for information navigation: Teleporting and Orienteering.

Teleporting occurs when a person jumps directly to the information they are seeking.

Orienteering consists of narrowing the search space through a series of steps (e.g., selecting links) based on prior and contextual information to hone in on the target. Most often, participants took an initial "large" step to the vicinity or information source and then refined the search space further through smaller steps based on local exploration.

Teevan, argue that orienteering provides three benefits for the user over teleporting: Orienteering is less cognitively demanding. It does not require discrete articulation of the searched-for item at the onset of the search. It allows users to rely on habit to get to the information target space, effectively reducing the search space. Orienteering provides the user a greater sense of control and location. Small, incremental steps in orienteering provide additional context for interpreting results.

In their study, Teevan and colleagues observed significantly more orienteering than teleporting behavior. Three additional interesting observations emerged. First, participants in their study consciously chose not to teleport, even when teleporting appeared viable. Second, participants tended not to use keyword searching. Third, on some occasions when participants employed keyword search, it was used as a tactic within orienteering. That is, at least one participant used iterative keyword searches to incrementally narrow the search space in small steps.

By analyzing ECM references form Central and Eastern Europe I'd like to confront previous thesis with actual Content Management routines.

2 Traditional way

Enterprise Content Management (ECM) is the strategy, methods and tools used to capture, manage, store, preserve, and deliver content and documents related to organizational processes. ECM covers the management of information within the entire scope of an enterprise whether that information is in the form of a paper document, an electronic file, a database, print stream, or even an email.

ECM aims to make the management of corporate information easier through simplifying storage, security, version control, process routing, and retention. The benefits to an organization include improved efficiency, better control, and reduced costs.

Information is organized into entity consist of metadata and binary object. ECM tools and strategies enable the management of an organization's unstructured information, wherever that information exists.

Users of first ECM platforms took advantage of the tools and methodology of the unstructured information management to create the document classes, metadata model and information structures. In most of the cases customers used content classification to organize the information into dedicated structures (example taken from customer from Energy & Utilities sector)

Taxonomy provided a formal structure for information, based on the individual needs of a business. Categorization tools automate the placement of content (document images, email, text documents) for future retrieval based on the taxonomy. Users could also manually categorize documents to ensure that content is properly stored.

Most of the earlier '90s ECM indexing/classification systems focus on the hierarchical model, as represented by the familiar paradigm:

However, not every information falls into this paradigm. Engineering information is massively interrelated and interdependent, which does not lend itself to such neat, flat hierarchies. For example: a drill-down for a energy plant engineer expecting to find the data for a pump at the bottom of a *plant -> area -> system* hierarchy may not be as intuitive for a purchasing engineer who would expect to find the same pump data at the bottom of a *supplier -> requisition -> contract* hierarchy (without obviously duplicating the data). Therefore, the engineering indexing/classification system can be more likened to orienteering model.

In early ECM systems (Image Services, Content Services + eProcess), typical scenario of gaining the business information was to browse the structure and prepare it for optimal processing. Typical searching of information was less used. Users who browse have different purposes than those who search, and we can't assume that those who browse can find content the same way through a search.

- Browsers (users who attend to browse the structures of information) had typically less knowledge about the whole business process they

were part of. They focus on working with recurrent business scenarios without any process exceptions

- Searchers have more knowledge and demands for finding information. A user relies on search to find specific information he or she already knows or suspects to exist. Rarely does a user search for something he or she doesn't even know to search for.

Often customers using the phrase browse&search but "browse" and "search" are clearly not the same thing. To browse is to skim or scan a displayed list of taxonomy terms, whether arranged alphabetically, hierarchically, or a combination. To search is to enter search terms into a search box (which may then be matched against a controlled vocabulary for more accurate results).

After Rosenfeld and Morville's 1998 book Information Architecture for the World Wide Web, the field began to differentiate between two discrete behaviors — "browse" and "search." People who browse the TC Library's *Careers > Technical Writing* section are behaving differently and have different goals than people searching for a particular article. This rule is also true with regard to ECM systems.

A user browses a table of contents when he or she is looking around to see what's there, usually moving with a mind open to discover new information. In contrast, a user searches content when he or she needs to locate a specific piece of information.

In The Future of Search and Discovery, Peter Morville notes that one limitation of search is finding out the "unknown unknown." This term, the "unknown unknown," comes from Donald Rumsfeld, Morville notes. The full Rumsfeld quote is as follows:

There are known knowns. These are things we know that we know. There are known unknowns. That is to say, there are things that we now know we don't know. But there are also unknown unknowns. These are things we do not know we don't know.

Implementation of ECM document metadata allows the user to dynamically restrict, filter, or limit a data set, based on selecting values from each of multiple properties that are displayed, typically in the right-hand margin, while references to content is displayed in the main screen area. This implementation of information can be considered "browse" because the user browses the displayed results and the displayed terms within each document.

In the early decade of ECM systems browsing and discovering the information was the scenario which dominated typical users routines for managing the information or completing workflow tasks.

3 New approach – baskets, filters, cases

As the fresh releases of ECM tools arrived it bought new concept of work. Business Process Framework changed the way of working with information and business processes. Framework for the development and deployment of business solutions based on the integration of content and workflow management.

Approach provided in Business Process Framework (BPF) – early case management system - optimized time of operating on case data. The tasks required by a case usually involve creating a case folder or container for all required artifacts. Another important step in the processing of a case involves following business procedures, both determined and ad hoc, ensuring the delivery of the service. Due to the dynamic and unique nature of each case, the requested services usually require collaboration with other specialized workers both inside and outside of the servicing organization. When combined, case management is highly collaborative, dynamic, and contextual in nature, with events driving a long lived case-based business process. The aggregation of many cases with a semblance of consistency, insight, and optimization becomes a challenging effort for both the caseworker and the organization.

In the beginning f the first decade of 21th century - in most of the cases users searched or filtered data instead of typical browsing scenario. The information set that is filtered by the metadata could be the entire set of content, but more likely it is a subset, based on a prior execution of either a category (for example document class) selection or a search. If the user's first step was to initiate a search to obtain search results, and then uses filters to limit the search results, this might be called "faceted/filtered search." Even though the user browses the results, because the facets are introduced as a second step following search, this step might be called "filtered search." If, however, the user's first step were to browse subject categories and select a category to obtain the initial data set, then the use of filters in the second step would more likely be called "faceted/filtered browse." Second step has more impact on the users behavior and BPF users spent more time on it.

Another implementation of filters and task-baskets is to allow the user to select among limiting criteria from the beginning, without first selecting a subject by browse or search. In order to achieve usable results (result sets that are not too large), the facets need to contain relatively large taxonomies: a large number and deep set of terms. While it is certainly possible to display a large taxonomy for browsing, it may be difficult to display multiple large, browsable taxonomies, one for each facet. Therefore, if facets are made available to the user from the start (without first requiring the user to select a limited data set based on a search or browse selection), it is more likely that not all the facets will display the terms to the user. The user must then execute a search within a filtered data. This would correctly be called "faceted/filtered search."

The distinction between filtered browse and filtered search is lost, however, where the distinction between browsing and searching is becoming blurred. Newer user interface implementations of taxonomies are combining search and browse, so that the difference is no longer as obvious. For example, there are examples where is a search box, and as the user types in something, a type-ahead feature matches the search string against controlled vocabulary terms, which are displayed in a short list under the box, and the user can browse the list to select a term. There are also business scenarios where case user may be presented with a search box to enter search terms, and there is a button next to the search box, which the user may optionally click, and then the search box becomes a scroll box to view and browse the entire controlled vocabulary for that field. When these kinds of advanced taxonomy-enhanced search boxes correspond to facets, the distinction between "filtered search" and "filtered browse" truly no longer exists.

4 Advanced Case Management

What we can observe for the last 3 years is another phase of ECM evolution. Adaptive Case Management is the different approach for managing the information.

The facilitation of knowledge work or what is increasingly known as "Case Management" represents the next imperative in office automation. The desire to fully support knowledge workers within the workplace is not new. What's new is that recent advances in Information Technology now make the management of unpredictable circumstances a practical reality.

There's now a groundswell of interest in a more flexible, dynamic approach to supporting knowledge. This approach results with reversal of the typical user scenarios of work with information. Comparing to the previous ECM generation – users of ACM requires 360 degrees view of the case. Now the speed of information gathering is the ultimate criterion.

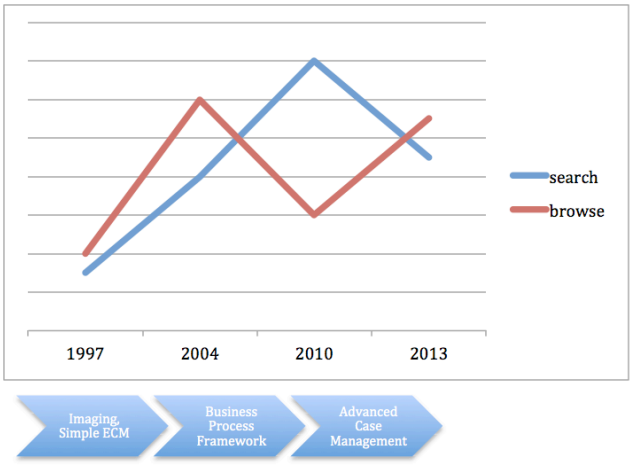
Most of the ACM systems do not build complex structure of data. Rather focuses of providing efficient searching or filtering mechanisms. Structures like hierarchical folders or compound documents have technical limitation (for example Windows barriers for catalogs number, IBM Filenet limitation for virtual folders etc). This is also premise in process of designing new efficient systems for managing information for enterprise scale customers. However the visible tendency of ACM is to use equally browse or discover mechanisms with searching/ filtering. Knowledgeworkers works with the system according to their specific needs and habits. Some may search. And some may be very good searchers, stringing together AND/OR operations, enclosing unique phrases in quotation marks problem of search is to know what we are searching for… Memorizing the names of perhaps several hundred documents (for example invoices) to know what to type into the box is not so easy.

Search deprives users from discovering the unknown unknown. But there's another benefit to browsing. Let's take the browsing behavior in help content as an example. Users who browse the help will realize that it contains empowering information to make them smarter users with the application. Users who sink their souls into the help for a while can walk away with a renewed confidence about it contents and what they can do in the application. Users who default to search modes only may never come to see all the valuable information in help. They may search for the wrong terms several times and give up with an impatient frustration, dismissing help entirely. But users who browse will "see more and learn more."

This tendency is very important for the knowledge workers who can benefit form the better insight of the data and complete 360-degree view of case data. However using filters and searching for information is also strong in ACM environment.

5 Summary

The article presents real examples to illustrate the phenomenon of the evolution of typical information structures in enterprise systems for managing the unstructured information, as well as the change in user's routines and way of working with data (metadata model and binary content). Paper presents examples of early ECM implementation associated with defined patterns for working with unstructuralized information. Participants of ECM and BPM systems were browsing the documents hierarchy to discover necessary data and enable better decisions faster. Below picture illustrates the most important phases of the data management evolution in terms of search and browse information:



Article covers as well the evolution of ECM systems (approach to work changes). It shows the transition from

typical for early systems scenarios of discovering information – to modern approach with searching and filtering data. This article presents differences in browsing and searching for information in Enterprise Content Management systems – which could illustrate the actual trend for production, IT systems.

6 References

[1] http://www.infoworld.com/d/data-explosion/datacenter-challenges-include-social-networks-rising-energy-costs-614

[2] „How Much Information? 2010 Report on Enterprise Server Information“, James E. Short, Roger E. Bohn Chaitanya Baru

[3] „Information architecture“, After Rosenfeld and Morville's 1998

[4] "In The Future of Search and Discovery", Peter Morville, 2002

[5] "Advanced Case Management with IBM Case Manager", IBM Redbooks, 05.09.2013

[6] "Mastering the Unpredictable", Keith Swenson, 2009

[7] "The perfect search engine is not enough: a study of orienteering behavior in directed search", Teevan, Alvarado, Ackerman and Karger, 2004