

Research Article

A Balanced Heuristic Mechanism for Multirobot Task Allocation of Intelligent Warehouses

Luwei Zhou, Yuanyuan Shi, Jiangliu Wang, and Pei Yang

Department of Control and System Engineering, School of Management and Engineering, Nanjing University, Nanjing 210093, China

Correspondence should be addressed to Pei Yang; yangpei@nju.edu.cn

Received 19 August 2014; Revised 21 October 2014; Accepted 21 October 2014; Published 11 November 2014

Academic Editor: Hari M. Srivastava

Copyright © 2014 Luwei Zhou et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper presents a new mechanism for the multirobot task allocation problem in intelligent warehouses, where a team of mobile robots are expected to efficiently transport a number of given objects. We model the system with unknown task cost and the objective is twofold, that is, equally allocating the workload as well as minimizing the travel cost. A balanced heuristic mechanism (BHM) is proposed to achieve this goal. We raised two improved task allocation methods by applying this mechanism to the auction and clustering strategies, respectively. The results of simulated experiments demonstrate the success of the proposed approach regarding increasing the utilization of the robots as well as the efficiency of the whole warehouse system (by 5~15%). In addition, the influence of the coefficient α in the BHM is well-studied. Typically, this coefficient is set between 0.7~0.9 to achieve good system performance.

1. Introduction

Autonomous guided vehicles (AGVs) have been used to perform tasks in warehouses for more than fifty years [1, 2]. In the past few years, they have been mostly used to transport large or heavy objects such as rolls of uncut paper or engine blocks. Recently, as autonomous robots [3–5] become cheaper, smaller, and more capable, they are widely used in the logistic industry, for example, being substituted for the manual labor in the typical pick-pack-and-ship process in warehouses [6]. Moreover, in [7, 8], it is pointed out that the multirobot coordination, if being really efficient and robust, will put significant impact on improving the logistical efficiency. The multirobot coordination involves the allocation and execution of individual tasks with an efficient mechanism. In this paper, we mainly focus on the task allocation process in which a team of autonomous robots must fulfill a set of orders in optimal routes meeting with certain criteria. This problem is also known as multirobot task allocation problem (MRTA) [9].

There are already some methods to deal with MRTA. In 1998, Yamauchi presented a greedy strategy [10] to coordinate robots, with which each robot chooses the closest task as soon as they finish their previous tasks. Gerkey and Mataric

introduced a dynamic task allocation method [11] for groups of autonomous robots, with which the robots bid on all unallocated targets, and the bids depend on the distance between their last targets and those unallocated targets. The algorithm proposed by Sandholm [12], known as the combinatorial auction method, is considered more efficient because it divides tasks into different clusters according to correlations between tasks; then robots bid on clusters of tasks [13]. In 2004, Lagoudakis et al. used Prim Allocation [14] to generate the MSF (minimum spanning forest) of tasks for robots and then transformed the forest in paths by the depth-first traversal. Later, some researchers [15, 16] improved the depth-first traversal process in the prim allocation method and better results are obtained.

As a matter of fact, most of current attempts focus on minimizing the travel cost of the robots group without concerning the travel time, that is, balancing the individual travel cost of each robot. There are few research works putting emphasis on improving the utilization of multirobots, especially in many time sensitive applications, such as the logistics industry, rescue, and exploration environments. Puig et al. [17] improved Yamauchi's greedy algorithm to solve the balancing-target allocation problem in the planetary exploration. By this method, after finishing its current task,

every robot moves to the frontier cell until all the tasks have been explored. However, since every robot only acts for itself with no coordination with other robots, this method may lead to a rather suboptimal solution in travel distance, which is not acceptable in our logistic application. In 2011, Elango et al. [18] proposed the K -means clustering and auction based mechanism (referred as KCAB mechanism) to balance the task allocation in multirobot systems. This approach considers the dual goal of both minimizing the travel distance and efficient sharing of the workload. Nevertheless, the complexity of this algorithm is relatively high and we will compare this method with our proposed approaches later.

Considering the specific characteristics of our problem, (i) balancing the task allocation as well as minimizing the travel distance, (ii) the cost of tasks are unknown, and (iii) large scale and dynamic environment, a balanced heuristic mechanism (BHM) is proposed to coordinate the autonomous robots and is applied to several traditional approaches, such as the auction allocation method and the K -means clustering method. To evaluate the performance of different strategies, three criteria are provided as the travel time (TT), coefficient of variation (CV) between different robots, and total travel cost (TTC). Simulation results strongly prove that the BHM has a better balancing performance compared with traditional methods.

The rest of this paper is organized as follows. In Section 2, we give the formal description and model of the order fulfillment process in the warehouse. In Section 3, we describe the overall control structure of the system and present the balanced heuristic mechanism (BHM). Then simulated experimental results are provided and compared with several traditional methods in Section 4.

2. Problem Formulation

2.1. Problem Statement. In this paper, we focus on the dynamic task allocation problem for multirobots in an intelligent warehouse. To characterize the system, some assumptions are listed as follows:

- (i) the system is composed of large-scale physical embedded robots;
- (ii) the robots and the tasks are homogenous;
- (iii) the environment of the system is known to each robot;
- (iv) each robot in the system is self-interested and fulfills its own tasks independently;
- (v) the robots may conflict with each other when fulfilling their tasks and they communicate with each other when being close to other vehicles (in purpose of obstacle avoidance);
- (vi) the time consuming at the station is the same for all robots.

Usually, the multirobot task allocation is described as follows: given a sequence of tasks and a set of robots, the goal is to assign tasks to robots in an efficient manner based on what the system is trying to optimize. In our approach, we strive to optimize the system in a dual objective function to

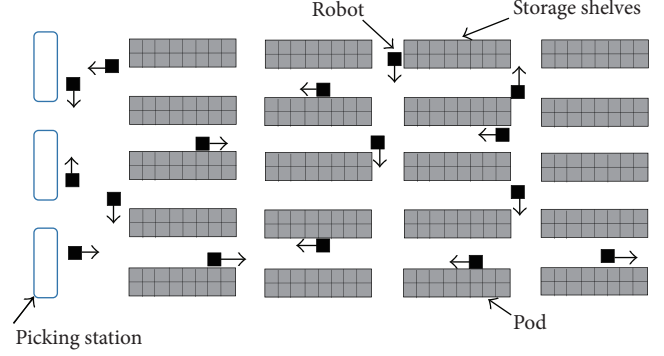


FIGURE 1: Configuration of a warehouse system.

keep the total travel cost as low as possible and keep the individual travel time as equal as possible, respectively. These two objectives correspond to a balance between time expenses and robot expenses, both of which deserve consideration in a concrete physical system. As shown in Figure 1, each storage shelf consists of several inventory pods and each pod consists of several resources. By order, a robot lifts and carries a pod at a time along a preplanned path, deliver it to the specific station which is appointed in the order and finally returns the pod back. The main consideration is how to allocate the tasks to the robots efficiently and equally while avoiding conflicts.

Another point that must be taken into account is that traditional task allocation methods are based on the assumption that the cost of each task is already known. However, in this problem, the input to the warehouse system is a sequence of orders, from which we can only know the location of the task and the specific station to which the inventory pod is supposed to be delivered. The cost of each task as well as the travel cost from one location to another is totally unknown.

2.2. Modelling. The model formulated to solve the balanced multirobot task allocation problem mentioned above in the warehouse system is as follows (as shown in Figure 2).

Let a set of robots $R = r_1, r_2, \dots, r_m$ to complete a set of tasks $T = t_1, t_2, \dots, t_n$. The cost of task t_i is w_i , which refers to the travel cost of carrying the inventory pod to the specific station and then delivering it back, ignoring the time cost of obstacle avoidance, and C_{ij} ($i, j \in R \cup T$) is the travel cost between every two locations (usually from the robot to the inventory pod assigned in the order). Suppose r_i have k tasks $T = t_{i1}, t_{i2}, \dots, t_{ik}$; then the total cost of all the assigned tasks in T_i is expressed as $W(r_i) = w_{i1} + w_{i2} + \dots + w_{ik}$. More details about the basic parameters and indexes using in the model are shown in Figure 2.

Rewrite T as $\Gamma = T_1, T_2, \dots, T_m$, which means a partition of the set of tasks where task set T_i are allocated to robot r_i . Then, define the individual travel cost as $ITC(r_i, T_i)$, which represents the sum cost for r_i to fulfill its task set T_i and can be calculated by

$$ITC(r_i, T_i) = C_{r_i, t_{i1}} + \sum_{j=1}^{k-1} C_{t_{ij}, t_{i(j+1)}} + W(r_i), \quad (1)$$

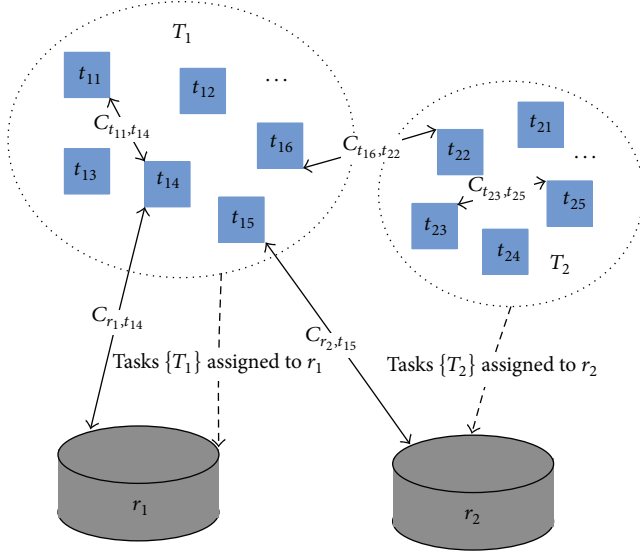


FIGURE 2: Basic parameters and indexes.

where $C_{r_i, t_{i1}}$ represents the travel cost for r_i to come to its first task t_{i1} , and $\sum_{j=1}^{k-1} C_{t_{ij}, t_{i(j+1)}}$ represents the total travel cost for the rest $k-1$ tasks. Also, $W(r_i)$ represents the total task cost of all the assigned tasks in T_i . Our dual goal can be expressed as

$$S = \min_{\Gamma} \max_i \text{ITC}(r_i, T_i), \quad (2)$$

which aims to minimize the travel cost as well as balances the individual travel cost of each robot.

Furthermore, we provide three criteria to evaluate the performance of different strategies: travel time (TT), which equals to the maximum of all the individual robots' travel costs; total travel cost (TTC), which reveals the total travel cost of all the robots, evaluates the power consumption and mechanical loss of robots; coefficient of variation (CV), which is a statistical measurement commonly used for comparing the diversity in work groups [19]:

$$\begin{aligned} \text{TT} &= \max_i \text{ITC}(r_i, T_i), \\ \text{TTC} &= \sum_{i=1}^m \text{ITC}(r_i, T_i), \\ \text{CV} &= \frac{\text{standard deviation}(\sigma)}{\text{mean}(\mu)}, \end{aligned} \quad (3)$$

where σ represents the standard deviation of $\text{ITC}(r_i, T_i)$ and μ is the mean of $\text{ITC}(r_i, T_i)$, $i = 1, 2, \dots, m$.

It is worth to point out that we use TTC to evaluate the power consumption of the system. On one hand, compared to the travel energy cost of the robots, other energy costs such as the communication cost and central controller cost are relatively small so that they can be neglected without affecting

the overall power consumption. On the other hand, $\sum_{i=1}^m W(r_i)$ (distance that robots running with pods) is constant for a system, which means the power consumption of this part is unchanged for a certain system. So we need to minimize $\sum_{i=1}^m (C_{r_i, t_{i1}} + \sum_{j=1}^{k-1} C_{t_{ij}, t_{i(j+1)}})$ (distance that robots running without pods) which can be fulfilled by minimizing TTC. Thus, TTC can be used to evaluate the power consumption of the system.

3. Method

In this section, we develop the BHM based approach for the task allocation of multirobots for the intelligent warehouse. We firstly provide an overview of the whole control system. Then the BHM principles are described in detail. By introducing this mechanism into the traditional auction allocation method and the K -means clustering method, two improved methods based on BHM are presented and analyzed for the task-allocation problem.

3.1. Overall System. The work space of the warehouse can be divided into several grids with inventory storage zones in the middle surrounding with the stations and workers. Autonomous robots transport movable inventory pods from storage locations to stations, where workers can pick items off and pack them up; then those packages are sent to the customers.

In technical aspects, the task allocation process is regulated by the central controller while the path planning and motion planning, for example, the obstacle avoidance, are fulfilled by robots. In the task allocation process, many current allocation mechanisms tend to use the dynamic method [20, 21]; however it may result in suboptimal performance. Thus, an approach is proposed combining both dynamic and static methods by means of the *task pool*. The continuously arriving orders accumulate in the task pool. The controller requires a certain number of tasks from the task pool after the previous batch of tasks has been completed, and uses a certain approach to allocate those tasks to robots. Then, the controller sends sets of tasks to corresponding robots by wireless communication such as ZigBee and the robots complete their tasks sequentially by a standard A^* algorithm. Usually, we use the A^* algorithm to find the shortest path between storage locations and stations in a static environment; while in an uncertain environment, the learning methods (e.g., reinforcement learning [22, 23]) may be necessary to find the shortest path, which will be our future work and is beyond the scope of this paper.

In addition, as for the obstacle avoidance process, robots firstly use infrared detectors to detect other robots within certain distance (usually several grids in front of them). Then, they communicate with each other by wireless devices so that they can verify their relative priorities. Generally speaking, the robot with more tasks left to be completed has priority to occupy front grids. Therefore, the robot with fewer tasks left is likely to give way to the robot with more tasks to be completed. So the actual balancing performance

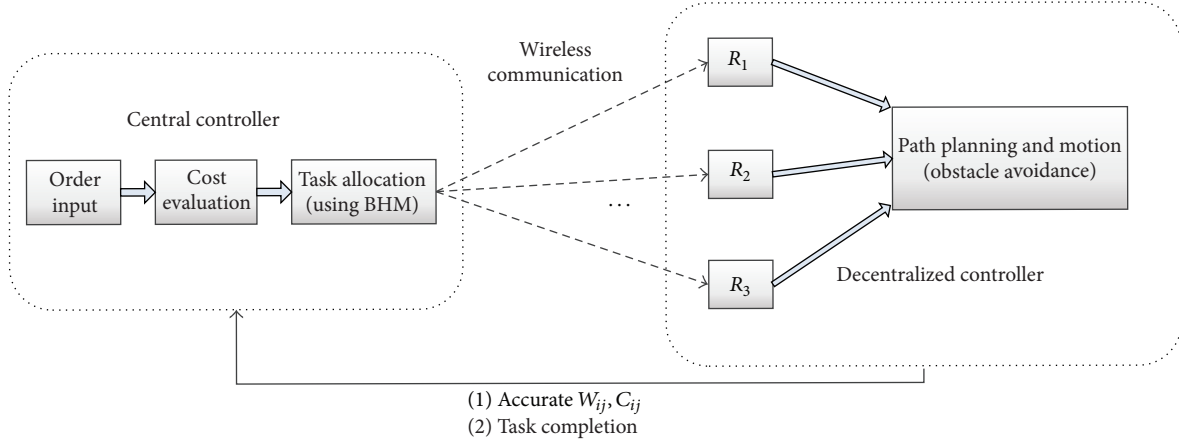


FIGURE 3: Control system structure.

could be further enhanced in the real world compared with the simulated balancing performance of the simulated experiment. This mechanism is referred as more-task-prior mechanism in this paper. The whole process is shown in Figure 3.

3.2. Balanced Heuristic Mechanism (BHM). Considering the goal of minimizing the travel cost as well as balancing the individual travel cost of each robot, a heuristic function is given in the following for guiding the task allocation process:

$$C(r_i, t_j) = \alpha \times \text{dist}(r_i, t_j) + (1 - \alpha) \times CW(r_i), \quad (4)$$

where $C(r_i, t_j)$ describes the cost (considering both the time and distance) for robot r_i to finish task t_j , $\text{dist}(r_i, t_j)$ is the travel cost from robot r_i to task t_j , and $CW(r_i)$ is the current value of $W(r_i)$ (refer to Section 2).

More specifically, as both task cost and travel cost from one location to another are unknown, we evaluate w_i (the cost of each task) and C_{ij} (the travel cost from one location to another) before the task allocation by using the standard A* algorithm, a simple but significantly effective path planning algorithm. When a robot fulfills a task or moves from one location to another in real environment, more complex motions should be taken into account, such as the obstacle avoidance. Then, the more accurate value is shared with the central controller and the original estimated value stored in the controller is replaced with the newest value, which can help the controller to make a better task allocation. Thus the following allocation process is in the light of new information and can be more accurate and predictable.

BHM makes tasks to be assigned to the robot who has the lowest $C(r_i, t_j)$. By introducing a parameter α , targets are more inclined to be assigned to the nearby robot with fewer allocated tasks. In practice, an effective value of α can be set between 0.7 and 0.9 according to the numbers of robots and tasks. The detailed relationship between α and the system performance will be given in Section 4.

At the beginning stage, as the value of $W(r_i)$ is quite small, this mechanism has limited effects. Thus the performance

of the system is not significantly superior to the one by traditional methods that merely focus on minimizing the total travel distance. However, as the allocation proceeds, the role of $(1 - \alpha) \times CW(r_i)$ comes into effect by sharing the workload with those who have fewer assigned tasks. Above all, the method can obtain an approximately optimal result by considering the total travel cost and the workload balance of robots. A sample illustration of this heuristic mechanism is given in Figure 4.

3.3. Application of BHM. Recently the auction-based task allocation method and the cluster-based task allocation method have been widely used to solve the MRTA problems [18, 24]. So we introduce our BHM approach into the classic single-item auction method and the K-means clustering method, respectively, and proposed two BMH-based methods to solve the balancing MRTA problem.

3.3.1. An Improved Auction Based Task Allocation Method Using BHM. A typical seal-bid single-round single-item auction process generally consists of four steps:

- (i) task announcement,
- (ii) matrix evaluation,
- (iii) bid submission,
- (iv) close of auction (to determine the winning bids and notify the winning robot).

At the second stage, the matrix is usually defined as a function of the optimistic travel cost [11] as shown in the following matrix:

$$\begin{pmatrix} d_{r_1, t_1} & d_{r_1, t_2} & \cdots & \cdots & d_{r_1, t_n} \\ d_{r_2, t_1} & d_{r_2, t_2} & \cdots & \cdots & d_{r_2, t_n} \\ \vdots & \vdots & \ddots & & \vdots \\ \vdots & \vdots & & \ddots & \vdots \\ d_{r_m, t_1} & d_{r_m, t_2} & \cdots & \cdots & d_{r_m, t_n} \end{pmatrix}, \quad (5)$$

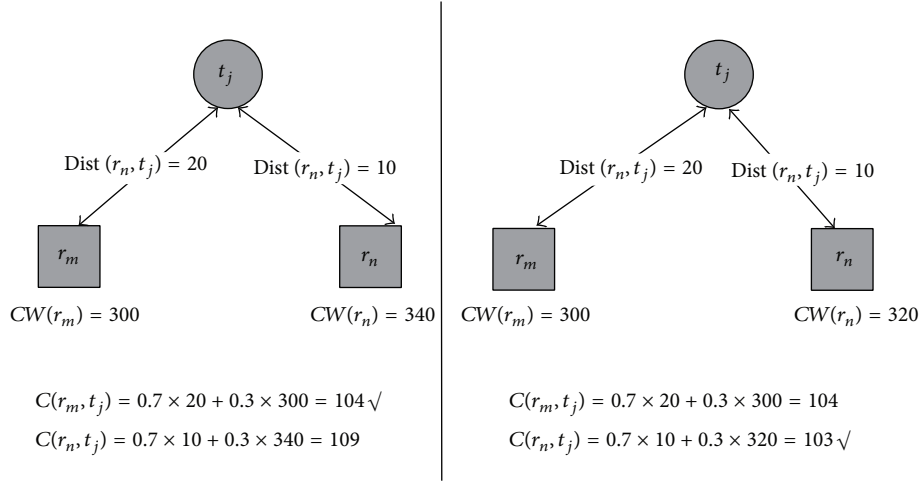


FIGURE 4: Example of the balanced heuristic mechanism.

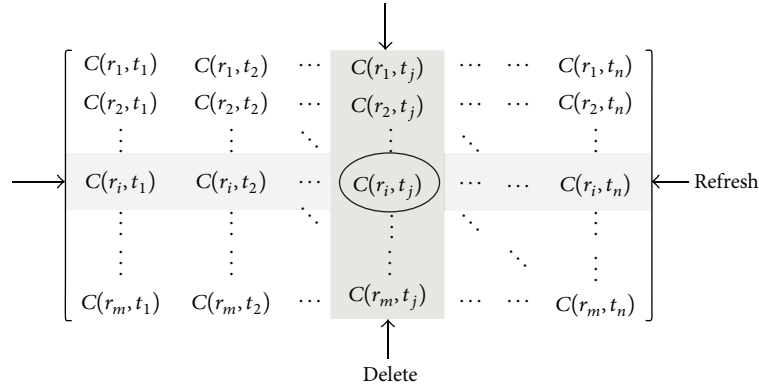


FIGURE 5: Evaluation matrix refreshment.

where $\{t_1, t_2, \dots, t_n\}$ are all the open tasks which have not been carried out by robots. In our method, the BHM is introduced into the matrix evaluation process. In order to strengthen the balancing performance of the auction, the matrix evaluation is defined by C_{r_i, t_j} rather than the optimistic travel distance between r_i and t_j .

In each round, after all of the robots submit their bids, the single-round single-item auction is closed. The controller will decide the winning bidder with the smallest bid value and notify the winner. Then the bid matrix will be refreshed with following steps (as shown in Figure 5).

Step 1. Delete the task t_k from the open-task set.

Step 2. Recalculate the row $(C_{r_i, t_1}, C_{r_i, t_2}, \dots, C_{r_i, t_n})$.

As described in (4), the value of $C(r_i, t_j)$ is both related to the distance between r_i and t_j as well as the already assigned tasks of r_i . Firstly, the current position of r_i is replaced by the location of t_j , since r_i wins the bid and has to move from its former position to t_j in order to carry out the order. Secondly, the $CW(r_i)$ increases by the cost of the newly

allocated task, which can be mathematically expressed as $CW(r_i) = CW(r_i) + w_{ij}$. After the matrix is refreshed, next single-item auction process is proceeding until all the tasks have been allocated.

3.3.2. A Balanced K-Means Clustering Method Using BHM. A clustering method can be generally divided into two stages, that is, firstly finding the cluster centers and then assigning the sample data into the proper cluster. The K-means clustering is a commonly used partitioning method [25], the main idea of which is to find K cluster centers (c_1, c_2, \dots, c_k) , which minimize D in the following by iteration:

$$D = \sum_{i=1}^n \min_{r=1,2,\dots,k} d(x_i, c_r)^2. \quad (6)$$

In (6), $d(x_i, c_r)$ represents the distance between the point x_i and c_r , known as the Euclidean distance. When D converges to a stable value, the appropriate (c_1, c_2, \dots, c_k) are the cluster centers for those K subgroups.

At the second stage, the sample data x_i is assigned into the proper cluster with the smallest $d(x_i, c_r)$, $r = 1, 2, \dots, k$.

However, this method only guarantees an optimal allocation scheme minimizing the travel cost, while the size differences between each cluster are ignored.

In order to solve the balancing MRTA problem, we make some modifications on both of the two stages. As for the first stage, given that robots can only move along the grid lines, we substitute the Euclidean distance with the Manhattan distance in (6). In a rectangular coordinate system, the Manhattan distance between point $P_1 : (x_1, y_1)$ and point $P_2 : (x_2, y_2)$ can be expressed as

$$d_{\text{manhattan}}(P_1, P_2) = |x_1 - x_2| + |y_1 - y_2|. \quad (7)$$

Thus, the main purpose is to find K cluster centers (c_1, c_2, \dots, c_k) , which minimize $D_{\text{manhattan}}$ in the following by iteration:

$$D_{\text{manhattan}} = \sum_{i=1}^n \min_{r=1,2,\dots,k} d_{\text{manhattan}}(x_i, c_r). \quad (8)$$

Then, to reduce the value $D_{\text{manhattan}}$ iteratively, we find the Manhattan center rather than the center of the collected tasks. Finally, a partial optimal solution is formed in a finite number of iterations. Generally speaking, there are three steps of the first stage.

- (1) *Initial step*: give an initial set of K cluster centers.
- (2) *Assignment step*: assign each point to the cluster center which has the shortest Manhattan distance.
- (3) *Update step*: calculate the Manhattan Center, which can reduce the overall value of $D_{\text{manhattan}}$. The Manhattan Center is defined as follows.

It is supposed that there are m points in a rectangular coordinate system as p_1, p_2, \dots, p_m , coordinates of which are denoted as $(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)$. Then, the descending order of the sequence x_1, x_2, \dots, x_m is denoted as x'_1, x'_2, \dots, x'_m and the descending order of the sequence y_1, y_2, \dots, y_m is denoted as y'_1, y'_2, \dots, y'_m . Our purpose is to find the point $P_m = (x_{mc}, y_{mc})$, which minimizes (8):

$$\min_{x_{mc}, y_{mc}} \sum_{i=1}^m (|x_{mc} - x_i| + |y_{mc} - y_i|). \quad (9)$$

Since the values of x_{mc} and y_{mc} do not affect each other, the value of $\sum_{i=1}^m |x_{mc} - x_i|$ and $\sum_{i=1}^m |y_{mc} - y_i|$ is minimized separately. It is easy to prove that when m is odd, x_{mc} is $x'_{(m+1)/2}$ so that $\sum_{i=1}^m |x_{mc} - x_i|$ is minimum; when m is even, x_{mc} is between $x'_{(m+1)/2}$ and $x'_{((m+1)/2)+1}$ so that $\sum_{i=1}^m |x_{mc} - x_i|$ is minimum. We can get y_{mc} in the same way. Therefore, the Manhattan center p_{mc} can be calculated by following equations:

$$x_{mc} = \begin{cases} x'_{(m+1)/2} & m \text{ is odd} \\ \frac{x'_{(m+1)/2} + x'_{((m+1)/2)+1}}{2} & m \text{ is even} \end{cases} \quad (10)$$

$$y_{mc} = \begin{cases} y'_{(m+1)/2} & m \text{ is odd} \\ \frac{y'_{(m+1)/2} + y'_{((m+1)/2)+1}}{2} & m \text{ is even.} \end{cases}$$

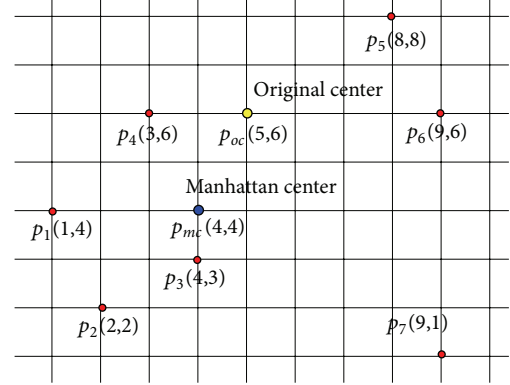


FIGURE 6: Example of the manhattan center.

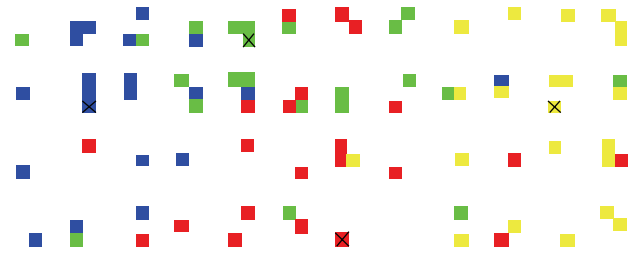


FIGURE 7: Example of the clustering result.

As mentioned before, the Manhattan center has the minimum overall Manhattan distance to these m points, by which the value of $D_{\text{manhattan}}$ is reduced. An example is shown in Figure 6. For the original center p_{oc} , $\sum_{i=1}^m (|x_{oc} - x_i| + |y_{oc} - y_i|) = 37$, while, for the Manhattan Center p_{mc} , $\sum_{i=1}^m (|x_{mc} - x_i| + |y_{mc} - y_i|) = 34$.

Now, we concisely prove the convergence of the algorithm. Given that the $D_{\text{manhattan}}$ generated by the algorithm is strictly decreasing and there only exist a finite number of such partitions, the algorithm will reach a partial optimal solution in a finite number of iterations.

After obtaining the cluster centers, the balanced heuristic mechanism is introduced into the second stage of clustering. In traditional methods, when the m cluster centers (r_1, r_2, \dots, r_m) are found in the first stage of the K -means algorithm, the remaining active tasks are allocated into the proper cluster with the smallest $d(x_i, c_r)$, $r = 1, 2, \dots, k$. However, in our approach, remaining active tasks are allocated into the proper cluster with the smallest $C(r_i, t_j)$ defined in (4).

The main idea is that, instead of merely considering the travel cost, the balance of the cluster size is taken into account. The task tends to be allocated into the nearby cluster with fewer tasks. When a new task is added into the cluster j , the $CW(r_i)$ will be refreshed. After all the active tasks have been allocated, the genetic-based TSP algorithm [26] is used to plan the route for tasks in each cluster. One example of the task allocation is demonstrated in Figure 7 ($m = 4, n = 50, \alpha = 0.8$), where different colors represent tasks allocated to different robots. Blocks with X are clustering centers.

TABLE 1: Balance behavior of single-item auction versus BHM auction.

$R = 8$ m	Single-item auction			BHM auction ($\alpha = 0.8$)		
	μ	σ	$CV = \sigma/\mu$	μ	σ	$CV = \sigma/\mu$
50	685	312	45.6%	711	61	8.6%
100	1374	315	22.9%	1469	53	3.6%
150	1969	553	28.1%	2128	41	1.9%
200	2591	449	17.3%	2827	50	1.8%
250	3234	439	13.6%	3555	32	0.9%
300	3897	609	15.6%	4221	21	0.5%
350	4418	701	15.9%	4794	26	0.5%
400	5085	775	15.2%	5499	19	0.4%

TABLE 2: Balance behavior of typical K -means versus BHM K -means.

$R = 8$ m	K -means			BHM K -means ($\alpha = 0.8$)		
	μ	σ	$CV = \sigma/\mu$	μ	σ	$CV = \sigma/\mu$
50	714	310	43.4%	749	50	6.7%
100	1390	513	36.9%	1508	57	3.8%
150	2058	764	37.1%	2204	49	2.2%
200	2751	1116	40.6%	2949	63	2.1%
250	3441	1076	31.3%	3684	69	1.9%
300	4095	1254	30.6%	4440	65	1.5%
350	4776	1484	31.1%	5214	83	1.6%
400	5465	1882	34.4%	5895	66	1.1%

4. Simulated Experiments

4.1. Simulation Description. Several groups of simulated experiments are implemented with OPENGL and the experimental process is as follows.

Step 1. Initialization.

Step 2. Creating orders in the task pool while the arriving rate of orders meets the Poisson distribution.

Step 3. Sending a specific number (e.g., 200) of orders from the task pool to the controller.

Step 4. Using different methods to allocate the orders (tasks). More specifically, there are four approaches, namely, K -means, BHM K -means, auction, and BHM auction, and their performances are compared.

Step 5. Sending the allocated tasks by the controller to the corresponding robots, which begin to fulfil their tasks (using the A^* algorithm to calculate the path and using the more-task-prior mechanism to avoid collision). Then go to Step 3.

4.2. Simulation Results Analysis. In this part, three criteria mentioned in Section 2 are used to evaluate the performance of the proposed balanced heuristic mechanism, that is, coefficient of variation (CV), travel time (TT), and total travel cost (TTC).

First of all, we compare the balance behavior of the improved methods with the typical auction method and the K -means method, results of which are given in Tables 1 and 2, where the data are collected by average during fifty times of simulation experiments. Both Tables 1 and 2 show that using the proposed BHM, the diversity degree in work groups is significantly reduced, which indicates the more equal workload distribution among robots.

As shown in Figures 8 and 9, the travel time (TT) of the multirobot system is provided by both the traditional method and the improved method. We can conclude that by applying the BHM based methods, the time spent to finish

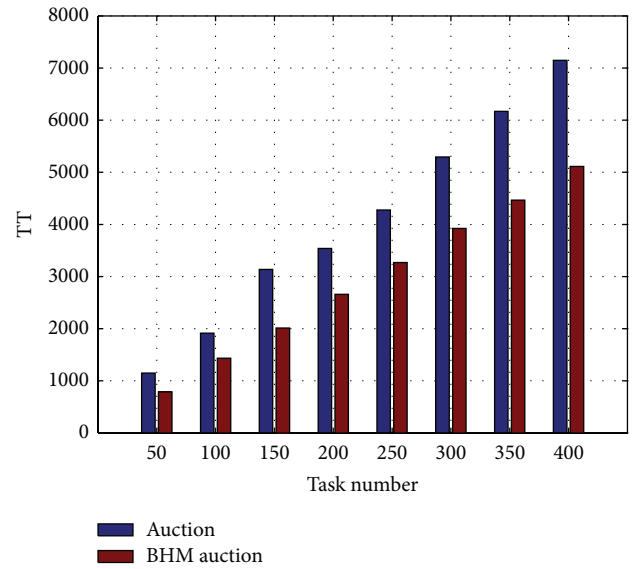


FIGURE 8: Time spent by single-item auction versus BHM auction.

the given task is much less than the one by using the traditional method, due to an efficient sharing of workload.

Comparison results for traditional methods and BHM based methods are demonstrated in Figure 10 by means of the total travel cost (TTC), from which we can conclude that in balanced heuristic cases, although robots pay more attention to balancing the workload rather than merely minimizing the travel cost as traditional methods do, the TTC only increases by about 8%. Concerning the obvious increasement of the robots utilization, this slightly increased power consumption is quite acceptable.

With the traditional auction method and K -means method, the task allocation process only concentrates on minimizing the travel cost. Once there is a robot nearest to an open task, other robots do not attempt to compete for the task even if they are idle. Thus the utilization of the system is not enough and the time cost to fulfill the given tasks is high. With BHM based methods proposed in this paper, all robots

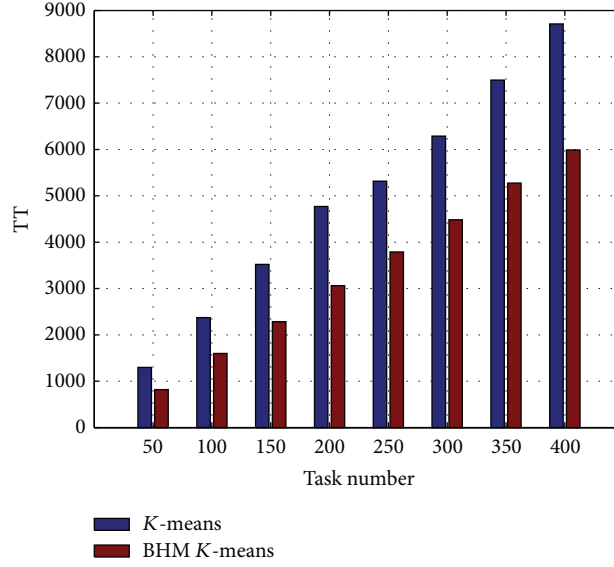


FIGURE 9: Time spent by K-means versus BHM K-means.

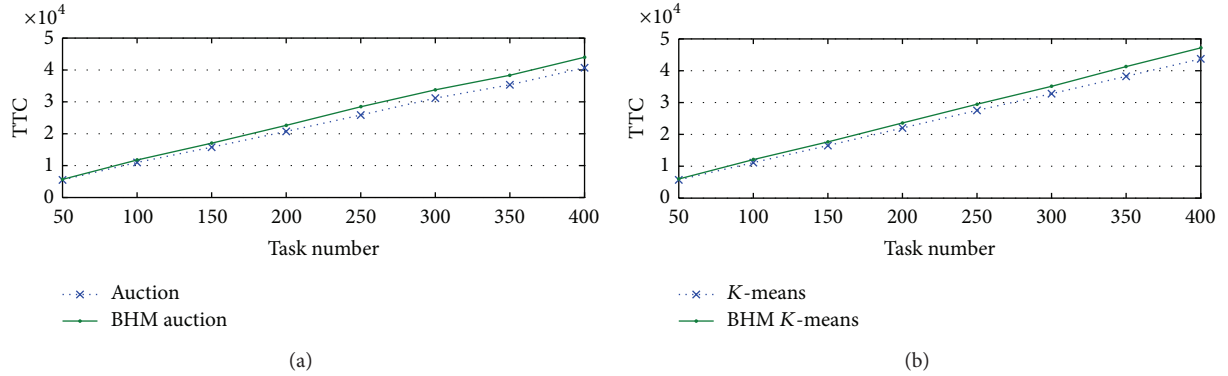


FIGURE 10: Total travel cost by traditional methods versus BHM adapted methods.

can be utilized in a balanced and effective manner while the travel cost is quite low.

Then we study the influence of the variable α on the system performance. The value of α describes the weight of the system energy saving demand and time saving demand. Figure 11 demonstrates the relationship between α and the travel time (TT). It shows that in order to keep a good balance between minimizing the travel distance and efficient sharing of the workload, the coefficient is usually set between 0.7 and 0.9 to achieve the minimum of TT.

Finally, besides comparing BHM-based approaches with the traditional auction method and K-means method, we use a more competing algorithm (KCAB mechanism in [18]) to testify the scientific merit of the BHM scheme (as shown in Figures 12 and 13). We can conclude that by BHM based methods, the time spent to finish the given task is 5~15% less than the KCAB mechanism; meanwhile, their total travel costs are almost the same. Thus, the benefit of the balanced heuristic mechanism to warehouse systems is significant.

5. Conclusion

In this paper, a novel balanced heuristic mechanism for the multirobot task allocation problem is presented with respect to the task cost and travel cost. To evaluate the proposed approach, we give three criteria, namely, travel time (TT), total travel cost (TTC), and coefficient of variation (CV), and compare some traditional methods with BHM-based methods under those criterions. Simulation results show that the proposed approach greatly enhances the balance behavior of the multirobot system, reduces the total travel time while achieves almost the same total travel cost as the traditional methods do. In addition, the influence of the variable α on the system performance is also studied in detail. In our future work, BHM will be applied into more complicated environments, which consist of heterogeneous tasks and robots. Furthermore, we will dedicate to enhance the robustness of the proposed approach in dynamic environments. Finally, it is worth to point out that this new mechanism is of great

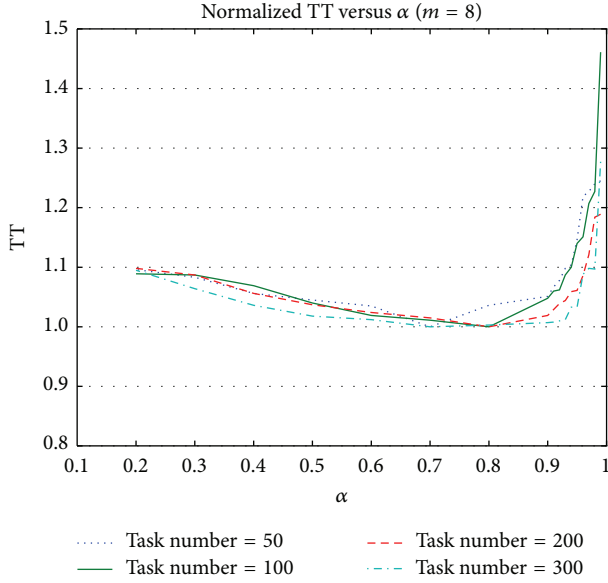
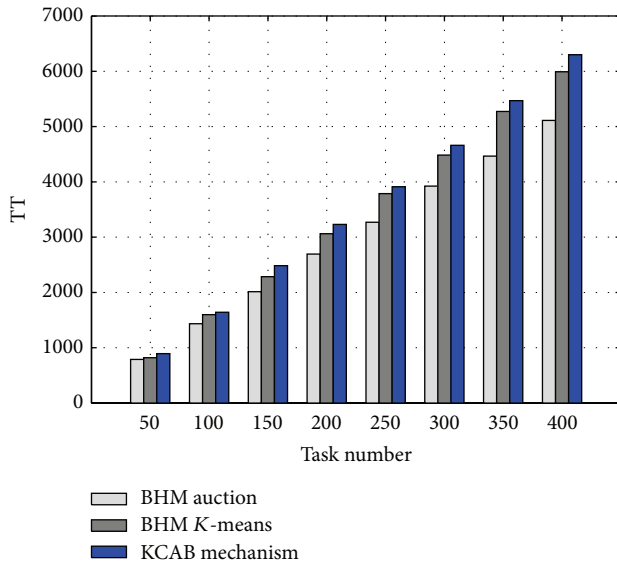
FIGURE 11: Relationship between α and travel time (TT).

FIGURE 12: Comparison between method provided in [15] versus BHM-based approaches (TT).

applicability to other task allocation problems, such as the WSNs problem in [27].

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Authors' Contribution

Luowei Zhou and Yuanyuan Shi contributed equally to this work.

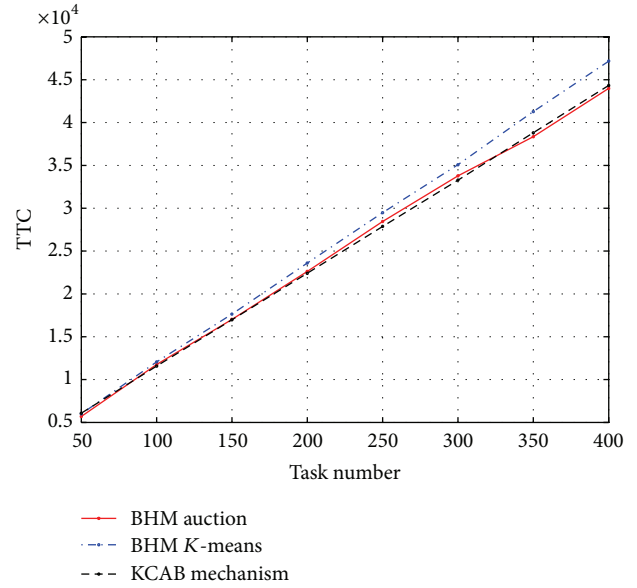


FIGURE 13: Comparison between method provided in [15] versus BHM-based approaches (TTC).

Acknowledgment

This work was supported by the Natural Science Foundation of China under Grants no. 61273327 and no. 61432008.

References

- [1] P. R. Wurman, R. D'Andrea, and M. Mountz, "Coordinating hundreds of cooperative, autonomous vehicles in warehouses," *AI Magazine*, vol. 29, no. 1, pp. 9–19, 2008.
- [2] X. Zhai, J. E. Ward, and L. B. Schwarz, "Coordinating a one-warehouse N-retailer distribution system under retailer-reporting," *International Journal of Production Economics*, vol. 134, no. 1, pp. 204–211, 2011.
- [3] C. Chen, H.-X. Li, and D. Dong, "Hybrid control for robot navigation—a hierarchical Q-learning algorithm," *IEEE Robotics & Automation Magazine*, vol. 15, no. 2, pp. 37–47, 2008.
- [4] S. Chen and C. Chen, "Probabilistic fuzzy system for uncertain localization and map building of mobile robots," *IEEE Transactions on Instrumentation and Measurement*, vol. 61, no. 6, pp. 1546–1560, 2012.
- [5] D. Dong, C. Chen, J. Chu, and T.-J. Tarn, "Robust quantum-inspired reinforcement learning for robot navigation," *IEEE/ASME Transactions on Mechatronics*, vol. 17, no. 1, pp. 86–97, 2012.
- [6] T. Niemueller, D. Ewert, S. Reuter et al., "Towards benchmarking cyber-physical systems in factory automation scenarios," in *KI 2013: Advances in Artificial Intelligence*, vol. 8077 of *Lecture Notes in Computer Science*, pp. 296–299, 2013.
- [7] M. Brambilla, E. Ferrante, M. Birattari, and M. Dorigo, "Swarm robotics: a review from the swarm engineering perspective," *Swarm Intelligence*, vol. 7, no. 1, pp. 1–41, 2013.
- [8] C. Nieto-Granda, J. G. Rogers, and H. I. Christensen, "Coordination strategies for multi-robot exploration and mapping," *Experimental Robotics*, vol. 33, no. 4, pp. 519–533, 2013.

- [9] B.-Y. Shih, C.-Y. Chen, and W. Chou, "An enhanced obstacle avoidance and path correction mechanism for an autonomous intelligent robot with multiple sensors," *Journal of Vibration and Control*, vol. 18, no. 12, pp. 1855–1864, 2012.
- [10] B. Yamauchi, "Frontier-based exploration using multiple robots," in *Proceedings of the 2nd International Conference on Autonomous Agents*, pp. 47–53, ACM, May 1998.
- [11] B. P. Gerkey and M. J. Mataric, "Sold!: auction methods for multirobot coordination," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, pp. 758–768, 2002.
- [12] T. Sandholm, "Algorithm for optimal winner determination in combinatorial auctions," *Artificial Intelligence*, vol. 135, no. 1-2, pp. 1–54, 2002.
- [13] M. Berhault, H. Huang, P. Keskinocak et al., "Robot exploration with combinatorial auctions," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 1-4, pp. 1957–1962, October 2003.
- [14] M. G. Lagoudakis, M. Berhault, S. Koenig, P. Keskinocak, and A. J. Kleywegt, "Simple auctions with performance guarantees for multi-robot task allocation," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '04)*, pp. 698–705, IEEE, October 2004.
- [15] L. Liu and D. A. Shell, "Large-scale multi-robot task allocation via dynamic partitioning and distribution," *Autonomous Robots*, vol. 33, no. 3, pp. 291–307, 2012.
- [16] S. Öztürk and A. E. Kuzucuoğlu, "Optimal bid valuation using path finding for multi-robot task allocation," *Journal of Intelligent Manufacturing*, 2014.
- [17] D. Puig, M. A. Garcia, and L. Wu, "A new global optimization strategy for coordinated multi-robot exploration: development and comparative evaluation," *Robotics and Autonomous Systems*, vol. 59, no. 9, pp. 635–653, 2011.
- [18] M. Elango, S. Nachiappan, and M. K. Tiwari, "Balancing task allocation in multi-robot systems using K-means clustering and auction based mechanisms," *Expert Systems with Applications*, vol. 38, no. 6, pp. 6486–6491, 2011.
- [19] A. G. Bedeian and K. W. Mossholder, "On the use of the coefficient of variation as a measure of diversity," *Organizational Research Methods*, vol. 3, no. 3, pp. 285–297, 2000.
- [20] S. S. Chiddarwar and N. R. Babu, "Conflict free coordinated path planning for multiple robots using a dynamic path modification sequence," *Robotics and Autonomous Systems*, vol. 59, no. 7-8, pp. 508–518, 2011.
- [21] J. Capitan, M. T. J. Spaan, L. Merino, and A. Ollero, "Decentralized multi-robot cooperation with auctioned POMDPs," *The International Journal of Robotics Research*, vol. 32, no. 6, pp. 650–671, 2013.
- [22] D. Dong, C. Chen, H. Li, and T.-J. Tarn, "Quantum reinforcement learning," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 38, no. 5, pp. 1207–1220, 2008.
- [23] C. Chen, D. Dong, H.-X. Li, J. Chu, and T.-J. Tarn, "Fidelity-based probabilistic Q-learning for control of quantum systems," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 5, pp. 920–933, 2014.
- [24] M. Elango, G. Kanagaraj, and S. G. Ponnambalam, "Sandholm algorithm with K-means clustering approach for multi-robot task allocation," in *Swarm, Evolutionary, and Memetic Computing*, vol. 8298, pp. 14–22, Springer International Publishing, 2013.
- [25] M. E. Celebi, H. A. Kingravi, and P. A. Vela, "A comparative study of efficient initialization methods for the k-means clustering algorithm," *Expert Systems with Applications*, vol. 40, no. 1, pp. 200–210, 2013.
- [26] M. Shao, "Research on improved convergence co-evolution genetic algorithm for TSPs," *International Journal of Convergence Computing*, vol. 1, no. 2, pp. 118–126, 2014.
- [27] I. Mezei, M. Lukic, V. Malbasa, and I. Stojmenovic, "Auctions and iMesh based task assignment in wireless sensor and actuator networks," *Computer Communications*, vol. 36, no. 9, pp. 979–987, 2013.

