An Agent-Based Intentional Multi-Robot Task Allocation Framework

Savas Ozturk¹, Ahmet Emin Kuzucuoglu²
¹TUBITAK BILGEM, Gebze, Kocaeli, Turkey
²Department of Computer and Control Education, Marmara University, İstanbul, Turkey

Abstract - When multiple robots are supposed to operate together, coordination and communication issues arise. "Which robot should execute which task?" is the key question of Multi-Robot Task Allocation (MRTA) problem. Among several autonomy level approaches being centralized or distributed, hybrid approaches have become popular at reaching optimal task allocation combining advantages of both centralized and distributed approaches. Hybrid approaches like auction-based or market-based are also named as intentional according to their cooperation model. Intentional type allocation model differs from swarm-type basically for number of robots employed. This study presents development of an intentional MRTA model based on multi-agent concepts using Java Agent Development Environment (JADE). A simple model is built and proposed on prior approaches in market-based task allocation in order to improve optimality. Experiments are performed using selected cases of MRTA literature and results prove that proposed model in this study produces optimal solution at each case.

Keywords: Multi-Robot Task Allocation(MRTA), market-based, intentional, auction, multi-agent, Java Agent Development Environment (JADE)

1 Introduction

MRTA problem has become a key research topic in the field of distributed multi-robot coordination in recent years. The goal of MRTA is to choose on time a proper action for every robot to ensure that they can effectively complete all the tasks. Supposing that any individual of the system cannot complete a complicated problem, this problem can be decomposed to the series of sub-problems on some principles and these sub-problems can distributed to corresponding individual to solve the problem. Presently, there are many methods for task allocation grouped in three main approaches: fully centralized, fully distributed and hybrid approaches.

A centralized task allocation approach used for multi-robot system suffers from limitations in both system scalability and robustness. If the central robot fails, the whole system fails. On the other hand, a fully distributed system produces suboptimal solution. Hybrid approaches combine both centralized and distributed approaches combining their advantages. Each individual in the system works for its own plan and benefit and at the end global benefit is emerged. Market-based approaches are basically built on buying and selling goods to the planner or other robots. Another classification about multi-robot task allocation is based on cooperation which groups task allocation types as swarm-type or intentional-type [1]. Swarm-type task allocation is performed by large number of robots and it depends on fully distributed approach. Intentional-type task allocation is either auction-based or market-based. This study mainly investigates intentional task allocation mechanisms and builds a taxonomy including the related literature and proposed model. Proposed model is a mix of prior techniques and includes some improvements, finally a novel MRTA task allocation mechanism is built which an objective as minimizing the overall has cost of given task allocation problems.

Due to some difficulties of procurement and execution of physical robots, agents are reckoned as real robots and experiments are carried out. Some of the most known MRTA techniques have been implemented at JADE [2]. A framework has been built similar to Contract Net Protocol (CNP) [3], MURDOCH [4] and TraderBots [5]. Despite the similarities between multi-agent and multi-robot systems, the transition from agents to robots is not straightforward. Some assumptions are made under some constraints. Optimization problem is focused instead of localization, mapping, sensory or other multi-robot concerns.

To summarize the contributions, this study has three main challenges: An agent-based system development study specified to multi-robot domain, a practical taxonomy compilation and an optimization study that minimizes overall cost by using hybrid techniques.

Section 2 of this study summarizes related literature. At section 3 it is told why JADE is chosen, how it is used and development of agent-based systems. Section 4 defines the MRTA problem and provides proposed model with comparisons to similar models including a novel taxonomy. Results on the experiments are discussed in Section 5 and study is concluded in Section 6.
2 Related Work

For the last decade, the number MRTA studies have been increasing year by year in order to handle the growing demands. Unmanned air and underwater vehicles, exploration, Robocup, security, disaster response, intelligent house, assembly lines and factories are some of the recent examples employed MRTA principles. MRTA studies have been classified according to autonomy level.

Centralized approaches employ optimal assignment and Hungarian algorithm [6][7]. The Hungarian algorithm allows a "minimum matching" to be found. This can be used in instances where there are multiple quotes for a group of activities and each activity must be done by a different agent, to find the minimum cost to complete all of the activities. This method helps finding optimum solution but as activities and agents are increased, it turns to be solved in a long time (O (n^3)). The Hungarian algorithm is provably complete and optimal. A central planner is employed which has all the information about the environment (robots, tasks, map and the obstacles). Planner makes calculations and assigns tasks to each robot. Task execution agenda of each robot can be calculated and sent by central planner as well. At this type of autonomy level; robot does exactly whatever planner wants. On the other hand, distributed approaches give all decision responsibility to the robots. To give an example, some models are based on emergent behaviour like swarm systems which robots don't work explicitly together [8]. Robots don't need to obey a planner, they use their nearby information. Local optima can be succeeded but it is hard to reach global optimum.

Some models depend on intentional behaviour which robots cooperate for a common purpose [1]. Intentional task allocation can be executed either auction-based [4] or market-based [5]. The nuance between both is represented at Fig. 1. Planner can be a central agent like a computer or it can be any of the robots.

![Fig. 1 Intentional task allocation; a) auction-based b) market-based](image)

Examples to intentional task allocation have been increasingly appeared since the end of 90's. BLE (Broadcast of Local Eligibility) system uses a Publish/Subscribe method to allocate tasks that are hierarchically distributed [9]. M+ proposes a scheme for negotiated task allocation, it allows for dynamic task reallocation of subcomponents of complex tasks [10]. ALLIANCE is developed for robust, fault tolerant, fully distributed heterogeneous multi-robot systems. As there is no negotiation process, communication cost is low [1]. Gerkey and Mataric published a taxonomy study and presented a framework based on Publish/Subcribe and first price auction mechanism, MURDOCH. According to taxonomy, robots are classified as single-task robots (ST) and multi-task robots (MT) and tasks are classified as single-robot tasks (SR) and multi-robot tasks (MR) [4][11]. TraderBots is the first introduced market-based MRTA study in the literature. Robots exchange their tasks if they negotiate. An individual seeks for its own benefit and then optimum result is reached. Market economy principles can be applied to multi robot systems according to this approach [5]. The most comprehensive study on task structures are held by Zlot which deals with complex tasks built on task trees and how to decompose them [12]. As an extension to TraderBots, Kalra proposed HOPTILES for tightly coupled robots [13]. Aforementioned approaches are based on single robot tasks; however some researchers attacked multi-robot task problems. Vig and Adams coined the term "multi robot coalition formation (MRCF)" [14]. MRCF is allocating multi-robot teams to different tasks (ST-MR problem). Tang and Parker proposed a model named ASyMTRe that enables switching between single robot tasks and multi-robot tasks dynamically [15]. Hasgul et al. applied project management principles for task scheduling to multi-robots successfully [16]. Kaleci et al. combines market-based approach and optimal assignment [17].

This study is focused especially on CNP, MURDOCH and TraderBots among all these approaches. CNP is also a de facto protocol of FIPA [18] and closely related to agent world. MURDOCH is examined as an auction based model and TraderBots as a market based model.

3 Multi-Agent Systems and JADE

Multi-robot and multi-agent research areas include similar concepts, sometimes robot experiments are performed in agent domain. Therefore at the level of explicit coordination among multiple individuals, the differences in techniques used by multi-agent and multi-robot systems are in fact very few [4]. Although robotics researchers employ sophisticated techniques while designing single robot control systems, they have tended to use techniques that are already well known in the agent community when designing explicitly coordinated multi-robot systems. Although it has some constraints and fidelity problems, agent are advantageous at developing large teams of robots, scalability, cost, development time, demonstration, flexibility. One can easily develop a robot-like agent and integrate it any agent system that is FIPA-compliant. There are several agent development environments which can be classified according to their purpose. RETSINA [19], SEAGENT [20] and JADE [2] are presented as general purpose ones whereas CONSENSUS [21] and iJADE [22] are developed for a specific purpose.

JADE is selected as an agent development environment among several agent development framework alternatives taking into below factors account:
- FIPA compliance
- Development environment (Eclipse support)
- Comprehensive documentary
- Technical support and live forum
- Amount of developers worldwide

Messaging system and behaviours of JADE are frequently used in this study. It is quite easy to send and receive messages. Messages are queued in a message list and executed by order. JADE encapsulates developer from the complexity of communication; there is no need for an expertise at TCP-IP or sockets. JADE behaviours are easily developed as well. JADE isolates developers from parallel programming and threading issues by bringing behaviours into use. Examples to some JADE behaviour are OneShot, Cyclic, Ticker, Waker, Simple, Sequential, and Parallel. The only requirement to use behaviours efficiently is to build a feasible schedule. Activity design is an important phase of agent development. Agents can be initiated manually by executing its code, or dynamically by calling from another agent. JADE agents are FIPA-compliant, to this end, these agents provide interoperability. It is possible to join a JADE agent container using not only computers but also mobile phones or tablets from any geographical location that has internet connection. Four types of agents have been developed for this study:

Robot Agent:

Robot Agent behaves like a autonomous real robot. It senses, decides and actuates. Obstacle sensing and avoidance have not been implemented yet, but robots are sensible to battery level and task announcements. For example, when battery level is low, it cannot bid to any tasks, it looks for a suitable charge station and goes to charging. Robot Agent is informed about map of the simulation world including walls, other robots and obstacles.

Auctioneer Agent:

Auctioneer Agent has a planner role. It is informed about all the tasks, charge stations and robots. When a robot is added to the system, Robot sends acknowledgement message to Auctioneer Agent in order to introduce itself. Auctioneer Agent is also responsible to help robots to find suitable charge station, to perform centralized optimal assignment and to inform graphical user interface agent.

Terrain Agent:

Terrain Agent shows the terrain, tasks and robots on a graphical user interface. Robots send their momentary information (location, budget, battery level, completed tasks, charge count) with messages where Auctioneer Agent sends task and charge station information.

SimManager Agent:

SimManager Agent is responsible for selecting terrain model, defining robots and tasks, Travelling Salesman Problem (TSP) type (tour or path), random seed for task creation and execution approach (Centralized, auction-based or market-based). Some information is read from configuration files. Creation and initialization of all other agents are triggered by SimManager Agent.

4 The Problem and Proposed Work

One of the optimization objectives at multi-robot task allocation is minimization of the cost, in other words maximizing the throughput as a definition used in industrial, manufacturing processes [23]. This study focuses on cost optimization of Multi-depot Travelling Salesman Path Problem (MD-TSPP) [5]. Differences of this problem from the well-known TSP are listed below:

a) There is only one salesman at TSP. MD-TSPP includes more than one salesman departing from different depots.
b) The initial city can be indefinite at TSP where departing depots of salesmen must be known at MD-TSPP.
c) TSP is often considered as a tour that salesman must return to its initial location. MD-TSPP is considered as a path, salesman stays at the last city visited waiting for a new task. (Tour means that the salesman must return to the starting city. Path means that the salesman will not return to the starting city, but just visiting each city once and only once.)

In this study, salesmen are the robots and tasks are the cities that robots must visit. The distance between two cities is the cost of travelling between two tasks. If a robot visits a city, that task is marked as completed. If all the tasks are completed by the robots, overall goal is accomplished. When the robots are not executing tasks, they remain stationary at their current locations.

Let \( c_{ij} \) be the cost for the \( j^{th} \) robot to visit the \( i^{th} \) city from the \((i-1)^{th} \) city in its tour (where the \( 0^{th} \) city is the starting location). The robot cost function for the \( j^{th} \) robot is computed as follows:

\[
rcost (j) = \sum_{i=1}^{n_j} c_{ij}
\]  

(1)

Where \( n_j \) is the number of cities in the tour for robot \( j \).

The team cost function is:

\[
tcost = \sum_{j=1}^{m} rcost (j)
\]  

(2)

where \( m \) is the number of robots [5].

This study differs from TraderBots that TraderBots approach condones the multiple visits for a city by a robot or different robots which means the global task is complete when all cities are visited by at least one robot. The approach in this study permits only one visit for each task considering that it would help minimizing the cost.

Another difference from TraderBots lays at the robot's situation when all of its tasks are completed. Robot makes a route plan when execution starts and this plan is dependent on its tasks. If it is wanted to return to its initial location, it must add that to its own route plan as an additional task.
The problem becomes more complicated if tasks are dynamically introduced into the system and the robots must reconfigure to execute the new tasks in real time.

Fig. 2 Task allocation steps

Task allocation steps are examined in depth according to several nuances.

(A) INITIAL TASK ASSIGNMENTS

Once the system is started, tasks are assumed as unassigned. For a fast start-up, tasks can be allocated to the robots in different ways. Most inconvenient but easy way is random allocation. This technique distracts reaching optimum solution. Greedy assignment is a feasible and practical solution that can fail in some circumstances. Optimal assignment is an expensive solution for initial assignments if followed with a trade process, but it eases finding better results. PRIM allocation, on the other hand, is proposed as an alternative to combinatorial assignment and it has the best score at cost/performance [24]. Random and greedy assignments may be preferred when large number of robots and tasks are used.

If random assignment is selected Tasks are assigned to robots using random number generator. For the other assignment types, following procedure is applied:

At the moment simulation is started, a central planner (Auctioneer Agent) has all the tasks. It announces all the tasks to all the robots. Robot agents bid for each task. Bids are calculated according to metrics, in this study the only metric is distance (task to robot). Each robot is equipped with a map of the world, which enables it to calculate the cost associating with visiting each of the tasks. Thus, the robots bid for each task based on their estimated costs to visit that city.

Distance is computed according to Euclidian distance. If \( \mathbf{u} = (x_1,y_1) \) and \( \mathbf{v} = (x_2,y_2) \) are two points on the plane, their Euclidian distance is given by

\[
d(u,v) = \sqrt{(x_1-x_2)^2 + (y_1-y_2)^2}
\]  

Robot bids are recorded into a proposal matrix. When all the proposals are delivered to the Auctioneer Agent, it executes selected assignment algorithm ( \( m \): number of robots).

i. **Greedy**: Best matching \( m \) tasks are assigned to robots (one task for each robot). Shortly; \( m \) tasks are assigned to the their closest robots.

ii. **PRIM**: Best task is assigned to the robot with best offer. Shortly, a task that has lowest cumulative distance to the robots is assigned to the closest robot. Only one task is assigned at each iteration.

iii. **Optimal**: Hungarian Algorithm is executed to make task assignments.

As soon as first assignment is completed, second assignment process (announcement + bidding + assignment) is executed. Assignments are done iteratively until all the tasks are owned by the robots. At each iteration, proposal matrix is cleared in order to catch changing conditions.

(B) TASK EXCHANGE BETWEEN ROBOT (TRADE PROCESS)

Initial task assignment is followed by a trading phase at market-based task allocation. But this process can be bypassed through task execution phase.

**Role Assignment**

Robots can bid for tasks, can announce tasks to sell or can do both of these at the same time concurrently. If all the robots try to sell all of their tasks at the same time, a complicated situation may be emerged.

After revising several studies, sequential objective-based seller robot employment model is preferred. At this model a robot is selected as a seller and others are informed to wait task offers. Robots define their most unsuitable tasks. Unsuitable task is worst task for the robot according to the cost. Robots ask themselves the question “How much my way is shortened if I get rid of this task?” for each task and the answer is sent to Auctioneer Agent. Auctioneer Agent assigns the robot which is candidate for maximum cost reduction as seller robot for the current trade.

**Auction Type**

Seller robots can announce for only one task, more than one task or combinatorial task bundles at each trade. Combinatorial trade requires much processor power, because all the task pair combinations should be taken into account [25].

In this study, single item-objective based method has been used which is related to proposed seller robot selection technique; unsuitable task is announced.

**Bid Valuation and Auction Clearance**

At a market, one has to protect its own profit. Each trade should satisfy both buyers and sellers, bids should be efficient. Bid can be calculated simply as Euclidian distance between the robot and the task. If number of tasks for a robot is lower than 20, TSP calculation can be used. Seller robot announces the task with the price as cost reduction assuming the task is
sold. Candidate robots calculate their cost increment assuming the task is bought. If amount of increment is lower than announced price, candidate robot bids for that. Fig. 3 illustrates that Euclidian distance calculation fails at some situations. WP2 (Way Point 2) is very close to Robot 2 but it is not reachable easily. To this end, TSP approach is suitable for this case.

There are several bid valuation strategies in the literature as well. Joundeau et al. Used wavefront calculation [26]. Kaleci et al. has taken dynamical and sensor models of robots into account [17]. Auction usually results as selecting best bid owner as the winner. Accepting incoming bids and transition to auction clearance last until all bids are OK or until time is out.

**Termination of Trading Phase**

When market reaches to stability and there exists no other tasks to trade, robots should begin execution as soon as possible. Whether the system is stable or not can be decided after searching trade situations. If there is no trade executed for while, it means that there is no profit for a robot to buy or sell any tasks anymore.

**C** **T** **A** **S** **K** **E** **X** **E** **C** **U** **T** **I** **O** **N**

After all auctions and task exchanges between robots end, each robot has a task list to do. Tasks are executed one by one according to one of the below ways:

i. **Next task from the list**: Robot executes the task list from the beginning to the end without sorting.

ii. **Nearest task**: Robot executes the nearest task to its current location.

iii. **Schedule**: Robot creates a plan calculating either TSP path or TSP tour.

Some limitations and assumptions must be considered as well. Each items in the below list complicates the solution a bit more.

- Battery life, locations of charge stations and charging time
- Unknown or newly introduced tasks
- Unknown or newly introduced robots
- Failure of a robot or lost communication
- Obstacle avoidance

**5 Experiments and Results**

Proposed model is applied into a developed agent based simulation environment. This software is convenient to perform experiments with below variables:

- Terrain model (dimensions, walls, obstacles)
- Number of robots
- Number of tasks
- TSP type (tour or path)
- Bid Valuation (Euclidian or TSP)
- Intention type (With or without trade)

Examples of well-known MRTA studies in the literature are applied and results have shown that proposed model always reaches the optimal solution. Hungarian Algorithm is used for initial task assignments for all the experiments presented in this section. After initial assignments, four types of experiment configurations established:

- TSP tour with trade
- TSP tour without a trade
- TSP path with trade
- TSP path without a trade

The number at the mid-bottom of each figure represents the final total cost for the related experiment setup. Fig. 4 represents four snapshots of Zlot example therefore 470 is the total cost of the simulation of Fig4.a. Money unit in these trade examples is pixels. Dimension of each map is 400 * 300 pixels. There are three robots and 5 tasks to do for Zlot example. It can be seen that trading is more advantageous at TSP tour problems.

**Fig. 4 Snapshots from the simulation: Zlot example [12]**
(a- TSP Path without a trade, b- TSP Path after trade, c- TSP tour without a trade, d- TSP tour after trade)
Fig. 5 illustrates an example from TraderBots warehouses. Robots exchange their tasks in order to achieve optimality. Although initial task assignments here are different from TraderBots, result is the same, optimal solution. This means that, proposed model fits for all similar problems.

Table 2. Cost savings for examples

<table>
<thead>
<tr>
<th>Example</th>
<th>Without trade</th>
<th>With trade</th>
<th>Saved (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Zlot – TSP Path</td>
<td>470</td>
<td>458</td>
<td>3</td>
</tr>
<tr>
<td>Zlot – TSP Tour</td>
<td>882</td>
<td>743</td>
<td>18</td>
</tr>
<tr>
<td>TraderBots</td>
<td>2146</td>
<td>1624</td>
<td>24</td>
</tr>
<tr>
<td>PRIM</td>
<td>252</td>
<td>146</td>
<td>42</td>
</tr>
</tbody>
</table>

Table 2 lists cost saving for each example experiment. It can be said that benefits of task exchange between robots incredibly increase when a) there are some obstacles in the area b) TSP tour approach is used. Also some experiments are performed with randomly distributed tasks and robots in an empty area (no obstacle).

As a final example, two rooms example can be shown (Fig. 6). It can be seen clearly from the figure what if market-based approach is not selected. Robot1 and Robot2 stay at their own rooms and they save 35% according to the total cost of their initial task assignments executed.

Experiment results clearly show that the combination of trade based approach with transition method between different regions using TSP gives promising results. Transition method should be improved by using legacy and novel motion planning techniques.

6 Conclusions

As a result of the focus on multi-robot systems, multi-robot coordination has received significant attention. In particular, MRTA has recently risen to prominence. The complexity of the multi-robot systems being studied has increased.

An agent-based MRTA model has been presented which is a hybrid approach combines trade-based task allocation, optimal assignment and TSP solution. This model produces more optimal results and suitable for larger teams and task collections as well. A simple way to select most unsuitable task for the robot has been presented.

Among several alternatives, JADE is chosen as development environment. CNP, MURDOCH and TraderBots principles are applied to the system. Tackling with delayed messages and conflicting behaviours, it is possible to implement a real-like multi-robot system with JADE. In this study, optimization strategy has been focused on instead of specific nature of multi-robot domain like sensory system, localization, mapping and etc.

Taxonomy is compiled including literature research and proposed techniques. Example cases of well-known multi-robot research are tested on the developed framework and it is seen that framework gives optimal results. Proposed model fits for all similar problems. Future work will include combining trading and execution phases to work concurrently and developing a fault-tolerant mixed environment (indoor/outdoor) multi-robot system. Improvement at efficient bidding and assignment strategies specialized to a warehouse system will be another research objective.
7 Acknowledgement

The authors would like to thank Software Testing and Quality Evaluation Center (YTKDM in Turkish) of Scientific and Technological Research Council of Turkey (TUBITAK in Turkish) for funding this study.

8 References