

Formation and adjustment of manned/unmanned combat aerial vehicle cooperative engagement system

ZHONG Yun^{1,*}, YAO Peiyang¹, ZHANG Jieyong¹, and WAN Lujun²

1. Information and Navigation College, Air Force Engineering University, Xi'an 710077, China;

2. Air Traffic Control and Navigation College, Air Force Engineering University, Xi'an 710051, China

Abstract: Manned combat aerial vehicles (MCAVs), and unmanned combat aerial vehicles (UCAVs) together form a cooperative engagement system to carry out operational mission, which will be a new air engagement style in the near future. On the basis of analyzing the structure of the MCAV/UCAV cooperative engagement system, this paper divides the unique system into three hierarchical levels, respectively, i.e., mission level, task-cluster level and task level. To solve the formation and adjustment problem of the latter two levels, three corresponding mathematical models are established. To solve these models, three algorithms called quantum artificial bee colony (QABC) algorithm, greedy strategy (GS) and two-stage greedy strategy (TSGS) are proposed. Finally, a series of simulation experiments are designed to verify the effectiveness and superiority of the proposed algorithms.

Keywords: manned combat aerial vehicle (MCAV), unmanned combat aerial vehicle (UCAV), cooperative engagement system, quantum artificial bee colony (QABC), greedy strategy (GS), two-stage greedy strategy (TSGS).

DOI: 10.21629/JSEE.2018.04.10

1. Introduction

Unmanned combat aerial vehicles (UCAVs), which possess low operational environmental requirements, strong battlefield viabilities and diversified tactics patterns, have presented the capability to perform multiple tasks, e.g., target search, intelligence detection, communication relay and synchronous missile strikes [1]. At present, UCAVs have become important air combat assets to Air Forces, and until 2011, there have been various UCAVs for military purpose [2].

The widespread use of UCAVs has given rise to a series of new air combat patterns, which include UCAV swarm engagement [3,4] and manned/unmanned combat aerial

vehicle (MCAV/UCAV) cooperative engagement [5,6]. Limited by the autonomous level of UCAVs, it is difficult for UCAV swarm engagement to be realized in a short term, while MCAV/UCAV cooperative engagement is an attainable and realistic goal.

MCAVs, which include airborne early warning aircraft (AEWA), transport aircraft, bombers, fighters and strike aircraft, take full advantage of superior operational performance, strong awareness capacity and short decision-making cycle, and together with UCAVs, they form an engagement system. In this human-in-the-loop system, MCAV commanders can give full play to their operational experience and wisdom, thus agilely respond to emergencies, e.g., pop-up tasks, pop-up threats and platform damages (platform mainly refers to UCAV), and consequently, realize maximum battlefield effectiveness [7,8].

The purpose of this research is to establish the mathematical models that are appropriate to generate and adjust task plans, and thereby make MCAV/UCAV cooperative engagement system adaptively cope with the change of battle-field situation. The proposed models in this paper are inspired by the recent research results in the related research field, which have broader research prospects.

This paper is organized as follows. In the next section, a detailed problem description and a review of related literature are provided. In Section 3, key attributes applied to establish the mathematical models are listed. Section 4 proposes a mathematical formulation for the static coalition formation (SCF) problem and then a novel algorithm called quantum artificial bee colony (QABC) is applied to solve the model. Then, for the dynamic coalition adjustment (DCA) problem, Section 5 establishes two distinct models to describe intra task-cluster coalition and inter task-cluster coalition adjustment, and then two novel algorithms respectively called greedy strategy (GS) and two-stage greedy strategy (TSGS) are proposed to solve these models. In Section 6, several simulation cases are

Manuscript received June 26, 2017.

*Corresponding author.

This work was supported by the National Natural Science Foundation of China (61573017), and the Doctoral Innovation Found of Air Force Engineering University (KGD08101604).

provided to verify the effectiveness and superiority of the proposed algorithms. Finally, in Section 7, a brief summary and prospect of future research are contained.

2. Problem description and related literature

2.1 Problem description

Assuming that there exists a mission coalition performing a combat mission [9], the mission coalition is made up of a limited number of MCAVs and a fleet of heterogeneous UCAVs. Usually, the mission includes multiple tasks performed in a directed or sequential sequence, and all tasks can be optimally aggregated to several task-clusters according to the sequence and geographical distribution of all tasks. Each task-cluster is performed by the corresponding MCAV/UCAV task-cluster coalition, and under the supervision control of MCAVs in different task-cluster coalitions, UCAVs form several task coalitions to perform the corresponding tasks.

It can be drawn a conclusion that the MCAV/UCAV cooperative engagement system is a typical hierarchical system, which contains three distinct levels, i.e., mission coalition, task-cluster coalition and task coalition. Fig. 1 is the logical relationship of these three types of coalition.

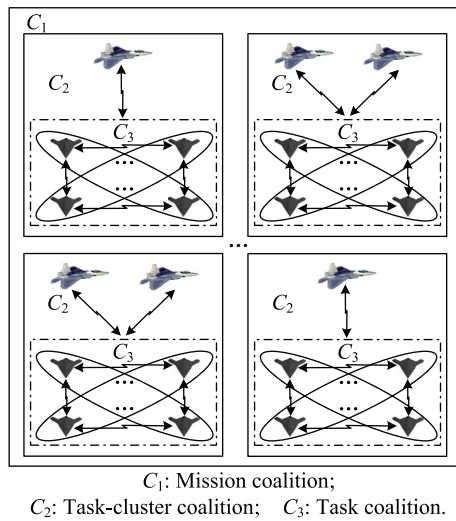


Fig. 1 Logical relationship of coalitions

Fig. 1 reveals that mission coalition is a virtual concept, in practical, once task-cluster coalition and task coalition, which correspond to task-cluster level and task level respectively, have been formed from top to bottom, then, mission coalition will be formed. A task-cluster coalition formation plan will determine which MCAVs command which UCAVs to perform which task-clusters, a task coalition will determine which UCAVs to perform which tasks in the corresponding task-cluster cooperatively. On the basis of above analysis, there are two key research areas in

the cooperative engagement system, i.e., (i) formation and adjustment of task-cluster coalition, (ii) formation and adjustment of task coalition.

The task-cluster coalition formation scheme is not the focus in this paper which assumes that task-cluster coalitions have already been formed. And the research focus is how to form task coalitions to perform geographically dispersed initial tasks in different task-cluster coalitions, and adjust task coalitions to perform pop-up tasks or tasks that have originally been allocated to the damaged platforms. In particular, during performing two types of new tasks mentioned above, adjustment is not limited to the task layer, but is also involved in the task-cluster layer, i.e., if the spare resources in a certain task-cluster coalition are insufficient to perform new tasks, it needs to adjust task-cluster coalition.

Furthermore, tasks in this research mainly refer to strike tasks, and different UCAVs may have different capabilities, which depend on payloads of UCAVs (e.g., missiles guided by different kinds of guidance system). In a task-cluster coalition, the number of task coalitions is equal to that of tasks, due to the limitation of UCAVs capabilities, each task is probably performed by one or more UCAVs, and each UCAV is likely to join different task coalitions to perform different tasks.

When emergencies, e.g., pop-up task and platform damage, have occurred in a particular task-cluster coalition, the engagement system will take different measures in accordance with different situations, i.e., (i) if spare resources of their own task-cluster coalition are enough to perform new tasks, then it will carry out internal adjustment, i.e., task coalition adjustment, (ii) if spare resources are not enough to perform new tasks, then external adjustment will be carried out, i.e., task-cluster adjustment. During the external adjustment, the task-cluster coalition, which needs to perform new tasks, has to ask UCAVs in the other task-cluster coalitions for help (this paper calls it “borrow”), thus satisfying the resource demand of emergent tasks.

2.2 Related literature

So far, previous researches on MCAV/UCAV coalition and UCAV coalition mainly focus on SCF, but are seldom involved in DCA. However, dynamic adjustment of action plan has been studied in some other areas, which mainly include the adjustment of C2 organization [10] and train operation [11,12].

In terms of coalition formation, aiming at applying heterogeneous UCAVs to perform search and attack tasks, a mathematical model is established to minimize the formation time and coalition size, and a polynomial time coalition formation algorithm (PTCFA) is proposed to solve the model [13]. To deepen the research findings of [13],

Sujit et al. [14] took target movement, communication range and delay of UCAVs into consideration, and designed a communication protocol over a dynamic UCAV network as well as a strategy for tracking and attacking moving targets.

Recently, a resource welfare based allocation method for search and attack tasks in unknown regions has been proposed [15]. And via adopting the balancing resource distribution strategy, a fleet of heterogeneous UCAVs with different types and quantities of resources could retain more assets available compared to the instantaneous task allocation method, thus UCAVs could respond to emergencies more effectively.

It should also be noticed that the concept of coalition has also been widely applied in the research field of networked air defense. For example, Young et al. [16] proposed the interception coalition (IC) concept, in which the whole air defense system could form an adaptive combat combination according to the battlefield environment, target properties and deployment situation.

In addition, Meng et al. [17] considered the cooperative coalition for the formation flight scheduling problem, which was only based on incomplete information and was optimized in a decentralized manner.

For coalition adjustment, to the best of our knowledge, research results are relatively rare, however, there are still some researches that could be referenced. For instance, two instantaneous task allocation methods were proposed in [18] and [19], the simulation results in these two papers similarly revealed that if UCAVs were instantaneously assigned to targets simply, it was a greedy fashion and could only obtain the local optimal solution. Jin et al. [20] presented a predictive assignment algorithm for UCAV coalition, and the algorithm could incorporate predictions for future tasks into the decision-making process.

In addition to this, some algorithms based on market mechanism were put forward to perform new tasks brought by emergencies. Choi et al. [21] proposed a consensus-based auction algorithm (CBAA) and its variation, consensus-based bundle algorithm (CBBA), to coordinate a fleet of autonomous vehicles. The similarity lay in that both of them adopted the synchronous information interaction method to generate conflict-free feasible solutions, while the main difference was that CBAA could only assign a task, but CBBA could assign multiple tasks. Furthermore, an asynchronous consensus-based bundle algorithm (ACBBA) [22] was proposed to solve the problem of information transmission chaos in [21].

To sum up, it is urgently needed to carry out research on formation and adjustment, especially adjustment, of MCAV/UCAV cooperative engagement system. The ob-

ject of this research is establishing unified scheduling and rescheduling models for MCAV/UCAV cooperative engagement system, and learning from existing research results, so as to improve the agility of the cooperative engagement system.

3. Attribute abstract

In this section, key attributes in formation and adjustment of cooperative engagement system are described.

Take a typical MCAV/UCAV task-cluster coalition for example, multiple task coalitions will be formed to perform initial tasks and a few adjustments will be made to perform emergent tasks. Firstly, several attributes will be defined, these attributes mainly consist of task-cluster coalition set TCC , besides, in the task-cluster coalition with new tasks, attributes include MCAV set M , UCAV set U , borrowed UCAV set U^b , initial task set IT , emergent task set ET and task coalition set TC . The detailed instructions are listed as follows.

(i) Task-cluster coalition set is expressed as $TCC = \{tcc_1, tcc_2, \dots, tcc_{N_{tcc}}\}$, and N_{tcc} is the number of task-cluster coalitions.

(ii) MCAV set is $M = \{m_1, m_2, \dots, m_{N_m}\}$, where N_m is the number of MCAVs.

(iii) UCAV set is $U = \{u_1, u_2, \dots, u_{N_u}\}$, and N_u is the number of UCAVs. The resources loaded by u_j ($j = 1, 2, \dots, N_u$) is expressed as $\mathbf{R}_j^u = (r_{j1}^u, r_{j2}^u, \dots, r_{jN_r}^u)$, and N_r is the number of resource types. Besides, the current position of u_j is $L_j^u = (lx_j^u, ly_j^u)$, the average velocity of u_j is v_j .

(iv) Borrowed UCAV set is expressed as $U^b = \{u_1^b, u_2^b, \dots, u_{N_u^b}^b\}$, and N_u^b is the number of borrowed UCAVs. The resources loaded by $u_{j'}^b$ ($j' = 1, 2, \dots, N_u^b$) can be expressed as $\mathbf{R}_{j'}^b = (r_{j'1}^b, r_{j'2}^b, \dots, r_{j'N_r}^b)$. Besides, the current position of $u_{j'}^b$ is $L_{j'}^b = (lx_{j'}^b, ly_{j'}^b)$, and the average velocity of $u_{j'}^b$ is $v_{j'}^b$.

(v) Initial task set is $IT = \{it_1, it_2, \dots, it_{N_{it}}\}$, where N_{it} is the number of initial tasks. The resource demand of it_k ($k = 1, 2, \dots, N_{it}$) is $\mathbf{R}_k^{it} = (r_{k1}^{it}, r_{k2}^{it}, \dots, r_{kN_r}^{it})$, and the location of it_k is $L_k^{it} = (lx_k^{it}, ly_k^{it})$. In addition, the processing time of it_k is dc_k^{it} , if the start time of it_k is sc_k^{it} , then the completion time of it_k satisfies $fc_k^{it} = sc_k^{it} + dc_k^{it}$.

(vi) Emergent task set is expressed to be $ET = \{et_1, et_2, \dots, et_{N_{et}}\}$, where N_{et} is the number of emergent tasks. And the resource demand of et_l ($l = 1, 2, \dots, N_{et}$) is represented by $\mathbf{R}_l^{et} = (r_{l1}^{et}, r_{l2}^{et}, \dots, r_{lN_r}^{et})$, the location of et_l is $L_l^{et} = (lx_l^{et}, ly_l^{et})$. In addition, the processing time of et_l is dc_l^{et} , if the start time of et_l is sc_l^{et} , then the completion time of et_l satisfies $fc_l^{et} = sc_l^{et} + dc_l^{et}$.

In order to simplify the problem, it defines $T = IT \cup$

$ET = \{t_1, t_2, \dots, t_{N_t}\}$, and N_t is the number of all tasks, then $N_t = N_{it} + N_{et}$ is established. Obviously, the resource demand, location, processing time, start time and completion time of t_o ($o = 1, 2, \dots, N_t$) is equal to the corresponding value of it_k or et_l in IT or ET respectively.

(vii) Task coalition set is expressed to be $TC = \{tc_1, tc_2, \dots, tc_{N_{tc}}\}$, where N_{tc} represents the number of task coalitions, in general, a task is performed by a task coalition, which contains a certain number of UCAVs, thereby there is $N_{tc} = N_t$ established. Assuming that the UCAV coalition set belonging to tc_q is expressed as $U_q = \{u_q^1, u_q^2, \dots, u_q^{N_q^{tc}}\}$, and N_q^{tc} is the number of UCAVs in U_q , so, there must have $U_q \subseteq U$. The combined resource of tc_q is $R_q^{tc} = (r_{q1}^{tc}, r_{q2}^{tc}, \dots, r_{qN_r}^{tc})$.

4. Static coalition formation

4.1 Definition of decision variables

Denote y_k^j to be an assignment variable that indicates whether it_k is performed by u_j , $y_k^j = 1$ means yes and $y_k^j = 0$ means no, so, all such y_k^j compose task coalition formation matrix $Y = (y_k^j)_{N_u \times N_{it}}$. Let $x_{k'k}^j$ represent the transition variable of u_j , if $x_{k'k}^j = 1$, then it means u_j will perform t_k after performing $t_{k'}$, while $x_{k'k}^j = 0$ means not, so, $x_{k'k'}^j = x_{kk}^j = 0$ holds.

In particular, all UCAVs should begin performing initial tasks from a fixed position, it assumes that there exists a virtual initial task t_0 locating in the fixed rendezvous location, thus, $IT = IT \cup \{t_0\}$ holds.

4.2 Objective function

From the perspective of assigning UCAVs to initial tasks, SCF mainly seeks to minimize the completion time of initial task set on the premise that the combined resource vector of formed task coalitions will satisfy the resource demand of all initial tasks. Thus, the objective function of the SCF problem is defined as

$$\min FT = \min \max(f_{c1}^{it}, f_{c2}^{it}, \dots, f_{cN_{it}}^{it}). \quad (1)$$

4.3 Programming constraints

Next, programming constraints will be listed as follows.

(i) For u_j and it_k , if $y_k^j = 1$, it contains two cases, one of which is that u_j is assigned to perform it_k after completing $it_{k'}$ ($k' \neq 0$), and $x_{k'k}^j = 1$ holds. The other is that it is the first time for u_j to perform a task and there exist no predecessor initial tasks, which means $x_{0k}^j = 1$. Thereby, the constraint indicates the relationship between $x_{k'k}^j$ and y_k^j is given by

$$\sum_{k'=0}^{N_{it}} x_{k'k}^j - y_k^j = 0,$$

$$k = 1, 2, \dots, N_{it}; j = 1, 2, \dots, N_u. \quad (2)$$

(ii) When it needs to form multiple task coalitions to perform multiple tasks, the same UCAV u_j can only belong to only one task coalition, i.e., after completing $it_{k'}$, u_j can only perform one task it_k , and the constraint is given by

$$\sum_{k=1}^{N_{it}} x_{k'k}^j \leq 1, \quad k' = 0, 1, \dots, N_{it}; j = 1, 2, \dots, N_u. \quad (3)$$

(iii) For it_k , the combined resources of all UCAVs assigned to it have to meet the demand of it_k . This is shown as

$$\sum_{j=1}^{N_u} y_k^j r_{jp}^u \geq r_{kp}^{it}, \quad p = 1, 2, \dots, N_r. \quad (4)$$

(iv) If it_k is performed by tc_q , then the start time of it_k depends on the maximum value of estimated time of arrivals (ETAs) of all UCAVs in tc_q , i.e.,

$$sc_k^{it} = \max(ETA_{u_q^1}, ETA_{u_q^2}, \dots, ETA_{u_q^{N_q^{tc}}}). \quad (5)$$

(v) If $x_{k'k}^j = 1$ holds for u_j , because it is needed for all UCAVs performing it_k to begin task execution until they all have arrived the task area, i.e., UCAVs arriving at the task area in advance should wait for the arrival of other UCAVs. Thus, the start time of it_k , i.e., sc_k^{it} , should satisfy

$$sc_k^{it} \geq sc_{k'}^{it} + dc_{k'}^{it} + zc_{k'k}^j, \quad k' = 0, 1, \dots, N_{it};$$

$$k = 1, 2, \dots, N_{it}; k' \neq k; j = 1, 2, \dots, N_u \quad (6)$$

where $zc_{k'k}^j$ is the time needed for u_j to fly from $it_{k'}$ to it_k . If $x_{k'k}^j = 0$, then let FT_{upp} be the upper bound of task completion time (a very large constant), so,

$$sc_k^{it} \geq sc_{k'}^{it} + dc_{k'}^{it} - FT_{upp}, \quad k' = 0, 1, \dots, N_{it};$$

$$k = 1, 2, \dots, N_{it}; k' \neq k \quad (7)$$

holds. According to (6) and (7),

$$sc_k^{it} \geq sc_{k'}^{it} + dc_{k'}^{it} + x_{k'k}^j \cdot zc_{k'k}^j +$$

$$FT_{upp} \cdot (x_{k'k}^j - 1), \quad k' = 0, 1, \dots, N_{it};$$

$$k = 1, 2, \dots, N_{it}; k' \neq k; j = 1, 2, \dots, N_u \quad (8)$$

holds.

(vi) For $zc_{k'k}^j$ in (8), in consideration of that tasks have a rather wide distribution, the turning radius is comparatively small compared with the distance between tasks, hence, it mainly adopts the distance between $it_{k'}$ and it_k to represent the flight distance for u_j from $it_{k'}$ to it_k , so,

$$zc_{k'k}^j = \frac{\sqrt{(lx_k^{it} - lx_{k'}^{it})^2 + (ly_k^{it} - ly_{k'}^{it})^2}}{v_j},$$

$$k' = 0, 1, \dots, N_{it}; k = 1, 2, \dots, N_{it}; k' \neq k \quad (9)$$

holds.

Let C_0 be all constraints mentioned in Section 4.3, it establishes the model for SCF as

$$\begin{aligned} & \min FT \\ & \left\{ \begin{array}{l} \sum_{k'=0}^{N_{it}} x_{k'k}^j - y_k^j = 0 \\ \sum_{k=1}^{N_{it}} x_{k'k}^j \leq 1 \\ \sum_{j=1}^{N_u} y_{jp}^j \geq r_{kp}^{it} \\ sc_k^{it} = \max(ETA_{u_q^1}, ETA_{u_q^2}, \dots, ETA_{u_q^{N_q^{tc}}}) \\ sc_k^{it} \geq sc_{k'}^{it} + dc_{k'}^{it} + x_{k'k}^j \cdot zc_{k'k}^j + FT_{upp} \cdot (x_{k'k}^j - 1) \\ zc_{k'k}^j = \frac{\sqrt{(lx_k^{it} - lx_{k'}^{it})^2 + (ly_k^{it} - ly_{k'}^{it})^2}}{v_j} \\ FT = \max(fc_1^{it}, fc_2^{it}, \dots, fc_{N_{it}}^{it}) \\ 0 < FT \leq FT_{upp} \\ x_{k'k}^j, y_k^j \in \{0, 1\} \\ j = 1, 2, \dots, N_u; k' = 0, 1, \dots, N_{it}; \\ k = 1, 2, \dots, N_{it}; k' \neq k; \\ p = 1, 2, \dots, N_r; q = 1, 2, \dots, N_{tc} \end{array} \right. \quad (10) \end{aligned}$$

4.4 Model solving

4.4.1 Task priority

To avoid deadlock of task allocation, a widely utilized weighted length algorithm (WLA) is applied to determine task priority, the priority of it_k is defined as

$$P(it_k) = dc_k^{it} + \max_{it_{k'} \in OUT(it_k)} P(it_{k'}) + \frac{\sum_{it_{k'} \in OUT(it_k)} P(it_{k'})}{\max_{it_{k'} \in OUT(it_k)} P(it_{k'})} \quad (11)$$

where $OUT(it_k)$ is the subsequent task set of it_k .

4.4.2 Resource distribution strategy

For it_k , if the corresponding task coalition is tc_q , it is a critical question that how to determine the resource distribution number of each UCAV according to UCAV resource load and task resource demand [15]. Take a typical case for example, if the resource demand vector of it_k is (3, 3), resource load vectors of u_q^1 , u_q^2 and u_q^3 in tc_q are (4, 0), (0, 5) and (6, 8) respectively. If the instantaneous distribution strategy is adopted, then resource load vectors of u_q^1 , u_q^2

and u_q^3 in tc_q will be (1, 0), (0, 2) and (6, 8) after performing it_k . And when the next coalition tc_{q+1} is formed to perform it_{k+1} , only u_q^1 and u_q^2 are in idle, while the resource demand vector of it_{k+1} is (2, 2), under this circumstance, u_q^1 and u_q^2 have no capacity to perform it_{k+1} and have to wait for other UCAVs. Thus, after the coalition formation, a balancing based resource distribution strategy is adopted and the detailed steps are as follows.

Step 1 For it_k , if the combined resource of tc_q is $R_q^{tc} = (r_{q1}^{tc}, r_{q2}^{tc}, \dots, r_{qN_r}^{tc})$, then, calculate the average remaining resource of UCAVs where the p th resource is nonzero, i.e.,

$$r_{qp}^{tc_remain} = \frac{r_{qp}^{tc} - r_{kp}^{it}}{N_{qp}^{tc'}}, \quad p = 1, 2, \dots, N_r \quad (12)$$

where $N_{qp}^{tc'}$ is the number of UCAVs where the p th resource is nonzero.

Step 2 If the p th resource of u_q^j is nonzero, judge if $r_{jp}^u > r_{qp}^{tc_remain}$ holds, if it holds, put u_q^j into the resource distribution UCAV set for the p th resource. Denote the number of UCAVs of the resource distribution UCAV set for the p th resource as N_{qp}^{tc} , so, $N_{qp}^{tc} \leq N_{qp}^{tc'} \leq N_q^{tc}$ holds.

Step 3 To ensure the balancing distribution of resource, it assumes the remaining resource of UCAVs, which need to distribute resource, is $\lfloor (r_{qp}^{tc'} - r_{kp}^{it}) / N_{qp}^{tc} \rfloor$ or $\lfloor (r_{qp}^{tc'} - r_{kp}^{it}) / N_{qp}^{tc} \rfloor + 1$, where $r_{qp}^{tc'}$ is the combined p th resource of resource distribution UCAV set, then

$$\begin{aligned} & K_1 \cdot \lfloor (r_{qp}^{tc'} - r_{kp}^{it}) / N_{qp}^{tc} \rfloor + \\ & K_2 \cdot (\lfloor (r_{qp}^{tc'} - r_{kp}^{it}) / N_{qp}^{tc} \rfloor + 1) = r_{qp}^{tc'} - r_{kp}^{it} \quad (13) \end{aligned}$$

holds. After calculation, it can be obtained that $K_1 = N_{qp}^{tc} \cdot (\lfloor (r_{qp}^{tc'} - r_{kp}^{it}) / N_{qp}^{tc} \rfloor + 1) + r_{kp}^{it} - r_{qp}^{tc'}$, $K_2 = r_{qp}^{tc'} - r_{kp}^{it} - N_{qp}^{tc} \cdot \lfloor (r_{qp}^{tc'} - r_{kp}^{it}) / N_{qp}^{tc} \rfloor$.

Step 4 Carry through the corresponding distribution operation to all N_r resources on the basis of Steps 1–3.

4.4.3 QABC algorithm

The artificial bee colony (ABC) algorithm has been extensively applied to solve the combinatorial optimization problem through simulating the foraging behavior of bees [23]. The traditional ABC algorithm usually divides bee colony into employed bees, onlooker bees and scout bees according to the division of labor. Each employed bee corresponds to a certain food source, which is actually a potential solution to the problem. After searching, employed bees will share information with onlooker bees, while onlooker bees select a food source with a certain probability to conduct neighborhood search. Within a given iteration of search, if employed bees or onlooker bees cannot find

better food source, then the food source will be abandoned and all bees will be transformed into scout bees to search new food source in a random way.

On count of discrete characteristics of the model, a QABC algorithm is proposed to solve the model.

4.4.3.1 Representation of quantum bits

In QABC, a food source is represented by a matrix, in which rows represent UCAVs, while columns represent tasks. In the g th iteration, the quantum bit represents whether the k th initial task is performed by the j th UCAV can be defined as $\mathbf{v}_{jk}^g = [\alpha_{jk}^g \ \beta_{jk}^g]^T$, where $|\alpha_{jk}^g|^2$ and $|\beta_{jk}^g|^2$ are the probabilities of the quantum bit collapsing to $|0\rangle$ and $|1\rangle$ state. Thus, $\alpha_{jk}^g \in [0, 1]$, $\beta_{jk}^g \in [0, 1]$ and $|\alpha_{jk}^g|^2 + |\beta_{jk}^g|^2 = 1$ hold. Obviously, the task allocation result of the j th UCAV can be encoded as

$$\mathbf{v}_j^g = [\mathbf{v}_{j1}^g \ \mathbf{v}_{j2}^g \ \cdots \ \mathbf{v}_{jN_{it}}^g] = \begin{bmatrix} \alpha_{j1}^g & \alpha_{j2}^g & \cdots & \alpha_{jN_{it}}^g \\ \beta_{j1}^g & \beta_{j2}^g & \cdots & \beta_{jN_{it}}^g \end{bmatrix}. \quad (14)$$

Therefore, the task allocation results of all UCAVs can be represented by

$$\mathbf{V}^g = [\mathbf{v}_1^g \ \mathbf{v}_2^g \ \cdots \ \mathbf{v}_{N_u}^g]^T. \quad (15)$$

Taking employed bees for example, the quantum bee colony $\mathbf{V}(g)$ can be expressed as

$$\mathbf{V}(g) = \left(\underbrace{\mathbf{v}^g, \mathbf{v}^g, \dots, \mathbf{v}^g}_{N_{eb}} \right) \quad (16)$$

where N_{eb} is the number of employed bees.

4.4.3.2 Quantum rotating gate

The update of quantum bit in QABC is mainly through adopting quantum rotating gate, in the g th iteration, the quantum rotating angle for the k th quantum bit of the j th bee is θ_{jk}^g , so, the quantum rotating gate can be denoted as

$$\mathbf{H}(\theta_{jk}^g) = \begin{bmatrix} \cos \theta_{jk}^g & -\sin \theta_{jk}^g \\ \sin \theta_{jk}^g & \cos \theta_{jk}^g \end{bmatrix}. \quad (17)$$

Then, the new quantum bit is calculated as

$$\mathbf{v}_{jk}^{g+1} = \mathbf{H}(\theta_{jk}^{g+1}) \mathbf{v}_{jk}^g \quad (18)$$

where the determination of θ_{jk}^{g+1} is mainly through the look-up table. Noticing that SCF is actually a minimum problem, so, it takes the reciprocal of the objective function value as the fitness value. Table 1 shows the rotation angle selection strategy.

Table 1 Rotation angle selection strategy

y_k^j	b_k^j	$f(y) > f(b)$	$\Delta\theta_{jk}^{g+1}$	$s(\alpha_k, \beta_k)$			
				$\alpha_k\beta_k > 0$	$\alpha_k\beta_k < 0$	$\alpha_k=0$	$\beta_k=0$
0	0	false	*	0	0	0	0
0	0	true	*	0	0	0	0
0	1	false	Δ	+1	-1	0	± 1
0	1	true	Δ	-1	+1	± 1	0
-1	0	false	Δ	-1	+1	± 1	0
1	0	true	Δ	+1	-1	0	± 1
1	1	false	*	0	0	0	0
1	1	true	*	0	0	0	0

It is observed that, $\theta_{jk}^{g+1} = s(\alpha_k, \beta_k) \Delta\theta_{jk}^{g+1}$, $\Delta = 0.01\pi$, $f(\cdot)$ is the fitness function, and $f = 1/FT$ holds, * represents that if $y_j^k = b_j^k$, quantum bit \mathbf{v}_{jk}^g is updated with probability P_m , and the calculation formula of \mathbf{v}_{jk}^g is

$$\mathbf{v}_{jk}^{g+1} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \mathbf{v}_{jk}^g. \quad (19)$$

QABC usually carries through quantization representation to the foraging behavior of bees in ABC, after the quantum measurement for $\mathbf{V}(g)$, certain solutions can be sought, i.e., food source position.

4.4.3.3 Constraint handling strategy

After mapping each food source position to solution, i.e., measuring each food source, matrix $\mathbf{W} = (w_{jk})_{N_u \times N_{it}}$ is obtained. And it is significant to note that \mathbf{W} is not the final solution because it may violate the resource constraints, so, it needs to deal with \mathbf{W} . The detailed steps are as follows.

Step 1 For each column of \mathbf{W} , let nonzero elements be 0 from top to bottom one by one, every time let one nonzero element be 0, judge that if

$$rs_{p_zb}^k = rs_{p_za}^k, \quad \forall p = 1, 2, \dots, N_r \quad (20)$$

holds, in which $rs_{p_zb}^k$ and $rs_{p_za}^k$ represent resource satisfaction degree of the p th resource of it_k before and after letting a particular element be 0 respectively, and the computational formula of rs_p^k is

$$rs_p^k = \min \left(\sum_{j=1}^{N_u} y_k^j r_{jp}^u / r_{kp}^{it}, 1 \right), \quad \forall p = 1, 2, \dots, N_r. \quad (21)$$

If (20) does not hold, then, let the element, which is set to be 0 previously, be 1 again.

Step 2 Judge whether resource satisfaction degrees of all resources are equal to 1, if they are, then, go to Step 4. Otherwise, for the p' th resource satisfies $rs_{p'}^k \neq 1$, let zero elements be 1 from top to bottom one by one, every time let just one zero element be 1, judge that if

$$rs_{p'_zb}^k = rs_{p'_za}^k, \quad \forall p' = 1, 2, \dots, N'_r \quad (22)$$

holds, in which $rs_{p'_zb}^k$ and $rs_{p'_za}^k$ represent resource satisfaction degrees of the p' th resource of it_k before and after letting a particular element be 1 respectively, and N'_r is

the number of resources whose satisfaction degree is not equal to 1 for it_k . If (22) is established, then let the element, which is set to be 1 previously, be 0 again. If

$$rs_p^k = 1, \quad \forall p = 1, 2, \dots, N_r \quad (23)$$

holds, then go to Step 3.

Step 3 Carry out the same operation in Step 1 again.

Step 4 Judge that whether all columns of W have been operated, if so, then terminate constraint handling strategy and output matrix Y , otherwise, return to Step 1.

4.4.3.4 Detailed solving process

To sum up, the detailed solving process of QABC is listed as follows.

Step 1 According to (11), prioritize initial tasks and generate task execution sequence.

Step 2 Initialize quantum bee colony to be $V(0)$, the value of each quantum bit is set as $[1/\sqrt{2}, 1/\sqrt{2}]^T$, i.e., each quantum bit is likely to collapse to $|0\rangle$ and $|1\rangle$ state with the same probability.

Step 3 Measure each food source position and adopt the constraint handling strategy to get the corresponding solution, and then, record the optimal food source position and fitness value.

Step 4 Determine whether to terminate the evolutionary computation on the basis of the termination condition, if it is, then terminate, otherwise, continue to calculate.

Step 5 Measure each food source position in colony $V(g)$ to obtain the corresponding solution and evaluate the fitness values.

Step 6 Update each food source position and obtain the new colony $V(g+1)$.

Step 7 Record the optimal food source position and its corresponding fitness value.

Step 8 Let $g = g + 1$, and return to Step 4.

5. Dynamic coalition adjustment

The catastrophe characteristics of battlefield environment, task and platform determine that the task plan of MCAV/UCAV cooperative engagement system cannot remain unchanged, but should always change in an adaptive way according to the change of environment, task and platform, thus realizing the self-organizing adjustment.

Hence, it respectively designs two distinct modules, i.e., internal coordination module and external collaboration module, to handle this kind of uncertainty.

5.1 Process flow

When emergent tasks occur, firstly, it needs to judge that if it is enough for the total spare resources to perform emergent tasks, then the internal coordination module will be

launched to reschedule a task plan for emergent tasks, otherwise, the external coordination module will be. Fig. 2 shows the process flow of coalition adjustment.

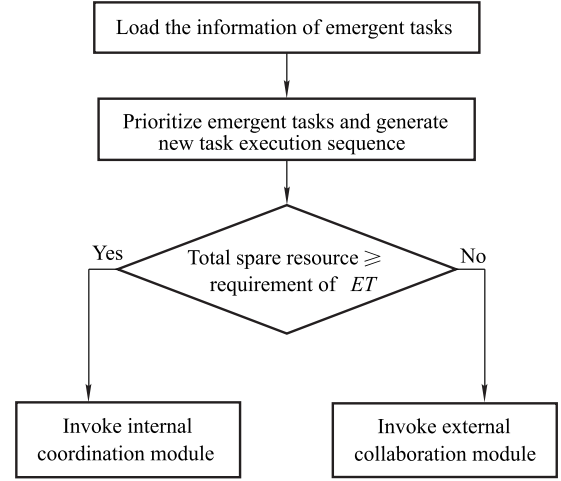


Fig. 2 Process flow of coalition adjustment

Actually, the internal coordination module and the external collaboration module are both model/algorithm modules designed to assign UCAVs to perform emergent tasks, the algorithms are respectively called GS and TSGS.

5.2 Internal coordination module

The first part is the internal coordination module, the function of which is to implement internal adjustment if the total spare resources of the corresponding task-cluster coalition can meet the demand of emergent tasks.

5.2.1 Internal coordination problem modeling

Assuming that at time epo , task set changes from $T = IT$ to $T = IT \cup ET$. Besides, let $Y' = (y'_{oj})_{N_u \times N_t}$ be the newly generated task coalition formation matrix, $y'_{oj} = 1$ indicates that u_j is assigned to perform t_o , and $y'_{oj} = 0$ represents the opposite meaning.

To maintain the stability of MCAV/UCAV cooperative engagement system, the rescheduling plan Y' should follow the principle of minimum original plan changes [24], i.e., the scheduling plan of initial tasks will remain unchanged. Denote FT' to be the completion time of T , the dynamic adjustment model of Y' is defined as

$$\begin{aligned} & \min FT' \\ \text{s.t. } & \begin{cases} y'_{oj} = y_{kj}, & it_k = t_o \\ y'_{oj} = \{0, 1\}, & o = 1, 2, \dots, N_t \\ C_0 \end{cases} \quad (24) \end{aligned}$$

where the first constraint ensures that the coalition formation plan of initial tasks is unchanged, and the second constraint describes the nature of decision variables for coalition adjustment.

5.2.2 GS

It is well known that, the most important thing for an algorithm designed to solve the online scheduling problem is its high stability and low time-consuming. Therefore, it is inappropriate to adopt intelligence algorithms [25], and a GS with excellent real-time performance is put forward to solve (24). Because GS is a deterministic algorithm, the stability requirement can be guaranteed. The detailed steps are listed as follows.

Step 1 Parameter initialization. Denote $\mathbf{R}^{\text{spare}} = (r_1^{\text{spare}}, r_2^{\text{spare}}, \dots, r_{N_r}^{\text{spare}})$ to be the total spare resource, so, $r_p^{\text{spare}} = \sum_{j=1}^{N_u} r_{jp}^u - \sum_{k=1}^{N_{it}} r_{kp}^{it}$. Besides, let $T_{\text{ready}} = ET$.

Step 2 Construct the initial solution of the model, i.e., \mathbf{Y}_1 , which satisfies $y_o^j = 1$ if $et_l = t_o$ and $y_o^j = y_k^j$ if $it_k = t_o$. Thus, for et_1 , available resource is $\mathbf{R}^{\text{spare}}$, while for et_2 , available resource is $\mathbf{R}^{\text{spare}} - \mathbf{R}_1^{\text{et}}$, and the resource distribution strategy can be referred to Section 4.4.2. By that analogy, available resource of $et_3, \dots, et_{N_{et}}$ can be calculated, and denote the current solution as $\mathbf{Y}_{\text{current}} = \mathbf{Y}_1$.

Step 3 Select any one of y_o^j satisfying $et_l = t_o$, and judge that whether after making $y_o^j = 0$, the total resource of UCAVs assigned to perform t_o still meets the demand of the emergent task. If it does, record all such y_o^j and go to Step 4, else, let $T_{\text{ready}} = T_{\text{ready}} \setminus \{et_l\}$.

Step 4 Judge that letting which y_o^j be 0 can make the objective function FT' minimum and record the corresponding solution \mathbf{Y}_2 , if $o = 1$, then, update available resource of $et_1, et_2, \dots, et_{N_{et}}$ according to Step 2, if $o = 2$, then update available resource of $et_2, \dots, et_{N_{et}}$. By that analogy, update available resource of all emergent tasks, and let $\mathbf{Y}_{\text{current}} = \mathbf{Y}_2$.

Step 5 Judge whether T_{ready} is empty, if so, output the current solution $\mathbf{Y}' = \mathbf{Y}_{\text{current}}$, else, return to Step 3.

5.3 External collaboration module

The external collaboration module is the second part, the function of which is to implement external support if the total spare resources of the corresponding MCAV/UCAV task-cluster coalition cannot meet the demand of emergent tasks. When emergent tasks occur, after loading information on emergent tasks, if task-cluster coalition is unable to perform new tasks, then it will borrow UCAVs from other task-cluster coalitions via a point-to-point communication paradigm [26]. Fig. 3 is the communication framework of external collaboration.

Through collaboration among task-cluster coalitions, the task-cluster coalition that needs to perform emergent tasks will obtain information about available UCAVs from

other task-cluster coalitions.

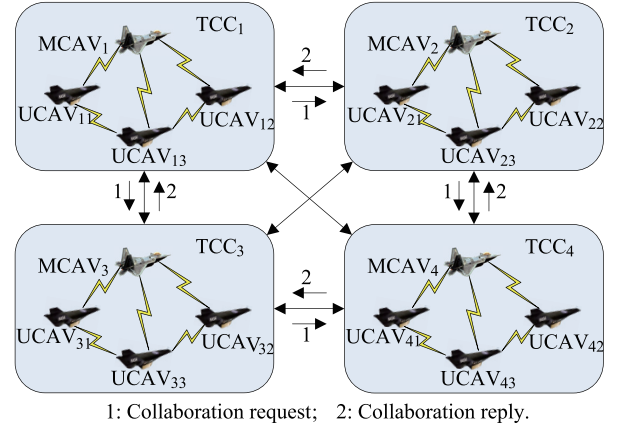


Fig. 3 Framework of external collaboration

5.3.1 External collaboration problem modeling

To reduce the impact on other MCAV/UCAV task-cluster coalitions, rescheduling plan \mathbf{Y}'' ought to follow the principle of minimum borrowed platform number and original plan changes. Let FT'' be the completion time of T , the dynamic adjustment model of \mathbf{Y}'' is defined as

$$\begin{aligned} & \min(FT'', N_u^b) \\ \text{s.t. } & \begin{cases} y_o^j = y_k^j, & it_k = t_o \\ y_o^j = 0, & it_k = t_o \\ y_o^j = \{0, 1\}; y_o^{j'} = \{0, 1\} \\ o = 1, 2, \dots, N_t; j = 1, 2, \dots, N_u; \\ j' = N_u + 1, N_u + 2, \dots, N_u + N_u^b \\ C_0 \end{cases} \end{aligned} \quad (25)$$

Different from j' in Section 3, the value of j' here ranges from $N_u + 1$ to $N_u + N_u^b$.

5.3.2 TSGS

TSGS contains two stages, the first stage is to determine which UCAV needs to be borrowed, and the second stage is to obtain the new task plan on the basis of the first stage.

Step 1 Parameter initialization. Denote the combined resource of available can-borrow UCAVs to be $\mathbf{R}_{b'}^{\text{spare}} = (r_{b'1}^{\text{spare}}, r_{b'2}^{\text{spare}}, \dots, r_{b'N_r}^{\text{spare}})$, $r_{b'p}^{\text{spare}} = \sum_{j''=1}^{N_u^{b'}} r_{j''p}^{u^{b'}}$ ($p = 1, 2, \dots, N_r$), subsequently, $\mathbf{R}^{\text{spare}} = (r_1^{\text{spare}}, r_2^{\text{spare}}, \dots, r_{N_r}^{\text{spare}})$ is the total spare resource.

Therefore, there must be $r_p^{\text{spare}} = r_{b'p}^{\text{spare}} + \sum_{j=1}^{N_u} r_{jp}^u - \sum_{k=1}^{N_{it}} r_{kp}^{it}$ ($p = 1, 2, \dots, N_r$). Then, let $T_{\text{ready}} = ET$.

Step 2 Construct the initial solution of the model, i.e., Y_3 , which satisfies $y''_o = y''_{j'} = 1$ in the condition of $et_l = t_o$ and $y''_o = y''_k$ in the condition of $it_k = t_o$. Hence, for et_1 , the available resource is R^{spare} , while for et_2 , the available resource is $R^{\text{spare}} - R_1^{\text{et}}$, where the specific resource distribution strategy can be referred to Section 4.4.2. By that analogy, the available resource of $et_3, \dots, et_{N_{et}}$ can be calculated, and denote the current solution as $Y_{\text{current}} = Y_3$.

Step 3 Select any $y''_{j'}$ with $et_l = t_o$, judge that whether after making $y''_{j'} = 0$, the total resources of UCAVs assigned to perform t_o still meet the demand of the task. If they do, then record all such $y''_{j'}$ and go to Step 4, else, let $T_{\text{ready}} = T_{\text{ready}} \setminus \{et_l\}$.

Step 4 Judge that letting which $y''_{j'}$ be 0 can make FT'' minimum and then record the corresponding solution Y_4 , if $o = 1$, then update available resource of $et_1, et_2, \dots, et_{N_{et}}$ according to Step 2, if $o = 2$, then update the available resource of $et_2, \dots, et_{N_{et}}$. Analogically, update the available resource of all emergent tasks, and let $Y_{\text{current}} = Y_4$.

Step 5 Judge whether T_{ready} is empty, if so, output the current solution $Y_5 = Y_{\text{current}}$, else, return to Step 3.

Step 6 Through Steps 1–5, minimum borrowed UCAV set U^b is obtained. Denote the combined resource of borrowed UCAVs as $R_b^{\text{spare}} = (r_{b1}^{\text{spare}}, r_{b2}^{\text{spare}}, \dots, r_{bN_r}^{\text{spare}})$,

where $r_{bp}^{\text{spare}} = \sum_{j'=1}^{N_u} r_{j'p}^u$ ($p = 1, 2, \dots, N_r$), thus, the total spare resource is $R^{\text{spare}} = (r_1^{\text{spare}}, r_2^{\text{spare}}, \dots, r_{N_r}^{\text{spare}})$,

where $r_p^{\text{spare}} = r_{bp}^{\text{spare}} + \sum_{j=1}^{N_u} r_{jp}^u - \sum_{k=1}^{N_{it}} r_{kp}^{it}$ ($p = 1, 2, \dots, N_r$). Besides, let $T_{\text{ready}} = ET$.

Step 7 Adopt GS to obtain the final task plan Y'' by taking Y_5 as the initial solution.

6. Simulation experiment

A series of simulation cases are designed to verify the correctness and effectiveness of the proposed mathematical model as well as the solving algorithm. The simulation platform is Matlab 2010a running on a Lenovo PC, the configuration of which is Inter(R) Dual-Core CPU 3.06 GHz.

In terms of task setting, assume that task sequence is shown in Fig. 4.

The specific task sequence is that, it_1 is destroying the long-range anti-aircraft missile, it_2 is destroying the medium-range anti-aircraft missile ②, it_3 is destroying the medium-range anti-aircraft missile ⑤, it_4 is destroying the medium-range anti-aircraft missile ④, it_5 is destroying the medium-range anti-aircraft missile ①, it_6 is destroying the

medium-range anti-aircraft missile ③, it_7 is destroying the communication center, it_8 is destroying the airport, it_9 is destroying the backup C2 center, it_{10} is destroying the C2 center, it_{11} is destroying the reconnaissance troop, it_{12} is destroying the tank force, it_{13} is destroying the bridge ①, it_{14} is destroying the bridge ②, and it_{15} is destroying the oil depot, it_{16} is destroying the powder warehouse, it_{17} is destroying the military camp ②, and it_{18} is destroying the military camp ①.

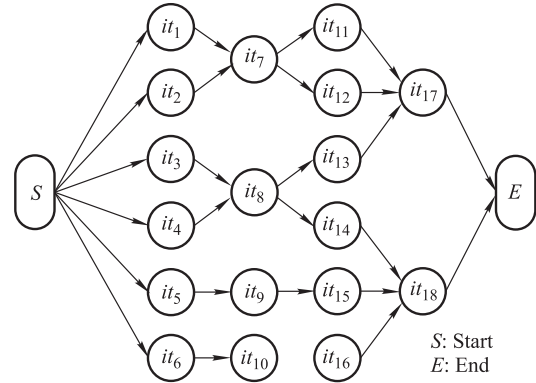


Fig. 4 Task sequence

In addition to that, let $N_u = 20$, $N_r = 8$. Tables 2–5 show resource demand of initial tasks, resource capability of UCAVs, location information of all initial tasks and initial position information of all UCAVs.

Table 2 Resource demand of initial tasks

Task	Resource demand							
	r_1	r_2	r_3	r_4	r_5	r_6	r_7	r_8
it_1	5	2	6	0	0	0	0	1
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots
it_4	0	0	7	0	1	0	0	0
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots
it_7	1	4	7	0	0	0	0	0
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots
it_{10}	0	0	0	0	5	0	0	0
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots
it_{13}	6	4	0	0	7	2	3	6
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots
it_{16}	7	1	2	0	0	0	6	0
it_{17}	0	0	7	0	0	0	0	0
it_{18}	0	0	2	6	2	2	0	0

In addition, the completion time set of initial tasks is (25, 15, 10, 12, 13, 17, 23, 15, 14, 7, 22, 16, 15, 8, 19, 14, 23, 18), and the velocity set of UCAVs is (2, 2.5, 4, 3, 3, 4.5, 3.5, 5, 6, 2.5, 3.5, 5, 4, 3, 4.5, 6, 4.5, 3.5, 4, 7).

Table 3 Resource capability of UCAVs

UCAV	Resource capability							
	r_1	r_2	r_3	r_4	r_5	r_6	r_7	r_8
u_1	1	0	0	0	5	9	3	0
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots
u_5	0	8	6	0	2	0	0	0
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots
u_{10}	1	0	8	7	9	0	0	0
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots
u_{15}	9	0	0	6	4	6	1	1
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots
u_{20}	0	0	9	0	4	1	0	0

Table 4 Location information of initial tasks

Task	Location	Task	Location	Task	Location	Task	Location
it_1	(70, 15)	it_6	(24, 60)	it_{11}	(35, 50)	it_{16}	(56, 15)
it_2	(44, 75)	it_7	(40, 60)	it_{12}	(25, 30)	it_{17}	(84, 35)
it_3	(15, 40)	it_8	(60, 30)	it_{13}	(15, 40)	it_{18}	(43, 25)
it_4	(30, 95)	it_9	(15, 25)	it_{14}	(20, 45)	—	—
it_5	(28, 73)	it_{10}	(45, 35)	it_{15}	(30, 10)	—	—

Table 5 Initial position information of UCAVs

UCAV	Position	UCAV	Position	UCAV	Position	UCAV	Position
u_1	(90, 55)	u_6	(80, 65)	u_{11}	(70, 75)	u_{16}	(60, 85)
u_2	(90, 55)	u_7	(80, 65)	u_{12}	(70, 75)	u_{17}	(60, 85)
u_3	(90, 55)	u_8	(80, 65)	u_{13}	(70, 75)	u_{18}	(60, 85)
u_4	(90, 55)	u_9	(80, 65)	u_{14}	(70, 75)	u_{19}	(60, 85)
u_5	(90, 55)	u_{10}	(80, 65)	u_{15}	(70, 75)	u_{20}	(60, 85)

6.1 Simulation, comparison and analysis of QABC as well as contrast algorithms

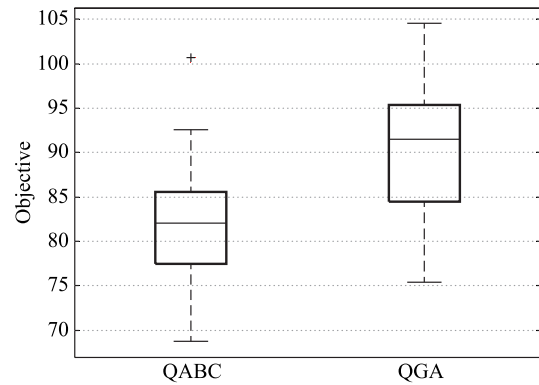
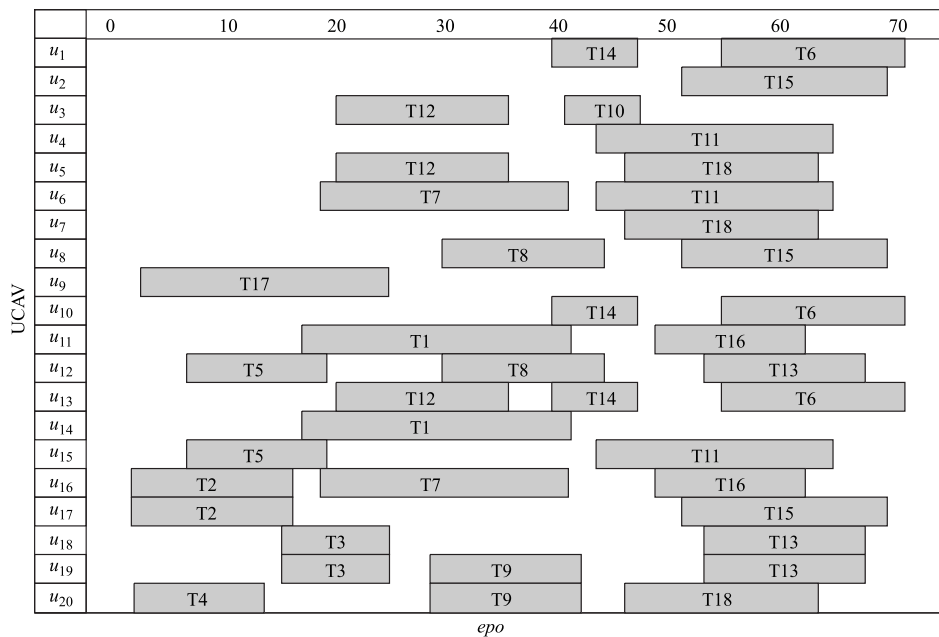
In terms of parameter setting, it sets $N_{eb} = N_{ob} = 40$, in which N_{eb} is the number of employed bees, and N_{ob} is the number of onlooker bees, besides, let $P_m = 0.01$.

Comparing QABC with the quantum genetic algorithm (QGA) [27] and PTCFA (redesign partly according to the model), Table 6 shows the comparison result of the three algorithms. In particular, QABC and QGA run 50 times, 200 iterations each time respectively, and the final results are averaged.

Table 6 Comparison of QABC, QGA and PTCFA

Algorithm	QABC	QGA	PTCFA
Object	81.797 8	89.927 4	150.722 6

Fig. 5 shows Box-and-Whisker plot reflecting the comparison of QABC and QGA. From Table 6 and Fig. 5, it can be got that QABC outperforms QGA and PTCFA in optimization capacity. And Fig. 6 is the Gantt chart under a typical solution, where the completion time of initial task set is 75.928 9.

**Fig. 5** Comparison of QABC and QGA**Fig. 6** Gantt chart of a typical solution

6.2 Simulation, comparison and analysis of GS as well as contrast algorithms

Take the typical solution in Section 6.1 as the static coalition formation result, to simplify, assuming that all emergent tasks are pop-up tasks, and all appear at $epo = 0$, besides, all information of emergent tasks have the stochastic nature. Fig. 7 shows the comparison of GS, QABC and QGA. In GS, the final objective function value comes up to 110.440 4, and for QABC and QGA, once the simulation result has reached 110.440 4, they will be terminated. Particularly, QABC and QGA run 50 times, and the results are averaged.

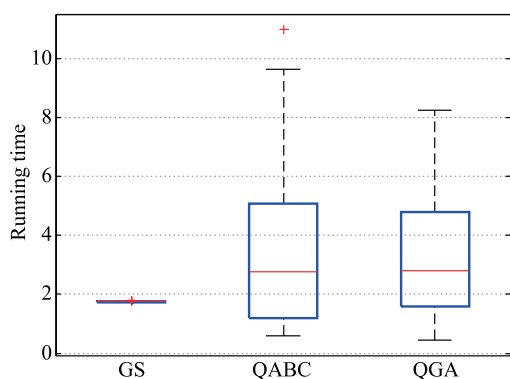


Fig. 7 Comparison of GS, QABC and QGA

From Fig. 7, it can be concluded that GS has better real-time and stability performance compared to QABC and QGA.

6.3 Simulation, comparison and analysis of TSGS as well as contrast algorithm

Considering (25) is a typical multi-objective optimization problem, hence, it adopts a multi-objective quantum artificial bee colony (MOQABC) as a benchmark. Take stochastic nature of emergent tasks as consideration, in TSGS, the final objective function value is (141.353 9, 3), once MOQABC obtains Pareto optimal solution, it will terminate the simulation. Particularly, MOQABC runs 50 times, and the results are averaged. Table 7 is the comparison of TSGS and MOQABC in terms of running time.

Table 7 Comparison of TSGS and MOQABC

Comparison item	TSGS	MOQABC
Average	2.392 2	19.826 3
Mean square error	0.052 4	31.495 5

From Table 7, it can be concluded that TSGS has better real-time as well as stability performance compared to MOQABC.

7. Conclusions

Most previous studies related to UCAV and MCAV/UCAV cooperative engagement system all made efforts to carry out research on static coalition formation. However, few studies have explored dynamic coalition adjustment. In this paper, task scheduling as well as rescheduling models is presented, and the corresponding algorithms are adopted to solve them.

Specially, a programming formulation for the problem of scheduling UCAVs to timing-constraints tasks is established and a QABC algorithm is proposed to solve it. Then, two distinct programming formulations for the problem of rescheduling UCAVs to timing-constraints emergent tasks are set up, and two algorithms called GS and TSGS are adopted to solve these two models respectively.

Admittedly, one limitation of the models is that they do not consider the time-sensitive characteristic of tasks [28]. In the aspect of algorithm design, GS and TSGS have relatively better real-time and stability performance, but with the increase of the number of emergent tasks, the running time of GS and TSGS would increase rapidly, so, designing a better real-time algorithm is still a difficult problem.

References

- [1] EVERS L, BARROS A I, MONSIEUR H, et al. Online stochastic UAV mission planning with time windows and time-sensitive targets. *European Journal of Operational Research*, 2014, 238(1): 348–362.
- [2] MURRAY C C, PARK W. Incorporating human factor considerations in unmanned aerial vehicle routing. *IEEE Trans. on Systems, Man and Cybernetics: Systems*, 2013, 43(4): 860–874.
- [3] ZHU X P, LIU Z C, YANG J. Model of collaborative UAV swarm toward coordination and control mechanisms study. *Proc. of the International Conference on Computational Science*, 2015: 493–502.
- [4] ERIGNAC C. An exhaustive swarming search strategy based on distributed pheromone maps. *Proc. of the AIAA Infotech & Aerospace Conference and Exhibit*, 2007: 1–16.
- [5] SCHOUWENAARS T, VALENTI M, FERON E, et al. Linear programming and language processing for human/unmanned aerial vehicle team missions. *Journal of Guidance, Control and Dynamics*, 2006, 29(2): 303–313.
- [6] CHINGTHAM T, SAHOO G, GHOSE M. An unmanned aerial vehicle as human-assistant robotics system. *Proc. of the IEEE International Conference on Computational Intelligence and Computing Research*, 2010: 1–6.
- [7] HU X X, MA H W, YE Q S, et al. Hierarchical method of task assignment for multiple cooperating UAV teams. *Journal of Systems Engineering and Electronics*, 2015, 26(5): 1000–1009.
- [8] ZENG J, YANG X K, YANG L Y, et al. Modeling for UAV resource scheduling under mission synchronization. *Journal of Systems Engineering and Electronics*, 2010, 21(5): 821–826.
- [9] LIN L, SUN Q B, WANG S G, et al. Research on time window based coalition formation for multi-UAVs task assignment. *Journal of Electronics & Information Technology*, 2013,

- 35(8): 1983–1988. (in Chinese)
- [10] HAN X, MANDAL S, PATTIPATI K R, et al. An optimization-based distributed planning algorithm: a blackboard-based collaborative framework. *IEEE Trans. on Systems, Man and Cybernetics: Systems*, 2014, 44(6): 673–686.
 - [11] WANG L, QIN Y, XU J, et al. A fuzzy optimization model for high-speed railway timetable rescheduling. *Discrete Dynamics in Nature and Society*, 2012, doi:10.1155/2012/827073.
 - [12] ARIANO A D, CORMAN F, PACCIARELLI D, et al. Re-ordering and local rerouting strategies to manage train traffic in real time. *Transportation Science*, 2008, 42(4): 405–419.
 - [13] MANATHARA J G, SUJIT P B, BEARD R W. Multiple UAV coalitions for a search and prosecute mission. *Journal of International Robotic System*, 2011, 62(1): 125–158.
 - [14] SUJIT P B, MANATHARA J G, GHOSE D, et al. *Handbook of unmanned aerial vehicles*. New York: Springer Verlag, 2015.
 - [15] KIM M H, BAIK H, LEE S. Resource welfare based task allocation for UAV team with resource constraints. *Journal of Intelligent & Robot System*, 2015, 77(3): 611–627.
 - [16] YOUNG B W. Future integrated fire control. *Proc. of the 10th International Command and Control Research and Technology Symposium*, 2005: 1–21.
 - [17] MENG L H, XU X H, ZHAO Y F. Cooperative coalition for formation flight scheduling based on incomplete information. *Chinese Journal of Aeronautics*, 2015, 28(6): 1747–1757.
 - [18] LIN L, ZHENG Z Q. Combinatorial bids based multi-robot task allocation method. *Proc. of the IEEE International Conference on Intelligent Robots and Automation*, 2005: 18–22.
 - [19] SUJIT P B, GEORGE J M, BEARD R W. Multiple UAV coalition formation. *Proc. of the American Control Conference*, 2008: 2010–2015.
 - [20] JIN Y, LIAO Y, MINAI A, et al. Balancing search and target response in cooperative UAV teams. *IEEE Trans. on Systems, Man and Cybernetics: Systems*, 2006, 36(3): 571–587.
 - [21] CHOI H L, BRUNET L, HOW J P. Consensus-based decentralized auctions for robust task allocation. *IEEE Trans. on Robotics*, 2009, 25 (4): 912–926.
 - [22] JOHNSON L, PONDA S S, CHOI H L, et al. Asynchronous decentralized task allocation for dynamic environments. *Proc. of the AIAA Guidance, Navigation and Control Conference*, 2011: 1441–1453.
 - [23] KARABOGA D. An idea based on honey bee swarm for numerical optimization. *Kayseri, Turkey: Erciyes University*, 2005.
 - [24] MU L. *Dynamic adaptive optimization methodology of C2 organization structure under uncertainty mission environment*. Changsha, China: National University of Defense Technology, 2011. (in Chinese)
 - [25] ZHU B, ZHU J F. Flight schedule recovery under uncertain airport capacity. *Transactions of Nanjing University of Aeronautics and Astronautics*, 2016, 33(4): 479–490.
 - [26] PARK C, PATTIPATI K R, AN W. Quantifying the impact of

information and organizational structures via distributed auction algorithm: point-to-point communication structure. *IEEE Trans. on Systems, Man and Cybernetics: Systems*, 2012, 42(1): 68–86.

- [27] LABOUDI Z, CHIKHI S. Comparison of genetic algorithm and quantum genetic algorithm. *The International Arab Journal of Information Technology*, 2012, 9(3): 243–249.
- [28] ROSENBERG B, BURGE J, GONSALVES P G. Applying evolutionary multi-objective optimization to mission planning for time-sensitive targets. *Proc. of the Genetic and Evolutionary Computation Conference*, 2005: 1–8.

Biographies



ZHONG Yun was born in 1990. He is currently a Ph.D. student of Air Force Engineering University. He received his B.S. degree in network engineering, M.S. degree in communication and information system from Air Force Engineering University, in 2012 and 2014, respectively. His research interests include decision support system and MAV/UAV cooperative engagement.
E-mail: 718227697@qq.com



YAO Peiyang was born in 1960. He is currently a professor and Ph.D. student supervisor of Air Force Engineering University. His research interests include command and information system and combat modeling.
E-mail: ypy_664@163.com



ZHANG Jieyong was born in 1983. He is currently a lecturer of Information and Navigation College, Air Force Engineering University. His research interests include mission planning technique and military organizational analysis.
E-mail: dumu3110728@126.com



WAN Lujun was born in 1986. He is currently a lecturer of Air Force Engineering University. His research interests include operational organization design and air traffic control.
E-mail: pandawlj@126.com