2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

Task Allocation in Heterogeneous Multi-Robot Systems Based on Preference-Driven Hedonic Game

Liwang Zhang†, Minglong Li†, Wenjing Yang and Shaowu Yang*

Abstract— **Multiple preferences between robots and tasks have been largely overlooked in previous research on Multi-Robot Task Allocation (MRTA) problems. In this paper, we propose a preference-driven approach based on hedonic game to address the task allocation problem of muti-robot systems in emergency rescue scenarios. We present a distributed framework considering various preferences between robots and tasks to determine the division of coalitions in such problems and evaluate the scalability and adaptability of our algorithm through relevant experiments. Furthermore, considering the strict communication limitations in emergency rescue scenarios, we have verified that our algorithm can efficiently converge to a Nash-stable coalition partition even in conditions of insufficient communication distance.**

I. Introduction

In Multi-Robot Systems (MRS), multiple robots cooperate and form coalitions to perform tasks efficiently and reliably. Multi-Robot Task Allocation (MRTA) as a key technology in MRS has been widely used in various fields such as load balancing [1] [2], search and rescue [3] [4], intelligent transportation [5], national defense construction [6], and so on. Let's consider a scenario with a group of electric vehicles and several charging stations. Each electric vehicle exhibits a brand preference for charging stations and tends to travel to the nearest available charging station. The challenge is to match these vehicles with available charging stations, which can be abstracted as an MRTA problem. There are various classical solutions to solve MRTA problems, including market-based methods [3] [7], optimization-based methods [8] [9], learning-based methods [10] [11] and so on. However, there are few articles based on these methods that deal with multiple preferences issues similar to the scenario mentioned above.

In this paper, we model the task allocation problem as a hedonic game [12]. In this game, robots are regarded as players who make decisions; the task allocation constitutes their strategy, and the rewards they acquire are referred to as payoffs. In the hedonic game, players selectively join the coalition based on their preferences to maximize their benefits. When every player in the game has no motivation to change their strategy, such a profile is a Nash-stable coalition partition. In MRS, game theory has been widely applied to solving task allocation problems. In [13], the authors propose a distributed multi-agent dynamic task allocation method based on the potential game, where each agent makes decisions independently based on local information. In [14], the authors use the Shapley value to assess the performance of sets of robots and tasks and partition the original problem into smaller subproblems based on the Shapley value rankings and clusters. Furthermore, in [15], a game-theoretic framework for anonymous decision-making is introduced to address the task allocation issue for multiple agents, taking into account the collaboration among self-interested agents. Based on this work, a distributed hedonic game formulation is proposed to solve the challenge of assigning multiple diverse robots to tasks. In this context, each robot determines whether to participate in a team based on the robots already assigned to a specific task [16]. However, it is noteworthy that these previous studies have yet to consider multiple types of preferences or the limitations of communication distance.

The primary research of this paper focuses on the formation of coalitions among a large number of heterogeneous rescue robots to perform various tasks in emergency rescue scenarios. We adopt a distributed approach to form coalitions. Each robot will choose to join the most suitable coalition, considering its preferences and task requirements. Robots will negotiate with each other within their communication range to reach an agreement. The goal is to find a task allocation scheme that satisfies all tasks and robot preferences while maximizing overall utility. In addition, we also consider communication limitations in emergency rescue scenarios. If the communication distance is insufficient, the communication network will not be strongly connected, resulting in the formation of multiple connected components. The robots within each connected component will establish coalitions to accomplish tasks based on identical rules. This considerably strengthens the robustness of our algorithm framework.

The main contributions of this paper are as follows:

1) It is the first time the hedonic game is applied to solve task allocation problems in emergency rescue scenarios, which has good social application value.
2) We have considered different preference relationships from the perspective of robots and tasks, which makes task allocation results more practical and effective.
3) The algorithmic framework we propose takes into account the communication distance constraints in emergency rescue scenarios, and can converge to a Nash-stable coalition partition even when communication distances are insufficient. Additionally, the allocation

This work was partially supported by the National Natural Science Foundation of China (No. 91948303).

†Equal contribution, *Corresponding author.
All authors are with the Department of Intelligent Data Science, College of Computer Science and Technology, National University of Defense Technology, Changsha 410073, China {liwangzhang22, liminglong10, wenjing.yang, shaowu.yang}@nudt.edu.cn

effect of this algorithm can also reach about 70% of the centralized algorithm allocation effect.

The structure of the remainder of this paper is as follows: In Section II, we articulate the task allocation problem and cast it as a hedonic game featuring transferable utility. In Section III, we propose a hedonic game decision algorithm to seek the Nash-stable coalition partition of the game. In Section IV, we design an emergency rescue scenario to evaluate our algorithm framework and analyze the simulation results. The concluding remarks and future plans are presented in Section V.

II. Problem Formulation

Suppose a set of robots $R = \{r_1, r_2, r_3, ..., r_I\}$, which are divided into $C = \{c_1, c_2, c_3, ..., c_k\}$ types. All robots have the ability to communicate, and they can communicate with other robots within a distance of d. Each robot has different abilities and different contributions to the task. Robots of the same kind are the same, and there is no difference.

Suppose a set of tasks $T = \{t_1, t_2, t_3, ...t_J\}$ represents J tasks that need to be completed by robots. Each task has a demand value for various types of robots and a completed reward value.

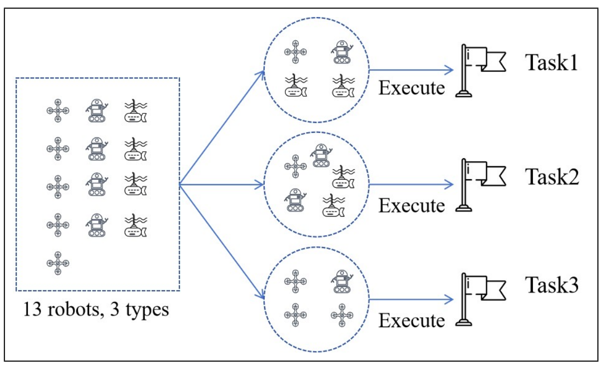


Fig. 1. Coalition Formation by Various Robot Types for Task Execution.

Definition 1 (Coalition)**.** A coalition $S_j \subseteq \Pi$ is a collection of different types of robots. These robots gather together to complete the task in the coalition through their preferences. Each coalition is mutually independent and non-overlapping.

Unlike most existing studies, we consider multiple aspects of robot and task preferences, such as the cost of robots performing tasks, the expected number of robots for the task, and the type of robot required for the task. Next, we will introduce the preferences of tasks and robots:

1) *Cost:* Robots prefer to complete tasks with less cost, so the cost of performing tasks is an essential factor for robots to choose tasks. In the actual scenario, the cost of performing tasks includes the power consumption of the robot, the wear and tear of the robot, and the driving distance of the robot. For the convenience of calculation, only the driving distance of the robot is considered as the cost in this paper.
2) *Expected number of robots:* The expected number of robots for a task is one of the preference factors. Due to the limited number of robots required for each task, when the number of robots in the coalition executing the task exceeds the expected number, the reward value of the task will decrease. This leads to a diminishing shared reward for each robot, causing robots to prefer joining coalitions with fewer members.
3) *Task type:* From the perspective of tasks, different situations of each task lead to different types of robots required, which reflects the preference relationship between tasks and robots. For example, maritime search and rescue requires more unmanned surface vessels instead of unmanned ground vehicles.

Each robot needs to choose a task to form a coalition S_j, all partitions form a set $\Pi = \{S_1, S_2, S_3, ..., S_m\}$, each robot has an individual utility $u_i : T \times S \to \mathbb{R}$. This is a function of the robot's ability, preferences, cost of performing the task, and the expected number of robots for the task. The objective of the algorithmic framework introduced in this paper is to identify a suitable partition set that maximizes the cumulative utility of all robots. The problem is formulated as follows:

$$\max_{\{x_{ij}\}} \sum_{\forall r_i \in R} \sum_{\forall t_j \in T} u_i\left(t_j, S_j\right) \tag{1}$$

subject to

$$\begin{gathered} \textstyle\sum_{\forall t_j \in T} x_{ij} \leq 1, \quad \forall r_i \in R \\ x_{ij} \in \{0,1\}, \quad \forall r_i \in R, \forall t_j \in T \\ S_j \cap S_{j'} = \emptyset, \quad \forall S_j, S_{j'} \in \Pi, j \neq j' \end{gathered}$$

where x_{ij} is a binary decision variable, if task j is assigned to robot r_i, then $x_{ij} = 1$, otherwise $x_{ij} = 0$; $u_i\left(t_j, S_j\right)$ represents the individual utility value obtained by the coalition S_j where the robot r_i belongs to jointly complete the task t_j. This utility value is the reward value obtained by the robot minus the cost consumed by the robot to complete the task,

$$u_i\left(t_j, S_j\right) = \frac{\sum_{c_k \in C} r(t_j, S_j^{c_k})}{|S_j|} - cost_{ij} \tag{2}$$

where $cost_{ij}$ is the cost that robot r_i needs to pay for task t_j, and $r(t_j, S_j^{c_k})$ represents the reward derived from task t_j when it is executed collectively by $S_j^{c_k}$, it can be defined as follows:

$$r(t_j, S_j^{c_k}) = \begin{cases} v_j log_{w_j^{c_k}}(|S_j^{c_k}| + \varepsilon_j), & |S_j^{c_k}| < w_j^{c_k} \\ v_j log_{w_j^{c_k}}(w_j^{c_k} + \varepsilon_j), & |S_j^{c_k}| \geq w_j^{c_k} \end{cases} \tag{3}$$

where v_j represents the base reward value for a task, $w_j^{c_k}$ is the expected value of task j for the number of robots of type c_k and $\varepsilon_j > 0$ denotes the design parameter associated with the diminishing marginal gain.

The reward value of the task gradually increases with the number of robots participating. Still, once the ability value of the robot exceeds the ability value required by the task, the reward value of the task will gradually decrease, which can reduce unnecessary waste of resources.

TABLE I
NOMENCLATURE

Symbol	Description
R	a set of I robots
T	a set of J tasks
r_i	the i-th robot
t_j	the j-th task
c_k	the type of robot r_i
Π	a partition
S_j	the coalition that performs task t_j
$\succ_i$	the strict preference relation of robot r_i
(t_j, S_j)	a task-coalition pair
$w_j^{c_k}$	task t_j requires the number of robots of type c_k
v_j	the basic task value of task t_j
$cost_{ij}$	the cost consumed by the robot r_i to perform the task t_j
D_i	a set of neighbors of the robot r_i.
d	the communication distance between robots

Assumption 1. This paper studies the ST-MR-IA problem, as discussed in [17]. The problem involves multiple robots collaborating to complete a task, with each robot handling one task at a time. Additionally, all the tasks that need to be assigned to the robots are known in advance.

Assumption 2. Each type of robot has different capabilities and preferences, and each type of task has different preferences for different robot types.

Assumption 3. The communication between robots can only be completed within a specific distance range, and indirect communication can also be carried out through other robots.

Algorithm 1: Decision-making Algorithm for each robot r_i

Input: R, T, D
Output: The partition Π

Initialize $\mathbf{r\_satisfied} \leftarrow 0$; $\xi^i \leftarrow 0$; $\Pi^i \leftarrow \{S_\phi = R, S_j = \emptyset \, \forall t_j \in T\}$;
while $\mathbf{r\_satisfied} = true$ **do**
    if $\mathbf{r\_satisfied^i} = 0$ **then**
        Based on Equation (2), calculate the utility to determine the S_{j*} and t_{j*} that maximize its utility;
        if $(t_{j*}, S_{j*}) \succ_i \left(t_{\Pi^i(\xi^i)}, S_{\Pi^i(\xi^i)}\right)$ **then**
            Join S_{j*} and update Π^i;
            $\xi^i \leftarrow \xi^i + 1$;
        end
        $\mathbf{r\_satisfied^i} = 1$;
    end
    Broadcast $M^i = \left\{\Pi^i, \mathbf{r\_satisfied}, \xi^i\right\}$ and receive M^k from its neighbors D_i;
    Collect all the messages and construct $\mathcal{M}_{all}^i = \left\{M^i, M^k\right\}$;
    for each message $M^k \in \mathcal{M}_{all}^i$ **do**
        if $\xi^k \geq \xi^i$ **then**
            $M^i \leftarrow M^k$;
            if $\Pi^i \neq \Pi^k$ **then**
                $\mathbf{r\_satisfied^i} = 0$;
            end
        end
    end
    if $\mathbf{r\_satisfied} = 1$ **then**
        return Π^0;
    end
end

III. Algorithm Description

A. *Hedonic Coalition Formation Algorithm*

In this subsection, the task allocation process is modelled as a hedonic game, in which participants cooperate according to their preferences to establish a stable coalition to improve the overall self-benefit.

Definition 2 (Preference)**.** The individual preference relationship can be given by the utility function, such as $u_i(t_1, S_1) > u_i(t_2, S_2)$, which means $(t_1, S_1) \succ_i (t_2, S_2)$.

In Algorithm 1, each robot has local variables for decision-making, such as $\mathbf{r\_satisfied}$, ξ^i, Π^i, where Π^i represents the local coalition partition of r_i, $\mathbf{r\_satisfied}$ indicates whether each robot satisfies the current allocation, and ξ^i is a positive integer, indicating the number of times Π^i is updated. Given an initial allocation Π^i, each robot will identify the task that maximizes its utility value from its respective preferences. If the new coalition is better than its original one, it updates the local partition and increases ξ^i. In any case, the robot has determined that the current coalition is ideal, so the current partition is satisfied. The robot then broadcasts local information to neighbour robots by sending messages and collecting information from all neighbour robots. Since each robot makes independent local decisions, conflicts are bound to arise. In such cases, we adopt a decentralized approach to selecting a common coalition, and if no consensus is reached, all robots will be dissatisfied.

Algorithm 1 realizes finding a Nash-stable coalition partition so that each coalition of tasks can be formed by robots with their preference. However, Algorithm 1 only completes the division of the coalition without considering the constraint of robot numbers. If the number of robots in the coalition exceeds the number required for the task, it will cause a waste of resources. Therefore, Algorithm 2 selects the most suitable robots from each coalition to complete their respective tasks based on the utility values and updates the partitions. The remaining robots are kept idle to reduce unnecessary consumption.

B. *Analysis of Existence and Convergence*

Subsequently, we will establish that the preference-driven hedonic coalition formation game we have proposed consistently yields a Nash-stable partition and guarantees conver-

Algorithm 2: Choose the most suitable number of robots for each task t_j

Input: $\Pi = \{S_1, S_2, S_3, ..., S_J\}$ from Algorithm 1
Output: The final partition Π

```
for each type c_k ∈ C do
    if |S_j^{c_k}| > w_j^{c_k} then
        Sort S_j^{c_k} by utility in descending order
        Select the top num_j most suitable robots to
          form a new S_j^*
        Update Π with S_j^*
    end
end
```

gence. First of all, it is necessary to introduce the definition of Nash stable coalition partition:

Definition 3 (Nash-stable Coalition Partition)**.** Suppose no robot in a partition can unilaterally move from the current coalition to another coalition. In that case, the partition is Nash-stable coalition partition, that is, $\left(t_{\Pi(i)}, S_{\Pi(i)}\right) \succ_i \left(t_j, S_j \cup \{r_i\}\right), \forall S_j \in \Pi$, where $\Pi(i)$ represents the index of the task allocated to robot i.

Before proving the Nash-stable coalition partition, we must first introduce the concept of the potential game:

Definition 4 (Exact Potential Game [18])**.** Given a game, if exist a potential function P such that $\forall r_i \in R, \forall S_{\Pi(i)}, S_{\Pi'(i)} \in \Pi$, satisfying the condition that follows, we can call this game an exact potential game.

$$\begin{aligned} &U_i\left(S_{\Pi(i)}, S_{\Pi(-i)}\right) - U_i\left(S_{\Pi'(i)}, S_{\Pi'(-i)}\right) \\ &= P\left(S_{\Pi(i)}, S_{\Pi(-i)}\right) - P\left(S_{\Pi'(i)}, S_{\Pi'(-i)}\right) \end{aligned} \tag{4}$$

This implies that we can associate the change in individual returns with a potential function. For any given robot i, if its optimal response (considering other robots' strategies) leads to an increase in payoff, the potential function value also increases. Consequently, we obtain a Nash equilibrium when the potential function reaches its maximum value. Therefore, we can approach equilibrium research by studying the potential function.

Theorem 1. The proposed preference-driven hedonic coalition formation game ensures the existence of Nash stable coalition partition and can eventually form a stable coalition partition.

Proof: First of all, the potential function is defined as follows:

$$P = \sum_{r_n \in R} u_n(\Pi(n), S_{\Pi(n)}) \tag{5}$$

which is the function that represents the sum of utility values for all robots. Assuming robot i changes its strategy from the current task t_j to task $t_j^{'}$, and the coalition partition Π changes to $\Pi^{'}$, the potential function can be represented as:

$$\begin{aligned} &P\left(S_{\Pi(i)}, S_{\Pi(-i)}\right) - P\left(S_{\Pi'(i)}, S_{\Pi'(-i)}\right) \\ &= \sum_{r_n \in R} u_n(\Pi(n), S_{\Pi(n)}) - \sum_{r_n \in R} u_n\left(\Pi'(n), S_{\Pi'(n)}\right) \\ &= \sum_{r_n \in S_j} u_n(t_j, S_j) + \sum_{r_n \in S_{j'}} u_n(t_{j'}, S_{j'}) \\ &- \sum_{r_n \in S_j \setminus \{r_i\}} u_n(t_j, S_j \setminus \{r_i\}) \\ &- \sum_{r_n \in S_{j'} \cup \{r_i\}} u_n\left(t_{j'}, S_{j'} \cup \{r_i\}\right) \\ &+ \sum_{r_n \in R \setminus S_j \setminus S_{j'}} \left(u_n(\Pi(n), S_{\Pi(n)}) - u_n(\Pi(n), S_{\Pi'(n)})\right) \end{aligned} \tag{6}$$

On the basis of Equation (2), robot strategy changes will only affect the original coalition S_j and the current coalition S_j'. Therefore, the utility values of other coalitions remain unchanged,

$$u_n(\Pi(n), S_{\Pi(n)}) - u_n(\Pi(n), S_{\Pi'(n)}) = 0, \quad \forall r_n \in R \setminus S_j \setminus S_{j'} \tag{7}$$

After calculating the variation of the utility function before and after the robot changes its strategy, we will now discuss the changes in the utility function. First, let's give the utility function before the robot changes its strategy:

$$\begin{aligned} U_i\left(S_{\Pi(i)}, S_{\Pi(-i)}\right) = &\sum_{r_n \in S_j} u_n(t_j, S_j) \\ &+ \sum_{r_n \in S_{j'}} u_n\left(t_{j'}, S_{j'}\right) \end{aligned} \tag{8}$$

Next, let's give the utility function after the robot changes its strategy:

$$\begin{aligned} U_i\left(S_{\Pi'(i)}, S_{\Pi'(-i)}\right) = &\sum_{r_n \in S_j \setminus \{r_i\}} u_n(t_j, S_j \setminus \{r_i\}) \\ &+ \sum_{r_n \in S_{j'} \cup \{r_i\}} u_n\left(t_{j'}, S_{j'} \cup \{r_i\}\right) \end{aligned} \tag{9}$$

Thus, the utility function's variation can be expressed as follows:

$$\begin{aligned} &U_i\left(S_{\Pi(i)}, S_{\Pi(-i)}\right) - U_i\left(S_{\Pi'(i)}, S_{\Pi'(-i)}\right) \\ &= \sum_{r_n \in S_j} u_n(t_j, S_j) + \sum_{r_n \in S_{j'}} u_n(t_{j'}, S_{j'}) \\ &- \sum_{r_n \in S_j \setminus \{r_i\}} u_n(t_j, S_j \setminus \{r_i\}) \\ &- \sum_{r_n \in S_{j'} \cup \{r_i\}} u_n\left(t_{j'}, S_{j'} \cup \{r_i\}\right) \\ &= P\left(S_{\Pi(i)}, S_{\Pi(-i)}\right) - P\left(S_{\Pi'(i)}, S_{\Pi'(-i)}\right) \end{aligned} \tag{10}$$

It indicates that the variation in the utility function is consistent with the variation in the potential function. According to Definition 4, we can conclude that it will eventually converge to a Nash equilibrium.

TABLE II
MINIMUM NUMBER OF ASSIGNED ROBOTS

Task Id	Task Type	UAV	UGV	USV
1	collapse	6	7	3
2	landslides	9	2	3
3	epidemics	2	9	7
4	tsunami	2	3	12
5	fire	6	6	3

IV. EXPERIMENTS

In this section, we employ simulation outcomes to validate the efficacy of the solution presented herein, demonstrating its superior performance in emergency response scenarios.

A. Mission Scenario and Settings

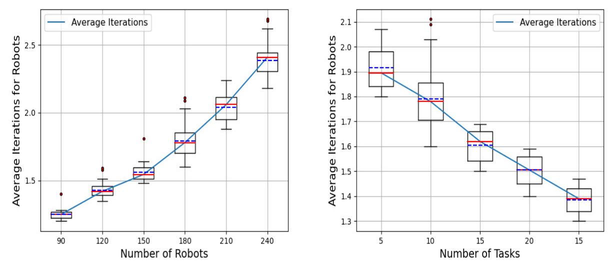
The experimental setup aims to replicate real-life situations as accurately as possible. Earthquakes are one of the most common natural disasters, causing immeasurable losses to human life and property. Strong earthquakes not only lead to building collapses but also give rise to secondary disasters such as landslides, tsunamis, fires, and epidemics. These disasters often involve complex and challenging relief tasks that require the coordination of various types of robots for rescue operations. In simulation experiments, five common types of disaster relief tasks in earthquake scenarios are considered, with three types of robots: unmanned aerial vehicles (UAVs) responsible for tasks like information gathering and communication relay, unmanned ground vehicles (UGVs) responsible for tasks such as transporting the injured and conducting geographical surveys, and unmanned surface vessels (USVs) responsible for firefighting rescue and water quality inspection tasks. Each task has different requirements for the type and quantity of robots. For instance, collapse disasters require more UAVs and UGVs, while the demand for USVs is relatively lower. However, tsunami disasters necessitate more USVs, while the requirements for UAVs and UGVs are comparatively lower.

In each run, I robots are randomly distributed within a region of size 300m $\times$ 300m, while J tasks are randomly distributed within an area of size 500m $\times$ 500m. The tasks are divided into five types, and the required robot types and quantities for each task are shown in Table II.

B. Analysis of Experimental Results

In this subsection, we will verify the scalability, adaptability, and robustness of the algorithm framework through designing different experiments.

1) Scalability: To validate the scalability of the algorithm framework, we conducted 100 Monte Carlo experiments in two groups. One group had a fixed number of tasks at 10 and varied the number of robots between 90, 120, 150, 180, 210, and 240. The other group had a fixed number of robots at 180 and varied the number of tasks between 5, 10, 15, 20, and 25. The communication distance between robots was 300 to ensure a strongly connected communication network. Fig. 2 presents the statistical outcomes through box plots, with the solid blue line indicating the average value of the 100 experiments for each condition. Based on Fig. 2(a), it is evident that the number of iterations increases linearly as more robots are involved. In Fig. 2(b), we can observe a decreasing trend in the number of iterations as the number of tasks increases. This could be attributed to a decrease in the probability of conflicts among the robots with a higher number of tasks.

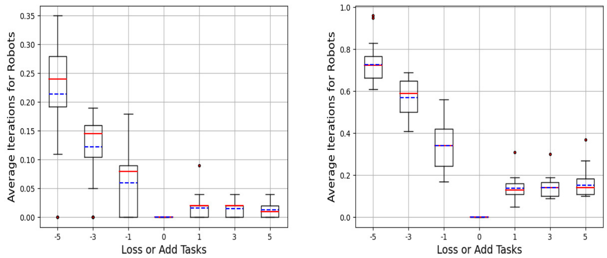


(a) (b)

Fig. 2. Comparison of Average Number of Iterations with Varying Task and Robot Numbers.

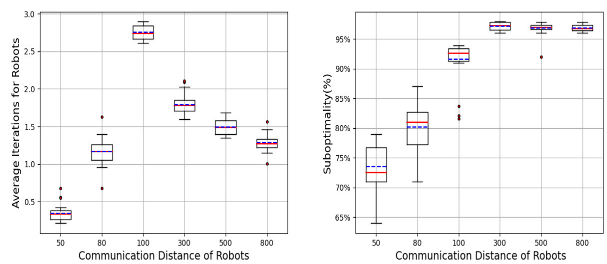
2) Adaptability: In this section, we discuss the adaptability of the proposed algorithm framework in dynamic environments when increasing or decreasing the number of tasks or robots. Assuming there are 10 tasks and 180 robots in a scenario with a strongly connected communication network. After finding a Nash-stable coalition partition, we divided the experiments into two groups. In one group, we fixed the number of tasks and increased or decreased the number of robots. In the other group, we fixed the number of robots and increased or decreased the number of tasks. For each dynamic environment, we conducted 100 Monte Carlo experiments, and the simulation results are shown in Fig. 3. Fig. 3(a) shows that increasing the number of robots has little impact, while decreasing the number of robots has some influence. The minimal effect of increasing the number of robots is due to the limitation imposed by the coalition size. The impact of decreasing the number of robots is that the missing robots for a task must be selected from the idle robots, requiring additional iterations. From Fig. 3(b), increasing the number of tasks has little impact, while decreasing the number of tasks has some influence. This is because increasing the number of tasks leads to a small portion of idle robots joining the coalition while decreasing the number of tasks releases all the robots responsible for those tasks.

3) Robustness: The previous experiments focused on strongly connected communication networks, ensuring direct or indirect communication between each robot. However, in real rescue scenarios, communication networks are often limited, making it difficult for robots to maintain a good communication environment. Therefore, this section evaluates the performance of the algorithm framework under a non-strongly connected communication network. In the simulation, there are 180 robots, 10 tasks, and communication distances of [50, 80, 100, 300, 500, 800]. Similarly,



(a) (b)

Fig. 3. Average Additional Iterations for Re-Convergence to Nash-Stable Partition with Partial Loss or Addition of Robots and Tasks.



(a) (b)

Fig. 4. Comparison of Average Iteration Count and Suboptimality (in Utility Value Ratio to Centralized Task Allocation) across Different Communication Distances.

Monte Carlo experiments were conducted 100 times, and the simulation results are shown in Fig. 4.

From Fig. 4(a), it can be observed that with insufficient communication distance, there are fewer iterations. This is because multiple connected components make decisions internally, and the number of robots in each connected component is relatively small, thus reducing conflicts. When the communication distance is moderate, the communication network between robots is strongly connected but not fully connected, resulting in an increased number of iterations. However, when the communication distance is sufficient for direct communication between every robot, the network becomes fully connected, enabling convenient information exchange among neighboring robots and subsequently reducing the number of iterations.

Furthermore, according to Fig. 4(b), although reducing the communication distance hampers the efficient exchange of information among robots, empirical evidence has demonstrated that the performance of task allocation is not significantly impacted. It can still achieve a performance level of at least 70% compared to centralized allocation.

V. Conclusion

The paper describes the application of the hedonic game framework in emergency rescue, addressing the problem of multi-robot task allocation with different preferences. Firstly, the Nash-stable coalition partition is determined based on the preferences of robots and tasks and then selected the most suitable robots to perform tasks. The proposed algorithm framework is theoretically proven to converge to a Nash-stable coalition partition. Finally, the algorithm's scalability, adaptability, and robustness are verified through experiments, ensuring efficient task allocation in emergency rescue scenarios with diverse robot types, varying task demands, and a non-strongly connected communication network.

In the future, we plan to consider the different priorities of tasks and the possibility of communication loss between robots during the task allocation process. It will enable us to better adapt to the requirements of various scenarios.

References

[1] B. Kopras, B. Bossy, F. Idzikowski, P. Kryszkiewicz, and H. Bogucka, "Task Allocation for Energy Optimization in Fog Computing Networks with Latency Constraints," IEEE Transactions on Communications, 70(12): 8229-8243, 2022.

[2] S. Shiekh, M. Shahid, M. Sambare, R. A. Haidri, and D. K. Yadav, "A load-balanced hybrid heuristic for allocation of batch of tasks in cloud computing environment," International Journal of Pervasive Computing and Communications, 2022.

[3] N. Hooshangi and A. A. Alesheikh, "Developing an agent-based simulation system for post-earthquake operations in uncertainty conditions: A proposed method for collaboration among agents," ISPRS International Journal of Geo-Information, vol. 7, no. 1, 2018.

[4] R. S. de Moraes and E. P. de Freitas, "Distributed Control for Groups of Unmanned Aerial Vehicles Performing Surveillance Missions and Providing Relay Communication Network Services," Journal of Intelligent and Robotic Systems: Theory and Applications, vol. 92, no. 3-4, pp. 645–656, 2018.

[5] S. Bae, I. Jang, S. Gros, B. Kulcsar, and J. Hellgren, "A Game Approach for Charging Station Placement Based on User Preferences and Crowdedness," IEEE Transactions on Intelligent Transportation Systems, pp. 3654–3669, 2022.

[6] X.-W. Liu, Q. Zhang, Y. Luo, Y.-J. Chen, and X. Lu, "ISAR Imaging Task Allocation for Multi-Target in Radar Network Based on Potential Game," IEEE Sensors Journal, vol. 19, no. 23, pp. 11192–11204, 2019.

[7] M. Bernardine Dias, R. Zlot, N. Kalra, and A. Stentz, "Market-based multirobot coordination: A survey and analysis," Proceedings of the IEEE, vol. 94, no. 7, pp. 1257–1270, 2006.

[8] S. Yoon and J. Kim, "Efficient multi-agent task allocation for collaborative route planning with multiple unmanned vehicles," IFAC-PapersOnLine, vol. 50, no. 1, pp. 3580–3585, 2017.

[9] J. Parker, A. Farinelli, and M. Gini, "Lazy max-sum for allocation of tasks with growing costs," Robotics and Autonomous Systems, vol. 110, pp. 44–56, 2018.

[10] M. Khani, A. Ahmadi, and H. Hajary, "Distributed task allocation in multi-agent environments using cellular learning automata," Soft Computing, vol. 23, no. 4, pp. 1199–1218, 2019.

[11] K. Jiang et al., "A Reinforcement Learning-Based Incentive Mechanism for Task Allocation Under Spatiotemporal Crowdsensing". IEEE Transactions on Computational Social Systems, 2023.

[12] J. H. Dr'eze and J. Greenberg, "Hedonic coalitions: Optimality and stabil- ity," Econometrica, vol. 48, no. 4, pp. 987–1003, 1980.

[13] H. Wu and H. Shang, "Potential game for dynamic task allocation in multi-agent system.," ISA Transactions, vol. 102, pp. 208–220, 2020.

[14] J. G. Martin, F. J. Muros, J. M. Maestre, and E. F. Camacho, "Multi-robot task allocation clustering based on game theory," Robotics and Autonomous Systems, vol. 161, p. 104314, 2023.

[15] I. Jang, H. S. Shin, and A. Tsourdos, "Anonymous Hedonic Game for Task Allocation in a Large-Scale Multiple Agent System," IEEE Transactions on Robotics, vol. 34, no. 6, pp. 1534–1548, 2018.

[16] A. Dutta, V. Ufimtsev, T. Said, I. Jang, and R. Eggen, "Distributed Hedonic Coalition Formation for Multi-Robot Task Allocation," in 2021 IEEE 17th International Conference on Automation Science and Engineering (CASE), 2021.

[17] B. P. Gerkey and M. J. Matarić, "A Formal Analysis and Taxonomy of Task Allocation in Multi-Robot Systems," The International Journal of Robotics Research, pp. 939–954, 2004.

[18] Q. La, B.-H. Soong, and Y. Chew, "Potential Game Theory: Applications in Radio Resource Allocation," Springer, 2016.