Computers and Electrical Engineering 118 (2024) 109300

Contents lists available at ScienceDirect

Computers and Electrical Engineering

journal homepage: www.elsevier.com/locate/compeleceng

A cooperation and decision-making framework in dynamic confrontation for multi-agent systems

Lexing Wang [a,b], Tenghai Qiu [a,*], Zhiqiang Pu [a,b], Jianqiang Yi [a,b]

[a] Institute of Automation, Chinese Academy of Sciences, Beijing, 100190, China
[b] University of Chinese Academy of Sciences, Beijing, 100490, China

ARTICLE INFO

Keywords:
Multi-agent system
Target allocation
Decision making
Swarm motion control

ABSTRACT

This paper investigates a dynamic confrontation problem where a swarm of agents with weak capabilities confronts multiple targets with strong capabilities. In such hostile situations, the agents typically strive to survive by cooperating to offend the targets and defend against potential attacks from the targets. To enhance the capability of such multi-agent systems, it is necessary to develop an efficient mechanism in which each agent autonomously makes decisions and cooperatively confronts targets. In this paper, a cooperation and decision-making framework is proposed for a multi-agent system confrontation, which is comprised of target allocation, tactical decision-making, and swarm motion control algorithms. First, to address the exponential growth problem of possible behavioral interactions with increasing numbers of agents, a hedonic coalition formation game algorithm is designed for the agents forming disjoint coalitions corresponding to different targets, i.e., the agents cooperating to confront the targets in the form of coalitions. Then, during the stage of attacking or defending the targets, each agent interacts with coalition members through information exchange. To implement effective cooperative behavior and explainable autonomous decision-making, a fuzzy cognitive map is designed to fuse situational information and obtain decision reference information for each agent's tactical decision-making. Moreover, to design effective attack/defense tactics and strategies for each agent, the tactical pursuit point method is utilized to develop a tactical pursuit point for each agent based on the decision reference information. Finally, a swarm motion control algorithm, including decision-oriented and swarm behavior rules, is designed to drive each agent towards the assigned target. Simulation results show the effectiveness of the designed framework and algorithms. In the confrontation, each agent adjusts its strategies according to different situations, occupies advantageous positions, and accomplishes cooperative attack/defense strategies to reduce casualties.

1. Introduction

The multi-agent system (MAS) is a swarm of agents to solve collaborative actions and behaviors between agents to achieve specific objectives. Compared to single-agent system, MAS has stronger distribution, collaboration, and robustness [1]. Through cooperation, the agents in MAS can overcome the limitations of single agent in perception and execution, complete complex tasks such as dynamic task allocation, collaborative reconnaissance, and confrontation [2–4]. Among these tasks, the confrontation problem, which simultaneously involves cooperation and competition tasks (where agents cooperate with teammates and compete with the

* Corresponding author.
E-mail addresses: wanglele2017@ia.ac.cn (L. Wang), tenghai.qiu@ia.ac.cn (T. Qiu), zhiqiang.pu@ia.ac.cn (Z. Pu), jianqiang.yi@ia.ac.cn (J. Yi).

https://doi.org/10.1016/j.compeleceng.2024.109300
Received 10 January 2024; Received in revised form 28 April 2024; Accepted 10 May 2024
Available online 23 May 2024

opponents), has received considerable attention in recent years. In this problem, MAS is required to expand the task capabilities and the overall performance through cooperation, and each agent should be able to make appropriate decisions autonomously based on current situation.

In nature, large numbers of individuals, such as birds, fishes, or animals, may work together to accomplish tasks that cannot be completed by an individual or any group of non-cooperative individuals [5]. For example, wolves hunting elk [6]. Inspired by these phenomenon, this study investigates an asymmetric confrontation problem where a swarm of agents with weak capability confronts multiple targets with strong capability. This problem can originate from assigning low-cost agents to confront high-value targets [7]. In such a hostile environment, environment faced by the agents is significant uncertainty and dynamic variability. Each agent typically strives to survive by cooperating to offend or destroy the targets while defending itself against potential attacks. Therefore, the autonomous decision-making of each agent and cooperation between teammates in attacking or defending against targets is crucial.

Different from one-to-one confrontation, various issues arise in MAS confrontation, such as resource allocation, task management, and action coordination. Moreover, the behavioral interactions of agents grow exponentially with the number of agents. How to address the curse of dimensionality caused by behavioral interactions is a challenge in current research on MAS confrontation.

Generally, decisions of agents in confrontation mainly include tactical decisions [8] and maneuver decisions [9]. Currently, research on agents' decision-making primarily focuses on maneuvering actions during a confrontation or on decision-making in specific mission scenarios, such as swarm strike missions [10]. However, there is a lack of research on autonomous decision-making from the perspective of tactical and strategic aspects, as well as the diversity and complexity of tasks. For multi-agent systems operating in complex and dynamic environment, it is challenging to determine effective tactics and strategies based on situational information and implement efficient cooperative behaviors.

In confrontation scenarios, the pursuer-evader game [11] represents a typical situation that can present some features of asymmetric confrontation. Modeling this problem requires identifying the pursuers and the evaders. However, this ignores the frequent role reversals that occur in actual confrontations. Therefore, the methods involved in the pursuer-evader game cannot be directly used to solve the asymmetric confrontation problem.

The asymmetric confrontation of agents bears a resemblance to some biological predation behaviors, such as the cooperative hunting of prey by wolf packs. In nature, many predators do not gather together in a disorganized manner; instead, they form small-scale groups and autonomously make decisions through various interactions. Compared to larger groups, smaller groups are more prone to cooperation, and their abilities are fully showcased. In larger groups, individuals interfere with each other, and some individuals cannot contribute to the hunt [12]. In dynamic confrontations, drawing inspiration from the wolf packs grouping to pursue and attack large prey, the multi-agent system can engage in confrontations in the form of coalitions. This grouping attack strategy has been proven effective in some literature [12,13].

Moreover, distributed strategies [14] have been proven robust in dynamic scenarios without needing assistance from centralized nodes. In a distributed MAS, each agent operates autonomously and cooperates with others to achieve common objectives. Of course, this requires complex communication and coordination strategies to ensure that each agent is aware of the surrounding environment and the actions of other agents.

Inspired by the above analysis, this paper studies a confrontation problem for MAS, and an asymmetric capability case between agents and targets is considered. It is obvious that when faced a target with stronger capability, it is difficult for a single agent or any group of non-cooperative individuals to accomplish the confrontation task. Although a multi-agent system has an advantage in numbers, some individuals will be unable to participate in confrontation, or the casualties of agents will increase, due to the interference between individuals. Therefore, the main objectives of this article are: (1) developing an effective way to decrease behavior interactions by dividing a swarm of agents into small groups, and these groups make decisions through interaction, (2) designing an effective tactical decision-making mechanism to enable agents to cooperate with each other and to confront the targets while reducing casualties. It can be seen that target allocation, autonomous decision-making, and swarm motion control are the three critical issues in MAS confrontation. Therefore, considering the confrontation task in this paper, MAS autonomously forms different coalitions for target allocation to confront the targets. Then, each agent generates a tactical pursuit point through tactical decision-making. Finally, under the behavioral rule-oriented and decision-oriented stimulus, each agent is steered to perform an attack or defense purpose under a control protocol.

The main contributions of this article are listed as follows.

1. A cooperation framework is proposed for multi-agent system in the confrontation scenarios, which is comprised of target allocation, tactical decision-making, and swarm motion control algorithms.
2. Inspired by biological behaviors, a multi-agent coalition formation algorithm is designed for grouping agents against multiple targets. Unlike the cumulative allocation algorithm in our previous work [15], this paper only considers the targets within the detection range rather than all targets, which has high efficiency. This algorithm is more suitable for confrontations in practical, as enemy targets are unknown.
3. A decision fusion mechanism for autonomous decision-making of the agents is developed for implementing effective cooperative behavior and interpretable autonomous decision-making reasoning. A fuzzy cognitive map (FCM) is constructed for each agent, with inputs including agent, coalition, and nearest neighbor situation information, while the output value serves as a reference for tactical decisions.
4. A tactical pursuit point (TPP) mechanism based on situational information is established to develop effective tactics and strategies. Each agent makes autonomous decisions and adjusts maneuver decisions based on current situation, attempting to obtain a favorable position.

The rest of this article is organized as follows. In Section 2, we review the related works. In Section 3, a confrontation problem investigated in this article is defined, and the agents' motion model is formulated. Besides, we introduce the concept of fuzzy cognitive maps. In Section 4, we propose a framework where a hedonic game algorithm, a TPP-based decision-making algorithm, and a swarm motion control algorithm are designed. Illustrative simulation results are presented in Section 5. Finally, Section 6 concludes this article.

2. Related works

As mentioned earlier, the main tasks of MAS confrontation correspond to three loosely coupled subtasks, namely, target allocation, decision-making, and swarm motion control. Now, we will review the latest relevant studies on these subtasks.

2.1. Target allocation

In general, the target allocation methods of coalition formation can be divided into centralized and distributed methods. Centralized methods have been utilized to solve the coalition formation for target allocation, for example, bipartite matching [16], swarm methods [17], and genetic algorithms [18]. They have achieved good performance in solving the problem. However, these methods rely on global information, which are suitable for a small number of agents. More importantly, centralized methods are susceptible to the single point of failure [19]. For large-scale multi-agent systems, it is more likely for each agent to acquire local information rather than global information to make decisions.

In contrast, the distributed methods provide insight to solve the above problem, where each agent can only communicate with the agents within a certain communication range. Lin et al. [20] proposed a distributed double-clock consensus-based k-means algorithm (DCKA) for the agents to self-organize to choose the most 'self-interested' group, and the DCKA presented a better performance than the centralized k-means algorithm. However, it is hard to design social utility functions. In addition, market-based approaches have also been utilized to solve coalition formation problems [21]. However, such methods rely on bidding mechanisms that result in extensive communication burdens on agents.

Among the different methods to form coalitions, game-theoretical methods present some superior performances in efficiency, scalability, computational cost, etc. [22]. Especially, the cooperative game associated with coalition formation has attracted significant attention from researchers. In cooperative game theory, the hedonic game [23] provides an analytical and theoretical enlightenment for researchers to solve the coalition formation problem. A hedonic game (also known as a hedonic coalition formation game) refers to a problem where each agent holds a preference order over the coalitions. As a result, several disjoint coalitions will finally be formed. In [24], a hedonic game was utilized for self-organizing agents suited to wireless networks, i.e., modeling the data collection by agents from several arbitrarily located tasks. Jang et al. [25] proposed a hedonic game autonomous decision-making framework to divide the agents into several disjoint coalitions towards the tasks. This algorithm achieved a convergence to Nash stability and presented a good performance in asynchronous and dynamical environments. Though game-theoretic approaches provide superior performance in MAS, the above methods ignore the overabundance of agents or the requirement of targets in the target allocation. In [26], the authors developed a game-theoretical decision-making framework to solve the target allocation problem with minimum requirement constraints. However, this method will lead to an increase in the total cost of MAS.

2.2. Decision-making method in confrontation

The existing decision-making methods for MAS confrontation include the differential game-based method [27], the expert system-based method [28], and the data-driven method [29].

The differential game method can learn how to act without prior knowledge of agents, but this method has many problems such as too many state variables, complex differential equations, and challenging to solve the analytical equations, so it is difficult to be applied to complex multi-agent environments. Expert system method is based on the prior knowledge of human experts in related fields to establish its system model. However, it depends on the rules made by a large number of human experts. In the complex environment of MAS confrontation, it is difficult and time-consuming to get a complete rule base that can cover the diverse confrontation situations.

In recent years, decision-making methods based on multi-agent reinforcement learning (MARL) have drawn a lot of attention. The agents receive rewards and learn the strategy through their interactions with environment. Compared with other methods like differential game methods and expert system methods, MARL method cares less about system model and is easier to design.

In [29], a hierarchical MARL framework for UAV swarm confrontation is proposed. A simulation environment for UAV swarm confrontation is constructed in [30]. In this environment, the performance of the multi-agent soft actor critic method and the multi-agent deep deterministic policy gradient method are compared. The weighted mean effect of interactions between agents is considered, and a weighted mean field reinforcement learning method for MAS confrontation is proposed [31]. However, these methods mainly focus on increasing the success rate under the condition that the swarm size is fixed and small. For traditional MARL methods, the strategy trained for a certain number of agents does not have a good performance to be effective for an MAS of a different size. Thus, the strategy has to be retrained as the swarm size changes. Due to the increase in swarm size, the dimensions of the state space and action space increase, and the solution space becomes larger. As a result, the training time increases exponentially as the swarm size increases.

As an alternative to the above approaches, a novel virtual tactical pursuit point (TPP) concept was investigated in [32,33]. The tactical pursuit method refers to generating a decision space for pursuit points around the target and then searching for a suitable pursuit point in this space. These pursuit points can guide agents to gain an advantage by performing corresponding maneuvers, i.e., the tactical objectives of agents are controlled by the generated pursuit points. The TPP method is similar to the maneuver-library method. The difference is that the decision result of the maneuver-library-based method can be served as control stick commands of an agent, while the output of the TPP method corresponds to the command of the guidance system of an agent. In [8], Xu et al. proposed a novel tactical pursuit point (TPP)-based autonomous decision-making framework to improve agent decision-making abilities in confrontation. However, they only considered a scenario of one-to-one confrontation.

2.3. Swarm motion control

Self-organization observed in biological systems including fish joining together in schools, ants forming trails in foraging, and birds flying in formations provides an excellent model for autonomous agents to imitate. Inspired by Reynold's three rules [34] (i.e., collision avoidance, velocity matching, and flock centering), Olfati-Sabe [35] proposed a distributed flocking algorithm, which investigated swarm behaviors by presenting a sound mathematical model and theoretical framework.

The swarm motion control has attracted remarkable interests in recent years. Beaver [36] proposed an optimal control protocol to induce flocking for a group of agents while considering energy minimization and safety. The literature [37] focused on the convergence speed of the flocking algorithm. A distributed fast synchronization (DFS) algorithm combined with switching communication topologies in flocking control was utilized to effectively improve the convergence rate. Sakai et al. [38] proposed a novel flocking algorithm that did not make a distinction between obstacles and neighbor agents. The algorithm regarded all detected objects as obstacles in a unified way, which maintained the basic properties of the existing flocking method. Jing and Wang [39] proposed a distributed angle-based control strategy for a group of double-integrator agents to achieve flocking behavior while maintaining a desired triangulated formation shape. However, the expected distance between the connections in the above work is constant, which is not conducive to the flexibility of configuration and tasks. Therefore, it is not conducive to intelligent agent systems combating multiple dispersed targets.

3. Preliminaries

3.1. Problem formulation

Let $A = \{a_1, a_2, \dots, a_n\}$ denote a swarm of n agents. The set of targets is denoted as $T = \{t_1, t_2, \dots, t_m\} \cup \{t_\phi\}$. t_ϕ represents the virtual or void target. Namely, when t_ϕ is allocated to some agents, the agents keep current actions.

Problem 1. This paper investigates multi-agent system cooperation and decision-making in a confrontation problem. Consider a system consisting of n agents and m targets; individuals on both sides have the ability to attack and defend. Each agent in MAS typically strives to survive in this dynamic conflict by cooperating to offend or attack the targets while defending itself against potential attacks. Therefore, environment faced by the agents is significant uncertainty and dynamic variability.

Furthermore, an asymmetric confrontation case between the agents and the targets is considered. The term 'asymmetric' refers to the difference in perception ability and kill capability between the agents and the targets. The targets can perceive all agents in environment, while the agents have a limited perception radius. The targets' kill capability (p^j_{attack}) is larger than that of the agents'(p^a_{attack}), i.e., $p^t_{attack} > p^a_{attack}$. A single agent cannot complete the task of attacking a target alone. Therefore, the agents are supposed to cooperate to attack by increasing the number of the agents and make up for the shortcomings in their capabilities. The number of agents is greater than the number of targets, i.e., $n > m$.

To further model the MAS confrontation, some assumptions are made as following:

1. Each agent can only attack one target at a time, and each target can only attack one agent at a time.
2. The types of the agents are the same, that is, homogeneous agents. However, their attack strategies are different in confrontation.
3. The agents have limited perception and communication range. Each agent can only detect targets within perception range and communicate with neighbor agents.

Assumption 1 is an explanation of the ability of the agents and the targets to perform a single task. For Assumption 2, as this paper mainly focuses on studying the effectiveness of agent cooperation and decision-making strategies, it is necessary to ensure the principles of univariate analysis. Assumption 3 illustrates the capability of the agents and the targets, which is the premise of the condition for confrontation.

The attack area of the agents is denoted by a cone area, as shown in Fig. 1. The attack area consists of attackable distance R^a_a and attackable angle θ in front of each agent. When a target is in the attack zone of an agent, the target will be hit by the agent.

In confrontation, the number of possible behavioral interactions increases exponentially with the number of agents in MAS. Moreover, the situation is complex and variable, and the attack/defense strategies are diverse, resulting in a sharp increase in the difficulty of solving. Therefore, there is a need to develop an efficient approach to address the limitations of existing methods.

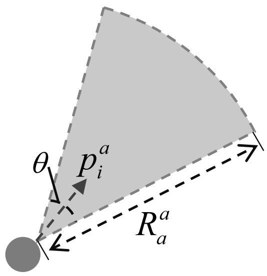


Fig. 1. Attack area.

3.2. Agent motion model

The motion model of agents is the foundation of confrontation. The control commands for each agent are executed through motion models to change the position and velocity, changing the individual's situation. In confrontation, the decision-making mainly considers the positional relationship and velocity vector between the agents and the targets in space, while body posture has little impact. Therefore, the agents are regarded as a particle, the dynamics of each agent are expressed by double integrators:

$$\begin{cases} \dot{q}_i^a = p_i^a \\ \dot{p}_i^a = u_i^a \end{cases}, \tag{1}$$

where $q_i^a, p_i^a, u_i^a \in \mathbb{R}^2$ are the position, velocity, and control input of agent a_i. The dynamics of the targets is consistent with the form of (1).

In this paper, it is assumed that each agent has a survival probability, and the initial value is 1. The survival probability of a given agent decreases if the agent is attacked by a target. In case the survival probability decreases below a particular threshold P_{th}, the agent is destroyed.

3.3. Fuzzy cognitive maps

Fuzzy Cognitive Map (FCM) [40] is an effective tool for modeling and simulating dynamic system knowledge representation and causal reasoning, which has a strong explanatory ability for the results. Therefore, it is suitable for analyzing the confrontation situation and intuitively describing the changes in the situation. Moreover, the result of FCM provides decision support for decision-making in confrontation.

The classic FCM model consists of concept nodes, directed edges, and their associated weight matrices. The nodes represent the attributes, characteristics, performance, etc., of the system. The directed edges represent the influence relationships between the nodes, and the degree of influence is described by the weight matrix. The FCM simulates complex system behavior through causal relationships and interactions between concepts. Inference model for FCM is formulated as:

$$C_j(t+1) = f\left(\sum_{i=1, i\neq j}^{n_c} w_{ij} C_i(t) + C_j(t)\right), \tag{2}$$

where C_j represents the state vector of j-node, and w_{ij} represents the weight value. $f(\cdot)$ represents the activation function of j-node, which has multiple forms of expression, such as sigmoid function. n_c is the number of nodes in FCM.

As an extended model, ABFCM (agent-based fuzzy cognitive map) [41] maps the nodes of FCM to c-agents (explanation of 'c-agent' can be found in Remark 1). In ABFCM, each node has different inference algorithms, and they can interact with other nodes through message-passing mechanisms to simulate the dynamic behavior of complex systems. Inference model for ABFCM is formulated as:

$$A_i^t = F_i(A_1^{t-1}, \dots, A_j^{t-1}, \dots, A_{n_c}^{t-1}; w_{11}, \dots w_{ji}, \dots w_{n_c n_c}), \tag{3}$$

Here, A_i^t is the state of i-node at time t. $F_i(\cdot)$ is the inference process of A_i^t under given information, rather than the activation function in FCM. A mapping between classical FCM and ABFCM is shown as Fig. 2.

Remark 1. Unlike an agent representing an entity in MAS (such as an unmanned vehicle), in ABFCM, an 'agent' represents a virtual concept. Thus, a 'c-agent' is used to distinguish it from the agents in MAS.

In a traditional FCM, a sigmoid function is utilized as the state transition function. However, generally, the parameters of the sigmoid are uniform, which dramatically limits the expression and simulation ability of FCM. Instead, an inspiration that ABFCM brings us is the design of updating functions of the node state. A personalized state update function can be designed for each node based on the concept type expressed by the node, according to the studied problem. Therefore, in Section 4.3, based on Problem 1, we design an ABFCM to handle the comprehensive impact of other information to an agent.

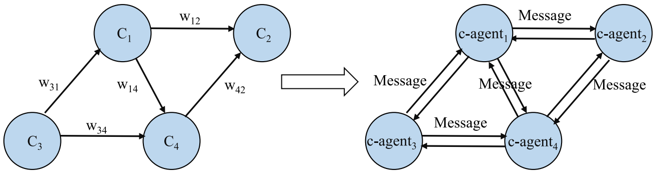


Fig. 2. Mapping between classical FCM and ABFCM.

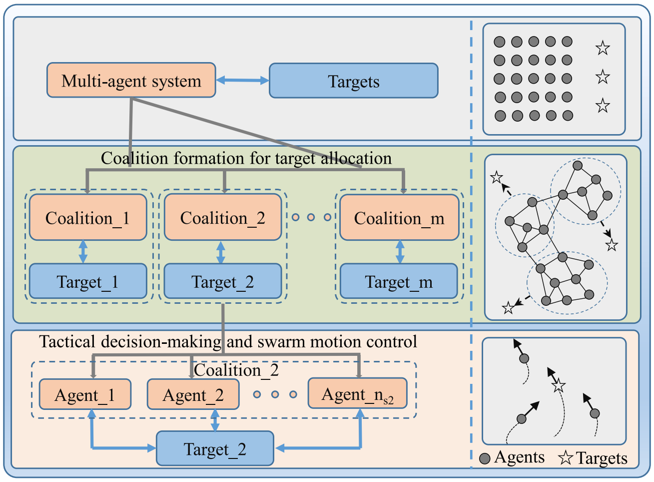


Fig. 3. The framework for MAS confrontation.

4. A framework for MAS confrontation

4.1. Framework

In this paper, the confrontation task is decoupled into several subtasks, including target allocation, tactical decision-making, and swarm motion control. To address these subtasks and achieve effective, intelligent, and autonomous task execution, this paper proposes a framework that is comprised of a target allocation algorithm, a tactical decision-making algorithm, and a swarm motion control algorithm. The framework is showed in Fig. 3.

Specifically, through target allocation, a swarm of agents is divided into several disjoint coalitions to confront multiple targets. Then, the agents confront the targets in the form of coalitions, and each coalition specifically confronts an assigned target. During the confrontation, each agent makes autonomous tactical decisions. Under the guidance of swarm behavior rules and tactical stimulus, each agent moves to attack or defend the target. It is worth noting that although the decomposition of the confrontation task is top-down, the behavior of agents in each algorithm is distributed.

The proposed framework and underlying algorithms appear as a feasible solution to the confrontation. The reasons are listed below:

1. System hierarchy. The framework is based on a hierarchical decomposition of the confrontation problem. This framework makes sense: (1) It follows logical steps for mission execution. (2) Effectively alleviate the exponential explosion problem caused by the increase in the number of agents.
2. Distributed decision-making. The uncertain information and dynamic environment require each agent to continuously adjust its intentions based on its current situation. Autonomous distributed decision-making enables each agent to quickly make orientation based on current situation and determine the tactical strategy for the next moment.
3. Emergent swarm behaviors. Based on some simple swarm behaviors, such as collision avoidance and attack/defense stimulus, many task-related functions can be completed, such as attack with tactical strategies. Due to the distributed allocation, decision-making, and swarm motion control of the system, it has good robustness against single-point faults. It can be extended to a large number of agents.

Therefore, the proposed framework has a potential to effectively address the problem of employing a swarm of agents in confrontation. Next, we will design specific algorithms in target allocation, tactical decision-making, and swarm motion control in confrontation.

4.2. Hedonic coalition formation game for target allocation

In a MAS confrontation, the number of possible behavioral interactions increases exponentially with the number of agents. Therefore, inspired by animal predation behaviors, the agents grouping to attack each target separately in the form of coalitions is considered as an effective scheme to solve this problem. Compared with a large group, it is easier for small groups to cooperate. However, it is crucial to decide the number of teammates in each coalition. Most existing methods adopt assigning all agents in MAS to targets, but they ignore the required number of agents to attack targets. A large number of members in the coalition not only interfere with each other, but some selfish agents also experience 'free-riding' and cannot contribute to the attack.

Remark 2. To determine coalition members, two concepts, reward V and demand F, are introduced. The 'reward' refers to a value that attracts agents to form a coalition with respect to confront a target. The 'demand' refers to the number of agents required to defeat a target, and it is evident that the stronger the target's ability is, the greater the demand is. These two concepts are positively correlated, which means the greater the demand to confront a target is, the greater the reward the corresponding coalition will acquire. This paper assumes that an agent can acquire the demand of a target when the agent perceives the target. The demand can be identified by the model or size of the target.

Here, a hedonic game algorithm is designed to form coalitions for target allocation. A marginal preference mechanism is used to determine the number of coalition members, according to the demand of targets.

Hedonic games are a subclass of cooperative games which concentrate on the competition between a swarm of agents. In a hedonic game, the agents form coalitions in accordance with an enforceable agreement and then choose their interested coalition to join in. The term 'hedonic' pertains that the utility of an agent only dependent on the members of the coalition to which the agent belongs. Therefore, the hedonic game provides a solution to form coalitions for target allocation.

A coalition $S_j \in A$ is a target-specific subset of agents executing to attack/defend target t_j. S_ϕ is an empty coalition, which is the set of agents who choose target t_ϕ. Hereinafter, this paper interchangeably uses S_0 to indicate S_ϕ.

Let $CS = \{S_0, S_1, S_2, \dots, S_m\}$ denote a set of disjoint coalitions and each coalition is matched with a target. For $\forall S_u, S_v \in CS$, $S_u \cap S_v = \phi$ and $\cup_{u=0}^{m} S_u = A$. As CS partitions the set A, CS is called coalition structure or coalition partition.

A target-coalition pair (t_j, S_j) is defined as "to attack/defend target t_j with coalition S_j", i.e., each coalition is assigned a particular target and is responsible for its execution.

A preference relation of agent a_i is represented by the symbol $\prec_i$ or $\sim_i$. For example, for two coalitions S_1 and S_2, $S_2 \prec_i S_1$ denotes that a_i has a strong preference on S_1 than S_2 while $S_1 \sim_i S_2$ denotes a_i has the same preference on S_1 and S_2. Likewise, $\preceq_i$ indicates the weak preference of agent a_i. The utility u_i is a quantitative description of preference relationships (as (4)), which indicate overall how well an agent fits with a coalition in terms of reward and cost.

$$u_i(S_1) \leqslant u_i(S_2) \Leftrightarrow S_1 \preceq_i S_2. \tag{4}$$

It is worth noting that preference relation is a core concept that affects the final formation process of the coalition.

The objective of coalition formation for target allocation is to determine a stable partition that all the agents agree with. In this paper, we seek for a Nash stable partition, which is defined as follows.

Definition 1. A coalition structure is **Nash stable** if no agent can benefit from moving from its coalition S_u to another (possibly empty) coalition S_v. It can be formulated as:

$$\forall a_i \in A, \forall S_v \in \{CS \backslash S_u\}, \{S_v \cup \{a_i\}\} \preceq_i S_u, \tag{5}$$

where ' \' represents the difference operation between two sets. Nash stable is beneficial to reduce communication burden among agents while yielding a social agreement among these agents even without having any negotiation [42]. Therefore, it is important to find a coalition partition with Nash stability.

In [6], MacDulty proposed that the relationship between hunting success and group size is non-linear when wolf packs hunt elk. When the group size is small, the hunting success increases with the increase of group size. However, hunting success tends to stabilize when the group reaches a specific size.

Inspired by the above phenomenon, we propose a mechanism of marginal preference to describe the reward of an agent joining in a coalition. For agents forming a coalition, the global reward of the coalition increases with the number of agents. When the coalition size peaks at a threshold, the global reward levels off. Once the size exceeds threshold, the global utility will be decreased.

The reward of a coalition is related to its number of participants in coalition. Suppose destroying a target t_j, the maximal reward is $V_j^{\max}$. Hence, while the agents form coalition for t_j, the reward function of S_j is designed as:

$$V_{S_j}\left(t_j, |S_j|\right) = \frac{V_j^{\max} \cdot |S_j|}{F_j} \cdot e^{-|S_k|/F_j+1}, |S_j| \in N_+, \tag{6}$$

where $|S_j|$ is the number of participants in S_j. $|\cdot|$ represents the cardinality of a set throughout this paper, i.e., the number of elements in the set. F_j represents the demand to defeat target t_j and $F_j > 0$.

The marginal reward is defined to describe the additional reward generated by agent a_i joining a coalition, which formulated as:

$$V_i(S_j) = V_S(S_j) - V_S(S_j \backslash \{a_i\}). \tag{7}$$

From (6) and (7), it can be seen that if the number of coalition members exceeds the demand of t_j, the reward for a_i will be non-positive.

Therefore, the utility of a_i is defined as the marginal reward minus the individual's cost required to the given target t_j. Thus, the utility function is formulated as:

$$u_i(t_j, |S_j|) = \begin{cases} V_S\left(t_j, |S_j|\right) - V_S\left(t_j, |S_j'|\right) - k_c \cdot\ cost_i(t_j), & if\ u_i > 0 \\ 0, & else \end{cases}, \tag{8}$$

where $S_j' = S_j \backslash \{a_i\}$, k_c is a constant. $cost_i(t_j)$ is the cost that agent a_i needs to pay for confronting t_j. We simply set the cost as a function of the distance from agent a_i to target t_j.

Notably, it is crucial for the utility to be non-negative in order to achieve a Nash-stable outcome [43]. Thus, $u_i(t_j, |S_j|) = 0$, $if\ u_i < 0$. In this paper, $u_{a_i}(t_\phi) = \alpha$ (α is a small positive number) is defined, which is beneficial to the convergence of the proposed algorithm. The reason is that if an agent is assigned to target t_j whose demand is already satisfied, the agent's utility will be zero. Because the utility that the agent chooses t_ϕ is bigger than t_j, thus, the agents will choose t_ϕ autonomously.

As mentioned above, the preference relation of agents affects the final formation of coalitions. Therefore, according to the utility function designed in (8), a preference relation is designed as follows:

Definition 2 (*Preference relation*)**.** Agents always prefer to join a coalition that makes their utility greater, i.e.,

$$S_1 \preceq_i S_2 \Leftrightarrow u(t_1, |S_1|) \leqslant u(t_2, |S_2|). \tag{9}$$

Theorem 1. *For a system consisting of the agents A and targets T, if the agents' utility is designed as (6)–(8) and each agent holds a preference relation defined by Definition 2, there always exists a Nash stable for the system.*

Proof. In this paper, the utility function is designed such that it tracks changes in its marginal improvement when one or more agents change their coalitions, which is a form of a potential function. Therefore, the *Theorem* 1 also can be corroborated by the conclusion proposed by [44], i.e., any such finite game would converge to a Nash-stable solution. ■

Through the proposed hedonic game coalition formation method above, forming coalitions for target allocation can be achieved, and the number of members in the coalitions can be determined based on the demands of the targets.

After designing the preference rule for agents and proving that a Nash stable coalition structure exists, a distributed algorithm is utilized for agents autonomously forming coalitions, where each agent makes decisions on its local information and shares information with its neighbor agents. The pseudocode of the algorithm can be traced back to Ref. [15].

4.3. Generating tactic pursuit points for tactical decision-making

Once multiple agents form coalitions corresponding to the targets, the next goal is to achieve an effective attack or defense against the targets for each agent. It is crucial to develop effective tactics and strategies. Essentially, confrontation for agents is to find the optimal position and obtain a favorable situation. Therefore, it is crucial to find effective strategic pursuit points based on situational awareness and evaluation.

In [8], a tactical pursuit point (TPP)-based autonomous decision-making framework is proposed for an agent to confront a target. The method enriches the tactical pursuit space of the agent, allowing for the emergence of tactical pursuit such as lead and lag, inside and outside pursuit. This paper builds on the work investigated in [8] and expands it to a cooperation strategy of multiple agents. We solve the problem of cooperation among multiple agents as a coalition to attack a common target. In addition to the agent's own information, the situation information of the coalition and nearest neighbors also be considered. An ABFCM is designed to integrate these information in order to generate an effective tactical pursuit point. Next, we will introduce how to generate a tactical pursuit point for each agent.

The tactical pursuit point method refers to generating a decision space for pursuit points around the target and then searching for a pursuit point in this space. This pursuit point can guide agents to gain an advantage by performing corresponding maneuvers, i.e., the tactical objectives of the agent are controlled by the generated pursuit points.

The diagram of generating a tactical pursuit point is shown in Fig. 4. A tactical pursuit point is linearly combined with the tactical pursuit basis (TPB) and the decision variables. The TPB is calculated by the state information of the target and agent. The decision variables are determined by an integrated situation information of the agent.

In order to attack/defense against a target, an agent should autonomously make decisions based on its situation awareness and cooperate with others. Therefore, situation assessment is the premise and basis for the agent to make autonomous decision-making. Unlike only calculating a comprehensive situational assessment value [45], the current situational type of the agent is also considered in this paper.

An agent's situation is divided into four types, namely, $\mathrm{Type} = 1, 2, 3, 4$, which are represented as: advantage, disadvantage, mutual safe, and mutual disadvantage, respectively [46]. An example of situation types is shown in Fig. 5(a).

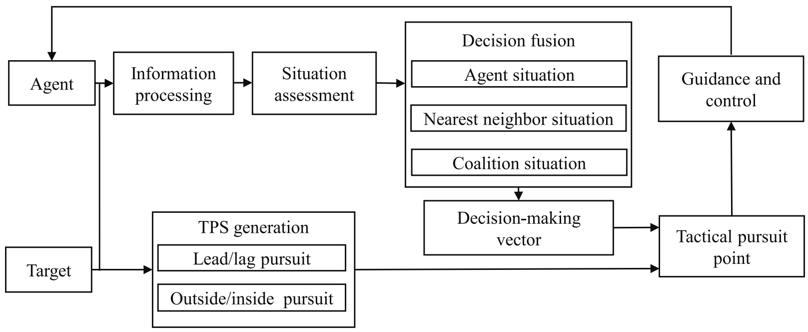
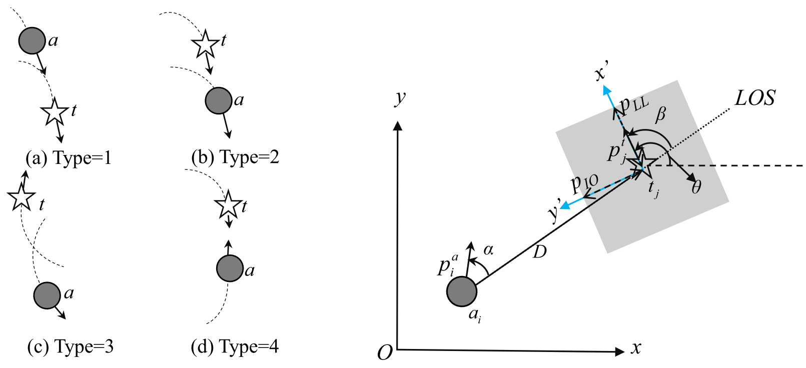


Fig. 4. Diagram of generating a tactical pursuit point.



(a) Situation types of an agent. (b) The geometric situation, TPS, and TPBs.

Fig. 5. Situation types and geometric relationship.

The confrontation situation mainly depends on the angle and distance between the agent and the target. The type of confrontation situation is mainly determined by the state variable $State = [\alpha, \beta, D]$, where α, β D are presented as Fig. 5(b). α represents the angle-off, which is the angle between p_i^a and line-of-sight (LOS). Besides, β represents the aspect angle, which is the angle between p_j^t and LOS. D represent the distance between agent a_i and target t_j.

The situation probability model of a_i at time k can be expressed as:

$$P_i\left(\text{Type}_k = j \,\middle|\, State = State_k\right), j \in \Omega = \{1, 2, 3, 4\}, \tag{10}$$

where it is obvious that $\sum_j^{\Omega} P_i\left(\text{Type}_k = j \,\middle|\, State = State_k\right) = 1$.

According to Bayes' theorem, (10) can be denoted as:

$$P_i(\text{Type}_k = j \,|\, State = State_k) = \frac{P_i\left(\text{Type}_k = j\right) P_i\left(State = State_k \,\middle|\, \text{Type}_k = j\right)}{\sum_l^{\Omega} P_i\left(\text{Type}_k = l \,\middle|\, State = State_k\right) P_i\left(State = State_k \,\middle|\, \text{Type}_k = l\right)}. \tag{11}$$

Since confrontation is a dynamic process, the result of the situation type assessment is only related to the current state. Therefore, the prior probability is independent of each other, namely $P_i\left(\text{Type}_k = j\right) = 0.25$.

Thus, (11) can be simplified as:

$$P_i\left(\text{Type}_k = j \,\middle|\, State = State_k\right) = \frac{P_i\left(State = State_k \,\middle|\, \text{Type}_k = j\right)}{\sum_l^{\Omega} P_i\left(State = State_k \,\middle|\, \text{Type}_k = l\right)}. \tag{12}$$

Situation assessment elements α, β, and D are assumed to be independent with a particular situation outcome $\text{Type} = j$. Thus, the conditional joint probability density function can be calculated by:

$$P_i(State = State_k \,|\, \text{Type}_k = j) = P_i\left(\alpha_k \,\middle|\, \text{Type}_k = j\right) \cdot P_i\left(\beta_k \,\middle|\, \text{Type}_k = j\right) \cdot P_i\left(D_k \,\middle|\, \text{Type}_k = j\right). \tag{13}$$

Therefore, by designing the conditional likelihood function of each state, the situation value and type of a_i at the k-moment can be calculated or identified by (12).

As shown in the rectangular shadow in Fig. 5(b), the tactical pursuit space is formed by the combination of lead–lag tactical pursuit basis (q_{LL}^j), the inside-outside tactical pursuit basis (q_{IO}^j). Therefore, the tactical pursuit space can be denoted as follows:

$$S = \left\{(x, y) : q_j^t + \lambda_1 q_{LL}^j + \lambda_2 q_{IO}^j\right\}, \tag{14}$$

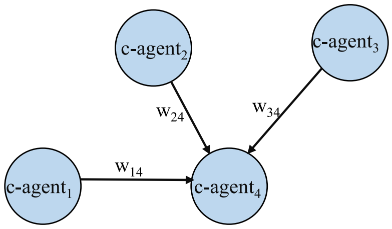


Fig. 6. An ABFCM for fusing information.

where q_j^t is the position of target t_j.

In addition, $\lambda_i \in [-1, 1]$ is the decision variable corresponding to the ith TPB in the maneuvering plane, where a positive λ_i represents lead or outside pursuit and a negative λ_i represents lag or inside pursuit.

Specifically, q_{LL}^j defines a pursuit point in the positive direction of the target's velocity vector, as follows:

$$q_{LL}^j = m_{LL} \cdot i_{x'}, \tag{15}$$

where $i_{x'}$ is a unit vector on the x-axis of body coordinate system of the target and is given by:

$$i_{x'} = \frac{p_j^t}{|p_j^t|}, \tag{16}$$

m_{LL} is the magnitude of q_{LL}^j and consists of two terms:

$$m_{LL} = k_{LL} \cdot R_a^t, \tag{17}$$

where k_{LL} is a constant, R_a^t is the attack radius of a target.

Similarly, q_{IO}^j defines a pursuit point on the y-axis in body coordinate system of the target. Its amplitude is determined by the attack radius of target too and can be derived as follows.

$$q_{IO}^j = m_{IO} \cdot i_{y'}, \tag{18}$$

where $i_{y'}$ is a unit vector on the y-axis of body coordinate system of the target and $m_{IO} = k_{IO} \cdot R_a^t$, where k_{IO} is a constant.

Finally, the tactical pursuit point q_i^* can be expressed as combinations of q_{LL}^j and q_{IO}^j in the form:

$$q_i^* = q_j^t + \lambda_1 q_{LL}^j + \lambda_2 q_{IO}^j. \tag{19}$$

Therefore, various types of tactical purposes can be obtained by adjusting λ_1 and λ_2. Meanwhile, a continuous decision-making space with multiple tactical objectives can be specified for autonomous decision-making. The decision variables are reduced to several normalized weight coefficients, greatly reducing autonomous decision-making's complexity.

Based on the above analysis, an agent's tactical intention is described through the linear combination of tactical pursuit basis. Thus, the autonomous decision-making problem becomes equivalent to finding the optimal decision-making vector λ. Next, we design a mechanism to calculate the decision-making vector based on the current situation of an agent.

The core issue of autonomous decision-making for a_i is to study how to use a fusion mechanism to integrate multiple decision information into its viewpoints. The research paradigm can be expressed as follows:

$$F(o^1, o^2, \ldots o^{m_o}) = o^i, \tag{20}$$

where o^b ($b \in \{1, 2, \ldots, m_o\}$) represents the input information of agent a_i. $F(\cdot)$ is the fusion function, which is the fusion decision-making mechanism of a_i.

As introduced in Section 3.3, we use an ABFCM to model the fusion process of (20). Fig. 6 shows the ABFCM for agent decision information fusion. The input nodes 1, 2, and 3 represent the coalition's situation information c-agent, the nearest neighbor's situation information c-agent, and the agent's own situation information c-agent, respectively. The output node is the fusion value c-agent. The weight is set by expert knowledge.

Remark 3. The situation information of a coalition is calculated by the situation between the geometric center of the coalition and the target. The method to calculate coalition situation information is the same as the previous situation assessment method.

Next, the value of output node serves as a decision reference information to calculate decision variable λ. λ is determined by two terms, i.e., its symbol and value. Specifically, its symbol influences whether the tactical pursuit point is on the left or right of the target, and in front or behind. And its value regulates the distance between the pursuit point and the target.

The value function is designed as:

$$\lambda_1^{value} = \begin{cases} e^{-k \cdot V_{on}}, \text{Type} = 1, 3 \\ 1 - e^{-k \cdot V_{on}}, \text{Type} = 2, 4 \end{cases}, \tag{21}$$

$$\lambda_2^{value} = \begin{cases} e^{-k\cdot V_{on}}, \text{Type} = 1,3 \\ 1 - e^{-k\cdot V_{on}}, \text{Type} = 2,4 \end{cases}, \tag{22}$$

where V_{on} is the value of output node in ABFCM. k is a constant.

The symbol function is designed as:

$$\lambda_1^{symbol} = \begin{cases} +1, \text{Type} = 1,3 \\ -1, \text{Type} = 2,4 \end{cases}, \tag{23}$$

$$\lambda_2^{symbol} = sign(y'_{a_i}), \tag{24}$$

where $q'_{a_i}(x'_{a_i}, y'_{a_i})$ is the coordinate of a_i in the body coordinate system of the target.

Finally, based on (21)–(24), the decision-making vector λ is determined by the two terms:

$$\begin{cases} \lambda_1 = \lambda_1^{symbol} \cdot \lambda_1^{value} \\ \lambda_2 = \lambda_2^{symbol} \cdot \lambda_2^{value} \end{cases}. \tag{25}$$

Therefore, based on (15), (18), and (25), the tactical pursuit point q_i^* can be obtained through (19).

4.4. Swarm motion control

Based on the target allocation and tactical decision-making results, each agent selects the corresponding behavioral rules and accordingly updates its state. The behavioral rules are underpinned by an efficient swarm motion control mechanism, which would steer each agent to attack or defend a target in a self-organized manner. Therefore, a behavioral rule-oriented and decision-oriented swarm motion control protocol is designed, leading to emergent tactical behaviors relevant to the confrontation.

The behavioral rules for generating swarming behaviors are cohesion, separation, and alignment [35]. On the basis of behavioral rules and the tactical attack purpose, a control protocol acting on an individual a_i is formulated as follows:

$$u_i = f_i^d + f_i^v + f_i^s. \tag{26}$$

(1) f_i^d is a gradient-based term, which is utilized to regulate the positions between agents, i.e., cohesion and separation. For MAS, an agent is supposed to stay away from the nearby agents within a safe distance. When MAS tends to cause fragmentation, the agent keeps close to others. Here, f_i^d is formulated by using a virtual potential field:

$$f_i^d = -\nabla_{q_i} V_i(q^a), \tag{27}$$

where $V_i(q)$ is a collective potential function based on the relative distance between agent a_i and its neighbors in N_i. It can be defined using the potential function as:

$$V_i(q^a) = \sum_{j\in N_i} \psi_i\left(\left\|q_j^a - q_i^a\right\|_\sigma\right). \tag{28}$$

The set of neighbors agents N_i of agent a_i is defined as:

$$N_i = \left\{ j | \left\|q_i^a - q_j^a\right\| \leqslant R_c \right\}, \tag{29}$$

where q_i^a, q_j^a is the positions of agent a_i, a_j, respectively. R_c is the communication radius.

The $\|\cdot\|_\sigma$ represents a map $R^n \to R^+$, which is defined as:

$$\|z\|_\sigma = \frac{1}{\varepsilon}\left[\sqrt{1+\varepsilon\|z\|^2} - 1\right], \tag{30}$$

where $\varepsilon \in (0,1)$. The map $\|z\|_\sigma$ is differentiable everywhere, while $\|z\|$ is not differentiable at $z = 0$. This property of σ-norm is used for the construction of smooth collective potential functions for individuals.

As the potential function is designed, the following conditions are considered:

(1) $\psi_i(q)$ is continuously differentiable for $q \in \left(0, R_c\right)$.

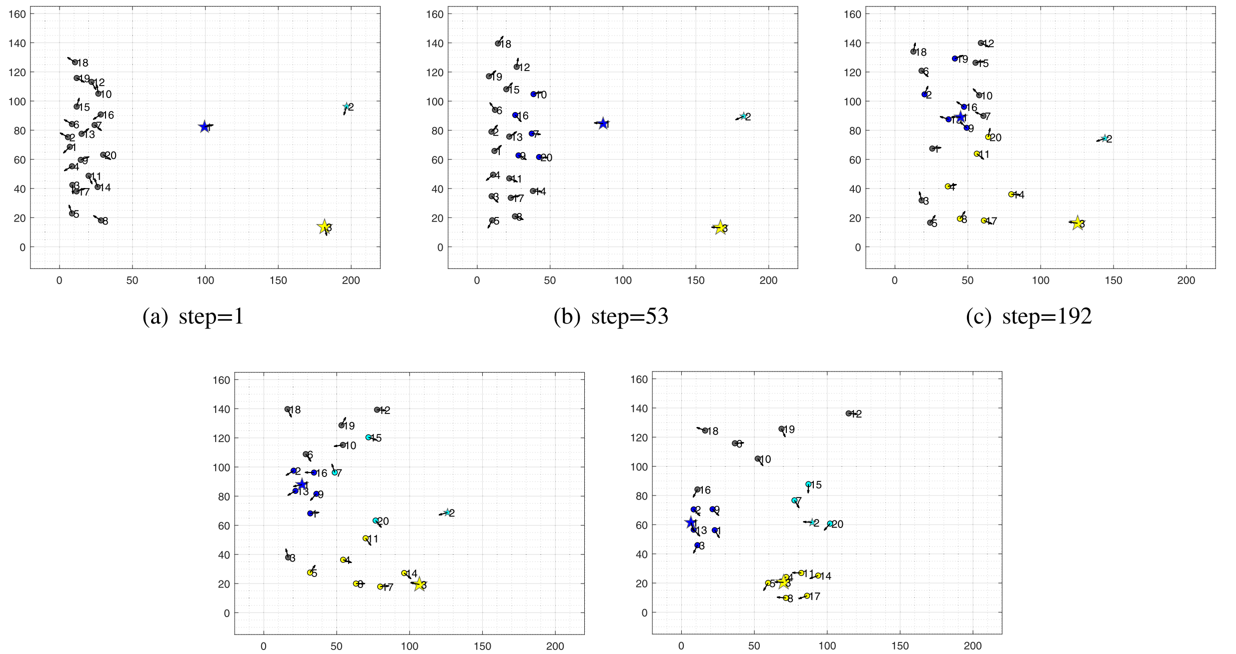
(2) $\psi_i(q)$ is monotonically decreasing for $q \in (0, \sigma_1]$, and increasing for $q \in (\sigma_2, R_c]$.

(3) $\psi_i(0) = \psi_{\max}$ and $\psi_i\left(R_c\right) = \psi_{\max}$.

Therefore, the pairwise attractive and repulsive potential $\psi_i(z)$ is defined as:

$$\psi_i(z) = \begin{cases} \frac{\tau^2}{z+\frac{\tau^2}{Q}} - z, & z \in (0, \sigma_1) \\ 0, & z \in [\sigma_1, \sigma_2] \\ \frac{(z-\sigma_2)^2}{R_c - z + \frac{(R_c-\sigma_2)^2}{Q}}, & z \in (\sigma_2, R_c) \end{cases}, \tag{31}$$

where τ and Q are fixed parameters. $[\sigma_1, \sigma_2]$ represent balance distances, satisfy $0 < \sigma_1 < \sigma_2 < R_c$. The reason for introducing balance distance is that if the expected distance between connecting agents is set to be constant, the energy gradient and control



(a) step=1 (b) step=53 (c) step=192

(d) step=255 (e) step=379

Fig. 7. Visualized target allocation results. Each star and its size represent each target's position and its reward (or demand), respectively. The circles indicate the positions of agents. The color of each circle implies that the corresponding agent is assigned to the same colored target. The grey circles represent agents joining an empty coalition. The demands of the three targets are $F_1 = 5$, $F_2 = 3$, and $F_3 = 6$, respectively.

term are both zero at this distance. The configuration of MAS is almost determined in this case, However, this expected distance is not conducive to the flexibility of configuration and performing tasks in confrontation. Therefore, the balance distance is used to replace the expected distance.

(2) f_i^v is responsible for the individual attempts to match velocity with its teammates, i.e., alignment, which is defined as:

$$f_i^v = k_v \sum_{j \in N_i} (p_j^a - p_i^a). \tag{32}$$

(3) f_i^s is responsible for moving towards tactical pursuit point (i.e., attack/defense stimulus). Based on the tactical pursuit point in (19), f_i^s is defined as:

$$f_i^s = k_{s_1}(q_i^* - q_i^a) + k_{s_2}(p_j^t - p_i^a). \tag{33}$$

Therefore, using (27), (32), and (33), the control protocol given in (26) can be rewritten as:

$$u_i = -\sum_{j \in N_i} \nabla_{q_i} \psi \left(\left\| q_j^a - q_i^a \right\|_\sigma \right) + k_v \sum_{j \in N_i} (p_j^a - p_i^a) + k_{s_1}(q_i^* - q_i^a) + k_{s_2}(p_j^t - p_i^a). \tag{34}$$

5. Simulation results

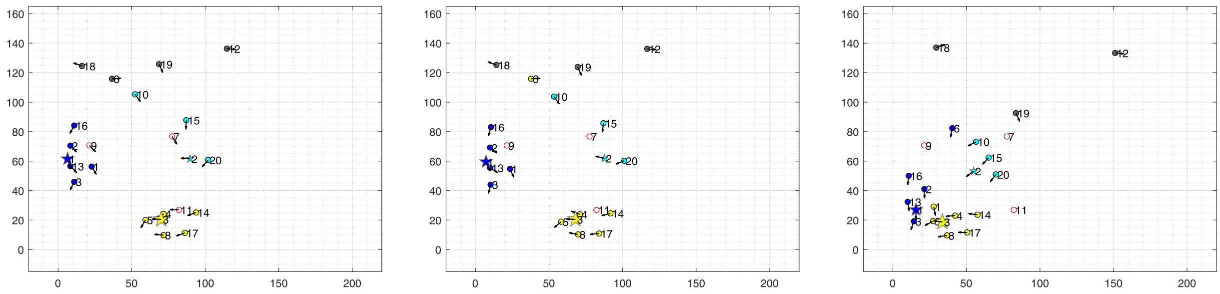
The simulation platform is MATLAB/SIMULINK R2019b. The simulation step size is 0.5s. Each step consists of three stages: target allocation, tactical decision-making, and swarm motion control.

In simulation, the circles and the stars indicate the agents and the targets, respectively. The size of the stars indicates the reward of the corresponding targets. For convenience, different colors of targets are used to distinguish different coalitions. The agents with the same color belong to the same coalition and attack/defend the same colored target. The grey circles represent the agents in empty coalition.

To evaluate the performance of the proposed framework and methods in confrontation, several simulations are conducted.

5.1. Target allocation result

In Section 4.2, we propose a target allocation method based on a hedonic game for forming coalitions to confront targets. Therefore, in this section, the effectiveness of the target allocation method will be validated.



(a) step=380 (b) step=386 (c) step=501

Fig. 8. Visualized re-allocation results after some agents are damaged. The red hollow circles indicate damaged agents.

In order to ensure the principle of univariate (or single task), in this section, the scenario is assumed to be that the targets and the agents do not hit each other. The targets only approach the agents, and the agents approach and follow the targets after sensing them. In the simulation, the number of agents is 20, and the number of targets is 3. The demands of the three targets are $F_1 = 5$, $F_2 = 3$, and $F_3 = 6$, respectively. The maximal reward is denoted as $V^j_{\max} = 1000 \cdot F_j$. the perception radius of the agent is $R_s = 50$, and it is randomly generated within an area 20×120 whose center is [20,75]. The target has prior knowledge of all agents' information and moves towards the agents with the largest situation superiority. The process of coalition is shown in Fig. 7. Because the agents are homogeneous, they are of the same size. The sizes of stars (targets) indicate different demands. Accordingly, the larger size the stars is, the greater the reward is. Thus, more agents will join the coalition. When the coalitions are formed, the agents with the same color belong to a same coalition and approach the same colored target.

Fig. 7(a) shows the initial state of agents and targets, then the targets moving towards the agents. As the target is outside the perception range of the agents, the agents maintain their current states. At step 53 (Fig. 7(b)), a_7 perceives t_1 and forms coalition S_1 with other agents through information transmission, and $S_1 = \{a_7, a_9, a_{10}, a_{16}, a_{20}\}$. In step 192 (Fig. 7(c)), a_{14} perceives t_3. a_{14} forms coalition S_3 with other agents, and $S_3 = \{a_4, a_8, a_{11}, a_{14}, a_{17}, a_{20}\}$. Notably, at this time, the members in S_1 change, $S_1 = \{a_1, a_2, a_9, a_{16}, a_{19}\}$. At step 255 (Fig. 7(d)), a_{20} perceives t_2 and forms coalition S_2 with other agents. At this time, $S_1 = \{a_1, a_2, a_9, a_{13}, a_{16}\}$, $S_2 = \{a_7, a_{15}, a_{20}\}$, and $S_3 = \{a_4, a_5, a_7, a_8, a_{18}, a_{14}\}$. Because the number of agents is redundant, the remaining agents join coalition S_0. At step 379 (Fig. 7(e)), it can be seen that members of each coalition are surrounding the targets.

It is worth noting that the members of coalition S_1 in Figs. 7(b)–7(e) may be different. The same case is in other coalitions. From (6) and (8), it can be seen that if the reward of a coalition changes (i.e., different targets) or the distance between an agent and a target changes, the utility of the agent also changes. Therefore, the coalition will change under the preference of (4). It indicates that agents can improve task execution efficiency by optimizing their utility. The results in Fig. 7 demonstrate that the target allocation method is effective as the number of targets increases.

Next, we will verify the effectiveness of the method when the agents in coalitions are damaged.

For the three coalitions in Fig. 7(e), it randomly specifies that one agent in each coalition is damaged, such as a_7, a_9, and a_{11}. The following allocation results are shown in Fig. 8. In 8(a), the three damaged agents are represented by hollow circles. In 8(b), a_{16}, a_{10}, and a_6 join in coalition S_1, S_2, and S_3, respectively. Afterward, these three agents began to approach their respective targets. At step 501 (Fig. 8(c)), as t_1 and t_3 are close, the members of coalitions S_1 and S_3 have changed. Agent a_1 deviates from S_1 to join S_3, while agent a_6 deviates from S_3 to S_1.

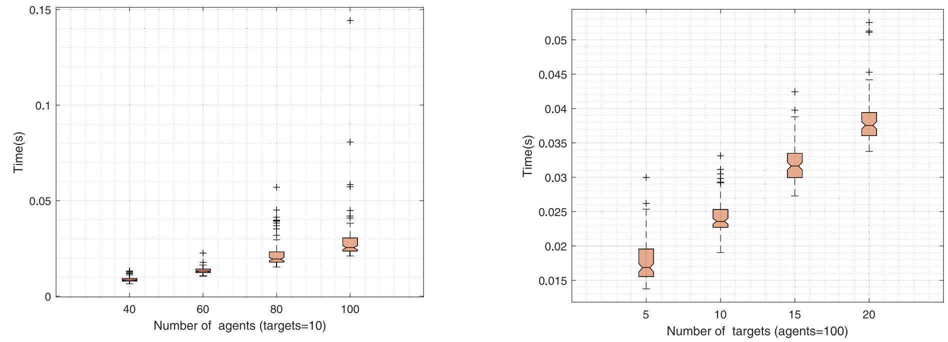
Besides, to evaluate the scalability of the proposed hedonic game method, we conduct Monte Carlo simulations with 100 runs for different scenarios involving different numbers of agents and targets. Fig. 9 shows the decision time of an agent in coalition formation, and the statistical results are represented by box-and-whisker plots. In Fig. 9(a), the number of targets is a fixed with $m = 10$ and the number of agents is various with $n \in \{40, 60, 80, 100\}$. In Fig. 9(b), the number of agents is a fixed with $n = 100$ and the number of targets is various with $m \in \{5, 10, 15, 20\}$.

In the two results, the decision time of an agent increases as the number of agents or targets increases. In Fig. 9(a), as the number of agents increases, they may have more conflicts while selecting the coalitions. Thus, an agent may spend more time resolving conflict. In Fig. 9(b), as the number of targets increases, the number of potential coalitions that the agent can join increases, resulting in an increase in decision time.

Finally, method comparisons are conducted to validate the proposed hedonic coalition formation game method.

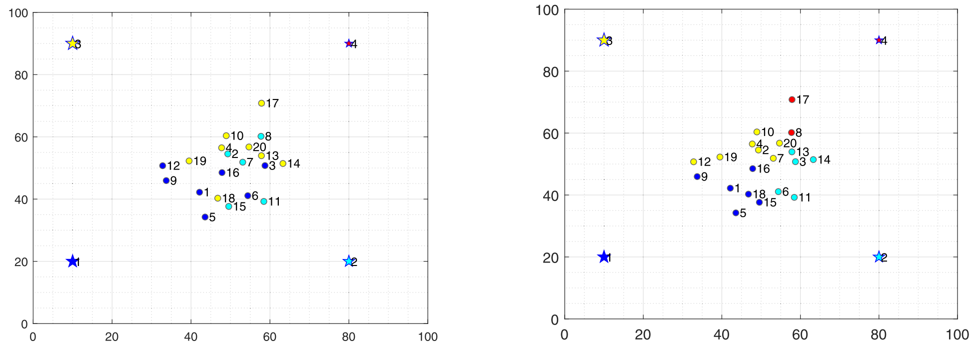
In [25], an anonymous hedonic coalition formation game method (represented as AHTA) is proposed, where the number of coalition members mainly depends on the reward of the targets. That is, the larger the reward is, the more coalition members there are. However, the shortcoming of AHTA method is that some targets may not be assigned to the agents. Fig. 10 shows a target allocation result between the proposed method (represented as HGTA) and AHTA method. In this simulation, the number of agents is 20, and the number of targets is 4. The demand of each target is $F_1 = 6$, $F_2 = 5$, $F_3 = 7$, and $F_4 = 2$. In both methods, the maximum reward for each target is the same, calculated by $V^j_{\max} = 1000 \cdot F_j$.

As shown in Fig. 10(a), due to the significant difference in rewards between target t_4 and other targets, no agents form coalition S_4, resulting in a problem of insufficient allocation. As a comparison, the proposed HGTA method effectively avoids the above



(a) Scenarios with different number of agents (b) Scenarios with different number of targets

Fig. 9. Decision time of an agent in scenarios involving different numbers of agents and targets.



(a) AHTA method in [25] (b) Proposed HGTA method

Fig. 10. Visualization results of method comparison.

problem. As shown in Fig. 10(b), each target has been assigned to corresponding agents, and the number of agents meets the demand of each target. Moreover, in the proposed method, the agents choose the coalition first based on distance, while the distribution of agents is filled with randomness in AHTA method. Therefore, the proposed HGTA method is more reasonable.

Meanwhile, we conduct another method comparison between the proposed HGTA method with the method proposed in [26] (represented as APTA). [26] investigates the same target allocation problem as our manuscript, where the minimum requirement constraint of each target is considered, and develops a game-theoretical decision-making framework to solve the problem. In this simulation, the number of targets is fixed with $m = 10$, and the number of agents is various with $n \in \{40, 60, 80, 100\}$. The demand for each target is the same, setting as 3. It is worth noting that in each scenario, the initial settings in the HGTA method and APTA method are identical. In each scenario, we conduct Monte Carlo simulations with 100 runs. Fig. 11 shows a comparative analysis of the total cost between the HGTA method and the APTA method. The total cost refers to the sum of distances among agents of all coalitions (except empty coalition) to their corresponding targets. A low total cost means fewer resources will be consumed to confront targets, which is beneficial to confrontation.

It can be observed from Fig. 11 that the total cost decreases with the increase of agents in both two methods. The reason is that as the number of agents increases, an agent closer to the target will prioritize joining the coalition, resulting in a decrease in total costs. However, in each scenario, the proposed HGTA method has a lower total cost than the APTA method. In the proposed HGTA method, the optimization of coalition cost is considered during the coalition formation algorithm [15], and each agent tends to join a coalition closer to itself. Therefore, the proposed HGTA method achieves a lower total cost.

Therefore, the proposed hedonic coalition formation game method for target allocation is effective, scalable, and adaptable to the dynamic changes in the number of agents and targets. Meanwhile, the number of members in coalitions satisfies the demands for confronting corresponding targets. When coalition members are damaged, other agents will join the coalition through re-allocation to meet the demands of targets. Moreover, during the process of system movement, the agents can also choose to join the most interested coalition, which has the maximal utility.

5.2. Tactical decision-making result

In this simulation, the number of agents is 20, and the number of targets is 3. The demands of the three targets are $F_1 = 5$, $F_2 = 3$, and $F_3 = 6$, respectively. The maximal reward is denoted as $V^j_{\max} = 1000 \cdot F_j$. Both the initial survival probabilities of agents

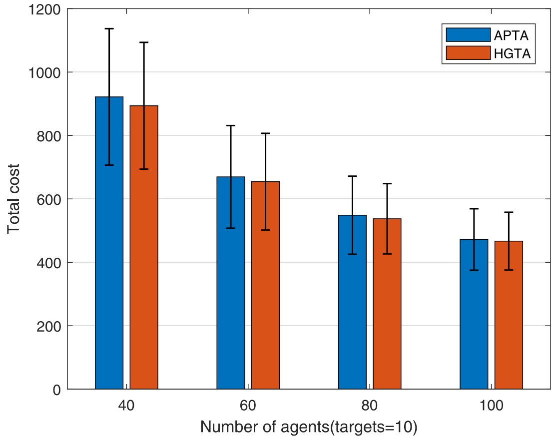
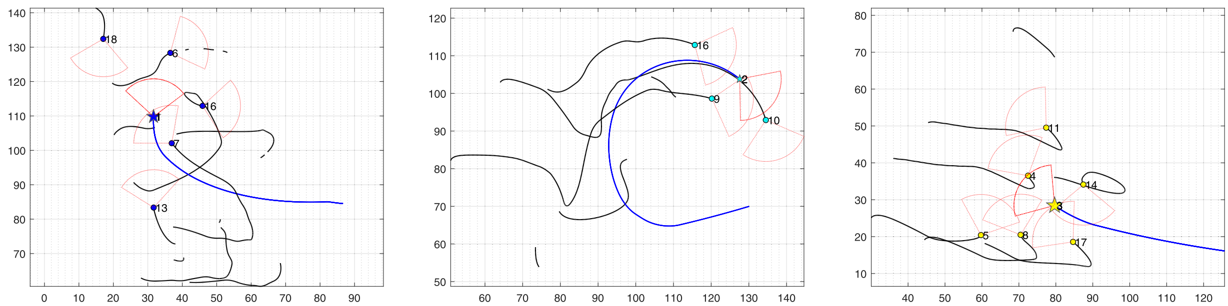


Fig. 11. Comparative results between the HGTA and APTA methods.



(a) Coalition S_1 confront t_1 (b) Coalition S_2 confront t_2 (c) Coalition S_3 confront t_3

Fig. 12. Visualized confrontation results. The black curve represents the trajectories of agents, the blue curve represents the trajectories of targets, and the red cone area represents the region of attack.

and targets are 1, and the survival probability thresholds of them are 0.1. The detect radius of agents is 50. The kill capability of agents is $p^{a}_{attack} = 0.1$ while the kill capabilities of targets are $p^{t_1}_{attack} = 0.3$, $p^{t_2}_{attack} = 0.2$, and $p^{t_3}_{attack} = 0.4$.

In the comparison, the pure-pursuit-based method is used for the targets. Each target selects an agent with the largest situation superiority to attack. Fig. 12 shows the trajectories of the three coalitions confront targets. Unlike saturation attack methods, in our approach, each agent attack target may not be synchronized, which is more in line with the actual situation.

As analyzed in Section 5.1, coalition members are not fixed in confrontation. During the process of attacking targets, different agents are allowed to join or leave the coalition. Therefore, in Fig. 12, there are some isolated trajectories, which are trajectories of agents who have previously joined the coalition and then left it later. Both targets and agents adjust their maneuver decisions through the situations and try to obtain a favorable situation. Therefore, their trajectories are intertwined.

Next, taking Fig. 12(b) as an example, we analyze the performance of the tactical decision-making algorithm in coalition S_2 attacking/defending target t_2.

The confrontation process between coalition S_2 and target t_2 varies from step 242 to step 579. The coalition members changed four times during the confrontation, as shown in Table 1. The situation type change curves of all agents in coalition S_2 who have participated in attacking/defending targets are shown in Fig. 13. It should be noted that when the target is outside the perception range of the agent or the agent does not belong to coalition S_2, the agent's situation type is 0.

In the first stage, agents a_9, a_{10}, and a_{20} are in a state of opposition to the target, and both sides individuals approach the other. Target t_2 approach the agent with the greatest advantage. At step 312, agent a_{20} senses the threat from the target and begins to take defensive measures, moving away from the target. At this point, its situation type of a_{20} is 2. At step 370, a_{20} left coalition S_2 and a_{11} joined coalition S_2. Meanwhile, target t_2 is moving towards a_{10}. The situation type of a_9 has become 1, so it gradually approaches the target from the rear to carry out an attack. At step 388, a_{11} left S_2 and a_{15} joined S_2. At step 426, a_{15} left S_2, and a_{16} joined S_2. At this point, target t_2 is approach agent a_{10}, while a_9 and a_{16} adjust their maneuvers to be behind t_2 and occupy a favorable position. Specifically, a_9 moves towards the inside of the target's movement path through inside pursuit, continuously reducing the distance from t_2. By taking a shorter path to gain tactical advantage, ultimately achieving a strike on t_2. Note that in our method, agent a_{10} is not simply running away when t_2 approaches it, as this may cause the target to turn to attack other agents

Table 1
Coalition formation process in confronting t_2.

Step	Coalition
242 – 369	$S_2 = \{a_9, a_{10}, a_{20}\}$
370 – 387	$S_2 = \{a_9, a_{10}, a_{11}\}$
388 – 425	$S_2 = \{a_9, a_{10}, a_{15}\}$
426 – 579	$S_2 = \{a_9, a_{10}, a_{16}\}$

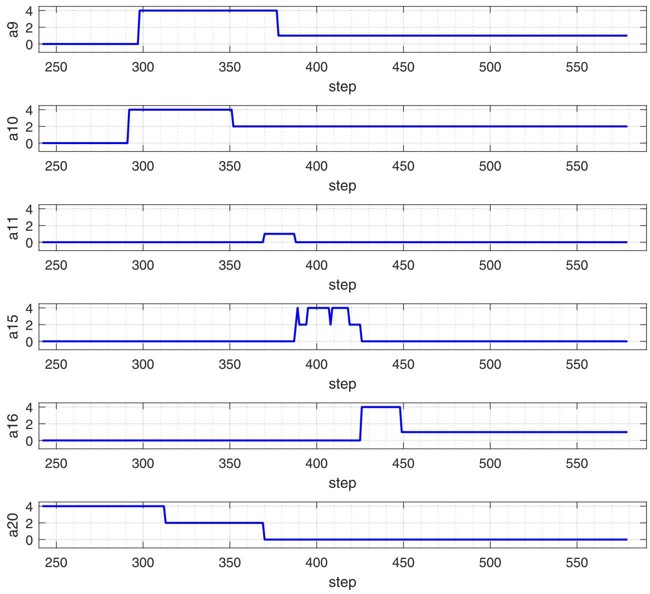


Fig. 13. The change curve of agents' situation type.

Table 2
Confrontation results with varying agent capabilities.

p^a_{attack}	Average number of surviving agents	Success rate
0.2	7.8	92%
0.3	10.3	95%
0.4	11.1	100%

or stay away from S_2. Here, agent a_{10} plays the role of luring t_2, moving towards the inside of the target's trajectory. It is beneficial for other agents to attack the target. Therefore, through tactical cooperation between coalition members, the target was destroyed.

From the above analysis, it can be seen that the agents in a coalition will adopt different maneuvering strategies when they are in different situations. The agents with advantageous situation types occupy a favorable position and are mainly responsible for attacking the target. The agents with disadvantageous situation types act as decoys to attract the target's attention, and they also try to switch their situation types. Through the cooperation of members in a coalition, the agents perform an effective attack on the target while reducing the casualties of agents. During the confrontation, the members of the agents in the coalition are also changing. From (8), it can be seen that agents closer to the target tend to join the coalition, which is beneficial for attacking the target.

Furthermore, the effectiveness of tactical decision-making algorithms in confrontation is verified by changing the capabilities of agents. In the simulation, the number of agents is 20, the attack capability of agents is various with $p^a_{attack} \in \{0.2, 0.3, 0.4\}$. The number of targets is 5, and the target's attack capability $p^t_{attack} = 0.6$. Set the maximum step to 1000 in the simulation. The success rate of the agents refers to all targets being destroyed within the maximum step. We conduct Monte Carlo simulations with 100 runs for these three scenarios. Table 2 shows the average number of surviving agents and success rate in the above three scenarios. When the agents' capability is low, even if they hit the targets, the damage to targets is limited. However, due to the targets' higher attack capability, they can quickly destroy the agents, resulting in a small average number of surviving agents and a low success rate for the agents. With the increase in the agents' capability, the probability of causing damage to the target after a single hit becomes higher, achieving a higher success rate. However, in these three scenarios, the success rates are more than 90%, which shows the effectiveness of the tactical decision-making algorithm in confrontation.

6. Conclusion

In this paper, we proposed a cooperation and decision-making framework for MAS confrontation. The framework is comprised of a target allocation algorithm, a tactical decision-making algorithm, and a swarm motion control algorithm. In the target allocation algorithm, a hedonic coalition formation game algorithm was designed to group the agents into disjoint coalitions corresponding to different targets. This grouping mechanism decreases some possible behavioral interactions that result from increasing numbers of agents. In the tactical decision-making algorithm, an ABFCM was designed to fuse the situation information of an agent. Then, based on the fusion result, a TPP-based method was designed to generate a pursuit point with tactical intention. Finally, a swarm motion control algorithm, including decision-oriented and swarm behavior rules-oriented, was designed to drive each agent towards the assigned target. Simulation results show the effectiveness of the designed framework and methods. The agents adjust their strategies according to the environment and cooperate with each other, defeating the stronger targets while reducing casualties.

Moreover, these proposed algorithms can also be applied to other scenarios, such as urban security, search and rescue missions, surveillance, and others. In these scenarios, agents cooperate to perform more complex tasks, thereby improving the efficiency of task completion.

Funding statement

This work was supported in part by the National Key Research and Development Program of China under Grant (2018AAA0102 404), in part by the National Natural Science Foundation of China under Grant (62073323), and in part by the Beijing Nova Program under Grant (20220484077).

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

References

[1] Ning B, Han Q-L, Zuo Z, Jin J, Zheng J. Collective behaviors of mobile robots beyond the nearest neighbor rules with switching topology. IEEE Trans Cybern 2017;48(5):1577–90.
[2] Yao P, Zhu Q, Zhao R. Gaussian mixture model and self-organizing map neural-network-based coverage for target search in curve-shape area. IEEE Trans Cybern 2020;52(5):3971–83.
[3] Zhen Z, Wen L, Wang B, Hu Z, Zhang D. Improved contract network protocol algorithm based cooperative target allocation of heterogeneous UAV swarm. Aerosp Sci Technol 2021;119:107054.
[4] Hu C. A confrontation decision-making method with deep reinforcement learning and knowledge transfer for multi-agent system. Symmetry 2020;12(4):631.
[5] Giles K, Giammarco K. Mission-based architecture for swarm composability (MASC). Procedia Comput Sci 2017;114:57–64.
[6] MacNulty DR, Smith DW, Mech LD, Vucetich JA, Packer C. Nonlinear effects of group size on the success of wolves hunting elk. Behav Ecol 2012;23(1):75–82.
[7] Xia W, Zhou Z, Jiang W, Zhang Y. Dynamic UAV swarm confrontation: An imitation based on mobile adaptive networks. IEEE Trans Aerosp Electron Syst 2023.
[8] Xu J, Zhang J, Yang L, Liu C. Autonomous decision-making for dogfights based on a tactical pursuit point approach. Aerosp Sci Technol 2022;129:107857.
[9] Koyuncu E, Inalhan G. Exploiting delayed and imperfect information for generating approximate UAV target interception strategy. J Intell Robot Syst 2013;69:313–29.
[10] Kong D, Chang T, Hao N, Zhang L, Guo L. Confrontation-based cooperative fire strike decision-making method of assault weapons and support weapons. Acta Armament 2019;40(3):629.
[11] Alexopoulos A, Schmidt T, Badreddin E. Cooperative pursue in pursuit-evasion games with unmanned aerial vehicles. In: 2015 IEEE/RSJ international conference on intelligent robots and systems. IROS, IEEE; 2015, p. 4538–43.
[12] Chi P, Wei J, Wu K, Di B, Wang Y. A bio-inspired decision-making method of UAV swarm for attack-defense confrontation via multi-agent reinforcement learning. Biomimetics 2023;8(2):222.
[13] Jiang F, Xu M, Li Y, Cui H, Wang R. Short-range air combat maneuver decision of UAV swarm based on multi-agent transformer introducing virtual objects. Eng Appl Artif Intell 2023;123:106358.
[14] Tu S-Y, Sayed AH. Distributed decision-making over adaptive networks. IEEE Trans Signal Process 2013;62(5):1054–69.
[15] Wang L, Qiu T, Pu Z, Yi J, Yuan W. A distributed hedonic coalition formation method for multi-agent tasks. In: 2022 IEEE symposium series on computational intelligence. SSCI, IEEE; 2022, p. 308–14.
[16] Dutta A, Asaithambi A. One-to-many bipartite matching based coalition formation for multi-robot task allocation. In: 2019 international conference on robotics and automation. ICRA, IEEE; 2019, p. 2181–7.
[17] Guerrero J, Oliver G. Multi-robot coalition formation in real-time scenarios. Robot Auton Syst 2012;60(10):1295–307.
[18] Rauniyar A, Muhuri PK. Multi-robot coalition formation problem: Task allocation with adaptive immigrants based genetic algorithms. In: 2016 IEEE international conference on systems, man, and cybernetics. SMC, IEEE; 2016, p. 000137–42.
[19] Chen X, Zhang P, Du G, Li F. A distributed method for dynamic multi-robot task allocation problems with critical time constraints. Robot Auton Syst 2019;118:31–46.
[20] Lin P, Wang Y, Qi H, Hong Y. Distributed consensus-based k-means algorithm in switching multi-agent networks. J Syst Sci Complexity 2018;31:1128–45.
[21] Dias MB, Zlot R, Kalra N, Stentz A. Market-based multirobot coordination: A survey and analysis. Proc IEEE 2006;94(7):1257–70.

[22] Skaltsis GM, Shin H-S, Tsourdos A. A survey of task allocation techniques in MAS. In: 2021 international conference on unmanned aircraft systems. ICUAS, IEEE; 2021, p. 488–97.
[23] Brandt F, Conitzer V, Endriss U, Lang J, Procaccia AD. Handbook of computational social choice. Cambridge University Press; 2016.
[24] Saad W, Han Z, Basar T, Debbah M, Hjorungnes A. Hedonic coalition formation for distributed task allocation among wireless agents. IEEE Trans Mob Comput 2010;10(9):1327–44.
[25] Jang I, Shin H-S, Tsourdos A. Anonymous hedonic game for task allocation in a large-scale multiple agent system. IEEE Trans Robot 2018;34(6):1534–48.
[26] Huo X, Zhang H, Huang C, Wang Z, Yan H. Task allocation with minimum requirements for multiple mobile robot systems: A game-theoretical approach. IEEE Trans Netw Sci Eng 2023.
[27] Park H, Lee B-Y, Tahk M-J, Yoo D-W. Differential game based air combat maneuver generation using scoring function matrix. Int J Aeronaut Space Sci 2016;17(2):204–13.
[28] Chin HH. Knowledge-based system of supermaneuver selection for pilot aiding. J Aircr 1989;26(12):1111–7.
[29] Wang B, Li S, Gao X, Xie T. UAV swarm confrontation using hierarchical multiagent reinforcement learning. Int J Aerosp Eng 2021;2021:1–12.
[30] Wang Z, Liu F, Guo J, Hong C, Chen M, Wang E, Zhao Y. UAV swarm confrontation based on multi-agent deep reinforcement learning. In: 2022 41st Chinese control conference. CCC, IEEE; 2022, p. 4996–5001.
[31] Wang B, Li S, Gao X, Xie T. Weighted mean field reinforcement learning for large-scale UAV swarm confrontation. Appl Intell 2023;53(5):5274–89.
[32] You D-I, Shim H. Design of an autonomous air combat guidance law using a virtual pursuit point for UCAV. J Korean Soc Aeronaut Space Sci 2014;42(3):199–212.
[33] Ramírez López N, Żbikowski R. Effectiveness of autonomous decision making for unmanned combat aerial vehicles in dogfight engagements. J Guid Control Dyn 2018;41(4):1021–4.
[34] Reynolds CW. Flocks, herds and schools: A distributed behavioral model. In: Proceedings of the 14th annual conference on computer graphics and interactive techniques. 1987, p. 25–34.
[35] Olfati-Saber R. Flocking for multi-agent dynamic systems: Algorithms and theory. IEEE Trans Autom Control 2006;51(3):401–20.
[36] Beaver LE, Kroninger C, Malikopoulos AA. An optimal control approach to flocking. In: 2020 American control conference. ACC, IEEE; 2020, p. 683–8.
[37] Wang F, Chen Y. Fast convergent flocking control of multi-agent systems with switching communication topology. In: 2020 American control conference. ACC, IEEE; 2020, p. 695–700.
[38] Sakai D, Fukushima H, Matsuno F. Flocking for multirobots without distinguishing robots and obstacles. IEEE Trans Control Syst Technol 2016;25(3):1019–27.
[39] Jing G, Wang L. Multiagent flocking with angle-based formation shape control. IEEE Trans Autom Control 2019;65(2):817–23.
[40] Papageorgiou EI, Salmeron JL. A review of fuzzy cognitive maps research during the last decade. IEEE Trans Fuzzy Syst 2012;21(1):66–79.
[41] Stula M, Stipanicev D, Bodrozic L. Intelligent modeling with agent-based fuzzy cognitive map. Int J Intell Syst 2010;25(10):981–1004.
[42] Czarnecki E, Dutta A. Scalable hedonic coalition formation for task allocation with heterogeneous robots. Intell Serv Robot 2021;14(3):501–17.
[43] Bilò V, Fanelli A, Flammini M, Monaco G, Moscardelli L. Nash stable outcomes in fractional hedonic games: Existence, efficiency and computation. J Artificial Intelligence Res 2018;62:315–71.
[44] Monderer D, Shapley LS. Potential games. Games Econ Behav 1996;14(1):124–43.
[45] Duan H, Lei Y, Xia J, Deng Y, Shi Y. Autonomous maneuver decision for unmanned aerial vehicle via improved pigeon-inspired optimization. IEEE Trans Aerosp Electron Syst 2022.
[46] Huang C, Dong K, Huang H, Tang S, Zhang Z. Autonomous air combat maneuver decision using Bayesian inference and moving horizon optimization. J Syst Eng Electron 2018;29(1):86–97.