Short Papers

An Optimal Task Allocation Approach for Large-Scale Multiple Robotic Systems With Hierarchical Framework and Resource Constraints

Liang Ren, Yingying Yu, Zhiqiang Cao, Zhiyong Wu, Junzhi Yu, Chao Zhou, and Min Tan

***Abstract*—In this paper, a hierarchical coalition approach is proposed. The proposed approach is conducted under a hierarchical framework, which is composed of individual robots in the bottom layer and managers in higher layers. A bottom-up resources vector updating process for each manager is executed. Next, a top-down resources comparison process between the task and the managers is used to generate candidate robot coalitions. Specifically, different managers may be combined when necessary to meet the resources requirement of some complex tasks. Furthermore, a matching degree between the task and candidate robot coalition is utilized for an optimized coalition selection. The effectiveness of the proposed approach is verified by simulations.**

***Index Terms*—Hierarchical framework, large-scale multiple robotic system, resource constraints, task allocation.**

I. Introduction

Multiple robotic systems are of great interest in recent decades [1]. As a typical yet challenging problem, multirobot task allocation (MRTA) has received much attention [2]. The objective of MRTA is to establish the relationship between robots and tasks with an effective resources deployment.

Traditional MRTA problems deal with tasks where each one can be accomplished by only a single robot, which have been solved [3], [4]. However, they fail when a task requires more resources that are beyond the capabilities of an individual robot. To deal with this situation, Shehory and Kraus proposed a multiagent coalition formation approach [5]. Han *et al.* proposed an allocation algorithm under the assumption that all robots are homogeneous [6]. In [7], one task is divided into several subtasks where each one can be executed by a single robot. Parker and Tang presented a coalition approach called ASyMTRe [8]. Vig and Adams tailored the approach in [5] to adapt to the multirobot domain [9]. The approaches in [8] and [9] achieve good assignment results; however, there exists the weakness of heavy computational cost.

To reduce computational cost, Chen and Sun developed a leader-follower based methodology [10], which ensures a greedy solution. Furthermore, they designed two coalition-based allocation methods with resource constraints [11]: sequential coalition and holistic coalition methods. These methods have achieved significant improvement in terms of computational complexity. A disadvantage of the methods in [10] and [11] is that the numbers of resources required by the robots for task accomplishment are assigned a priori. Actually, in some cases, it is difficult to acquire this knowledge in advance. In addition, these methods adopt the greedy solution, which possibly results in the waste of resources.

It is worth noting that the method in [10] implies the idea of hierarchical allocation. The hierarchical allocation where the robots in the bottom layer are divided into many groups can effectively shorten the allocation time. This provides a natural solution, especially for large-scale robotic systems. Cao *et al.* decomposed the task allocation problem into two levels: group-level and member-level [12]. First, it fails for some complex tasks whose requirements are beyond the resources possessed by a single robot group. Second, the centralized solution of group-level leads to a weak fault tolerance. The method in [13] improves the second drawback by a multilayer subdivision for the group level. However, no resources integration among these high layers is involved, which reduces the efficiency of task allocation. Besides, the first drawback of [12] still remains unsolved. In practice, the influence from complex tasks makes the existing hierarchical methods not always readily feasible.

Aiming at the challenges of complex tasks and resources integration among the managers, this paper proposes a task allocation solution based on hierarchical framework, without a priori knowledge about the numbers of the required resources. There are two roles for the members of this hierarchical framework: robots in the bottom layer, and managers in other layers. By a bottom-up resources vector updating process of managers, and a top-down resources comparison process between the task and managers, candidate robot coalitions are generated. The matching degree based on Euclidean distance is then utilized to obtain the optimized one. It shall be noted that, to meet the resources requirement of some complex tasks, different managers may be combined together to provide more possible candidate robot coalitions. Besides, this top-down comparison process can effectively reduce the searching scope of candidate robot coalitions. The proposed approach achieves a fast task allocation with optimal resources deployment for large-scale multiple robotic systems.

II. Problem Statement

We label $\mathbf{F}$ as a resource space with a dimension of dim($\mathbf{F}$). For a resources vector $F = [f_1 \ldots f_k \ldots f_{\dim(\mathbf{F})}]$ in $\mathbf{F}$, each f_k has a special integration rule according to the resource type including sum type, maximum type, and minimum type. For example, $F(a) + F\ (b) = [f_1(a) + f_1(b) \quad \max(f_2(a), f_2(b)) \min(f_3(a), f_3(b)) \ldots]$ according to the application. $f_k = 0$ means that there is no or no need of this resource.

Comparison of resources vectors is done by comparing component by component, and we have

$$F\ (a) \succcurlyeq F\ (b) \Leftrightarrow \forall k, f_k(a) \geq f_k(b) \tag{1}$$

Manuscript received March 9, 2017; revised June 9, 2017; accepted September 27, 2017. This work was supported in part by the National Natural Science Foundation of China under Grant 61633017 and Grant 61633020, in part by the National High Technology Research and Development Program of China (863 Program) under Grant 2015AA042201, and in part by the Key Research and Development Program of Shandong Province under Grant 2017CXGC0925. *(Corresponding author: Zhiqiang Cao.)*

The authors are with the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China, and also with the School of Computer and Control Engineering, University of Chinese Academy of Sciences, Beijing 101408, China (e-mail: renliang2015@ia.ac.cn; yuyingying2015@ia.ac.cn; zhiqiang.cao@ia.ac.cn; wuzhiyong2015@ia.ac.cn; junzhi.yu@ia.ac.cn; chao.zhou@ia.ac.cn; min.tan@ia.ac.cn).

Digital Object Identifier 10.1109/JSYST.2017.2773056

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

The matching degree of two resources vectors is expressed by Euclidean distance [14], as follows:

$$|F(a), F(b)| = \sqrt{\sum_{k=1}^{\dim(\mathbf{F})} \left[\left(\frac{2}{f_k^{\max} + f_k^{\min}}\right) \cdot (f_k(a) - f_k(b))\right]^2} \tag{2}$$

where $f_k^{\max}$ and $f_k^{\min}$ are the maximum and minimum values corresponding to f_k, respectively.

We consider an *N*-layer hierarchical framework and all robots are assumed to be located within a robotic base. The robot in the *N*th layer is denoted by R_{N,g_N,i_N}, where g_N and i_N are the group number and the index of the robot, respectively. The resources vector possessed by R_{N,g_N,i_N} is expressed by $F(R_{N,g_N,i_N})$. We label the robot's state as a Boolean variable $B(R_{N,g_N,i_N})$, and it is 1 when the robot is executing a task or in a breakdown status. The manager is denoted by M_{n,g_n,i_n} $(n = 1,\ldots,N-1)$, where g_n and i_n are its group number and index in the *n*th layer, respectively. Denote the resources vector possessed by M_{n,g_n,i_n} with $F(M_{n,g_n,i_n})$, which is obtained by integrating the resources of its subordinates.

The construction process of the hierarchical framework is as follows. For the robots in the *N*th layer, they are divided into many groups in advance. Initially, each group creates a manager $M_{N-1,g_{N-1},i_{N-1}}$ as a member of the *N*–1th layer. These managers are further divided to generate new managers in the *N*–2th layer. Obviously, the i_nth manager M_{n,g_n,i_n} in the *n*th layer is created from the managers $M_{n+1,g_{n+1},i_{n+1}}$ in the *n*+1th layer with $g_{n+1} = i_n$. This process is repeated until the top manager $M_{1,1,1}$ in the top layer is generated. Notice that a manager may be attached to a physical robot, and the robot with a powerful computing capability is preferable. Actually, it is not limited to a physical robot, and a manager can also be attached to a calculation unit. This distributed characteristics of managers provide the possibility of parallel computing.

The task to be allocated is denoted with *T*, and its resources requirement vector is labeled as $F(T)$. Constrained by task properties, not all individual robots are involved for the task. A robot-level resource constraint $\widetilde{F_T}(R_{N,g_N,i_N})$ imposed by the task *T* is considered. In addition, the coalition formed by the robots is expressed by A_T with its resources vector $F(A_T)$.

The objective of this paper is stated as follows. Given a large-scale multiple robotic system with a predefined grouping and a task *T*, under the robot-level resource constraint $\widetilde{F_T}(R_{N,g_N,i_N})$, find an optimal and fast assignment A_T^* of robot coalitions based on the hierarchical framework to satisfy the resources requirement $F(T)$ of the task.

III. Task Allocation Approach Based on Hierarchical Coalition

A. Robot-Level Screening and Resources Vector Updating of Managers

Based on the robot-level resource constraint $\widetilde{F_T}(R_{N,g_N,i_N})$, a spare robot will be qualified when the values of its resources vector $F(R_{N,g_N,i_N})$ are not lower than those in $\widetilde{F_T}(R_{N,g_N,i_N})$. The set Ω_{rl} including all qualified robots is then acquired. In the hierarchical architecture, each manager shall update its resources vector from bottom to top by integrating the resources of its subordinates, as follows:

$$F(M_{n,g_n,i_n}) = \sum_{g_{n+1}=i_n} F\left(M_{n+1,g_{n+1},i_{n+1}}\right). \tag{3}$$

Note that for the manager in the *N*–1th layer, its subordinates must be the spare qualified robots. Clearly, different tasks lead to different qualified robots. Therefore, we have

$$F\left(M_{N-1,g_{N-1},i_{N-1}}\right) = \sum_{\substack{R_{N,g_N,i_N} \in \Omega_{rl} \\ g_N = i_{N-1}}} F(R_{N,g_N,i_N}). \tag{4}$$

Algorithm 1: The selection of candidate robot coalitions.

Input: The resources vector $F(M_{n,g_n,i_n})$ of M_{n,g_n,i_n}, the resources vector $F(R_{N,g_N,i_N})$ of R_{N,g_N,i_N} in Ω_{rl}, and the resources requirement vector $F(T)$.

Output: robot coalitions set Ω_{rc} with the corresponding resources vectors.

$n = 1$, $\Omega_{rc} = \Phi$;
search (M_{n,g_n,i_n});
  for its each subordinate $M_{n+1,g_{n+1},i_{n+1}}$ with $g_{n+1} = i_n$
    if $F(T) \preccurlyeq F(M_{n+1,g_{n+1},i_{n+1}})$ and $n+1 < N-1$ **then** *search* $(M_{n+1,g_{n+1},i_{n+1}})$;
    else if $F(T) \preccurlyeq F(M_{n+1,g_{n+1},i_{n+1}})$ and $n+1 = N-1$
      Com$(\Omega_{rl}, M_{n+1,g_{n+1},i_{n+1}})$;
      for robot team C_r in *Com*$(\Omega_{rl}, M_{n+1,g_{n+1},i_{n+1}})$
        if $F(T) \preccurlyeq F(C_r)$ **then** $\Omega_{\mathrm{rc}} \leftarrow A_T = C_r$; $F(A_T) = F(C_r)$;
      end for
    end if
  end for
if $\Omega_{rc} = \Phi$
    Com$(\Omega_{rl}, M_{n,g_n,i_n})$;
    for each robot team C_r in *Com*$(\Omega_{rl}, M_{n,g_n,i_n})$
      if $F(T) \preccurlyeq F(C_r)$ **then** $\Omega_{rc} \leftarrow A_T = C_r F(A_T) = F(C_r)$;
    end for
end if
return;

B. Selection of Candidate Robot Coalitions

All managers compare the resources vector $F(M_{n,g_n,i_n})$ $(n = 1,\ldots,N-1)$ with $F(T)$ required by the task in a top-down order. If manager M_{n,g_n,i_n} possesses sufficient resources, then $F(T)$ is compared with the resources vector of its each subordinate. If there exists a subordinate with sufficient resources, this comparison shall be repeated until *N*–1th layer; in this case, those subordinates without sufficient resources shall be not considered, which reduces the searching scope. For each manager with enough resources in the *N*–1th layer, each robot coalition whose resources vector is no less than $F(T)$ is determined from its subordinate robots. All candidate robot coalitions constitute the set Ω_{rc}. If a manager without in the *N*–1th layer has sufficient resources to execute the task whereas its each subordinate is not competent to the task, its subordinates should be combined together to generate feasible robot coalitions on the premise of $\Omega_{rc} = \Phi$.

The detailed algorithm is shown in Algorithm 1, where *search* (M_{n,g_n,i_n}) is a recursive function by the depth-first search algorithm, *Com*$(\Omega_{rl}, M_{n,g_n,i_n})$ is used to enumerate each possible robot combination C_r from the qualified robots belonging to M_{n,g_n,i_n}. $F(C_r)$ is labeled as the resources vector of robot combination C_r.

C. Optimal Robot Coalition

The optimal robot coalition A_T^* is acquired from the set Ω_{rc}, which is given as follows:

$$\begin{aligned} A_T^* &= \operatorname*{argmax}_{A_T \in \Omega_{\mathrm{rc}}} |F(T), F(A_T)| \\ &\text{s.t.} B(R_{N,g_N,i_N}) = 0 \\ &\quad F(R_{N,g_N,i_N}) \succcurlyeq \widetilde{F_T}(R_{N,g_N,i_N}) \\ &\quad F(A_T) \succcurlyeq F(T) \\ &\quad \sigma_1 \le |F(T), F(A_T)| \le \sigma_2 \end{aligned} \tag{5}$$

where σ_1 and σ_2 are given thresholds. The first two items of the constraints in (5) are used to ensure that an individual spare robot should satisfy the robot-level resource constraint, and the third item guarantees that the robot coalition A_T has sufficient resources to execute the task, while the last item is applied to ensure the reasonability of the resources

TABLE I
ROBOTS AND THEIR RESOURCES

	$s_1\,(\mathrm{m/s})$	$s_2\,(^\circ)$	s_3	$s_4\,(^\circ)$	$s_5\,(\mathrm{m})$	$s_6\,(\mathrm{kg})$	$s_7\,(\mathrm{cm})$	state
$R_{4,1,1}$	0.5	40	1024×768	270	10	5	70	0
$R_{4,1,2}$	0.7	40	1280×1024	270	10	3	80	0
$R_{4,1,3}$	0.5	90	1280×1024	270	10	4	70	0
$R_{4,1,4}$	0.5	40	2048×1536	270	10	1	70	0
$R_{4,1,5}$	0.5	40	1280×1024	270	20	2	90	0
$R_{4,2,6}$	0.7	40	1280×1024	360	10	3	70	0
$R_{4,2,7}$	0.5	40	1280×1024	360	6	5	70	0
$R_{4,2,8}$	0.5	90	1280×1024	360	10	2	80	0
$R_{4,2,9}$	0.5	40	2048×1536	360	10	3	70	0
$R_{4,2,10}$	0.5	40	1280×1024	360	20	5	90	0
$R_{4,3,11}$	0.6	40	1280×1024	360	10	4	70	0
$R_{4,3,12}$	0.5	60	1280×1024	360	10	5	80	0
$R_{4,3,13}$	0.5	40	1600×1200	360	10	3	70	0
$R_{4,3,14}$	0.4	40	1280×1024	270	10	4	80	0
$R_{4,3,15}$	0.5	40	1280×1024	360	12	5	90	0
$R_{4,4,16}$	0.6	90	1600×1200	360	12	5	90	0
$R_{4,4,17}$	0.6	60	2048×1536	360	12	4	60	0
$R_{4,4,18}$	0.5	30	1280×1024	270	10	3	70	0
$R_{4,4,19}$	0.6	60	1600×1200	360	20	2	70	0
$R_{4,4,20}$	0.7	60	1600×1200	360	12	2	70	0

utilization. By restricting $|F(T), F(A_T)|$ in the interval $[\sigma_1,\ \sigma_2]$, some robot coalitions possessing much more resources than task requirement shall be excluded, which can avoid the waste of resources.

D. Computational Complexity

Consider a multiple robotic system whose number is *I*, and it is divided into *G* groups with each group having $n_j\ (j = 1, 2, \ldots, G)$ members. Consider that each group satisfies the task requirement, the computational complexity of the proposed approach mainly depends on four parts: robot-level screening, resources updating of managers, resources comparison, and selection of candidate robot coalitions. For the first part, each robot is required to be filtered, whereas the second and third parts only rely on the number of managers. In the last part, candidate robot coalitions within the *j*th group are obtained by checking all possible combinations [9], and the computational cost is $2^{n_j} - 1$. Therefore, the computational cost of the proposed approach may be expressed by $I + N_V + N_V + \sum_{j=1}^{G}(2^{n_j} - 1)$, where N_V is the number of managers. Clearly $\sum_{j=1}^{G} 2^{n_j}$ and *I* are much bigger than N_V and *G* according to the hierarchical framework. Thus, the computational complexity of our hierarchical coalition is given by $O\,(I + \sum_{j=1}^{G} 2^{n_j})$.

IV. SIMULATIONS

In the following simulations, $\sigma_1 = 0.5$ and $\sigma_2 = 1.5$. The robots are grouped, and a four-layer framework is built. The resources of an individual robot include motion ability, camera angle, camera resolution, laser angle, laser detecting distance, manipulator weight capacity, and manipulator length, which correspond to s_1–s_7, respectively. It shall be noted that the type of resource s_6 is a sum one, and other resources belong to the maximum type.

A. Results With Fixed Grouping for a 20-Robot System

In this simulation, we consider a system with 20 robots. The robots are divided into four groups with equal number, and their resources are given in Table I.

Next, we verify the proposed approach by a hunting task T_1. For this task, only the first five resources are necessary. The resource requirement vector of this task is given by $F(T_1) = [0.5\ \ 40\ 1280 \times 1024\ \ 270\ \ 10]$ with $\widetilde{F_{T_1}}(R_{N,g_N,i_N}) = F(T_1)$. According to our approach, there are 60 effective robot coalitions. Their matching degrees with the task T_1 are shown in Fig. 1, where the robot coalitions generated from the first group to the fourth group are expressed by circle, square, diamond, and downward-pointing triangle, respectively. It is seen that, no. 26 robot coalition, i.e., $\{R_{4,2,8}, R_{4,2,9}, R_{4,2,10}\}$, is regarded as the optimized one, due to a good matching degree of 1.496297.

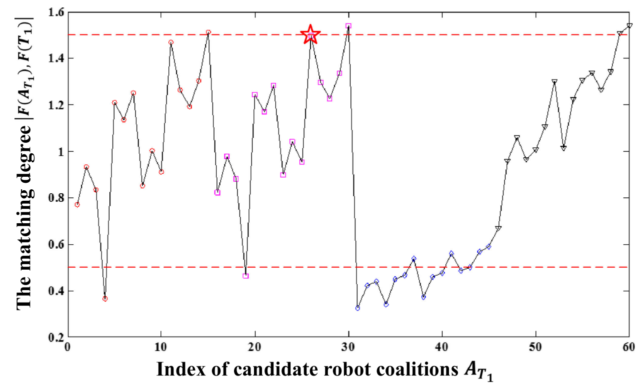

Fig. 1. Matching degrees of candidate robot coalitions with the task T_1.

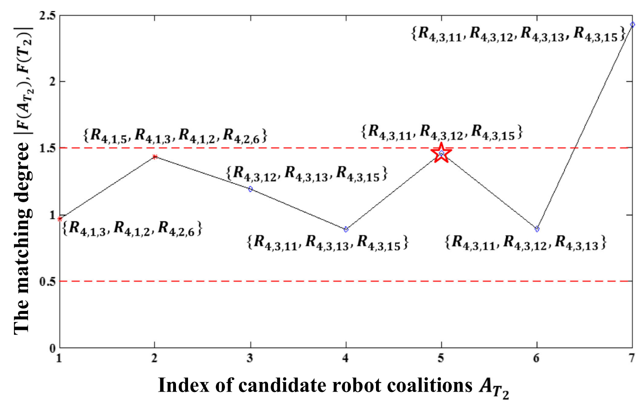

Fig. 2. Matching degrees of candidate robot coalitions with the task T_2.

During the task execution of T_1, a transportation task T_2 is assigned, which requires the resources s_1–s_7. $F(T_2) = [0.5\ \ 40\ \ 1280 \times 1024\ \ 270\ 10\ 10\ 70]$ with $\widetilde{F_{T_2}}(R_{N,g_N,i_N}) = [0.5\ 40\ 1280 \times 1024\ 270\ 10\ 2\ 70]$. According to Algorithm 1, there are seven candidate robot coalitions, and their matching degrees with the task T_2 are depicted in Fig. 2, where the robot coalitions expressed by star and diamond are generated from the combination of the first and second groups, and the third group, respectively. Clearly, no. 5 robot coalition becomes the winner with a matching degree of 1.462313.

B. Results With Different Groupings for a 180-Robot System

In this simulation, we consider 180 robots. The number n_j of robots within a group varies from 3 to 12, and we obtain the execution time of task assignment for the hunting task and transportation task on a PC with i7 CPU, as shown in Fig. 3. It can be seen that the execution time increases with the increasing of the robots number within a group, and the task is smoothly allocated in real-time for different groupings.

C. Comparison With the Existing Approaches

Table II gives the comparison of coalition-based task allocation approaches in terms of computational complexity.

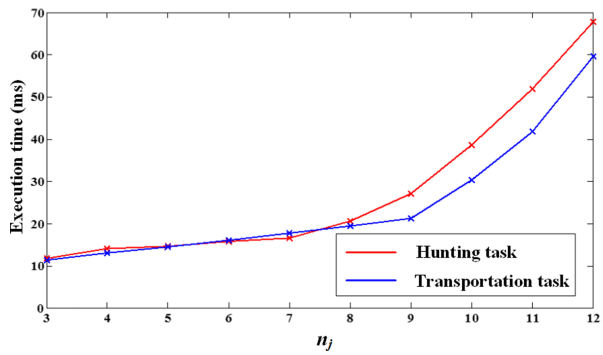

Fig. 3. Execution time of the proposed hierarchical coalition approach

TABLE II
COMPUTATIONAL COMPLEXITY COMPARISON

Coalition Approach	Computational Complexity
ASyMTRe [8]	$O(I!)$
Coalition Formation [9]	$O\left(2^I\right)$
Leader-Follower Coalition [10]	$\mathrm{O}(I^3)$
Sequential Coalition [11]	$\mathrm{O}(I^2)$
Hierarchical Coalition	$O\left(I + \sum_{j=1}^{G} 2^{n_j}\right)$

Next, we provide the quantitative comparison of the proposed approach with the approaches in [9] and [11] by the transportation task with $n_j = 3$. Notice that the approach in [11] relies on a priori knowledge about the numbers of the required resources, which is hard or impossible to be acquired in some situations. Therefore, the approach in [11] has to be improved and only quantitative values requirement of resources is used. Note that the improved sequential coalition approach is similar to the approach in [11] in computational complexity.

Under the same simulation conditions, the comparison results are demonstrated in Fig. 4, where Fig. 4(a) illustrates the execution time of the approach in [9], and Fig. 4(b) describes the execution time of the improved sequential coalition and our approach. Clearly, the approach in [9] has a heavy computation burden, while both the improved sequential coalition approach and the proposed approach achieve a fast allocation even for a robotic system with hundreds of robots. Compared with the approach in [11], the proposed approach has a broader application without the need of a priori knowledge about the numbers of the required resources. Besides, the proposed approach possesses a good resources deployment instead of a greedy solution in [11]. Moreover, our approach possesses the potential capability of parallel computing where different managers may be distributed in different physical units.

It is worth pointing out that candidate robot coalitions within a group are obtained by checking all possible combinations [9]. If the improved sequential coalition approach is applied to generate possible robot combinations within a group, the computational complexity of our approach shall be further reduced.

V. CONCLUSION

In this paper, the task allocation for large-scale multiple robotic systems is addressed under a hierarchical framework. The simulation results show that the proposed approach is fast with a satisfactory performance, and it can combine different managers to adapt to the resources requirement of complex tasks. In the future, we will extend the scope of the proposed approach, and consider more factors including current position of robot into the selection of optimal robot coalition. Meanwhile, we shall conduct deeper researches on grouping and layering for the hierarchical task allocation approach.

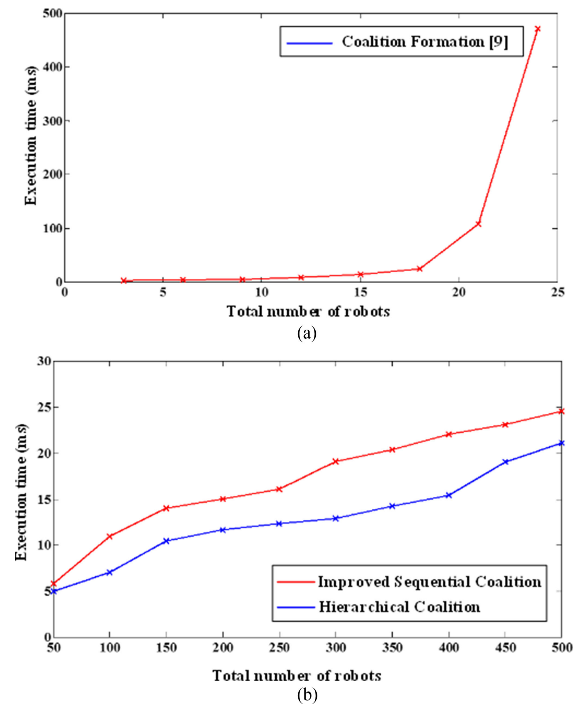

Fig. 4. Comparison of the proposed hierarchical coalition with the coalition formation and improved sequential coalition.

REFERENCES

[1] Z. Cao *et al.*, "A novel geometric transport approach for multiple mobile manipulators in unknown environments," *IEEE Syst. J.*, doi: 10.1109/JSYST.2016.2581171.

[2] C. Nam and D. A. Shell, "Assignment algorithms for modeling resource contention in multirobot task allocation," *IEEE Trans. Autom. Sci. Eng.*, vol. 12, no. 3, pp. 889–900, Jul. 2015.

[3] L. Luo, N. Chakraborty, and K. Sycara, "Distributed algorithms for multirobot task assignment with task deadline constraints," *IEEE Trans. Autom. Sci. Eng.*, vol. 12, no. 3, pp. 876–888, Jul. 2015.

[4] C. Nam and D. A. Shell, "Analyzing the sensitivity of the optimal assignment in probabilistic multi-robot task allocation," *IEEE Robot. Autom. Lett.*, vol. 2, no. 1, pp. 193–200, Jan. 2017.

[5] O. Shehory and S. Kraus, "Methods for task allocation via agent coalition formation," *Artif. Intell.*, vol. 101, no. 1/2, pp. 165–200, 1998.

[6] Y. Han, D. Li, J. Chen, X. Yang, and Y. Hu, "Task allocation algorithm based on robot ability and relevance with group collaboration in a robot team," in *Proc. Int. Conf. Intell. Netw. Intell. Syst.*, 2009, pp. 273–277.

[7] W. Shang, J. Zhong, Q. Yan, and J. Chen, "Resource constrained multirobot task allocation with an optimal solution," in *Proc. Int. Conf. Mechatron. Autom.*, 2012, pp. 2389–2394.

[8] L. E. Parker and F. Tang, "Building multirobot coalitions through automated task solution synthesis," *Proc. IEEE*, vol. 94, no. 7, pp. 1289–1305, 2006.

[9] L. Vig and J. A Adams, "Multi-robot coalition formation," *IEEE Trans. Robot.*, vol. 22, no. 4, pp. 637–649, Aug. 2006.

[10] J. Chen and D. Sun, "Resource constrained multirobot task allocation based on leader-follower coalition methodology," *Int. J. Robot. Res.*, vol. 30, no. 12, pp. 1423–1434, 2011.

[11] J. Chen and D. Sun, "Coalition-based approach to task allocation of multiple robots with resource constraints," *IEEE Trans. Autom. Sci. Eng.*, vol. 9, no. 3, pp. 516–528, Jul. 2012.

[12] L. Cao, H. Tan, H. Peng, and M. Pan, "Multiple UAVs hierarchical dynamic task allocation based on PSO-FSA and decentralized auction," in *Proc. Int. Conf. IEEE Robot. Biomimetics*, 2014, pp. 2368–2373.

[13] Y. Ma, Z. Cao, C. Zhou, and M. Tan, "A hierarchical task allocation method based on task case and its application to multi-robot hunting," in *Proc. 1st Int. Conf. Intell. Netw. Intell. Syst.*, 2008, 689–692.

[14] M. Deza and E. Deza. *Encyclopedia of Distances*. New York, NY, USA: Springer, 2013.