An Auction-Based Task Allocation Algorithm in Heterogeneous Multi-Robot System

Jieke Shi, Zhou Yang, and Junwu Zhu

1 Introduction

With the fast development of technologies, robots are applied to more and more field. However, in terms of current robotics development, the single-robot system has limitations in getting information and solving problems. What's more, nowadays the environments faced by the robots are always dynamic, real-time, complex adversarial, and stochastic. Facing such complex tasks and the changing working environment, it is hard for a single-robot system to perform operations.

Compared to a single-robot system, a multi-robot system has the features about the distribution of time, space, information, and resources. In order for multiple robots to cooperate efficiently, the key problem we need to research on is how to allocate tasks properly according to working environment and situations. This is a fundamental problem in robots system researchers, also called multi-robot task allocation, MRTA.

When the number and states of the robots are clear, MRTA problem can be thought of as an assignment problem if the number and states of tasks are also clear. Usually, an accomplished task can provide some payoff to the system. In some special working environment, such as the robot rescue tasks, accomplishing a task will not provide the system with any positive payoff. But if the task is always delayed, it will cause damage to the system continuously.

J. Shi · Z. Yang
College of Information Engineering, Yangzhou University, Yangzhou, Jiangsu, China

J. Zhu (✉)
College of Information Engineering, Yangzhou University, Yangzhou, Jiangsu, China

Department of Computer Science, University of Guelph, Guelph, ON, Canada
e-mail: jwzhu@yzu.edu.cn

H. Lu, L. Yujie (eds.), *2nd EAI International Conference on Robotic Sensor Networks*, EAI/Springer Innovations in Communication and Computing,
https://doi.org/10.1007/978-3-030-17763-8_14

In this paper, we focus on task allocation problem in rescuing environment. So our aim is to minimize the whole damage to the system caused by sudden catastrophic events.

In dynamic, changing, and stochastic working environments, a successful task scheduling strategy fulfills the following two points:

– *Optimization*: Finding a nearly optimized task allocation scheme within a certain time constraint.
– *Dynamic*: Ensuring the efficiency during the whole working time. In other words, transferring the allocation outcome properly according to the changing of the working environment.

In this paper, in response to the above two requirements, based on auction, we propose a merging method between centralized and distributed methods.

We make three contributions in this work. Firstly, we propose a dynamic auction method for differentiated tasks under cost rigidities, DAMCR. When the number of tasks waiting to be assigned is up to some certain threshold value, DAMCR is activated. Secondly, we prove that DAMCR is ϵ-optimal. Finally, we design detailed experiments and demonstrate the results to analyze our method.

2 Related Work

In the field of multi-robot system, we can use the dual programming representation process for solving decision problems [1]. The traditional dual programming algorithm is generally based on the global static environment known and determined, but the results of task allocation cannot necessarily achieve the goals in the ideal time [5]. In order to improve the efficiency of search and rescue (SAR) in disaster relief, some researchers used auction-based task allocation scheme to develop a cooperative rescue plan [2, 10].

As to dealing with the problem of robot collaboration in the distributed environment, auction algorithms are a feasible method for task assignment that have been shown to efficiently produce suboptimal solutions [4]. The traditional way of computing the winner is to have a central system acting as the auctioneer to receive and evaluate each bid in the fleet. Once all of the bids have been collected, a winner is selected based on a predefined scoring metric [9]. Considering multiple resources of the robots and limited robot communication range, Lu [8] applied the idea of second-price auction to determine the final price and the number of provisioned VMs in the double auction.

The downside of these approaches is that the bids from each agent must somehow be transmitted to the auctioneer [7, 11]. In the rescue robot system there exist a number of rescue robots, the robots need to complete the corresponding task to ensure the system loss minimum. In order to obtain an ideal task allocation scheme, we must take full account of the benefits and costs of completing the task [3]. A common method to avoid communication limitation is to sacrifice mission performance by running the auction solely within the set of direct neighbors of the auctioneer [6].

3 Model Description and Notations

3.1 Definitions of Notations and Model

The dynamics of the environment are mainly reflected in the changes in some factors of the work environment as time progresses. We define E_t to represent the multi-robot system work environment at time t. Mathematically speaking, E_t can be formalized as triples:

$$E_t =< R, T_1, T_2 > \tag{1}$$

R is the set of all the robots in this system, it can be formalized as below:

$$R =< r_1, r_2, \ldots, r_N > \tag{2}$$

r_i is the ith robot in the system. $N = |R|$ represents the total number of robots. A robot r_i can be described as below:

$$r_i =< robID_i, state_i, place_i, t > \tag{3}$$

In this formula, $robID_i$ uniquely identifies a robot. $place_i$ represents the robot's location. $state_i$ indicates the status of the robot r_i, whose value rules are demonstrated below:

$$state_i = \begin{cases} 0, & \text{if the robot is free;} \\ taskID_j, & \text{if } task_j \text{ is assigned to it.} \end{cases} \tag{4}$$

In formula (1), T_1 is the set of tasks that have been allocated while T_2 is the set of the tasks waiting to be assigned. Apparently, the set T consisting of all the tasks in the system is $T = T_1 \cup T_2$. An element in those sets, in other words, a task

$$task_j =< taskID_j, state_j, place_j, C_j(AR_j^t), t > \tag{5}$$

$taskID_j$ uniquely identifies a task. $place_j$ represents the robots location. Similar to a Cartesian coordinate system, we use (x_j, y_j) to represent $place_i$. $state_i$ indicates the status of the robot t_j, whose value rules are demonstrated below:

$$state_j = \begin{cases} 0, & \text{if the task is not assigned;} \\ taskID_j, & \text{if } robID \text{ takes care of it.} \end{cases} \tag{6}$$

We need to point out that AR_j^t is the accomplishment rate of $task_j$ at time t. Clearly, $AR_j^t \in [0, 1]$. We also define a new parameter called damaging rate function, labeled with $C_j(AR_j^t)$. It represents how much damages $task_j$ will cause to the system in per unit of time when the accomplishment rate equals to AR_j^t.

We use auction as a basic method to allocate tasks, a formalized auction can be represented as below:

$$A =< B, T > \tag{7}$$

B is the set of bidders, in other words, robots waiting for tasks. T is the set of tasks waiting to be assigned in this auction.

For a certain task t_j, bidder b_i will calculate two parameters: t_{ij}^1 and t_{ij}^2. t_{ij}^1 represents the time it takes for b_i to go to task t_j. t_{ij}^2 represents the time it takes for b_i from starting working on t_j to accomplishing it. So far, if t_j is assigned to b_i at time t_0, then under such allocation result, the damage that $task_j$ will cause to the system can be formalized as below:

$$cost_i^j = \int_{t_0}^{t_0+t_1^{ij}} C_j(0)dt + \int_{t_0+t_{ij}^1}^{t_0+t_{ij}^1+t_{ij}^2} C_j(AR_j^t)dt \tag{8}$$

4 Static Auction Algorithm Description

In this part, we consider one to one auction model of static task allocation with N robots matching N rescue tasks. The assignment problem to be discussed here is the mathematical problem of minimizing the damage of the whole system by matching N robots with N tasks. Constraint conditions of the algorithm can be expressed in a mathematical formula as below:

$$\min \quad \sum_{i \in B} \sum_{j \in T} x_{ij} C_{ij} \tag{9}$$

$$\text{s.t.} \quad \sum_{j|(i,j) \in T} x_{ij} = 1, \forall i = 1, 2, \ldots, n \tag{10}$$

$$\sum_{i|(i,j) \in T} x_{ij} = 1, \forall j = 1, 2, \ldots, n \tag{11}$$

$$x_{ij} = 0, 1, \forall (i, j) \in T \tag{12}$$

In this format group above, we use set T to represent the tuples of all possible distributions. It can be formalized as below:

$$T = \{(i, j) | j \in T(i), \forall i = 1, 2, \ldots, n\} \tag{13}$$

We use $T(i)$ to represent the set of all tasks that can be assigned to the robot i. In order to reduce the algorithm complexity, we use $\bar{t}$ to set a time limit, if the total time of robot i solving task j is more than t, task j cannot be assigned to robot i, which can be described as below:

$$T(i) = \{j | t_{ij} \leq \bar{t}\} \tag{14}$$

We use the set A to represent the two-tuples (i, j) consisting of robots and tasks. Each robot can have one two-tuples $(i, j) \in A$ at most. Each robot can have one two-tuples $(i, j) \in A$ at most either. For the set A, if there is a two-tuples$(i, j) \in A$, it means that task j is assigned to robot i.

In the algorithm, we set a positive value called ϵ and a price set $p = \{p_1, \ldots, p_n\}$. For robot i, if the difference between its absolute value of the relative gains obtained from task j and the optimal relative gains obtained from all the allocation schemes is not greater than ϵ, we call robot i and task j satisfy complementary slackness condition. This two-tuples (i, j) is the optimal result, which can be described as below:

$$|p_j - C_{ij} - \max_{k \in T(i)} \{p_k - C_{ik}\}| \leq \epsilon \tag{15}$$

The specific process of auction algorithm is described as follows.

Step 1 Select a $\epsilon > 0$, set $p_k = 0, \forall k = 1, 2, \ldots, n$, the set of robots that are not assigned tasks, is denoted as N, and the set of robots which are tender to the task j in the bidding phase is denoted as $B(j)$;

Step 2 This step is an iterative process:

– *Decision phase*: For each robot i in the set N, get the maximum relative gains u_i and the assigned task j_i when the maximum relative gain is obtained:

$$u_i = \max_{k \in T(i)} \{p_k - C_{ik}\} \tag{16}$$

And its second relative gains v_i:

$$v_i = \max_{k \in T(i), k \neq j} \{p_k - C_{ik}\} \tag{17}$$

If $T(i)$ has only one task j, we define v_i as $-\infty$.

– *Bidding phase*: All robots bid for j_i which is the most gainful task, the bidding price of the robot is determined as:

$$a_{ij_i} = p_{j_i} - u_i + v_i - \epsilon = C_{ij_i} + v_i - \epsilon \tag{18}$$

– *Allocation phase*: For each task j, if the $B(j)$ is not empty, we update the price of the task j to the highest bid price:

$$p_j = \max_{i \in B(j)} a_{ij} \tag{19}$$

The task is assigned to the highest bidding robot i_j, and at the same time, the two-tuples related to the robot i_j and the task j are removed from the A which is an infeasible allocation, and the new two-tuples (i_j, j) are added to the set A.

In this iterative process of the algorithm, we use p_j and p'_j, respectively, to represent the price corresponding to the task before and after the iteration. In the iterative process, if robot i bids to the task j_i and is successfully assigned to the task j_i, its price will be updated, which can be described as follows:

$$p'_{j_i} = C_{ij_i} + v_i - \epsilon \tag{20}$$

For each task j, there are $p_j \geq p'_{j_i}$, so we can get the format below:

$$|p'_{j_i} - C_{ij_i} - \max_{k \in T(i), k \neq j} \{p_k - C_{ik}\}| \leq \epsilon \tag{21}$$

It can be seen that after every iteration process, every two-tuples (i, j) always satisfies the complementary slackness condition.

5 Experimental Results and Future Work

In this paper, we propose a new auction model and use the auction algorithm to study multi-robot task allocation problem. Through experiments, we find that the task allocation can be effectively carried out by our algorithm.

We choose the classic Hungarian algorithm to do contrastive experiments. For all cost matrices, the same optimal assignment results can be obtained by iteration. The time spent on auction algorithm and Hungarian algorithm is as follows (see Fig. 1).

On the different scale of task allocation, the speed advantage of auction algorithm is obvious. The number of tasks has a greater impact on the running time of the Hungarian algorithm than the algorithm we proposed.

Due to the limited ability of the author, there are many shortcomings in this paper. On the one hand, new tasks may occur during the execution of a task, robots may need to stop their current work and carry out new tasks. On the other hand, a mechanism must be added to the auction algorithm to find out the infeasibility of the problem when the problem is insoluble. All of these are discussed in the future work.

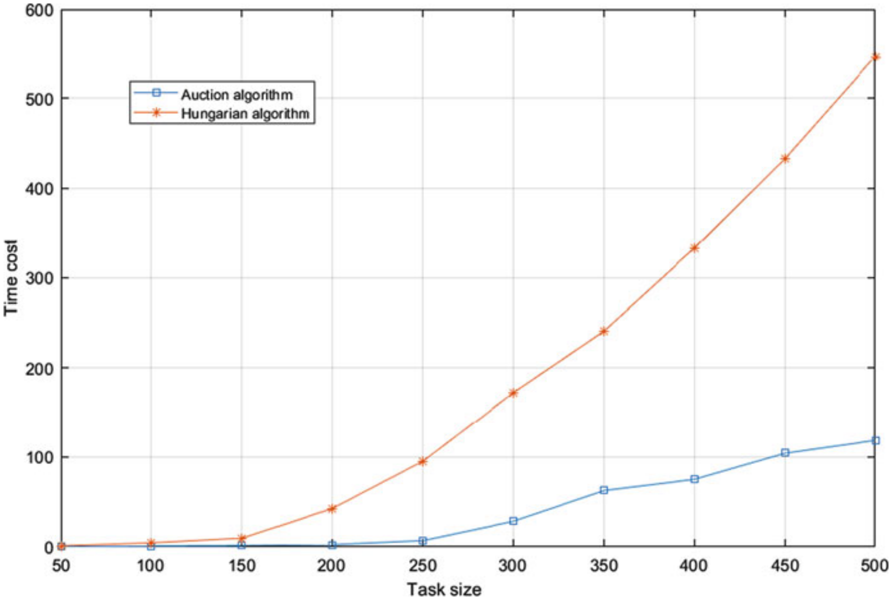

Fig. 1 Time cost comparison between auction algorithm and Hungarian algorithm

Acknowledgements This work was supported by the National Nature Science Foundation of China under Grant 61170201, Grant 61070133, and Grant 61472344, in part by the Innovation Foundation for graduate students of Jiangsu Province under Grant CXLX12 0916, in part by the Natural Science Foundation of the Jiangsu Higher Education Institutions under Grant 14KJB520041, in part by the Advanced Joint Research Project of Technology Department of Jiangsu Province under Grant BY201506106 and Grant BY2015061-08, and in part by the Yangzhou Science and Technology under Grant YZ2017288 and Yangzhou University Jiangdu High-end Equipment Engineering Technology Research Institute Open Project under Grant YDJD201707.

References

1. Booth, K. E., Nejat, G., & Beck, J. C. (2016). A constraint programming approach to multi-robot task allocation and scheduling in retirement homes. In *International Conference on Principles and Practice of Constraint Programming* (pp. 539–555). Berlin: Springer.
2. Das, G. P., McGinnity, T. M., Coleman, S. A., & Behera, L. (2015). A distributed task allocation algorithm for a multi-robot system in healthcare facilities. *Journal of Intelligent & Robotic Systems, 80*(1), 33–58.
3. Elango, M., Nachiappan, S., & Tiwari, M. K. (2011). Balancing task allocation in multi-robot systems using k-means clustering and auction based mechanisms. *Expert Systems with Applications, 38*(6), 6486–6491.
4. Garg, R., & Kapoor, S. (2006). Auction algorithms for market equilibrium. *Mathematics of Operations Research, 31*(4), 714–729.

5. Hooshangi, N., & Alesheikh, A. A. (2017). Agent-based task allocation under uncertainties in disaster environments: An approach to interval uncertainty. *International Journal of Disaster Risk Reduction, 24*, 160–171.
6. Jiang, L., & Zhang, R. (2011). An autonomous task allocation for multi-robot system. *Journal of Computational Information Systems, 7*(11), 3747–3753.
7. Liu, Y., Yang, J., Zheng, Y., Wu, Z., & Yao, M. (2013). Multi-robot coordination in complex environment with task and communication constraints. *International Journal of Advanced Robotic Systems, 10*(5), 229.
8. Lu, L., Yu, J., Zhu, Y., & Li, M. (2018). A double auction mechanism to bridge users' task requirements and providers' resources in two-sided cloud markets. *IEEE Transactions on Parallel and Distributed Systems, 29*(4), 720–733.
9. Ponda, S. S., Johnson, L. B., & How, J. P. (2012). Distributed chance-constrained task allocation for autonomous multi-agent teams. In: *American Control Conference (ACC), 2012* (pp. 4528–4533). Piscataway: IEEE.
10. Tang, J., Zhu, K., Guo, H., Gong, C., Liao, C., & Zhang, S. (2018). Using auction-based task allocation scheme for simulation optimization of search and rescue in disaster relief. *Simulation Modelling Practice and Theory, 82*, 132–146
11. Wang, L., Liu, M., & Meng, M. Q. H. (2017). A hierarchical auction-based mechanism for real-time resource allocation in cloud robotic systems. *IEEE Transactions on Cybernetics, 47*(2), 473–484.