



Relation Adaptation of the Grouped Multiagent Network in Task Allocation

Kai Li¹ , Sentang Wu¹ , and Ying Wang² 

¹ Beihang University, Beijing 100191, China
likai1030@buaa.edu.cn

² Beijing Normal University, Beijing 100875, China

Abstract. In a large-scale multiagent system, agents are usually organized into groups by the network and participate in tasks in units of groups. However, according to the environment and task requirements, the original network relation is not necessarily optimal, and an adaptive evolution of this relation is needed. Therefore, a network relation adaptation method for grouped multiagent systems based on Q-learning is proposed in this paper. Each group decides whether to change neighbors and which neighbor to replace by evaluating local and global factors, such as task completion rate, task participation rate, and average revenue. With iterative learning, groups can converge at the optimal action strategy, and the optimal network structure in the current environment can also be formed. The simulation results show that compared with the network without relation adaptation, the method presented in this paper can effectively improve the task allocation success rate of grouped multiagent systems.

Keywords: Relation adaptation · Multiagent system · Task allocation

1 Introduction

With the rapid development of self-organizing networks and agent technology, the coordination of multiagent systems (MASs) has been widely studied. In the research on cooperative MASs, task allocation is one of the most important topics [1].

Many previous studies used the method based on microtasks and decomposition, such as Crowd Flower and TurKit [2]. In this method, complex tasks are decomposed into simple tasks (i.e., subtasks), which are then allocated to agents independently. Then, the results of all agents are combined by the controller to obtain the final result. This method can simplify the complex task; however, it brings about a heavy decomposition burden to the central node, and the optimal result of the tasks cannot be ensured [3].

The team-formation-based approach can reduce the load of the center and give full play to the autonomous collaboration of MASs. When a task appears, the agent organizes or joins a team according to its own resources. Then, the task is executed when all the task requirements are satisfied by the team. The agent in this model can coordinate autonomously to assign tasks and can adapt to the failures of some agents and different system scales, thus achieving great robustness. A smart factory model was presented in

[4], where smart factory objects were defined as different types of agents, who could achieve efficient collaboration with a self-organizing method based on big data. For large-scale robots in intelligent logistics scheduling, the task allocation approach was studied in [5]. The problem was established as a multiobjective optimization model, and a hybrid algorithm was designed to achieve multiobjective competitive optimization. Furthermore, the authors in [6] proposed a similarity-based method for MAS task allocation, where the similarity between historical and new tasks was utilized to avoid redundant computations and speed up the process of allocation.

In recent years, social networks and social teams have attracted the attention of researchers, who take into account reputations and social relations. Team formation in social networks was first presented in [7], with the aim of finding a social team that was able to satisfy all task requirements while minimizing the cost of communication. Considering the selfishness of members, a management mechanism of relation and reputation was introduced in [8], which was able to greatly improve the quality of task allocation and social sensing. In general, a large-scale MAS is often composed of several groups, and agents cooperate better with members in their own group than with agents in other groups. Therefore, a community-aware task allocation method was proposed in [9], in which the agent could communicate only with the neighbors in its communities. This community-based model was able to achieve better performance in task allocation compared with previous approaches and reduce the computational load and communication costs. The performance of this method is very close to that of the benchmark algorithm; however, the computational and communication loads were reduced. Moreover, a task allocation model for social multiagent groups was presented in [10], where multiple factors, such as capability, reputation, communication cost, and social background, were comprehensively considered, and great performance was achieved.

The above research is considered under the condition that the network relation, that is, the network topology, remains unchanged. In fact, the original network relation is not necessarily optimal. Therefore, it is necessary to dynamically adjust the network relation according to the tasks to fundamentally improve the system efficiency. Relation rewiring is one of the most common network adaptation methods. For example, a self-organizing mechanism was designed in [11] to enable the agent to adjust its relation, and high levels of task allocation and utility maximization of each action could be achieved. On the basis of the team formation mechanism, a reputation-based neighbor rewiring method was proposed in [12]. Members would choose an agent with the highest reputation as a new neighbor of their team and abandon the that agent with the lowest reputation among their original neighbors. In [13], each agent updated the probability of cooperation by aggregating local and historical information and rewired the links of defective neighbors to other nodes. Furthermore, a composite self-organization mechanism of the agent network was proposed in [14], which enabled the agent to dynamically adjust the relation and weight with other members to achieve efficient task allocation.

However, the previous network adaptation methods considered only a single agent and did not consider a grouped network in large-scale systems. Therefore, based on the task allocation of grouped multiagent networks, the relation adaptation method based on Q-learning is considered in this paper to improve the performance of task allocation. The rest of this paper is organized as follows. In Sect. 2, the model of task allocation

and relation adaptation in a grouped multiagent network is described. Then, the relation adaptation method and algorithm based on Q-learning are presented in Sect. 3. Then, in Sect. 4, experiments are conducted to verify the performance of the proposed method. Finally, we conclude the paper in Sect. 5.

2 Model Description

2.1 Task Allocation in Multiagent Groups

Given a grouped MAS, $MAG = \langle G, A, E \rangle$, where $G = \{G_1, G_2, \dots, G_m\}$ represents the groups in the system, $A = \{a_1, a_2, \dots, a_n\}$ represents the set of agents, and $e_{ij} \in E$ is the relation between a_i and a_j . In the network system, an agent can belong only to one group, and there must be a leader in each group who is responsible for communication between groups (Fig. 1).

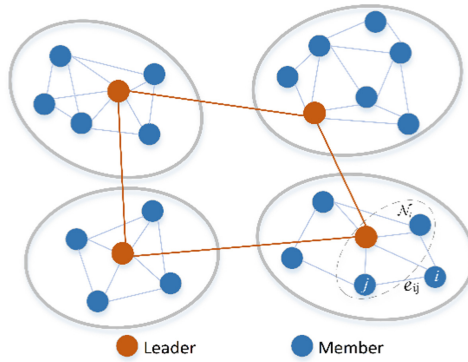


Fig. 1. Groups in the multiagent network

The neighbor agents of a_i are defined as $\mathcal{N}_i$; then, $e_{ij} = 1$ if $a_j \in \mathcal{N}_i$ and $e_{ij} = 0$ otherwise. Similarly, the neighbor groups of G_x are defined as $\mathcal{N}_{G_x}$, and the relations among groups can be represented by leaders. Furthermore, the communication cost between a_i and a_j is defined as

$$c_{ij} = \begin{cases} \frac{\theta}{c_i + c_j} & \text{if } e_{ij} = 1 \\ \infty & \text{elsewise} \end{cases} \quad (1)$$

where c_i, c_j represent the communication capabilities of the two agents, and $\theta = \{1, 5\}$ is the coefficient. There is a low communication cost when the two agents are neighbors in a group, and thus, $\theta = 1$; when the two agents are the leaders of the two groups, $\theta = 5$.

Assume that there are l types of skills in the MAS, which are denoted as $Skill = \{s_1, s_2, \dots, s_l\}$ and that each agent has one skill $S_i \in Skill$. The capability of agent a_i

when using its skill is defined as b_i . There are several tasks $T = \{t_1, t_2, \dots, t_n\}$, and the skill required by $t_k \in T$ is S_{t_k} . If all requirements of the task can be satisfied, then a successful task allocation has been carried out. Then, the problem of task allocation can be considered as choosing a group with the best performance to carry out the task. However, it may be difficult for an agent group to perform a complex task independently, and in such cases, the assistance of neighbor groups may be needed. Therefore, not only the group self but also the relation context should be considered in task allocation. The more high-quality neighbor groups a group has, the easier it is for it to find help in performing a task. To achieve greater efficiency in a MAS, the factors that can be considered in task allocation include the satisfaction of skill requirements, capability b_i , reputation R_i , and the communication cost.

The reputation mechanism is introduced in task allocation, where agents can accumulate their reputations by successful task execution. The reputation mechanism can improve the success rate and completion quality of tasks, thereby improving the group efficiency of MASs. Suppose that each task has a budget, ρ_k , for reputation reward and that the reputations obtained by agents are calculated from the contribution of each agent and the quality of task completion [10]:

$$R_i = R_i + \frac{b_i}{|W(t_k)|\mu_k} \rho_k \chi_k \quad (2)$$

where b_i is the capability of agent a_i , $W(t_k)$ is the set of agents that participate in the task, μ_k is the average capability of $W(t_k)$, and χ_k is the quality of task completion.

We propose a heuristic algorithm in [10] to address the problem of task allocation for social multiagent groups and provide a detailed description and analysis. Since the focus of this paper is the problem of network relation adaptation, the specific process of task allocation is not described in detail here.

2.2 Relation Adaptation of Multiagent Groups

Since the leader agent has particularity compared with common agents in and between groups, a hybrid network model is used in this paper, in which the characteristics of centralized and decentralized networks are combined. In the hybrid model, the information of each agent is summarized by the leader, who is also responsible for coordination with other groups. Meanwhile, task allocation within the group is still decentralized. Moreover, the hybrid model can reduce the load of leader agents in large-scale centralized systems, and the problem of decisions in a decentralized model being uncontrollable can be solved [10].

Relation adaptation is the self-organization of agent groups to adjust the network relation in the absence of external instructions to obtain better task allocation efficiency. The decisions made by the group for relation adaptation are called actions, and two separate action types are defined in our model, where

$$Action = \{adaption, no_action\}$$

represent the relation adaptation of group networks and no action, respectively.

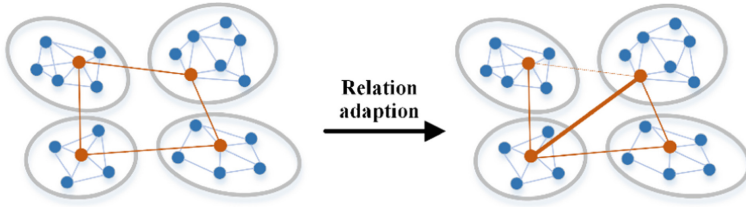


Fig. 2. Relation Adaptation of Multiagent Groups

Definition 1: Relation adaptation of agent groups. An agent group breaks the relation with one of its neighbor groups and establishes a new neighbor relation with another group.

The action of relation adaptation is shown in Fig. 2, in which the dotted line represents the original relation, and the thick solid line represents the newly established relation. Since the agent group needs to find assistants from its neighbor groups to complete complex tasks, the network relation of the group is particularly important. When a group believes that a neighbor group cannot provide effective assistance according to historical tasks, it will abandon that neighbor group and choose a new neighbor group. However, if all groups tend to establish relations with the best group, then the distributed network evolves into a centralized structure, which greatly increases the burden of communication and decision making of leader agents. Therefore, to avoid this problem, an upper limit for the number of neighbors should be set. When a group takes the action of adaptation but the number of neighbors of the target group has reached the upper limit, the group rewires its relation to its second-best target group.

3 Relation Adaptation Based on Q-Learning

Q-learning is well-understood and the most widely used reinforcement learning algorithm [15]. This method is utilized in the relation adaptation of multiagent networks so that each agent group can automatically and dynamically adjust the network relation in the process of completing tasks to improve the performance of the MAS.

3.1 Stateless Q-Learning

The standard Q-learning method can be described as a Markov decision process (MDP), where the aim of each agent is to find the optimal action strategy that maximizes the expected future reward [16]. In Q-learning, all actions in each state correspond to a Q-value, which is the expected reward that an agent can obtain after taking a certain action in that state. Whenever an agent takes an action, the Q-value is updated according to the reward of the current action until the estimated Q-value is approximately equal to the actual future reward.

However, in large-scale scenarios, the action solution of each distributed agent requires a long process of trial and error. Moreover, it is difficult to accurately define the state in complex network systems, so standard Q-learning cannot be used to optimize

the relation adaptation strategy. To address this problem, the standard Q-learning algorithm was simplified to stateless Q-learning in [17], which can also achieve the expected learning performance. In the problem studied in this paper, a complex network relation cannot be clearly defined as a certain state, so relation adaptation can be regarded as a stateless Q-learning process.

In stateless Q-learning, each agent group G_x updates its Q-value according to the reward, the update equation of which is

$$Q_x(act) \leftarrow \underbrace{Q_x(act)}_{\text{old value}} + \alpha \underbrace{[r_x^t - Q_x(act)]}_{\text{learned value}} \quad (3)$$

where act represents the action, $\alpha \in (0, 1]$ is the learning rate, and r_x^t is the reward in the t th learning.

Each time the agent group makes an action selection, the greedy strategy is used to choose an action with the most expected reward; that is,

$$\overline{act} = \arg \max_{act \in Action} (Q_x(act)) \quad (4)$$

When the Q-values of two actions are equal, such as the initial condition where all Q-values are 0, the group randomly picks one of the actions to try, thus continuing the learning process.

In addition, the learning rate has an important influence on the convergence process of Q-learning. We use the “Win or Learn Fast” (WoLF) method to adjust the learning rate and achieve the convergence of the action strategy. The basic idea of WoLF is that the agent should learn slowly when it is winning and learn fast when losing [19]. This variable learning rate can be described as

$$\alpha = \begin{cases} 0.1, & r_x^t < 0 \\ 0.01, & r_x^t > 0 \end{cases} \quad (5)$$

where the lower learning rate α is used for successful trials and the higher learning rate α is used for failed trials.

3.2 Evolution Reward

In addition to the model and learning rate of stateless Q-learning, it is also necessary to introduce an evaluation method to estimate the result of the action taken by the group. Then, the reward function is designed as

$$r_x^t = value_x^T - value_x^{T-1} \quad (6)$$

where $value_x^T$ is the evaluation of the MAS by group G_x in the T th iteration. It should be noted that t here refers to the iteration times of a certain action, and T refers to the sum of iteration times of two actions. It can be seen from Eq. (6) that if the system valuation

decreases after an action is taken, then the trial of the action fails, and the reward is negative.

The evaluation of the current network by group G_x is defined as

$$value_x^T = \underbrace{\tau_1\varphi_x + \tau_2\vartheta_x}_{local\ group} + \underbrace{\tau_3\phi + \tau_4\beta + \tau_5\psi}_{global} \quad (7)$$

where φ_x is the task participation rate of G_x , ϑ_x is the average reputation of G_x , ϕ is the task success rate, β is the average task completion quality, ψ is the inverse of average communication cost in tasks, and $\tau_1, \tau_2, \tau_3, \tau_4, \tau_5$ are coefficients.

It can be seen from Eq. (7) that the evaluation of the system depends on two aspects: the first two items are considered from the local perspective of the group, which represents its own benefit, and the last three items are considered from the global perspective, which represents system efficacy. This evaluation method means that the agent group should not only improve its own benefits but also have the overall consciousness to realize win-win cooperation between the part and whole.

3.3 Algorithm of Relation Adaptation

Algorithm 1 is designed to adjust the relation between groups. First, the parameters and Q-values need to be initialized, and then, the action decisions and iterative evolutions are carried out by agent groups within the iterative time T_d . The leaders of each group make action decisions according to the Q-value. If relation adaptation is selected, then the method of second-order neighbor rewiring can be used for evolution. The second-order neighbors can be described as

$$\mathcal{N}_{G_x}^2 = \bigcup_{G_y \subseteq G} \mathcal{N}_{G_y} - \mathcal{N}_{G_x} - G_x \quad (8)$$

That is, the second-order neighbors are the neighbors of a group's neighbors.

However, it should be noted that if a group directly abandons a neighbor and connects to a second-order neighbor without considering the relation between the new and old neighbors, then it easily leads to global topology disconnection. Therefore, the method of second-order neighbor recommendation is used in relation adaptation here (Lines 11–15). Group G_x first selects neighbor G_r , who has the lowest reputation, as the neighbor to be abandoned; then, the new neighbor is recommended by G_r among its neighbors. The candidate set of new neighbors is

$$\mathcal{N}_{G_r}^{re} \ominus = \mathcal{N}_{G_x}^2 \cap \mathcal{N}_{G_r} \quad (9)$$

Algorithm 1: Relation Adaptation of Groups in the MAS

```

01: Initialize parameters  $T_d, size\_neigh, k$ ;
02: Set  $T = 0$  and  $Q_i(act) = 0$ ;
03: while  $T \leq T_d$ 
04:   for  $G_x \subseteq G$  and  $T \neq 0$  do
05:     if  $Q_x(adaption) = Q_x(no\_action)$ 
06:        $act_m = randm(adaption, no\_action)$ ;
07:     else
08:        $act_m = \arg \max_{act \in Action} (Q_x(act))$ ;
09:     end if
10:     if  $act_m = adaption$ 
11:        $G_r = \arg \min_{G_s \in \mathcal{N}_{G_x}} (R_{G_s})$ ;
12:        $\mathcal{N}'_{G_r} = \{G_y \mid G_y \subseteq \mathcal{N}_{G_r}^{re}, |\mathcal{N}_{G_y}| < size\_neigh\}$ 
13:        $G_m = \arg \max_{G_s \in \mathcal{N}'_{G_r}} (R_{G_s})$ 
14:        $\mathcal{N}_{G_x} = \mathcal{N}_{G_x} \cup G_m$ ,  $\mathcal{N}_{G_m} = \mathcal{N}_{G_m} \cup G_x$ ;
15:        $\mathcal{N}_{G_x} = \mathcal{N}_{G_x} - G_r$ ,  $\mathcal{N}_{G_r} = \mathcal{N}_{G_r} - G_x$ ;
16:     end if
17:   end for
18:   execute  $k$  tasks, and obtain reputations;
19:   for  $G_x \subseteq G$  do
20:     calculate  $value_x^T$  according to Eq. (7);
21:     if  $T \neq 0$ 
22:       calculate the reward  $r_x^t$  according to Eq. (6);
23:        $Q_x(act_m) = (1 - \alpha)Q_x(act_m) + \alpha r_x^t$ ;
24:     end if
25:   end for
26:    $T = T + 1$ ;
27: end while

```

The group with the highest reputation in the candidate new neighbor set will be the new neighbor of G_x , where the reputation of a group is the sum of the reputations of all its agents. Global topological connectivity can be ensured with the method of second-order neighbor recommendation. In Line 12 of Algorithm 1, $|\mathcal{N}_{G_y}|$ represents the number of neighbors of group G_y , and $size_neigh$ is set as the upper limit of neighbors. Here, $size_neigh$ is utilized to avoid all groups tending to establish relations with the best group, which increases the burden of communication of the group leader. After each action decision, the system executes k tasks and updates the Q-value to optimize the action strategy.

It is known that second-order neighbor recommendation is a special case of second-order neighbor rewiring. Since the group communicates only with the recommender (i.e., the abandoned neighbor), there is no need to find new neighbors in all second-order neighbors. Therefore, while ensuring global connectivity, the communication cost of relation adaptation can also be reduced by the proposed method.

4 Experiments and Analysis

We conduct experiments to verify the performance of the proposed algorithm for relation adaptation. There are 160 agents in our experiments, which are divided into 16 groups at random, and the upper limit of neighbors is set to 8. The initial network is randomly generated and guaranteed to be connected.

There are 15 types of skills defined in the experiment. Based on these skills, 500 tasks are created, with each task containing 15 skill requirements. The agent has a skill randomly, and the capability is determined by a normal distribution $N(\mu_0, \sigma_0^2)$, where $\mu_0 = 10$ and $\sigma_0 = 2$. The reputations of agents are 0 at the beginning. Besides, the reputation budget of each task is set to 10, and the expected agent capability of each task is $\mathcal{B}_k = \mu_0$. In the experiment, agent groups execute relation adaptation and update Q-values after every 50 tasks. All the factors are considered equally in this paper, so the parameters are set as $\tau_1 = \tau_2 = \tau_3 = \tau_4 = \tau_5 = 1$.

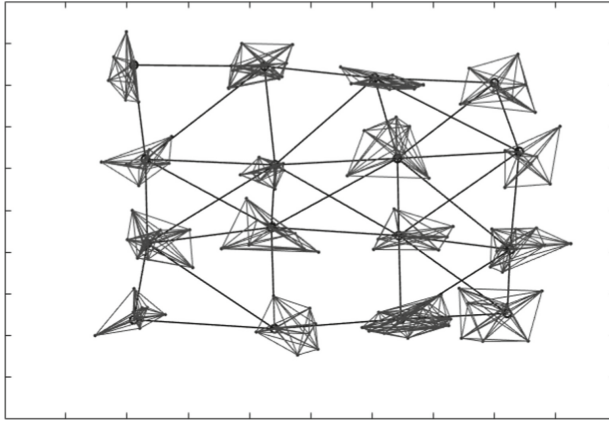


Fig. 3. Initial topology of the MAS

To show the performance advantages of the network adaptation, we compare and analyze task performance after adaptation and that without adaptation. The initial network relation is shown in Fig. 3, and the network relation after adaptation with 500 tasks is shown in Fig. 4. The success rate of task allocation is shown in Fig. 5, and it can be seen that the success rate of the MAS is significantly improved after relation adaptation. In addition, we also analyze the communication cost of tasks. From the result in Fig. 6, there are lower communication cost after relation adaptation than that without adaptation. Therefore, the network relation adaptation proposed in this paper can effectively improve the efficiency of task allocation for MASs.

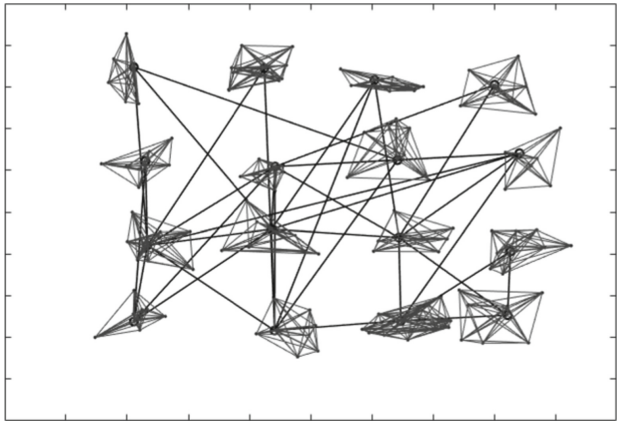


Fig. 4. Adapted topology of the MAS

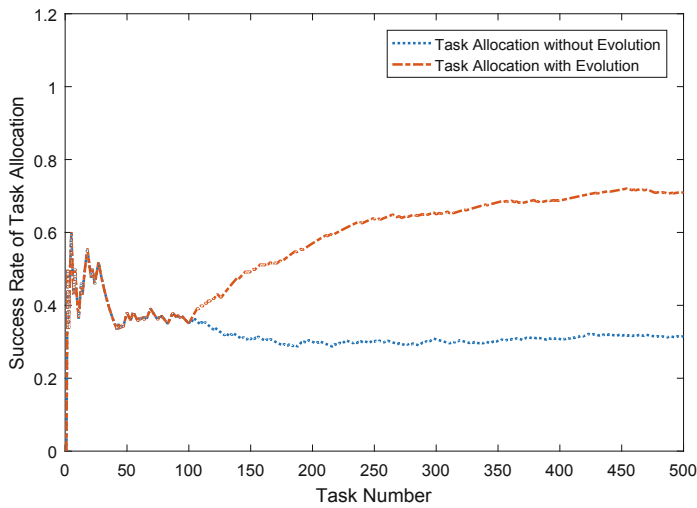


Fig. 5. Success rate of task allocation with relation adaptation and that without adaptation

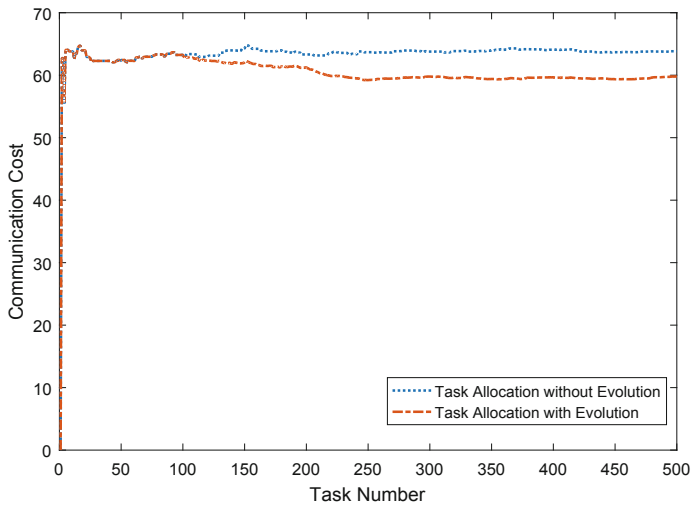


Fig. 6. Communication cost of task allocation

5 Conclusions

This paper addresses the relation adaptation of grouped multiagent networks. According to the environment and task requirements, the original network relation is not necessarily optimal. Therefore, a network relation adaptation method for grouped MASs based on Q-learning is proposed in this paper. Each group decides whether to change neighbors and which neighbor to replace through system evaluation and iterative learning, thus allowing for the optimal network structure to be formed. Furthermore, the results of the experiments show that the method proposed in this paper can effectively improve the success rate of task allocation for grouped MASs .

Acknowledgments. This work was supported by the Key Project of Strengthened Basic Research under Grant 2020-JCJQ-ZD-02212.

References

1. Ye, D., Zhang, M., Vasilakos, A.V.: A survey of self-organization mechanisms in multiagent systems. *IEEE Trans. Syst. Man Cybern. -Syst.* **47**(3), 441–461 (2016)
2. Luz, N., Silva, N., Novais, P.: A survey of task-oriented crowdsourcing. *Artif. Intell. Rev.* **44**(2), 187–213 (2014). <https://doi.org/10.1007/s10462-014-9423-5>
3. Jiang, J., An, B., Jiang, Y., Zhang, C., Bu, Z., Cao, J.: Group-oriented task allocation for crowdsourcing in social networks. *IEEE Trans. -Syst. Man Cybern. Syst.* **51**(7), 4417–4432 (2021)
4. Wang, S., Wan, J., Zhang, D., Li, D., Zhang, C.: Towards smart factory for industry 4.0: a self-organized multi-agent system with big data based feedback and coordination. *Comput. Netw.* **101**, 158–168 (2016)

5. Xue, F., et al.: A hybrid many-objective competitive swarm optimization algorithm for large-scale multirobot task allocation problem. *Int. J. Mach. Learn. Cybern.* **12**(4), 943–957 (2020). <https://doi.org/10.1007/s13042-020-01213-4>
6. Zhou, Y., Fei, C., Wang, W.: Task similarity-based task allocation approach in multi-agent engineering software systems. *J. Inf. Sci. Eng.* **32**(4), 1021–1039 (2016)
7. Lappas, T., Liu, K., Terzi, E.: Finding a team of experts in social networks. In: *Proceedings of 15th ACM SIGKDD International Conference on Knowledge Discovery Data Mining*, pp. 467–475, Paris (2009)
8. Ren, J., Zhang, Y., Zhang, K., Shen, X.: SACRM: social aware crowdsourcing with reputation management in mobile sensing. *Comput. Commun.* **65**, 55–65 (2015)
9. Wang, W., Jiang, Y.: Community-aware task allocation for social networked multiagent systems. *IEEE Trans. Cybern.* **44**(49), 1529–1543 (2014)
10. Li, K., Wu, S., Wen, Y., Wang, Y.: Task allocation of multiagent groups in social networked systems. *IEEE Internet Things J.* **9**, 12194–12208 (2021). <https://doi.org/10.1109/JIOT.2021.3134681>
11. Kota, R., Gibbins, N., Jennings, N.R.: Decentralized approaches for self-adaptation in agent organizations. *ACM Trans. Auton. Adapt. Syst.* **7**(1), 1–28 (2012)
12. Peleteiro, A., Burguillo, J. C., Chong, S. Y.: Exploring indirect reciprocity in complex networks using coalitions and rewiring. In: *International Conference on Autonomous Agents Multiagent System*, pp. 669–676. Paris (2014)
13. Yang, G., Zhu, C., Zhang, W.: Adaptive and probabilistic strategy evolution in dynamical networks. *Phys. A* **518**, 99–110 (2019)
14. Ye, D., Zhang, M., Sutanto, D.: Cloning, resource exchange, and relation adaptation: an integrative self-organisation mechanism in a distributed agent network. *IEEE Trans. Parallel Distrib. Syst.* **25**(4), 887–897 (2014)
15. Morozs, N., Clarke, T., Grace, D.: Distributed heuristically accelerated Q-learning for robust cognitive spectrum management in LTE cellular systems. *IEEE Trans. Mob. Comput.* **15**(4), 817–825 (2016)
16. Lv, S., Xu, X., Han, S., Tao, X., Zhang, P.: Energy-efficient secure short-packet transmission in NOMA-assisted mMTC networks with relaying. *IEEE Trans. Veh. Technol.* **71**, 1699–1712 (2021). <https://doi.org/10.1109/TVT.2021.3133907>
17. Claus C., Boutilier, C.: The dynamics of reinforcement learning in cooperative multiagent systems. In: *National Conference on Artificial Intelligence*, pp. 746–752. Madison (1998)
18. Jiang, T., Zhao, Q., Grace, D., Burr, A.G., Clarke, T.: Single-state Q-learning for self-organised radio resource management in dual-hop 5G high capacity density networks. *Trans. Emerg. Telecommun. Technol.* **27**(12), 1628–1640 (2016)
19. Morozs, N., Clarke, T., Grace, D., Zhao, Q.: Distributed Q-learning based dynamic spectrum management in cognitive cellular systems: Choosing the right learning rate. In *IEEE Symposium Computers and Communications*, pp. 1–6. Funchal (2014)