Physica A 491 (2018) 127–141

Contents lists available at ScienceDirect

Physica A

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

Modeling and simulation of dynamic ant colony's labor division for task allocation of UAV swarm

Husheng Wu [a], Hao Li [b], Renbin Xiao [c,*], Jie Liu [b]

[a] School of Equipment Management and Support, Armed Police Force Engineering University, Xi'an 710086, China
[b] Air Force Early-Warning Academy, Wuhan 430019, China
[c] School of Automation, Huazhong University of Science and Technology, Wuhan 430074, China

HIGHLIGHTS

- In this paper, we introduce our method for dynamic task allocation problem of unmanned aerial vehicle.
- A new dynamic ant colony's labor division (DACLD) model is proposed, which also has a better practicability in various multi-agent systems.
- DACLD can meanwhile get both UAVs' states and real-time positions, and has high degree of self-organization and flexibility under dynamic environments.

ARTICLE INFO

Article history:
Received 24 January 2017
Received in revised form 29 June 2017
Available online 14 September 2017

Keywords:
Ant colony's labor division
Dynamic task allocation
Response threshold model
UAV swarm

ABSTRACT

The problem of unmanned aerial vehicle (UAV) task allocation not only has the intrinsic attribute of complexity, such as highly nonlinear, dynamic, highly adversarial and multi-modal, but also has a better practicability in various multi-agent systems, which makes it more and more attractive recently. In this paper, based on the classic fixed response threshold model (FRTM), under the idea of "problem centered + evolutionary solution" and by a bottom-up way, the new dynamic environmental stimulus, response threshold and transition probability are designed, and a dynamic ant colony's labor division (DACLD) model is proposed. DACLD allows a swarm of agents with a relatively low-level of intelligence to perform complex tasks, and has the characteristic of distributed framework, multi-tasks with execution order, multi-state, adaptive response threshold and multi-individual response. With the proposed model, numerical simulations are performed to illustrate the effectiveness of the distributed task allocation scheme in two situations of UAV swarm combat (dynamic task allocation with a certain number of enemy targets and task re-allocation due to unexpected threats). Results show that our model can get both the heterogeneous UAVs' real-time positions and states at the same time, and has high degree of self-organization, flexibility and real-time response to dynamic environments.

1. Introduction

With the rapid development of technology, unmanned aerial vehicle (UAV) can work in different complicated or uncertain environments, and has been extensively used to perform various military tasks including surveillance, recon-

* Corresponding author.
E-mail address: rbxiao@hust.edu.cn (R. Xiao).

http://dx.doi.org/10.1016/j.physa.2017.08.094

naissance, attack, and damage assessment, etc. But there are some limits in weapons, ammunition, detection equipment and speed of a single UAV, which makes it difficult to separately complete complex multi-objective reconnaissance and massive attack missions. Naturally, UAV swarm (which consists of many heterogeneous UAVs) combat gradually attract increasing military attention because of its better robustness through redundancy and faster tasks accomplishment, and has being done extensive research as a new style in the future battlefield [1]. As is known to all, the core of UAV swarm combat lies in its complex task allocation problems under dynamic environment. The objective of task allocation is to maximize total reward obtained by destroying various targets which have different attributes, and maximize the overall operational effectiveness of UAV swarm under the premise of meeting various tactical and technical indexes. This problem presents some complex features, such as highly nonlinear, dynamic, highly adversarial, multi-modal, etc. Previous studies mainly focused on static task allocation and gave more attention about two important parts, modeling and solving algorithms.

As for task allocation model, due to the uncertainties and complexities in the dynamic battlefield, some scholars extend and improve several optimization models such as multiple traveling salesman problem (MTSP) [2], job shop scheduling problem (JSSP) [3], mixed-integer linear programming (MILP) [4] and multiple processors resource allocation (MPRA) [5], vehicle routing problem (VRP), etc. In [6], the task assignment of cooperating UAVs is classified as the polygon visiting multiple traveling salesman problem (PVMTSP) which can be solved by two proposed algorithms using genetic fuzzy; In [7], multiple UAVs' mission planning is modeled as a vehicle routing problem with time window and precedence requirements; Moreover, capacitated vehicle routing problem with time windows (CVRPTW) model, which has better ability to describe complicated battlefield scenarios, has been used in modeling for the planning problem of cooperative search mission carried by "Predators" and "Global Hawk" [8]. In [9], a heuristic task allocation method combining VRP and MILP is proposed to carry out various tasks of different time windows at various locations using a fleet of heterogeneous UAVs. For more complex task allocation problems, in [10], a cooperative multi-tasks the allocation problem (CMTAP) model was proposed, which can give a good description of the temporal relationship among different tasks and scheduling constraints.

Algorithms of solving task allocation problem are mainly divided into two categories: the centralized algorithms and the distributed ones. There are many centralized algorithms such as enumeration method, dynamic programming, branch-and-bound procedure, etc. Moreover, increasing researchers give more extensive attention on intelligent optimization algorithms, such as genetic algorithm (GA) [11], particle swarm optimization (PSO) [12], ant colony optimization (ACO) [13], wolf pack algorithm (WPA) [14], and cat swarm optimization (CSO) [15], etc. Although the centralized algorithms are easy to understand and can theoretically obtain the optimal scheme of task allocation, the huge computational complexity is a great challenge for center node. In addition, without fully using self-adaptation and cooperative characteristic of intelligent agents, the performance of the results obtained by centralized methods may be very ordinary, even poor in the dynamic environment [16]. While, the distributed algorithms fully consider the information interaction between heterogeneous UAVs, which is helpful for solving dynamic task allocation problem. Generally speaking, the distributed algorithms can be divided into two types: top-down and bottom-up. The top-down ones are mainly based on the ideas of layered hierarchical and divide-and-conquer, which decomposes the original problem into several sub-problems so as to solve those sub-problems well through consultations and cooperation among each other. There are many top-down methods, such as decentralized Markov decision process (Dec-MDP), distributed model predictive control (DMPC), dynamic distributed constraint optimization (DDCOP), contract net (CN), auction algorithm and other methods based on market auction mechanism [17]; While the bottom-up algorithms mainly focus on the optimization and coordination strategies based on stress response and behaviorism. Moreover, the bottom-up algorithms emphasize individual dynamic response to environment, and can emerge the global self-organization behavior through individual's local awareness and response interaction. Because of its simple calculation and good robustness, many researchers use it to solve complex task allocation problems [18].

These previous studies can be adopted for solving the problem of static task allocation (to UAV) in a straightforward manner. However, for the dynamic task allocation of UAV swarm combat, these approaches are weak in some important considerations: (1) The battlefield is highly uncertain and dynamic, while many current related studies based on static model have very ordinary real-time performance [19]; (2) UAV Swarm is composed of heterogeneous UAVs, while the UAVs' resource consumption and multiple attributes (such as flight speed, fuel consumption, stealth or anti-stealth and attack ability) are often not fully taken into account; (3) It is difficult to deal with task re-allocation due to unexpected enemy targets of which situation often emerges in actual battlefield. Meanwhile, especially for task allocation problem with a large number of UAVs, which is not suitable to apply centralized control or global model, swarm intelligent methods give a good solution to these complex distributed problems [20].

Actually, there are many similarities between biological collective behavior and tactical actions of UAV Swarm, including formation flying, cluster attack, cooperative reconnaissance and defense. Therefore, inspiration by biological systems, we can get better models and solutions for task allocation problem of UAV Swarm under complex battlefield environment. As is known to all, there are various biological collective behaviors, such as wolf pack hunting, ant colony foraging, birds migrating, shoal feeding, cattle resisting attack, etc. [21]. Among them, the labor division is one of the most important characteristics of ant colony, bee colony, wolf pack, fish school and other social creatures [22]. In 1996, Bonabeau [23] et al. proposed fixed response threshold model (FRTM) which is a classical ant colony's labor division model. It has advantages of modeling and implementing easily, and can reflecting the flexibility of complex problems of task allocation. And then, it has been continuously improved and widely applied in task allocation problems [24], management of enterprise production [25], wireless sensor network (WSN) [26] and swarm robot system [27]. In [28], a generalized ACLD model with multiple constraints and multiple individuals is proposed, which is used to construct the ACLD model with some specific constraints to solve task allocation of resilient supply chains; In [29], by introducing reinforcement learning, the response threshold reinforcement model has been proposed and applied to a swarm robotic system that performs food foraging; Castello et al. [30] propose an extended model of the simple response threshold model using a discretized version of the attractor selection paradigm, and applied in foraging mission of multi-robot system. These previous researches provide valuable references for its further improvement. However, especially for the classic ACLD, there are also some defects more or less, such as single task, single constraint, fixed response threshold and single individual response. While, the task allocation problem of UAV swarm combat is highly complicated, and has some distinctive characteristics of dynamism, multi-goal, multi-task, multi-status and multi-constraints. The state or solution space of this problem is very huge and complicated. If the heuristic stochastic algorithms or the classic ACLD are applied to solve the problem, there will be a number of invalid solution or local minimum value.

In this paper, based on the analysis of the strengths and weaknesses of the classic ACLD, FRTM, we improve it in dynamic response threshold, dynamic environmental stimuli, multi-type tasks and multi-state transition probability. The main contributions of this paper lie in proposing of the dynamic ant colony's labor division and its implementation to solve the task allocation problem in dynamic environment for UAV swarm combat. Because each UAV can be abstracted and viewed as an agent with many characteristics and constraints, Naturally, the proposed approach can provide a reference for other dynamic task allocation problems, such as multi-robot task allocation [31], dynamic task scheduling problem of CPU–GPU heterogeneous multi-core system [32] and load balancing in distributed systems [33], just that the research object is also viewed as agents while their characteristics and constraint conditions should be changed [34].

The remainder of this paper is organized as follows: In Section 2, we introduce and analyze the classic ACLD, fixed response threshold model (FRTM). Section 3 proposes the dynamic ant colony's labor division model for dynamic task allocation problem of UAV swarm combat. In Section 4, a detail description for task allocation problem of UAV swarm combat is proposed, and then the effectiveness of the proposed algorithm is demonstrated through several simulated scenarios of UAV swarm combat. Finally, Section 5 concludes this paper.

2. Classic ACLD model and its analysis

2.1. Fixed response threshold model

As for ant colony, except for the queens and reproductive males, the other ants work together to create a favorable environment. Tasks includes nest maintenance, foraging, brood care and nest defense, etc. While some ants may specialize on one task, most of the workers are totipotent, meaning that they are able to perform all tasks and can therefore switch from one task to another in response to the need of the colony [35]. Fixed response threshold model (FRTM), a good description about ant colony's collaborative labor division, is the most classic and basic ACLD model. The basic idea of FRTM is easy to follow, according to each ant's capabilities, each one has a response threshold corresponding to each task. For each ant, there is a fixed response threshold, and is corresponded to a special task. The ants' response threshold may reflect the actual difference between their response behaviors or perceived difference between the stimulation ways of tasks. There exists an environment stimulus corresponding to each task, and relative to the need of ant colony. Once the stimulus of a task exceeds the ant's response threshold, this ant begins to undertake this task. When an active ant exits from a task, then the need of the related task will increase, which result in the environmental stimulus rising. Finally, an ant begins to take this task once the environmental stimulus' value reaches to its response threshold. The mathematical description of FRTM mode is as follows:

- Environmental stimulus' value is time-varying
  An ant is stimulated by the environment, and the stimulus value is corresponding to a task one by one. What is more, the stimulus value s determines whether the ant response to this task or not. The stimulus dynamics is therefore described by the equation:

$$s(t+1) = s(t) + \delta - \varphi \cdot n_{act} \tag{1}$$

  where t denotes the discrete time variable, and $s(t)$ is the environmental stimulus' value at time t. There is a constant increase δ with every time step and a decrease of φ with every active worker, where φ is the efficiency of work (how many units of work an individual can do per time step), and n_{act} is the number of active individuals with task at time step t.
- The inactive individuals' response to the environmental stimulus
  Each ant i have a fixed response thresholds θ_i which corresponds to its response to a certain task. Let ST_i be the state of the ant i ($ST_i = 0$ corresponds to inactivity, $ST_i = 1$ corresponds to performing the task), and n be the constant which controls the curve shape of threshold function, generally $n = 2$. The emergence of a new task will trigger a stimulus to ants, and each ant will decide whether performing this task or not according to the following probability p:

$$P\left(ST_i = 0 \rightarrow ST_i = 1\right) = \frac{s^n}{s^n + \theta_i^n}. \tag{2}$$

- The probability of a active individual transforming to the idle one During a time step T, an active ant i can transform to the idle one according to P:

$$P\left(ST_i = 1 \rightarrow ST_i = 0\right) = p \tag{3}$$

where p is usually a constant, and $ST_i = 0 \rightarrow ST_i = 1$, $ST_i = 1 \rightarrow ST_i = 0$ are mutually independent.

2.2. Similarity analysis between UAV swarm task allocation and FRTM

The essence of ant colony's labor division is the task allocation, while the FRTM can reflects the flexibility of task allocation, because of which it is successfully applied in the real task allocation problem [35]. In particular, there are many similarities between the labor division of ant colony and the task allocation of UAV swarm combat:

(1) Once there is a task, generally speaking, ants as well as UAVs will form a team to collaboratively perform this task [36]; And just as ants mobilize several teams for a complex task, UAV swarm will also carry out complicated tasks in the form of several UAV unions [37,38];
(2) The individual behavior has flexibility, no matter for ants or UAVs, and they can make decisions according to the environment, task and their own ability. Moreover, this behavioral flexibility will produce the plasticity of group labor division, which ultimately maintain a balance for the whole system;
(3) The time-varying characteristics of the environmental stimulus in ant colony is similar to the variety of the battlefield situation and combat needs of UAV swarm;
(4) There are numerous ants in ant colony, and there are also great number of UAVs which are small-sized and low-cost. Because of so many individuals, naturally, the centralized control method of top-down is obviously low efficient [39]. UAV swarm has the distributed command structure, just like ants, whether a task is performed or not has a certain spontaneity rather than mandatory, and result in swarm emergence phenomenon by behavior response of numerous individuals.

2.3. Analysis about FRTM

Note that the classic ACLD, FRTM, shows a simple description of transformation and the driving mechanism for labor division phenomena in insect society. What is more, there are many similarities between the dynamic task allocation of UAV swarm combat and the FRTM, as mentioned in Sections 2.1 and 2.2, which contributes to the original purpose of using FRTM to solve this dynamic task allocation problem.

However, there are many tasks, multi-state individuals and some quantitative constraints in UAV swarm. Obviously, FRTM is difficult to directly suitable for the complex combat of UAV swarm combat: (1) There is only one task in FRTM, while there are multiple tasks (such as surveillance, reconnaissance, attack, and damage assessment) in UAV swarm battlefield; (2) There are only two states for an ant ($ST_i = 0$ corresponds to inactivity, $ST_i = 1$ corresponds to performing the task), While there are multiple states for a UAV because of its multiple missions; (3) An ant may be free to withdraw from the task or perform at any time in FRTM, while there may be multiple constraints in UAV swarm battlefield, such as constraints from UAVs' capabilities, ammunition, detection equipment and speed; (4) Each ant has single ability attribute corresponds to a single task in FRTM, while each UAV has its own abilities, and correspond to multi-type tasks for multiple targets; (5) Each task only need single individual's response in FRTM, while there need multiple UAVs cooperatively response to perform a complicated combat task; (6) As for the FRTM, the individual has little learning and memory ability, and the response threshold is single and fixed no matter how many times this individual repeatedly perform the same task, while it is well known that the battlefield experiences are of great help to execute the same type of subsequent tasks.

3. Dynamic ant colony's labor division

3.1. Improvement ideas

With the analysis in Section 2.2, naturally, it is necessary to redesign the classic ACLD, FRTM, so as to solve some complicated problems just like task allocation problem of UAV swarm combat. These problems usually have some characteristics such as multi-tasks, multi-targets, multi-constraints, multi-states, multi-attributes, time-varying response threshold and accumulation of self-learning experiences. Specifically, the improvement ideas can be summarized as follows:

(1) The multi-states of individuals should be fully considered, which are determined by their own abilities and attributes. For example, a Reconnaissance/Attack UAV may be an idle state or perform reconnaissance for N_g targets as well as attack

N_m targets. Naturally, there will be $N_g + N_m + 1$ states at $t = (k+1)T$ for one UAV U_i at $t = kT$; (2) the response threshold should be initially determined by individuals' multiple attributes, and time-varying with the environment and the individual states; (3) multiple environmental stimulus and response thresholds, corresponding to multiple sub-tasks (reconnaissance or attack) respectively, should be time-varying; (4) Because of the antagonism and complexity of UAV swarm combat, many constraints need be seriously considered, such as ammunition loading, abilities of reconnaissance or attack, et al.; (5) The coordination between individuals should be fully considered when solving highly complex problems. When the ammunition demands for destroying targets are far away from the capacity of single UAV, it need more UAVs response at the same time to collaboratively carry out this task; (6) the individual self-learning should also be given fully consideration. UAVs can accumulate experience when they repeatedly carry out some non-resource consumptive task (e.g. scout missions); (7) In addition, there are multiple tasks and multiple constrains for heterogeneous UAVs. The response thresholds are corresponding to multi-tasks for multi-targets, and time-varying with actual quantity of their own ammunition, amount of available fuel, the scope of reconnaissance or attack, et al.

Then, dynamic ant colony's labor division (DACLD) is proposed in the following sections and has the characteristic of multi-targets, multi-tasks, multi-states, time-varying response threshold, multi-constraints, multi-individual responses.

3.2. Description of each variable

Dynamic ant colony's labor division (DACLD) involves the following variables:

- $s_j^k(t)$ is the stimulus value of task k for target j at time t, where $j = 1, 2, \ldots, N_A$ is the No. of targets, N_A is the number of targets; $k = 1, 2, \ldots, N_M$ is the type code of tasks, N_M is the type number of tasks.
- δ_j^k is the increment of environmental stimulus within each time step in performing the task k for target j;
- $\theta_{ij}^k(t)$ is the response threshold of agent i performing task k for target j at time t, where $i = 1, 2, \ldots, N_U$ is the No. of N_U agents;
- δ_{ij}^k is the factor measuring reduced demand caused by agent i carrying out the task k for target j, which also means the execution efficiency within time step;
- $n_{act}^{jk}(t)$ is the quantity of agents who perform task k for target j at time t;
- $x_{i,j}^k(t)$ is the state variable of agent i, where $x_{i,j}^k(t) = 1$ means that agent i is performing task k for target j at time t, while $x_{i,j}^k(t) = 0$ means agent i not performing task k for target j at time t;
- $P\left(x_{i,j}^k(t) = 0 \to x_{i,j}^k(t+T) = 1\right)$ is the probability of agent i performing task k aiming at target j at the next time $t + T$ while agent i not performing task k for target j at time t;
- $P\left[x_{i,j}^k(t) = 1 \to x_{i,j^*}^{k^*}(t+T) = 1\right]$ is the probability of agent i who currently performing task k for target j, and then transmitting to task k^* for target j^* at next time $t + T$;
- $\zeta_i^k(t)$ is the learning factor of agent i, which shows the accumulation of experiences after it repeatedly performs task k, often result in the improved efficiency of task execution;
- φ_i^k is the forgetting factor of agent i, that is, once agent i performs task k, the response threshold of agent i to perform other tasks will be increased.
- $\upsilon_i^k(t)$ is the resource consumptive factor of agent i performing task k (of course, the tasks are resource consumptive) at time t.

Except for $s_j^k(t)$ and $\theta_{ij}^k(t)$, the other variables are input variables. As for DACLD, ants can be view as agents, and each agent is corresponding to a set of response thresholds. What is more, the tasks divides into two categories: non-resource consumptive (e.g. reconnaissance task) and resource consumptive (e.g. attack task). If N_S targets need to perform reconnaissance task at time t, the response threshold set of the UAV U_i for performing reconnaissance is $\Phi_i^k = \left\{\theta_{i1}^k(t), \theta_{i2}^k(t), \ldots, \theta_{iN_S}^k(t)\right\}$, here $k = 1$; If N_A targets need to be attacked, then the response threshold set will be $\Phi_i^k = \left\{\theta_{i1}^k(t), \theta_{i2}^k(t), \ldots, \theta_{iN_A}^k(t)\right\}$, here $k = 2$. Moreover, each target j corresponds to stimulus $s_j^k(t)$ at time t. Generally, an agent will decide whether or not to perform task k by taking the environmental stimulus $s_j^k(t)$ and its response threshold $\theta_{ij}^k(t)$ both into consideration. In particular, the agent more likely perform the task k if $s_j^k(t)$ is stronger while $\theta_{ij}^k(t)$ is lower.

3.3. Dynamic environmental stimulus

In dynamic battlefield environment, the environmental stimulus $s_j^k(t)$ is comprehensively determined by the priority, value, risk factors of task k for target j. The environmental stimulus can change over time, and the higher stimulus will attract more agents. Specifically, urgency or priority usually reflects some tactical intention for UAV swarm combat. And the dynamic environment stimulus $s_j^k(t)$ mainly involves the following details:

(1) There is an environmental stimulus for each task, which reflects how urgent the task is; it is easier to attract more agents to perform this task if the stimulus value is higher. If the task is not completed, the corresponding stimulus

value will be increased over time as follows:

$$s_j^k(t+1) = s_j^k(t) + \delta_j^k - \sum_{i=1}^{N_U} \left[\partial_{ij}^k \cdot x_{i,j}^k(t) \right] \tag{4}$$

$$n_{act}^{jk}(t) = \sum_{i=1}^{N_U} x_{i,j}^k(t). \tag{5}$$

(2) Practically, for the same target, there is a certain executive order between different types of tasks. For example, only after UAV scouting the target so as to know the information of the enemy, UAV may successfully attack this target.
If $k = 1, 2, \ldots, N_M$ is the type code of tasks, $k^* = k_1, k_2, \ldots, k_{N_M}$ the sequential type code of the tasks, N_M is the number of tasks' type, there will be the following equation:

$$s_j^{k_1}(t) \gg s_j^{k_2}(t) \gg \cdots \gg s_j^{k_{N_M}}(t). \tag{6}$$

In particular, $s_j^{k_1}(t) = 0$ if the task k_1 has been completed. In addition, if the current calculation aims at the environmental stimuli of task k_1 for target j, then $s_j^{k_2}(t) = s_j^{k_3}(t) = \cdots = s_j^{k_{N_M}}(t) = 0$.

(3) If $x_{i,j}^k(t) = 1$, that is, the current state α of agent i is to carry out task k for target j at time t; while $x_{i,j*}^{k^*}(t+T) = 1$ means the state β of agent i at time $t+T$. The relative environmental stimulus $\widetilde{s}_{\alpha\beta}$, a comparison between state α to state β, can be calculated as follows:

$$\widetilde{s}_{\alpha\beta} = \frac{s_{j*}^{k*}(t+T)}{s_j^k(t)}. \tag{7}$$

3.4. Dynamic response threshold

The response threshold is relevant to the abilities, states, resources capability and battlefield of agent i, and can updated according to the following rules:

3.4.1. For the non-resource consumptive task

$\zeta_i^k(t) < 1$ is the learning factor of agent i and inversely proportional to the number of execution times for task k, which shows the accumulation of experiences after agent i repeatedly performs task k many times. Moreover, it is:

$$\zeta_i^k(t) = \begin{cases} \dfrac{1}{N_k(t-T)} \cdot Stu, & N_k(t-T) \neq 0 \\ Stu, & N_k(t-T) = 0 \end{cases} \tag{8}$$

where $N_k(t-T)$ shows the number of times that agent i has carried out the same type of task k before the moment t; $Stu \in (0, 1)$ is the initial learning factor.

$\varphi_i^k > 1$ is the forgetting factor of agent i, and can be calculated by the following equation:

$$\theta_{ij}^k(t+1) = \begin{cases} \zeta_i^k \cdot \theta_{ij}^k(t), & k = k^* \\ \varphi_i^k \cdot \theta_{ij}^k(t), & k \neq k^*. \end{cases} \tag{9}$$

3.4.2. For the resource consumption task

$\upsilon_i^k(t)$ is the resource consumptive factor, which shows the resource consumptive factor of agent i performing the task k at time t. $\upsilon_i^k(t)$ is relevant to the initial resource ownership D_{imax} and the current resource ownership $D_i(t)$:

$$\upsilon_i^k(t) = \begin{cases} \dfrac{D_{imax}}{D_i(t)}, & D_i(t) \neq 0 \\ \infty, & D_i(t) = 0. \end{cases} \tag{10}$$

Once agent i performs the resource consumptive task k for target j, its response threshold will be updated according to the following equation:

$$\theta_{ij}^k(t+T) = \upsilon_i^k(t) \cdot \theta_{ij}^k(t). \tag{11}$$

Moreover, agents must transform among states so that there also need to compare response thresholds in different states and calculate relative response threshold:

$$\widetilde{\theta}_{\alpha\beta} = \frac{\theta_{ij*}^{k^*}(t+T)}{\theta_{ij}^k(t)} \tag{12}$$

where $\theta_{ij}^{k}(t)$ and $\theta_{ij^*}^{k^*}(t+T)$ respectively correspond to the response threshold of agent i in the state α and state β.

In addition, once an agent performs a task, its response threshold corresponding to this type of task will be decreased. At the same time, its response threshold to other tasks will be increased. Considering the resources constraints, the response threshold corresponding to the resource consumptive task will increase. Moreover, the response threshold will be infinite if agent's resources exhausted so as to lose ability to perform such task.

3.5. Transition probability

Agents only transform between only two states in FRTM. However, as for the DACLD, each agent usually can carry out multi-type tasks for multiple targets. The transition probability of agent i, which is in the two states $ST_i(t)=\alpha$ and $ST_i(t+T)=\beta$, can be calculated by the next formula:

$$\begin{aligned} P\left[ST_i(t)=\alpha \rightarrow ST_i(t+T)=\beta\right] &= P\left[x_{i,j}^{k}(t)=1 \rightarrow x_{i,j^*}^{k^*}(t+T)=1\right] \\ &= \frac{\left[\widetilde{s}_{\alpha\beta}\right]^n}{\left[\widetilde{s}_{\alpha\beta}\right]^n+\left[\widetilde{\theta}_{\alpha\beta}\right]^n+\rho\cdot\left[\Delta\tau_i\left(j\rightarrow j^*\right)\right]^n} \end{aligned} \tag{13}$$

where n denotes the constant which controls the curve shape of threshold function, and generally $n=2$; $\Delta\tau_i(j\rightarrow j^*)$ is the response time of agent i who transit from task k for target j to task k^* aiming at target j^*, ρ is the delay penalty coefficient.

The probabilities of agent i transfer from $ST_i(t)=\alpha$ to other states are also calculated, and then make a comparison, agent i will transfer from state α to state β^* if $P\left[ST_i(t)=\alpha \rightarrow ST_i(t+T)=\beta^*\right]$ is the biggest one. Additionally, the idle agent i should meet one of the following two equations:

$$s_i(t)=\min\left\{s_j^k(t), j=1,2,\ldots,N_A, k=1,2,\ldots,N_M\right\} \tag{14}$$

$$\theta_i(t)=\max\left\{\theta_{ij}^k(t), j=1,2,\ldots,N_A, k=1,2,\ldots,N_M\right\}. \tag{15}$$

3.6. Simulation rules and steps

Simulation rules involved in DACLD should be considered: (1) Within every time step T, agents can comprehensively consider some factors such as their own state, abilities, resources capability, demand and benefits for external tasks, et al. And then agents can decide whether to join in or quit from a task; (2) Agent has the ability to learn and accumulate experiences through executing task, which embodies in the adjustment of the response threshold for the same task; (3) For the same target, different types of tasks have different sequences or priorities; (4) once a task is performed by an agent, if the ability of executing agent meets the needs of task k for the target j, then the stimuli of target j in the environment to other agent will rapidly decrease, thus the probability chosen by other agent to perform the task k for target j is greatly reduced; Reversely, if the ability of executing agent does not meet the needs of target j, then this stimulus will be rapidly increased so as to attract other agents to join in. When a target is non-execution state for a long time, its stimulus will continuously increase to until some agents perform this task.

Meanwhile, steps involved in DACLD are shown below.

Step 1. Initialize the discrete time variable $t=0\cdot T$, response threshold $\theta_{ij}^k(0)$, environmental stimuli $s_j^k(0)$, learning factor $\zeta_i^k(0)$, forgetting coefficient φ_i^k, resource consumptive factor $\upsilon_i^k(0)$, delay penalty coefficient ρ;

Step 2. Calculate the relative environmental stimulus $\widetilde{s}_{\alpha\beta}$ and relative response threshold $\widetilde{\theta}_{\alpha\beta}$ of each agent to every task, according to Eqs. (7) and (12);

Step 3. Conclude the state transition probability of each agent according to Eq. (13) and to make decisions on the next state of each agent;

Step 4. Update the environmental stimulus' values according to the formula (4), if $s_j^k(t)>0$, it will transfer to step 2, otherwise, turn to step 5;

Step 5. Output the results.

4. Numerical simulation for UAV swarm combat

In this section, firstly, there is a detail model description for dynamic task allocation problem of UAV swarm combat, including conflict resolution during a mission horizon; then, Section 4.3 addresses the simulation setup. Finally, simulation verification and results analysis are given under two practical cases (with a certain number of enemy targets and due to unexpected threats).

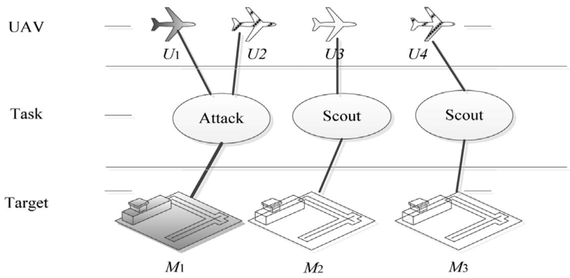

Fig. 1. The illustration of task allocation problem of UAV swarm (U_1, U_3, U_2 and U_4 denote the attack, reconnaissance and reconnaissance/attack UAV, respectively).

$$X = (\underbrace{1, 2, 3, 4, 5,}_{\text{reconnaissance}} \underbrace{6, 7, 8, 9, 10}_{\text{Attack}})$$

Fig. 2. UAV task coding.

4.1. Problem statement

The dynamic task allocation problem of UAV swarm combat can be described as follows: how UAV swarm collaboratively attack and destroy the N_A targets as soon as possible and maximize the total rewards. As shown in Fig. 1, UAV swarm, consisting of 4 heterogeneous UAVs, collaboratively perform reconnaissance and attack on three ground targets.

Meanwhile, we assume the following: (1) the battlefield does not have no-fly zones, obstacles and emergent threats; (2) UAVs all can freely communicate with each other; (3) The damage assessment to target is usually taken by the UAV who attack this target; (4) Targets are mutually independent, that is to say, the problems of collaborative anti-scout and firepower defense are not considered.

Thus it is desirable that UAV swarm destroys as many targets as quickly as possible so as to attain the maximal total reward, which can be described as follows:

$$\max f = \sum_{j \in J(t)} E_j(t) \tag{16}$$

$$J(t) = \left\{ j \middle| \sum_{i \in I(\tau)} D_i(\tau) > B_j \right\} \tag{17}$$

$$I(\tau) = \left\{ i | x_{i,j}^{k*}(\tau) = 1 \right\}, \tau \in \{0, T, 2T, \ldots, t\} \tag{18}$$

where $E_j(t)$ represents the value of target j at time step $t J(t)$ is the set of destroyed targets; k^* is the attack task that UAV performs, $I(\tau)$ is the set of performing task k^* for target j from the beginning to the moment τ; B_j is the ammunition quantity to destroy target j; $D_i(\tau)$ is the ammunition quantity of UAV U_i at time step τ.

In the process of performing task, some conditions (e.g. UAV's resource, ability, location information, etc.) change over time; moreover, the addition of new targets, the value change of target or the change of the tactical, all of them form the dynamic attributes of this problem. It is inevitable that many task conflicts need to be solved during dynamic task execution, which is shown in the following subsections.

4.2. Conflict resolution

Generally, if there is more than one UAV to perform the same task for the same target, there may be a kind of task conflicts. To encode easily, the same number of virtual targets are invented so as to be suitable for different targets and different tasks. For example, there are originally 5 targets. So, it encodes 5 targets which need to be scouted and 5 virtual targets (with the same attributes as 5 original targets) need to be attacked, and can get the task code as given in Fig. 2.

4.2.1. Conflict resolution of attack task

UAVs are allowed to attack the same target j, while the following conflicts and resolutions should be considered:

(1) If $D_j^*(t) < \min\{D_i(t)\}, i = 1, 2, \ldots, N_U, j = 1, 2, \ldots, N_A$, the UAV with the smallest response threshold will perform attack for target j, where $D_i(t)$ is the units amount of ammunition carried by UAV i, $D_j^*(t)$ means that there need $D_j^*(t)$ units of ammunition to destroy the target j at time t.

Table 1
UAV swarm information.

No	Reconnaissance capability	Attack	Ammunition quantity	Initial position	UAV type
1	0.7305	√	6	(0,0)	Reconnaissance/Attack
2	0	√	8	(0,0)	Attack
3	0.8288	×	0	(0,0)	Reconnaissance
4	0.8108	×	0	(0,0)	Reconnaissance
5	0	√	7	(0,0)	Attack
6	0	√	10	(0,0)	Attack

Table 2
Target information.

No	Position	Value	Ammunition requirement
1	(2.6906,6.8336)	7	4
2	(7.6550,5.4659)	9	3
3	(1.8866,4.2573)	4	3
4	(2.8750,6.4444)	7	4
5	(0.9111,6.4762)	4	4
6	(5.7621,6.7902)	4	3

(2) If $\min\{D_i(t)\} < D_j^*(t) < sum\{D_i(t)\}$, $i = 1, 2, \ldots, N_U$, $j = 1, 2, \ldots, N_A$, the UAV with less response threshold has more priority to execute attack mission for target j.
(3) If $D_j^*(t) > sum\{D_i(t)\}$, $i = 1, 2, \ldots, N_U$, $j = 1, 2, \ldots, N_A$, there needs more UAVs to join in attacking the target j.

In addition, if UAVs attack the same target simultaneously, whose ammunition will be first consumed is a tactical problem. In order to strive for greater flexibility in the battlefield, it should keep as many survival UAVs as possible. It should firstly consume the ammunition of Reconnaissance/Attack UAVs, and then consume attack UAVs' ammunition; and if there are two or more UAVs, then preferentially consume the one with less ammunition.

4.2.2. Conflict resolution of reconnaissance task
Generally speaking, it is unnecessary for multiple UAVs to scout the same target. The reconnaissance UAV with smaller response threshold have the priority to scout the target. Meanwhile, the environmental stimulus of is relevant to its value and ammunition requirement on the premise that target j has been scouted. And the larger its value and ammunition requirement is, the less environmental stimulus is.

4.3. Experimental setup

To validate the effectiveness of our proposed model and algorithm, it is tested under the following experimental setup: UAV swarm, consists of 6 heterogeneous UAVs, performs the collaborative combat missions aiming at six targets. The information of UAVs and targets is respectively shown in the following Tables 1 and 2.

The initial environmental stimulus $s_j^k(t=0)$ are only relevant to combat intention and values of targets and can be calculated by the following equation:

$$s_j^k(t=0) = {}^{E_j(t=0)}/_{E_{sum}} \tag{19}$$

where $s_j^k(t=0)$ is the initial stimulus of task k for target j at time step t, $E_j(t=0)$ is the value of target j at time $t=0$, and E_{sum} is the total value of all targets.

Generally speaking, the ability of UAV and its distance from target is the main basis of UAV decision. And stronger ability or closer distance will result in smaller response threshold, which can be calculated by the following equation:

$$\theta_{ij}^k(t) = d_{i,j}(t)/p_i^k \tag{20}$$

where $\theta_{ij}^k(t)$ is the response threshold of UAV i performing task k for target j at time t, and $d_{i,j}(t)$ is the relative distance from UAV i to target j at time t. If $k=1$ means reconnaissance missions, p_i^k denotes the relative reconnaissance capability; while $k=2$ means attack mission, p_i^k denotes the relative attack ability which is shown by relative amount of ammunition.

The parameters are set as follows: if $k=1$, ∂_{ij}^k is execution efficiency of agent i performing reconnaissance task aiming at target j; if $k=2$, ∂_{ij}^k is execution efficiency of agent i performing attack task aiming at target j, $\partial_{ij}^k(t) = D_i(t)$, where $D_i(t)$ is the ammunition quantity carried by U_i at time t; *Stu*, the initial learning factor of UAV is equal to the reconnaissance capability index; delay penalty coefficient $\rho = 0.2$. In addition, the UAV reconnaissance and combat all require a unit of time T. UAV flight speed in each unit of time T moves 2 units of distance. If the UAV has been assigned a task, it will go to the target airspace, then two actions should simultaneously be taken: perform the assigned tasks and assign tasks at the next time step; otherwise, it should move as follows: (1) If the UAV is the reconnaissance one, then returns; (2) If the UAV is the attack one, it will return if the amount of ammunition is 0; (3) If the UAV is the reconnaissance/attack one, if the amount of ammunition is 0 and no target need to be reconnoitered, then returns.

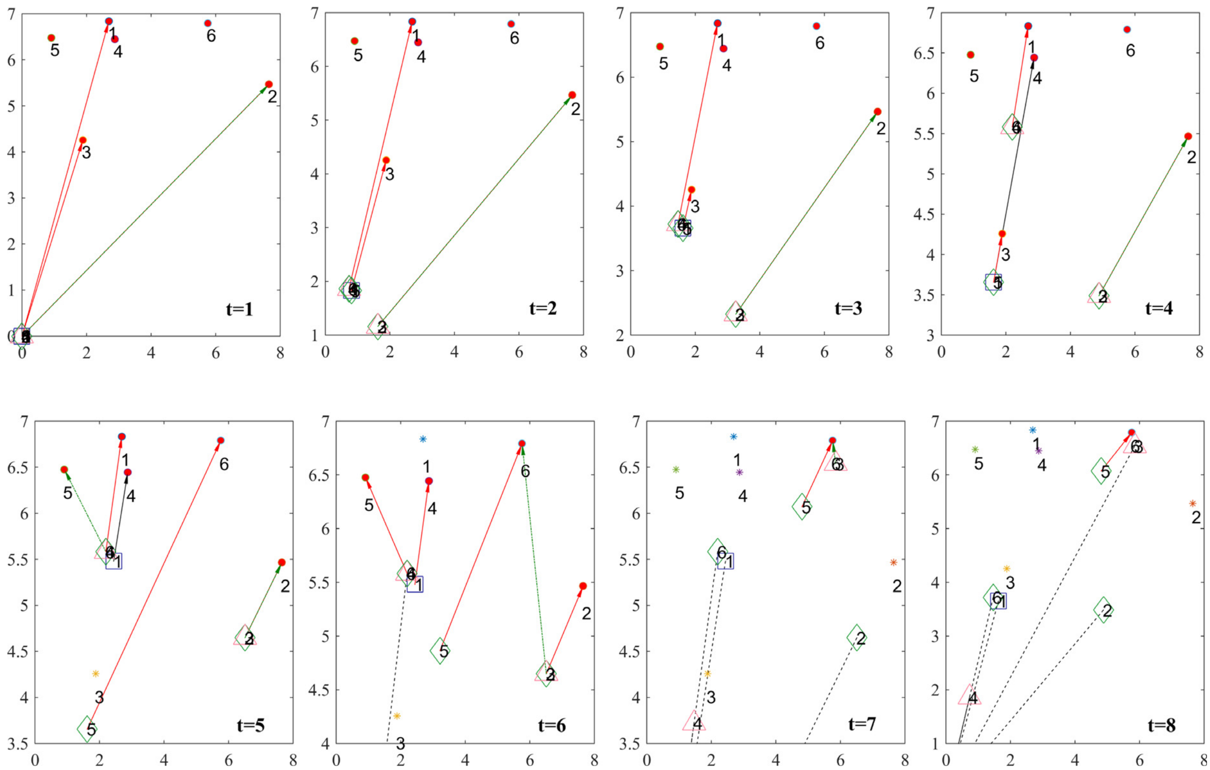

Fig. 3. The combat process diagram of UAV swarm (omit the process of UAVs returning to base) ('△' denotes reconnaissance UAV, '◇' means attack-UAV, '□' denotes Reconnaissance/Attack UAV, the number in the middle of each shape is UAV no; '•' indicates the survival targets, '*' denotes the destroyed targets; the green dotted lines are the trajectories of UAVs assigned reconnaissance missions, and the red solid lines are the trajectories for UAVs assigned attack missions, the black solid lines are the trajectories for UAVs assigned scout-attack missions, the black dashed lines are the trajectories of UAVs who return to the base).

4.4. *Simulation results*

In this subsection, we demonstrate the performance of the proposed DACLD for solving dynamic task allocation problems in two cases (task allocation with a certain number of targets and task re-allocation due to the unexpected threats).

4.4.1. *Dynamic task allocation problem with a certain number of targets*

It can assume that UAV swarm takes off from the base with the coordinate (0, 0). And then, after solve this task allocation problem using the proposed DACLD, there is a complete process diagram of the dynamic combat of UAV swarm, as is shown in Fig. 3. Then, the dynamic information of UAVs can be clearly and intuitively known at every moment, including their location, tasks, flight trajectory and target state.

To clear out the dynamic situation in which UAV swarms perform missions, the sequence diagram of UAV swarm operations is introduced in Fig. 4.

As shown in Figs. 3 and 4, the enemy targets are quickly assigned to UAV swarm, and the collaboration of multiple UAVs can also be reflected in the whole combat process. For example, the reconnaissance/attack UAV1 and the attack UAV5 form an alliance aiming at Target 3; the attack UAV2 and the reconnaissance UAV3 form an alliance aiming at Target 2; the reconnaissance UAV4 and the attack UAV6 form an alliance aiming at Target 1 and Target 5. The ability collaboration of UAVs greatly improves the efficiency of UAV swarm combat, which plays to the highlights of heterogeneous UAV cooperative combat.

Moreover, as shown in Fig. 4, the enemy targets are destroyed by UAV swarm after nine units of time, and the whole operational process is completed after 13 units of time, during which UAV swarm start from the base, take the cooperative reconnaissance and attack targets, and then return to the base. The whole combat process takes full advantages of the characteristics of heterogeneous UAVs, for instance, the reconnaissance/attack UAV1 successively performs the reconnaissance and attack missions aiming at target 3 and target 4; The reconnaissance UAV4 successively complete the scout mission aiming at target 1 and target 5; Attack UAV6 successively perform attack on target 1 and target 5. And the consistency of the combat mission is highly great, a drone attack will be taken right after the completion of reconnaissance. For example, after UAV4 finish its reconnaissance of target 1 at time $t = 5$, attack-UAV6 immediately launch attack on target

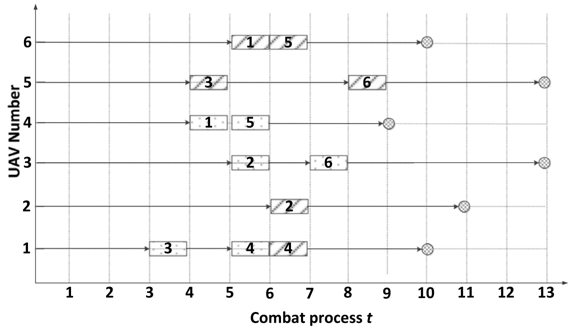

Fig. 4. The sequence diagram of UAV swarm operations ("X" shows UAV is performing the scout task aiming at target X, "Y" indicates the UAV is performing the Attack-task aiming at target, "" is the base).

1; reconnaissance UAV3 finish its mission for target 16 at time $t = 8$, attack-UAV5 immediately launch its attack on target 6. These fully explain that DACLD can make a highly efficient designation for a UAV swarm mission, and quickly complete its reconnaissance and attack task.

4.4.2. Task re-allocation due to the unexpected threats

However, in the dynamic battlefield, UAV swarm is supposed to face unpredictable dynamic events and should be able to smoothly respond to these events. In this subsection, to verify the robustness and flexibility of the proposed DACLD, we demonstrate the performance of our model in comparison with a task allocation method based on wolf pack algorithm (WPA) [40] under the same simulation environment. Assuming that there are 2 new enemy targets (No. 7 and No. 8, respectively) suddenly emerged at $t = 7$, whose values are 9 and 4, locations are (6.3571, 6.3571) and (5.4913, 5.0605) respectively, the amount of ammunition are 6 units and 3 units respectively, and we can get the combat process diagram of UAV swarm task execution under the situation of emerging unexpected threats at $t = 7$, shown as Figs. 5 and 6.

Compared with Figs. 5 and 6, we can know that the distributed DACLD can get a faster response for sudden threatens and can get a better task allocation scheme according the ability characteristics of UAVS. For example, as shown in Fig. 5, reconnaissance/attack UAV1 is fully utilized to perform reconnaissance and attack task aiming at target 4 during $t = 5$ and $t = 6$, and then UAV1 respond fast and perform the reconnaissance and attack task aiming at the unexpected enemy target 7, which is not expressed well for the task allocation method in [40]. Because the method based on WPA is a central task allocation method, which is good at pre-assignment and static task allocation, it may get a mediocre performance suffer the sudden threats in dynamic environment.

Fig. 7 shows the combat sequence diagram of UAV swarm under the situation of emerging unexpected threats. Compared with Figs. 3, 4 and 5, 7, we can know that at $t = 7$, the reconnaissance/attack UAV1, attack-UAV2 and attack-UAV6 should have returned to the base, but they were assigned new task (reconnaissance mission for target 7, attack task for target 7 and target 8, respectively) based on DACLD. Meanwhile, as shown in Fig. 5 and the left picture in Fig. 7, it also fully reflects the advantages of coordination between multi-UAVs. For example, aiming at the new unexpected target 8. UAV3 finish its reconnaissance mission for target 8 at time $t = 9$; while attack-UAV5 and attack-UAV6 are nearby target 8, the amount of ammunition carried by UAV6 is only 2 units after its attack on target 1 and target 5, and there is only one unit of ammunition carried by attack-UAV5 after its attack on target 3 and target 6. Based on the proposed DACLD, attack-UAV5 and attack-UAV6 are assigned to collaboratively attack on target 8, and destroy target 8 at time $t = 9$; During destroying the unexpected target 8, the collaboration between reconnaissance UAV3, attack-UAV5 and attack-UAV6 is fully presented. When dealing with the unexpected threat target 7, the Reconnaissance/Attack UAV1 and attack UAV2 also implements highly cooperation.

It also can be seen in Figs. 5 and 7 that, as for the results based on DACLD, except for UAV4, each UAV will almost synchronously return to the base after it performs its reconnaissance or attack on 8 targets. What is more, the ammunition quantity carried by UAVs which have the capacity to attack have been relatively evenly consumed before they returning. For instance, the ammunition quantity carried by UAV1, UAV2 and UAV5 is 1, 0, 0, 0, respectively while they returning.

Moreover, as is shown in Fig. 8, UAV swarm encounters the targets and launches its attack at $t = 4$, and there are two threats (unexpected target 7 and target 8) at $t = 7$. As compared with the method in [40], the proposed DACLD can make a rapid response, and 8 targets are destroyed at $t = 11$ after they have been scouted and attacked, which totally cost 7 units of time, while method in [40] cost more combat time, and 8 targets are destroyed at $t = 14$, which totally cost 10 units of time. These all show that DACLD has characteristic of good adaptability, self-organization and quick response. Without no global information or central guidance, ants can automatically make decisions according to their current status and environmental stimuli, which realize individual labor division and finally maintain a global dynamic balance. That means the number of ants that perform different tasks is always variable in response to the need of ant colony. On one hand, the plasticity of labor division of ant colony enables ants to adapt to environmental changes, which highlights its adaptability; On the other hand, it also makes the ant colony survive under the harsh circumstance or complete some complex tasks, which highlights its

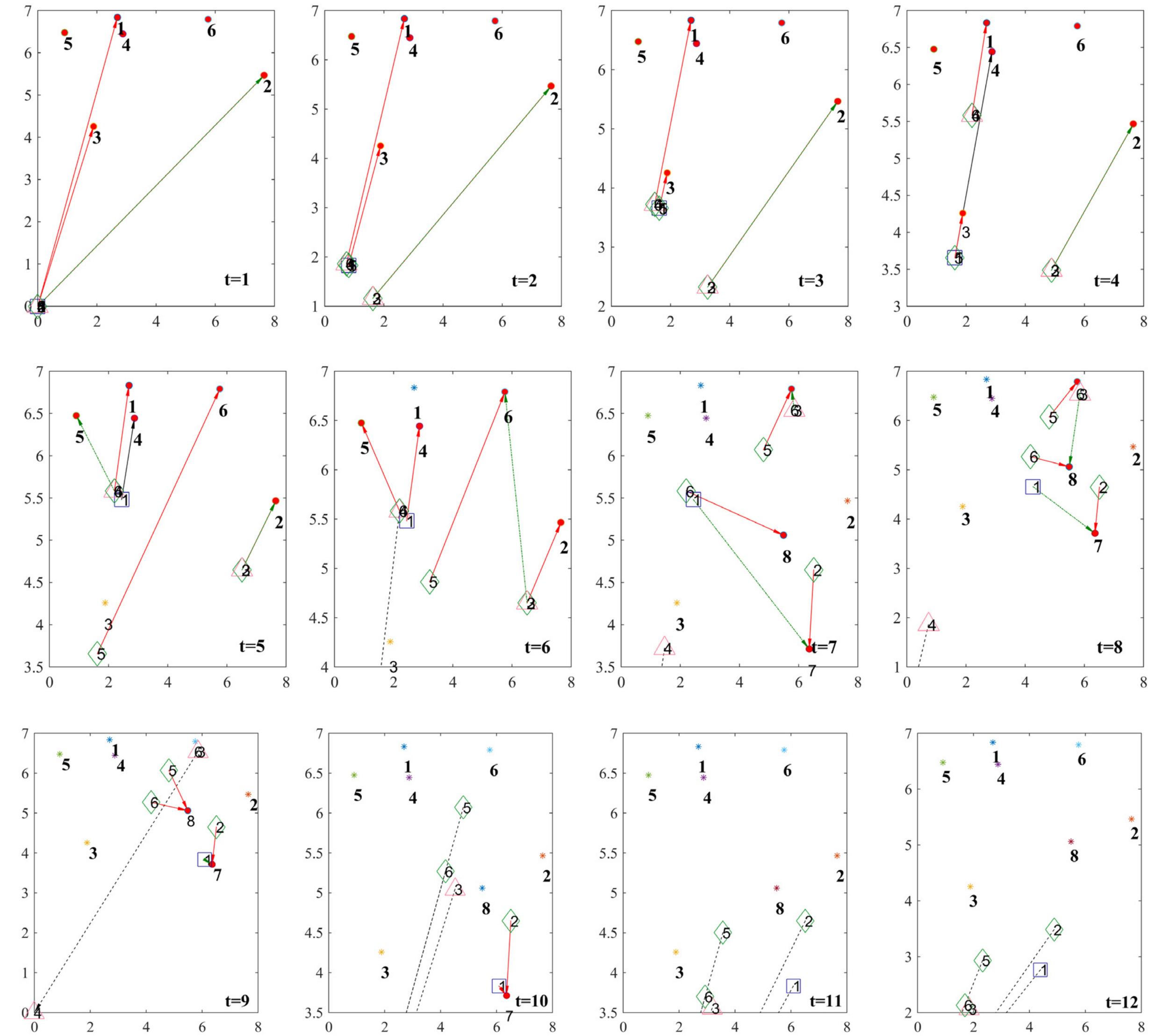

Fig. 5. The combat process diagram of UAV swarm based on DACLD (Omit the process of UAVs at beginning and returning to base. The symbols are the same as ones in Fig. 3).

robustness. Experimental results also indicate that the proposed DACLD can rapidly respond to some unexpected threatens, and can make timely adjustment according to the situations in battlefield, and can make a good task allocation scheme that beneficial for the global situations.

5. Conclusion

The dynamic task allocation problem of UAV swarm combat is highly complex, and its state space of task allocation will be very gigantic and complicated. If the heuristic random search algorithm is used to solve this problem, there will be a large number of invalid solutions and it is very easy to fall into the local minimum. Based on the full analysis about the classic ant colony's labor division, the new dynamic environmental stimulus, response threshold and transition probability are designed, and a "bottom-up" and distributed DACLD is proposed, which is characteristic of multi-task types, multi-states, self-adaptive response threshold and multi-individual responses. This method originates from the biological group, with high degree of self-organization and fast response. Then, this method is applied to the dynamic task allocation problem of UAV swarm combat, and two situations (with a certain number of targets and with the emergent threats) is given full

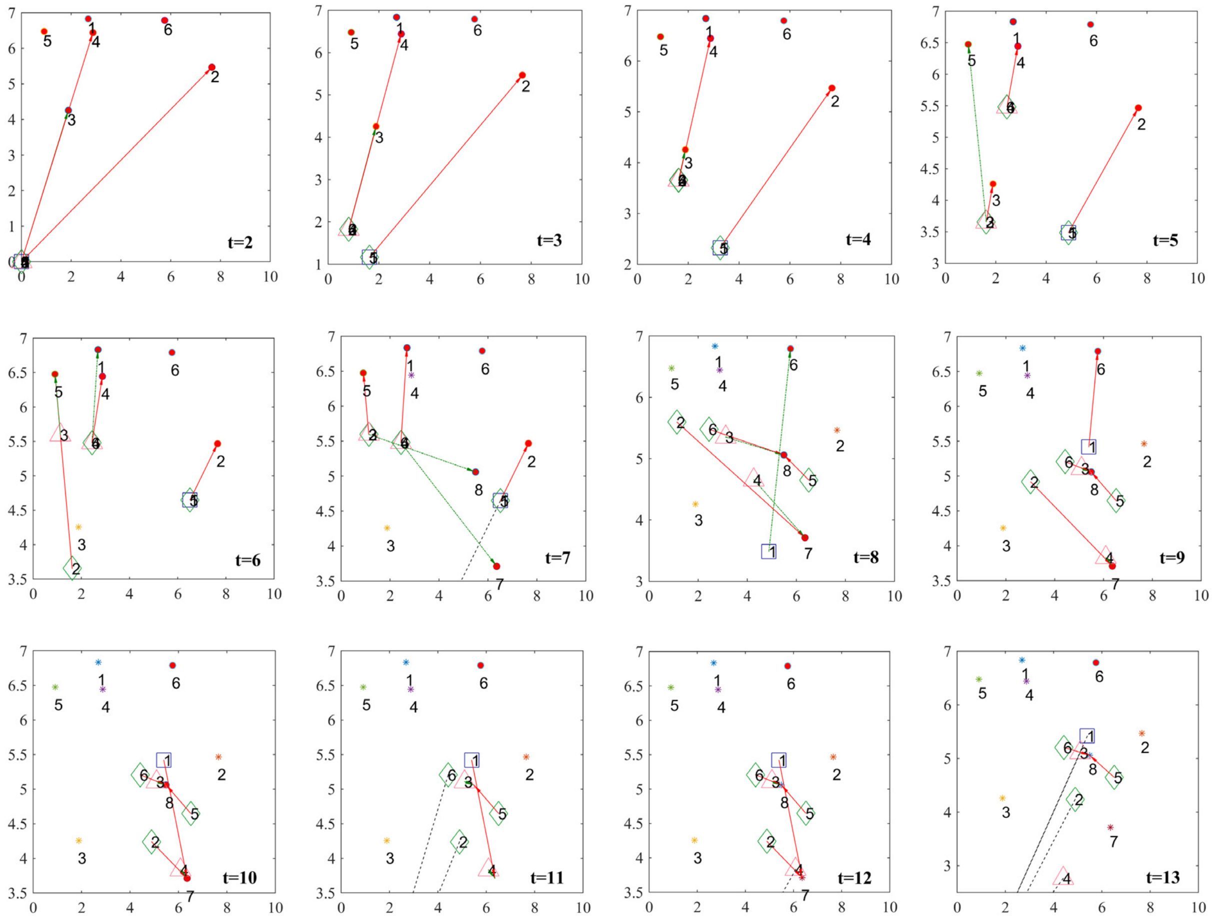

Fig. 6. The combat process diagram of UAV swarm based on the method in Ref. [40] (Omit the process of UAVs at beginning and returning to base. The symbols are the same as ones in Fig. 3).

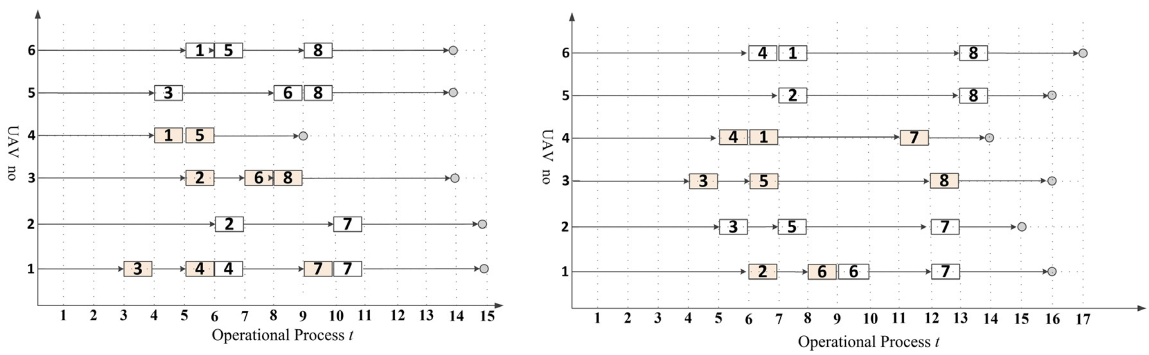

Fig. 7. The combat sequence diagram of UAV swarm (The left one is the result based on DACLD, and the right one is the result based on the method in Ref. [40]) ('X', 'Y' shows UAV is respectively performing reconnaissance or attack task aiming at target X or target Y, '◎' is the base).

consideration, which show that the proposed DACLD can not only get a better scheme of dynamic task allocation, which is characteristic of stronger robustness and flexibility, but also solve complicated problems of dynamic task allocation in battlefield scenarios. However, in future, we will develop and apply the proposed DACLD to solve the dynamic task allocation problems aiming at dynamic air targets. In conclusion, DACLD is an effective dynamic task allocation method characteristic of self-organization, which is also extendable and can be applied to other kinds task allocation problems of intelligent agents.

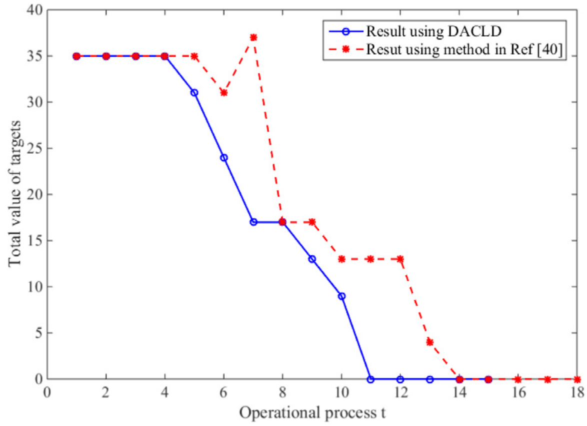

Fig. 8. The comparison curve of the total values of targets changing with the operational process *t*.

Acknowledgments

This work is supported by National Natural Science Foundation of China (Grant No. 61502522 and No. 61502534), Chinese Postdoctoral Science Foundation (No. 2016M603052), Open Funding Program of China Joint Laboratory of Flight Vehicle Ocean-based Measurement and Control (Grant No. FOM2015OF015), and Theoretical Research Foundation of Armed Police Force Engineering University (No. WJY201511, No. WXK2016-22).

References

[1] G.D. Goh, S. Agarwala, G.L. Goh, V. Dikshit, S.L. Singa, W.Y. Yeong, Additive manufacturing in unmanned aerial vehicles (UAVs): Challenges and potential, Aerosp. Sci. Technol 63 (6) (2017) 140–151.
[2] J. Ma, M. Li, Y. Zhang, H. Zhou, Firefly algorithm solving equal-task multiple traveling salesman problem, J. Comput. Theor. Nanos 12 (7) (2015) 304–307.
[3] S. Hanoun, A. Bhatti, D. Creighton, S. Nahavandi, Task assignment in camera networks: A reactive approach for manufacturing environments, IEEE Syst. J PP (99) (2016) 1–11.
[4] M. Radmanesh, M. Kumar, Flight formation of UAVs in presence of moving obstacles using fast-dynamic mixed integer linear programming, Aerosp. Sci. Technol 50 (3) (2016) 149–160.
[5] A.R.B. Behrouzian, D. Goswami, T. Basten, M. Geilen, Multi-constraint multi-processor resource allocation, in: International Conference on Embedded Computer Systems: Architectures, Modeling, and Simulation Samos, Greece, 2015, pp. 338–346.
[6] A. Sathyan, N.D. Ernest, K. Cohen, An efficient genetic Fuzzy approach to UAV swarm routing, Unmanned Systems 04 (02) (2016) 117–127.
[7] L. Geng, Y.F. Zhang, J.J. Wang, et al., Cooperative mission planning with multiple UAVs in realistic environments, Unmanned Systems 2 (01) (2014) 73–86.
[8] L.C. Shen, J. Chen, N. Wang, Overview of air vehicle mission planning techniques, Acta Aeronautica Et Astronautica Sinica 35 (3) (2014) 593–606.
[9] J.J. Wang, Y.F. Zhang, L. Geng, J.Y.H. Fuh, S.H. Teo, A heuristic mission planning algorithm for heterogeneous tasks with heterogeneous UAVs, Unmanned Systems 3 (3) (2015) 1–15.
[10] N. Ozalp, U. Ayan, E. Oztop, Cooperative multi-task assignment for heterogonous UAVs, J. Bone. Joint. Surg 64 (7) (2015) 1068–1073.
[11] C. Liu, A. Kroll, Performance impact of mutation operators of a subpopulation-based genetic algorithm for multi-robot task allocation problems, Springer Plus 5 (1) (2016) 1361.
[12] G. Oh, Y. Kim, J. Ahn, HL. Choi, PSO-based optimal task allocation for cooperative timing missions, IFAC-Papers on Line 49 (17) (2016) 314–319.
[13] H.R. Boveiri, An incremental ant colony optimization based approach to task assignment to processors for multiprocessor scheduling, Front. Inform. Tech. El 18 (4) (2017) 498–510.
[14] H.S. Wu, F.M. Zhang, Wolf pack algorithm for unconstrained global optimization, Math. Probl. Eng. 2014 (1) (2014) 1–17.
[15] R. Rautray, R.C. Balabantaray, Cat swarm optimization based evolutionary framework for multi document summarization, Physica A 477 (7) (2017) 174–186.
[16] P. Novoa-Hernández, C. Cruz, D.A. Pelta, Self-adaptation in dynamic environments-a survey and open issues, Int. J. Bio-Inspir. Com. 8 (1) (2015) 1–13.
[17] C.L. Shen, J. Chen, L. Wang, Overview of air vehicle mission planning techniques, Acta Aeronautica Et Astronautica Sinica 35 (3) (2014) 593–606.
[18] M.H. Kim, H. Baik, S. Lee, Response threshold model based UAV search planning and task allocation, J. Intell. Robot. Syst 75 (3) (2014) 625–640.
[19] X. Hu, H. Ma, Q. Ye, Hierarchical method of task assignment for multiple cooperating UAV teams, J. Syst. Eng. Electron. 26 (5) (2015) 1000–1009.
[20] O. Deepa, A. Senthilkumar, Swarm intelligence from natural to artificial systems: Ant colony optimization, GRAPH-HOC 8 (1) (2016) 9–17.
[21] H. Duan, Q. Luo, New progresses in swarm intelligence-based computation, Int. J. Bio-Inspir. Com 7 (1) (2015) 26–35.
[22] A. Arcuri, N. Lanchier, Stochastic spatial model for the division of labor in social insects, Math. Mod. Meth. App. S 27 (01) (2017) 45–73.
[23] E. Bonabeau, G. Theraulaz, J.L. Deneubourg, Quantitative study of the fixed threshold model for the regulation of division of labour in insect societies, Proceedings of the Royal Society B Biological Sciences 263 (22) (1996) 1565–1569.
[24] W. Lee, D.E. Kim, Task partitioning based on response threshold model in robot harvesting task, in: Artificial life 14: International Conference on the Synthesis and Simulation of Living Systems, 2014, pp.759–760.
[25] R.B. Xiao, W.M. Chen, T.G. Chen, Modeling of ant colony's labor division for the multi-project scheduling problem and its solution by PSO, J. Comput. Theor. Nanos 9 (2) (2012) 223–232.
[26] Y. Yang, X.S. Qiu, L.M. Meng, Task coalition formation and self-adjustment in the wireless sensor networks, Int. J. Commun. Syst 27 (10) (2014) 2241–2254.

[27] S.F. Dos, A.L.C. Bazzan, Towards efficient multi-agent task allocation in the Robocup rescue: A biologically-inspired approach, Auton. Agent. Mulit-AG 22 (3) (2011) 465–486.
[28] R.B. Xiao, T.Y. Yu, X.G. Gong, Modeling and simulation of ant colony's labor division with constraints for task allocation of resilient supply chains, Int. J. Artif. Intell. T 21 (03) (2012) 1240014–1–1240014–19.
[29] T. Yasuda, K. Kage, K. Ohkura, Response threshold-based task allocation in a reinforcement learning robotic swarm, in: IEEE, International Workshop on Computational Intelligence and Applications. Hiroshima, Japan, 2014, pp.189–194.
[30] E. Castello, T. Yamamoto, Y. Nakamura, Foraging optimization in swarm robotic systems based on and adaptive response threshold model, Adv. Robotics 28 (20) (2014) 1343–1356.
[31] C. Wei, K.V. Hindriks, C.M. Jonker, Dynamic task allocation for multi-robot search and retrieval tasks, Appl. Intell 45 (2) (2016) 1–19.
[32] T. Zhang, J. Li, Online task scheduling for LiDAR data preprocessing on hybrid GPU/CPU devices: A reinforcement learning approach, IEEE J-STARS 8 (1) (2015) 386–397.
[33] Y. Jiang, A survey of task allocation and load balancing in distributed systems, IEEE T. Parall. Distr 27 (2) (2016) 585–599.
[34] P. Sobkowicz, Agent based model of effects of task allocation strategies in flat organizations, Physica A 458 (9) (2016) 17–30.
[35] A. Arcuri, N. Lanchier, Stochastic spatial model for the division of labor in social insects, Math. Mod Meth. Appl. S 27 (01) (2016) 45–73.
[36] M.H. Kim, H. Baik, S. Lee, Resource welfare based task allocation for UAV team with resource constraints, J. Intell. Robot. Syst 77 (4) (2015) 611–627.
[37] X.X. Hu, H.W. Ma, Q.S. Ye, Hierarchical method of task assignment for multiple cooperating UAV teams, J. Syst. Eng Electron 26 (5) (2015) 1000–1009.
[38] S.S. Ponda, L.B. Johnson, A. Geramifard, J.P. How, Cooperative mission planning for multi-UAV teams [M], Handbook of Unmanned Aerial Vehicles (2015) 1447–1490.
[39] R.B. Xiao, Y.C. Wang, Z.W. Wang, Research on structure emergence based on cellular automata, Int. J. Bio-Inspir. Com 6 (2) (2014) 126–139.
[40] H. Li, R.B. Xiao, H.S. Wu, Modelling for combat task allocation problem of aerial swarm and its solution using wolf pack algorithm, Int. J. Innov. Comput. I 7 (1) (2016) 50–59.