



# Multi-UAV reconnaissance task allocation for heterogeneous targets using grouping ant colony optimization algorithm

Sheng Gao<sup>1</sup> · Jiazheng Wu<sup>1</sup> · Jianliang Ai<sup>1</sup>

Accepted: 8 February 2021 / Published online: 23 February 2021  
© The Author(s), under exclusive licence to Springer-Verlag GmbH, DE part of Springer Nature 2021

## Abstract

The study on multiple unmanned aerial vehicles (UAVs) reconnaissance task allocation problem is an important research field, which is significant for both military and civilian applications. This problem has often been considered as a multiple traveling salesman problem where the targets are considered as points. In this paper, we present a novel mathematical model that classifies heterogeneous targets as point targets, line targets and area targets to improve the fidelity of the model. It is a complex combinatorial optimization problem, for which we can hardly get an optimal solution as the scale of the problem expands. A new heuristic algorithm called grouping ant colony optimization algorithm is proposed for this new model. Compared with traditional ant colony algorithm, pheromone is divided into membership pheromone and sequence pheromone corresponding to grouping and permutation characteristics of the model, respectively. Also, negative feedback mechanism is introduced to accelerate convergence speed of the algorithm. The simulation results demonstrate that the new algorithm can consider comprehensively the performance of different UAVs and the characteristic of heterogeneous targets. It outperforms existing methods reported in the literature in terms of optimality of the result, and the advantage gets more obvious with the scale of reconnaissance task allocation problem expanding.

**Keywords** Unmanned aerial vehicles · Reconnaissance task allocation · Combinatorial optimization problem · Heuristic algorithm · Ant colony algorithm

## 1 Introduction

With the development of science and technology such as control theory, wireless network, and electronic engineering (Hoffmann et al. 2007; Wang et al. 2020; Hadi et al. 2014), autonomous performance of unmanned aerial vehicle (UAV) is constantly improving, making it more and more popular in military and civilian applications such as natural disaster relief (Luo et al. 2019), search and rescue (Yang et al. 2020), forest fire extinguishing (Merino et al. 2006), and agricultural irrigation (Albornoz and Giraldo 2017). However, a single UAV can hardly accomplish a complex task because of its limited capability. The research of multi-UAV system is necessary for the cooperation between UAVs, and the development of

single-UAV platform and multiple UAV networks makes it possible (Wang et al. 2017). Multi-UAV collaborative reconnaissance is a typical mission for better target finding and information acquisition. Task allocation is an important problem in multi-UAV reconnaissance mission, which can make UAVs conducting tasks in a good order and minimize system cost efficiently (Fu et al. 2019). Researches on multi-UAV task allocation are mainly focused on problem modeling and algorithm innovation.

The basic task allocation problem that aims to find the shortest flight path can be formulated as vehicle routing problem (VRP) (O'Rourke et al. 2001), multiple traveling salesman problem (MTSP) (Yousefikhoshbakht et al. 2013), and mixed integer linear programming (MILP) (Alighanbari 2004). Considering UAV endurance and multitasking, the task allocation problem can be formulated as multiple traveling salesman problem with time window (MTSP-TW) (Kona et al. 2015) and dynamic network flow optimization (DNFO) (Nygard et al. 2001). Although algorithms for solving these problems have been fully

✉ Sheng Gao  
differentthink@163.com

<sup>1</sup> Department of Aeronautics and Astronautics, Fudan University, Shanghai 200433, China

studied, the task allocation problem is always greatly simplified without consideration of the nature of the problem when using these models. Besides, the existing models only concentrate on the features of UAVs, while targets are always simplified as points and the features and heterogeneity of targets are ignored. Zhu et al. (2018) firstly introduced the heterogeneity of targets in task allocation problem, but only considered differences between targets at the stage of entering and leaving after allocation. In this paper, we present a novel multi-UAV reconnaissance task allocation model, considering the heterogeneity of both UAVs and targets. Targets are classified as point targets, line targets, and area targets, where line targets are modeled as two symbiotic point targets and area targets are modeled as four mutually exclusive point targets according to the characteristic of targets.

The task allocation problem is a typical NP-hard combinatorial optimization problem. Algorithms to solve the combinatorial optimization problem can be divided into two categories, the exact methods and the approximate methods. The exact methods such as branch and bound (BNB) (Vincent et al. 2013) and dynamic programming (DP) (Zhang et al. 2019) can obtain local optimal solutions for low-dimensional problems, but they can hardly find feasible solution as the number of UAVs and targets grows, because of the exponential increase of computational cost. On the contrary, the approximate methods with low computational complexity can efficiently acquire feasible solutions, which are mainly utilized to solve combinatorial optimization problems. Most of approximate methods are inspired by biological or physical phenomenon, such as simulated annealing (Mafarja and Mirjalili 2017), neural networks (Somhom et al. 1999), bee colony algorithm (Dokeroglu et al. 2019), ant colony optimization (ACO) (Junjie and Dingwei 2006), and genetic algorithm (GA) (Yuan et al. 2013). ACO and GA are widely applied for the task allocation problem and have been verified to be potential to find superior solutions (Pendharkar 2015; Srikanth and Geetha 2018; Ramirez-Atencia et al. 2017). The no free lunch theorem indicates that no universal heuristics outperform other heuristics across all possible problems, though one heuristic may be more suitable for a particular problem than others (Wolpert and Macready 1997; Pandiri and Singh 2018). Therefore, it is necessary to modify algorithms according to the characteristics of task allocation problem. Deng et al. (2013) proposed multi-type genes to modify the genetic algorithm for meeting specific demands of tasks. Zhu et al. (2018) developed the opposition-based genetic algorithm using double-chromosomes encoding and multiple mutation operators (OGA-

DEMMO) which used opposition-based learning and multiple mutation operators to enhance the global exploration capability of genetic algorithm. Fei et al. (2008) presented a multi-ant colony algorithm based on the job-division mechanism for a better construction of task allocation solution.

The multi-UAV reconnaissance task allocation problem has two characteristics: (1) grouping: targets need to be parted into different groups, each group corresponds to a UAV; (2) permutation: a given set of targets needs to be arranged into a particular order. Previous ACO-based algorithms for task allocation problem do not make full use of grouping information, causing a lot of uncertainty in the process of state transition. In this paper, we propose a new grouping ant colony optimization (GACO) algorithm based on ACO to solve the multi-UAV reconnaissance task allocation problem. In GACO, pheromone is divided into membership pheromone and sequence pheromone corresponding to grouping and permutation characteristic, respectively. Due to the large search space, negative feedback mechanism is incorporated into traditional ant colony algorithm, which can increase convergence speed. The new algorithm can improve search efficiency by learning the correlation between UAVs and targets.

The main contributions of this paper are as follows. Firstly, we present a novel multi-UAV reconnaissance task allocation model, which considers the heterogeneity of targets and performance of different UAVs. Secondly, to produce high-quality solutions, we develop an ACO meta-heuristic-based algorithm to solve the new multi-UAV reconnaissance task allocation model. Based on the characteristics of the problem, we divide the pheromone into membership pheromone and sequence pheromone and introduce a competition mechanism to expand new nodes. The negative feedback process is utilized to increase convergence speed of the new algorithm as well. Finally, we analyze the performance of our algorithm experimentally and conduct comparative experiments of different scale scenarios, and the results show that our algorithm has better global exploration capability.

The remainder of this paper is organized as follows: Sect. 2 describes the heterogeneity of UAVs and targets, and Sect. 3 introduces the formal model of the multi-UAV reconnaissance task allocation problem. Our proposed algorithm to solve the model is presented in Sect. 4. Numerical simulation is performed in Sect. 5 to illustrate the effectiveness and optimality of the proposed method. Finally, Sect. 6 outlines some concluding remarks to summarize the contribution of this paper.

## 2 Basic models

### 2.1 Sensor model

In this paper, UAVs are equipped with photoelectric reconnaissance sensors, which can reconnoiter ground targets. When the area of targets is fully covered by a sensor's field of view, the reconnaissance task on this target is completed. As illustrated in Fig. 1, a sensor's field of view is assumed as a circle with radius  $r$ ,  $d = 2r$  is the reconnaissance width of the sensor, and  $H$  is the flight altitude. To simplify the model, it is assumed that the sensor's field of view is not influenced by the attitude of UAVs.

### 2.2 Targets model and reconnoiter strategies

According to their features of geometry and the sensor's field of view, reconnaissance targets in real-world mission environment are classified as point targets, line targets, and area targets. As illustrated in Fig. 2, point targets refer to those targets whose size is smaller than the sensor's field of view, such as buildings and ground vehicles. The reconnaissance task for a point target is performed when a UAV flies over it.

Line targets represent targets whose length is longer than the reconnaissance width  $d$ , while width is shorter than  $d$ . As shown in Fig. 3, when a UAV flies over the center line along the longer side, a line target can be covered. In this paper, line targets are considered as two point targets: the entry point and the exit point.  $L_1$  is the entry point and  $L_2$  is the exit point; this couple of point targets is symbiotic, which means once one is reconnoitered, the other one must be the next target. Typical line targets include railways, rivers, and aircraft run ways.

As for area targets, both their length and width are longer than reconnaissance width  $d$ . There are many reconnaissance strategies for area targets, and spiral search route is adopted in this paper with the shortest route length,

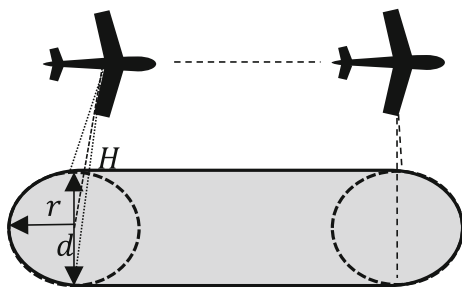


Fig. 1 Field of view of sensors

Fig. 2 Schematic diagram of point target

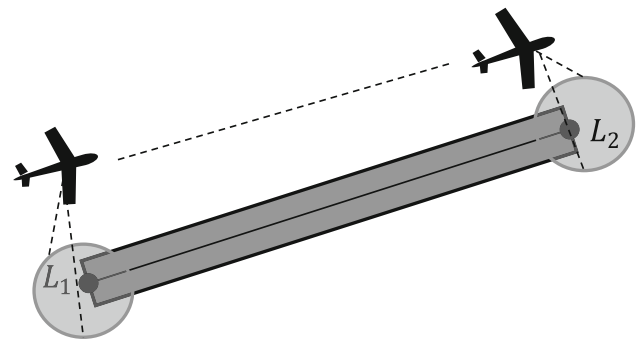
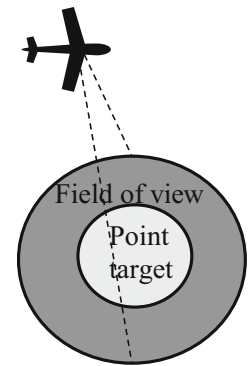


Fig. 3 Schematic diagram of line target

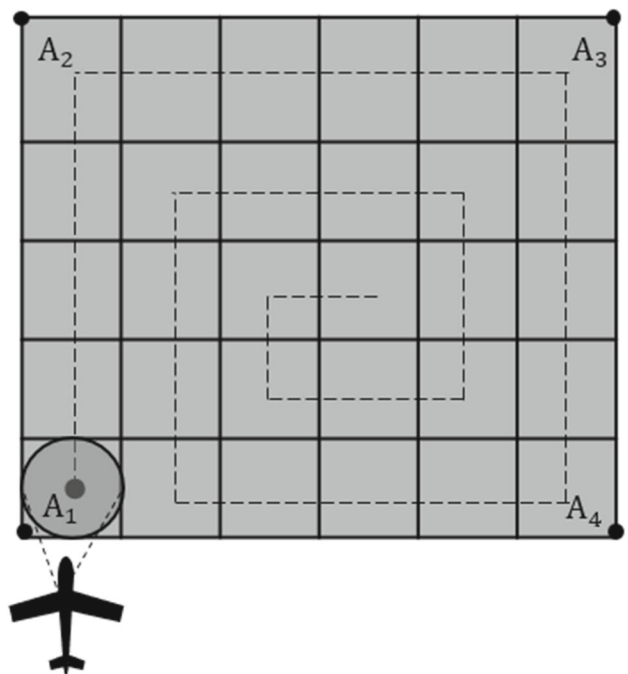


Fig. 4 Spiral search route of area target

as shown in Fig. 4. We consider the area target as four mutually exclusive point targets, which are the vertices of the area target. An entry point and an exit point should be chosen from the four vertices, the entry point and the exit point can be the same one. For the spiral search route, it can be proven that the search length is the same for different entry point and exit point combinations. The search length is calculated as follows:

$$L_s = \frac{L_a \cdot W_a}{d} - d + \frac{\sqrt{L_a^2 + W_a^2}}{2} \tag{1}$$

where  $L_s$  is the total search route length for the area target;  $L_a$ ,  $W_a$  are the length and width of the area target, respectively;  $d$  is the reconnaissance width of the sensor. Typical area targets include lakes, squares, and other wide-area targets.

### 2.3 UAVs model

To perform the reconnaissance task, there are  $N_u$  UAVs with specific features, which are presented in Table 1. In this paper, the UAV is considered as a particle that cruises.

## 3 Problem description and formulation

The multi-UAV reconnaissance task allocation problem aims at assigning  $N_t$  targets to  $N_u$  UAVs with lower cost, where  $N_u < N_t$ . The UAVs start at departure airports and land at landing airports, each target needs to be reconnoitered by a single UAV, and each UAV can reconnoiter any targets within its capability constraint. In this paper, targets are heterogeneous, so are the UAVs. The targets include point targets, line targets, and area targets. The line target is considered as two symbiotic point targets represented by two endpoints. The area target is considered as four mutually exclusive point targets represented by four vertices. The cost contains two parts: the maximum time to complete reconnaissance task and the total fuel consumption of UAVs, which is related to the properties of different UAVs. Thus, the multi-UAV reconnaissance task

allocation problem is similar to the multi-depot multiple traveling salesman problem (MDMTSP). Different from the MDMTSP, target points in this problem can be symbiotic and mutually exclusive to each other.

Take  $U = \{U_1, U_2, \dots, U_{N_u}\}$  as a set of  $N_u$  UAVs,  $D = \{D_1, D_2, \dots, D_{N_u}\}$  as a set of designated departure airports, and  $E = \{E_1, E_2, \dots, E_{N_u}\}$  as a set of designated landing airports.  $T = \{T_1, T_2, \dots, T_{N_p}\}$  is a set of  $N_p$  points representing  $N_t$  targets,  $T = T_p \cup T_L \cup T_A$ , where  $T_p, T_L, T_A$  are sets of points representing point targets, line targets, and area targets, respectively. The goal of UAVs is to reconnoiter all the targets from their designated departure airports to their designated landing airports in a minimum time with minimum fuel consumptions. Thus, it is a multi-objective optimization problem, the weighted-sum approach is applied in this paper Deb (2014), the objective function including two subobjectives is defined as follows:

$$\min J = \varepsilon \cdot w_1 \cdot \max_{u=1,2,\dots,N_u} t_u + w_2 \cdot \sum_{u=1}^{N_u} t_u f_{r_u} \tag{2}$$

where  $w_1, w_2 \in [0, 1]$  are the weight factors of the two subobjectives;  $\varepsilon$  is scale factor that makes two subobjectives the same order of magnitude. Since the minimum of the above problem does not change if all weights are multiplied by a constant, it is the common practice to choose weights such that their sum is one, or  $w_1 + w_2 = 1$ .  $f_{r_u}$  is the fuel consumption at cruising speed,  $t_u$  is the time of completing the allocated task for the  $u$ th UAV, and is determined by the flight speed and path length, as defined in the following equation:

$$t_u = \left( \sum_{k \in D} \sum_{j \in T} RL_{kj} X_{kj}^u + \sum_{j \in T} \sum_{h \in E} RL_{jh} X_{jh}^u + \sum_{i \in T} \sum_{j \in T} RL_{ij} X_{ij}^u \right) / v^u \tag{3}$$

where  $v^u$  is the cruising speed of  $u$ th UAV; if  $i, j \in T_A$ ,  $RL_{ij}$  is the search length of area targets formulated by Eq. (1), else,  $RL_{ij}$  is the path length from location  $i$  to location  $j$ ; the binary matrix  $\mathbf{X}^u$  is called allocation matrix representing

**Table 1** Different UAV features considered

Feature	Description
Initial position	The position of the UAV at the beginning of the task
End position	The position of the UAV after completing the task
Autonomy	The maximum time that the UAV can keep flying
Cruising speed	Speed of the UAV for a flight profile
Fuel consumption ratio	Fuel consumption per hour of the UAV for a flight profile

the task allocation results of the  $u$ th UAV. If the  $u$ th UAV needs to move from  $i$  to  $j$ , then  $X_{ij}^u$  is 1, otherwise  $X_{ij}^u$  is 0.

Additionally, the following constraints are imposed on this problem:

$$\sum_{u \in U} \sum_{j \in T} X_{kj}^u = \sum_{u \in U} \sum_{j \in T} X_{jh}^u = 1 \quad k \in D, h \in E \tag{4}$$

$$\sum_{u \in U} \sum_{k \in D} X_{kj}^u + \sum_{u \in U} \sum_{i \in T} X_{ij}^u + \sum_{u \in U} \sum_{h \in E} X_{jh}^u = 1 \quad j \in T \tag{5}$$

$$\sum_{k \in D} \sum_{h \in E} X_{kh}^u = 0 \quad u \in U \tag{6}$$

$$X_{kj}^u + \sum_{i \in T} X_{ij}^u - X_{jh}^u - \sum_{i \in T} X_{ji}^u = 0 \tag{7}$$

$u \in U, k \in D, h \in E, j \in T$

$$u_i - u_j + L \sum_{u \in U} X_{ij}^u + (L - 2) \sum_{u \in U} X_{ji}^u \leq L - 1 \tag{8}$$

$i \neq j, i, j \in T$

$$\sum_{u \in U} X_{ij}^u = 1 \quad i, j \in T_L^m \tag{9}$$

$$\sum_{u \in U} \sum_{i \in T_A^m} \sum_{j \in T_A^m} X_{ij}^u \leq 1 \tag{10}$$

In these formulas, constraint (4) ensures that each UAV takes off from a designated departure airport, and lands at a designated landing airport. Constraint (5) ensures each target is reconnoitered exactly once. Tours without any target points are prohibited with constraint (6). Route continuity is represented by constraint (7). In constraint (8),  $u_i$  is the number of nodes visited on the path from the departure airport up to node  $i$ ; this constraint can prevent subtours with no departure airport and landing airport. Constraint (9) represents the symbiotic relationship between  $i$  and  $j$ , while  $i, j$  belong to the  $m$ th line target,  $T_L^m$  is the point set representing  $m$ th line target. Constraint (10) represents the mutual relationship between  $i$  and  $j$ , while  $i, j$  belong to the  $m$ th area target,  $T_A^m$  is the point set representing  $m$ th area target.

## 4 Grouping ant colony optimization algorithm

### 4.1 Group ant colony

In the multi-UAV reconnaissance task allocation problem, different UAVs do not reconnoiter the same target, thus we can map each UAV to an ant subgroup, each ant subgroup constructs a task assignment plan for its corresponding UAV, and subgroups interact with each other by pheromone to achieve task coordination.

Dividing ant colony into different subgroups is based on the fact that different UAVs do not reconnoiter same targets. Let ant colony  $AC = \{AC_i, i = 1, 2, \dots, N_{AC}\}$ ,  $N_{AC}$  is the number of subgroups,  $N_{AC} = N_u$ , the  $AC_i$  is subject to:

$$\begin{cases} \bigcup_{i=1}^{N_u} AC_i = AC \\ AC_i \cap AC_j = \emptyset, \quad i \neq j \end{cases} \tag{11}$$

The ant clan  $AG_i = \{Ant_{1,i}, Ant_{2,i}, \dots, Ant_{N_u,i}\}$ ,  $i = 1, 2, \dots, N_{AG}$  is an ant set that contains  $N_u$  ants; each  $AG_i$  is an allocation solution of UAVs.  $AG_i$  is subjected to:

$$\begin{cases} \bigcup_{i=1}^{N_{AG}} AG_i = AC \\ AG_i \cap AG_j = \emptyset \quad i \neq j \end{cases} \tag{12}$$

In conclusion, the ant colony can be divided into  $N_u$  subgroups in accordance with  $N_u$  different UAVs, or  $N_{AG}$  ant clans in accordance with  $N_{AG}$  allocation solutions. It can be illustrated in Fig. 5.

### 4.2 Pheromone construction and state transition

In the grouping ant colony optimization algorithm, the task allocation solution is formulated by ant clan  $AG_i$ ,  $i = 1, 2, \dots, N_{AG}$ . Firstly, each ant in  $AG_i$  will find its next

AG				
AC	AG <sub>1</sub>	AG <sub>2</sub>	...	AG <sub>N<sub>AG</sub></sub>
AC <sub>1</sub>	Ant <sub>1,1</sub>	Ant <sub>1,2</sub>	...	Ant <sub>1,N<sub>AG</sub></sub>
AC <sub>2</sub>	Ant <sub>2,1</sub>	Ant <sub>2,2</sub>	...	Ant <sub>2,N<sub>AG</sub></sub>
⋮	⋮	⋮	⋮	⋮
AC <sub>N<sub>u</sub></sub>	Ant <sub>N<sub>u</sub>,1</sub>	Ant <sub>N<sub>u</sub>,2</sub>	...	Ant <sub>N<sub>u</sub>,N<sub>AG</sub></sub>

Fig. 5 Division of ant colony

target based on transition probability, and the transition probability  $p_{ij}$  is the probability of target  $j$  been chosen by the ant from target  $i$  and is formulated as follows:

$$p_{ij} = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha \cdot [\eta_{ij}]^\beta}{\sum_{k \in \text{allowed}_t} [\tau_{ik}(t)]^\alpha \cdot [\eta_{ik}]^\beta} & f j \in \text{allowed}_t \\ 0 & \text{otherwise} \end{cases} \quad (13)$$

where  $\text{allowed}_t = \{T - \text{tabu}_t\}$  is the target set containing all the targets that have not been chosen at time  $t$ ,  $\text{tabu}_t$  is the target set containing all the targets that have already been chosen at time  $t$ ;  $\tau_{ij}(t)$  is the sequence pheromone value between target  $i$  and  $j$ ;  $\eta_{ij}$  is the inverse distance between target  $i$  and  $j$ , it is used as heuristic information.  $\alpha$  and  $\beta$  are parameters that control the relative importance of pheromone versus greediness of the algorithm.

Secondly, after all ants in  $AG_i$  have chosen their next target, they will compete with each other for the permission to extend. It means that only one of the  $N_u$  ants in  $AG_i$  can choose the target at a time. The probability of ant  $m$  winning the permission to extend is:

$$p_m = \frac{\gamma_{j,m}(t)}{\sum_{k=1}^{N_u} \gamma_{j,k}(t)} a \quad (14)$$

where  $\gamma$  is the membership pheromone matrix;  $\gamma_{j,m}$  is the membership pheromone in  $j$ th row and  $m$ th column of  $\gamma$ , which stands for the probability that target  $j$  belongs to UAV  $m$ . The formula means that the greater the value of membership pheromone is, the greater the probability is.

### 4.3 Pheromone update

In ACO, the construction of the solution depends on the accumulation of pheromone, and it is a positive feedback process. In this paper, negative feedback mechanism is introduced to GACO algorithm. The update of sequence pheromone and membership pheromone is different from each other.

Once all ants in ant colony finished node extending, the sequence pheromone is updated as follows:

$$\tau_{ij}(t + 1) = \max(0, (1 - \rho) \cdot \tau_{ij}(t) + \Delta\tau_{ij}) \quad (15)$$

where  $\rho$  ( $0 < \rho < 1$ ) is the evaporation coefficient of pheromone, and  $1 - \rho$  the persistence coefficient,  $\Delta\tau_{ij}$  is the pheromone increments between target  $i$  and  $j$  after each iteration. The value of  $\tau_{ij}$  is non-negative.

$$\Delta\tau_{ij} = \sum_{k=1}^{N_{AG}} \Delta\tau_{ij}^k \quad (16)$$

where  $\Delta\tau_{ij}^k$  is the quantity of pheromone laid on edge  $(i, j)$  by the  $k$ th ant clan after each iteration. It is given by:

$$\Delta\tau_{ij}^k = \begin{cases} pf^k - nf^k & \text{if } k\text{th ant clan uses edge } (i, j) \\ 0 & \text{otherwise} \end{cases} \quad (17)$$

$$pf^k = \frac{Q}{C_k} \quad (18)$$

$$nf^k = \begin{cases} \frac{3 \cdot Q}{2 \cdot C_k} & \text{if } C_k = \max_k \{C_k \mid k = 1, 2, \dots, N_{AG}\} \\ 0 & \text{otherwise} \end{cases} \quad (19)$$

where  $Q$  is a constant, and  $C_k$  is cost value of ant clan  $k$  calculated from formula (2). Formulas (17)–(19) mean that in each iteration  $N_{AG}$  ant clans leave pheromone trail on edges they traveled through. The ant clan leaves more pheromone trails when its cost function value is smaller. If the ant clan got the largest cost function value, the pheromone value will be subtracted by  $nf^k$ , and it is just the process of negative feedback.

The membership pheromone is updated by the grouping result of  $k$ th ant clan with lowest cost value in current iteration and  $l$ th ant clan with lowest cost value up until current iteration.  $k$ th ant clan represents local best solution, and  $l$ th ant clan represents currently global best solution. If target  $j$  belongs to UAV  $m$  in the  $k$ th ant clan or  $l$ th ant clan, then  $\gamma_{j,m}$  is updated as follows:

$$\gamma_{j,m} = \gamma_{j,m} + K \quad j = 1, \dots, N_p \quad (20)$$

Then, each row element in  $\gamma$  is updated as follows to ensure the sum of each row is 1:

$$\gamma_{j,m} = \frac{\gamma_{j,m}}{\sum_{m=1}^{N_u} \gamma_{j,m}} \quad j = 1, \dots, N_p; \quad m = 1, \dots, N_u \quad (21)$$

where  $K$  ( $0 < K < 1$ ) is a constant. Formulas (20)–(21) mean that the update of the membership pheromone is a positive feedback and it is also a negative feedback process when  $N_u > 2$ , it is proved as follows:

**Theorem 1** *Update of the membership pheromone is not only a positive feedback but also a negative feedback when  $N_u > 2$ .*



**Proof** The sum of each row of membership pheromone matrix is 1. According to formulas (20)–(21), the membership pheromone can be updated to:

$$\begin{cases} \gamma_{j,m}(t+1) = \frac{\gamma_{j,m}(t) + K}{1 + K} & \text{if } j \text{ belongs to } m \\ \gamma_{j,m}(t+1) = \frac{\gamma_{j,m}(t)}{1 + K} & \text{otherwise} \end{cases} \quad (22)$$

The membership pheromone increment is:

$$\begin{cases} \Delta\gamma_{j,m} = \frac{K(1 - \gamma_{j,m}(t))}{1 + K} & \text{if } j \text{ belongs to } m \\ \Delta\gamma_{j,m} = -\frac{K \cdot \gamma_{j,m}(t)}{1 + K} & \text{otherwise} \end{cases} \quad (23)$$

Since  $0 < \gamma_{j,m}(t) < 1$ , it is proved that when  $j$  belongs to  $m$ ,  $\Delta\gamma_{j,m}(t) > 0$ , and  $\Delta\gamma_{j,m} < 0$  otherwise. It means that when target  $j$  belongs to UAV  $m$  in the  $k$ th ant clan or  $l$ th ant clan its membership pheromone will increase.  $\square$

Also, formula (23) indicates another advantage of membership pheromone updating strategy, making a finer search of GACO algorithm: when the membership pheromone is close to 0, it increases fast while decreases slowly.

#### 4.4 Procedure of GACO

The procedure of GACO algorithm for solving multi-UAV reconnaissance task allocation problem is as follows:

*Step 1* initialize the  $N_u \times N_{AG}$  ant colony, construct the ant subgroup  $AC_u$ ,  $u = 1, 2, \dots, N_{AC}$  and ant clan  $AG_v$ ,  $v = 1, 2, \dots, N_{AG}$ . Each  $AC_u$  maps to a UAV. Set the iteration counter  $c = 0$  and the maximum number of iterations  $c_{\max}$ .

*Step 2*  $\forall AG_v$ ,  $v = 1, 2, \dots, N_{AG}$ .

1.  $\forall Ant_{k,v}$ ,  $k = 1, 2, \dots, N_u$ , choose a target according to formula (13).
2. each ant in  $AG_v$  competes for extending according to formula (14).
  - If the extending point is a point target, put it in the tabu table.
  - If the extending point belongs to a line target, extend another point of the line target. Put the line target in the tabu table.
  - If the extending point belongs to area target, extend another point of the area target according to the sequence pheromone. Put the area target in the tabu table.
3. repeat (1) and (2) until all targets are extended.

*Step 3*  $\forall AG_v$ ,  $v = 1, 2, \dots, N_{AG}$  calculate the cost value. If the ant clan exceeds the autonomy of UAV, then remove the ant clan.

*Step 4* update the best ant clan in current iteration denoted as  $k$ th ant clan.

*Step 5* if the allocation plan constructed by the  $k$ th ant clan is better than the current global best allocation plan,  $k$ th ant clan is used as the new global best task allocation plan.

*Step 6* update sequence pheromone and membership pheromone according to formulas (15)–(21).

*Step 7*  $c = c + 1$ , if  $c > c_{\max}$ , break the algorithm and return the global best task allocation solution, else return step 2.

The pseudo-codes of GACO are shown in Algorithm 1.

**Algorithm1** GACO for multi-UAV task allocation**Begin**


---

1. initialize the  $N_u \times N_{AG}$  ant colony
2. construct the ant subgroup  $AC_u, u = 1, 2, \dots, N_{AC}$  and ant clan  $AG_v, v = 1, 2, \dots, N_{AG}$ .
3. set the iteration counter  $c = 0$  and the maximum number of iterations  $c_{max}$ .
4. **while**  $c < c_{max}$
5.   **for** each ant clan in  $AG_v, v = 1, 2, \dots, N_{AG}$
6.     **for** each ant in  $Ant_{k,v}, k = 1, 2, \dots, N_u$
7.       choose a target according to formula (13)
8.     **end for**
9.   each ant in  $AG_v$  competes for extending according to formula (14)
10.   **if** the extending point is a point target
11.     put it in the tabu table
12.   **else if** the extending point belongs to a line target
13.     extend another point of the line target.
14.     put the line target in the tabu table.
15.   **else** the extending point belongs to area target
16.     extend another point of the area target according to the sequence pheromone
17.     put the area target in the tabu table.
18.   **if** all targets in tabu table
19.     calculate the cost value
20.   **if** the ant clan exceed the autonomy of UAV
21.     remove the ant clan
22.   **end for**
23. find the best ant clan in current iteration denoted as  $k^{th}$  ant clan
24. **if**  $k^{th}$  ant clan is better than the global best ant clan denoted as  $l^{th}$  ant clan
25.   replace  $l^{th}$  ant clan with  $k^{th}$  ant clan
26. update sequence pheromone and membership pheromone according to formula (15)-(21).
27.  $c = c + 1$
28. **end while**
29. **return** the task allocation solution

**end**

---

## 5 Simulation experiments

In this section, the proposed multi-UAV task allocation algorithm is verified through simulation experiments, and it is compared with some other methods of multi-UAV task allocation problem through comparison tests. All simulations are coded by MATLAB in Windows 10 Enterprise Edition 64-Bits System, and the hardware environment is in a computer equipped with Intel(R) Core (TM) i5-7300HQ 2.50 GHz and 8 GB RAM. All the parameters

with physical units in the simulations are normalized. The task region is limited in a square area  $[0, 500] \times [0, 500]$ . The initial position of each UAV is (1,1), while the end position is (350,200). The reconnaissance width of each UAV is 2. Table 2 shows the parameters used throughout the experimental phase.

The scale factor  $\varepsilon$  is set to be 30; the weight factors  $w_1$  and  $w_2$  of subobjectives are both set to be 0.5. The initialization of sequence pheromone  $\tau$  and membership pheromone  $\gamma$  is  $\tau_{ij} = 1$  ( $i = 1, \dots, N_p, j = 1, \dots, N_p$ ),



**Table 2** Parameters in GACO

Parameter	Formula	Value
Scale factor $\varepsilon$	(2)	30
Weight of time cost $w_1$	(2)	0.5
Weight of fuel cost $w_2$	(2)	0.5
Weight of sequence pheromone $\alpha$	(13)	3
Weight of heuristic information $\beta$	(13)	4
Evaporation coefficient $\rho$	(15)	0.05
Sequence pheromone increment $Q$	(18)	10
Membership pheromone increment $K$	(20)	0.1
Number of ant clans $N_{AG}$	Null	40
Number of iterations $c_{max}$	Null	100

**Table 3** Performance of UAVs

	UAV1	UAV2	UAV3
Autonomy	7.0	8.0	8.0
Cruising speed	300	340	360
Fuel consumption ratio	25	34	36

$\gamma_{j,m} = 0.5$  ( $j = 1, \dots, N_p$ ,  $m = 1, \dots, N_u$ ). Other parameters of the GACO algorithm are set empirically as follows: number of iterations  $c_{max} = 100$ ; number of ant clans  $N_{AG} = 40$ ; coefficients  $\alpha = 3$ ,  $\beta = 4$  that control the probability of an ant visiting a target node; pheromone evaporation coefficient  $\rho = 0.05$ ; coefficients  $Q = 10$ ,  $K = 0.1$  that control the pheromone increment.

## 5.1 Simulation I

Firstly, the effectiveness of the proposed method is verified. In this simulation scenario, there are 3 UAVs, 50 point targets, 3 line targets, and 2 area targets. The performance of each UAV is shown in Table 3.

The task allocation result of GACO is given in Fig. 6. The point indexes are from 1 to 66, and point indexes 1 and 66 represent initial position and end position of all UAVs, respectively. Area targets and line targets are indexed by their vertexes, area targets are represented by red solid squares, line targets are represented by red solid lines. It can be seen that all the targets are visited only once by UAVs, and the line targets and area targets are covered. It is in accord with the requirements of the targets reconnaissance. Meanwhile, the reconnaissance sequences of the assigned targets for the UAVs are in a good order. Thus, the GACO algorithm can obtain satisfying task allocation

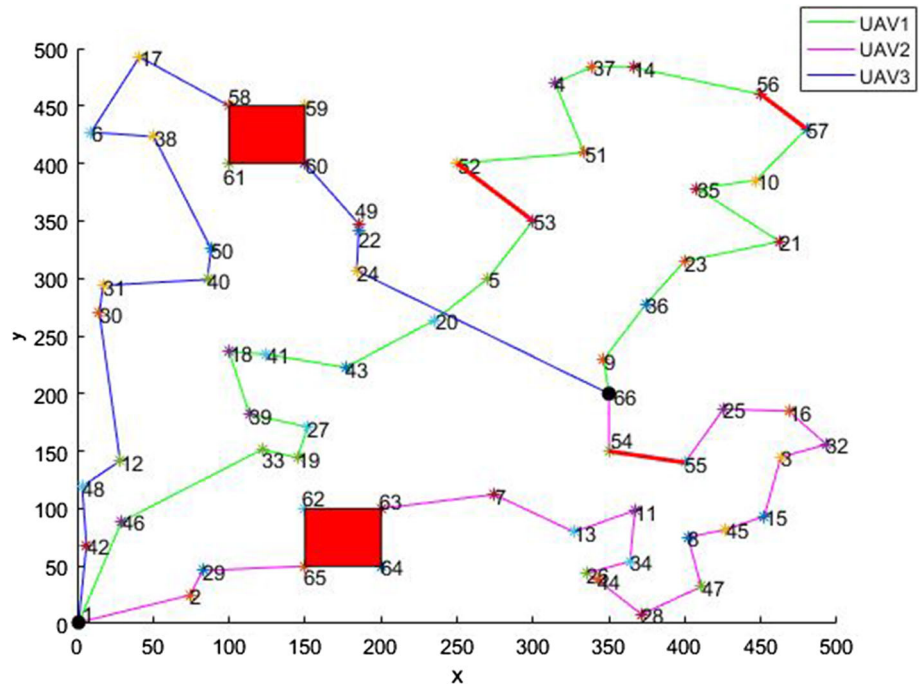
result for multi-UAV cooperative reconnaissance problem on heterogeneous targets. The total cost of task allocation result is 376.81. The detailed allocation results are provided in Table 4; it can be seen that all the UAVs meet their autonomy constraints. UAV1 reconnoiters the largest number of targets, its time cost is minimal, though its cruising speed is the smallest. This is because area targets occupy most of the time cost of UAV2 and UAV3, the time cost required to complete area targets of UAV2 is 3.77, while the time cost of UAV3 is 3.56. The result allocates area targets to UAVs with high cruising speed to save total task time. Although the time cost of UAV2 is larger than UAV3, its fuel consumption is less, due to the lower fuel consumption ratio. From the result, targets are appropriately assigned to different UAVs, and UAVs can cooperatively accomplish the reconnaissance task to reduce the task execution time.

Figure 7 is the convergence curve of objective function cost, the green curve represents the mean cost of ant subgroups in each iteration, and the red curve is the minimum cost up to each iteration. It can be seen that the curve of cost value converges after 39 iterations, and the ant cost decreases from 483.62 to 376.81, the reduction rate is 22.15%, thus, its convergence rate is satisfactory. Figure 8 is the membership pheromone convergence of four targets, and their indexes are corresponding to Fig. 6. Point indexes 13 and 12 represent point targets, point index 55 represents line target, and point index 63 represents area target. The green curves, pink curves, and blue curves represent the possibility that the targets belong to UAV1, UAV2, and UAV3 in each iteration, respectively. It can be seen that there is at least one membership pheromone rise and one drop in each iteration, which is due to the positive and negative feedback update mechanism of the membership pheromone. The iterations to reach convergence of membership pheromone are 20, 16, 18, and 13, which are smaller than the convergence iteration in Fig. 7, and it means that the membership pheromone should converge faster than the objective function.

## 5.2 Simulation II

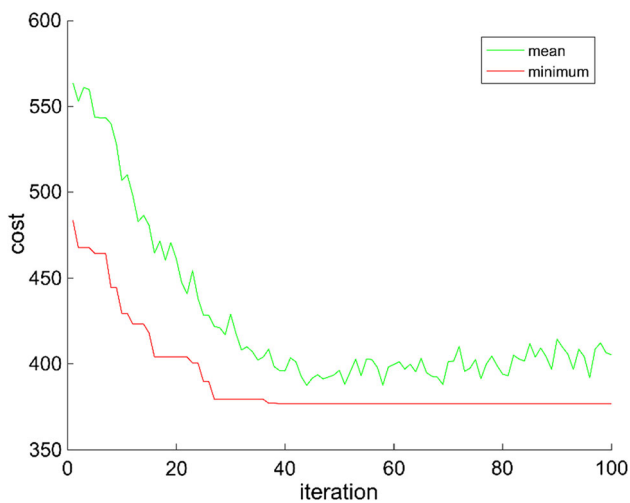
To verify the optimality of GACO in solving multi-UAV task allocation problem, firstly, we compare the GACO algorithm with the state-of-the-art ant colony algorithm, which is so-called ant colony-partheno genetic algorithm (AC-PGA) (Jiang et al. 2020). The initial parameters of AC-PGA are set as: the residual coefficient of the pheromone is 0.1; the importance of the pheromone and the visibility is 2 and 8; natural selection ratio is 0.5; size of population and iteration number are both 100. There are 40 point targets, 5 line targets, and 3 area targets in the test scenario. We run each algorithm 20 times and choose the

**Fig. 6** Task allocation result of GACO



**Table 4** Task allocation result

	Time	Number of point targets	Number of line targets	Number of area targets	Fuel consumption
UAV1	4.58	20	2	0	114.58
UAV2	6.43	17	1	1	218.62
UAV3	6.32	13	0	1	227.52



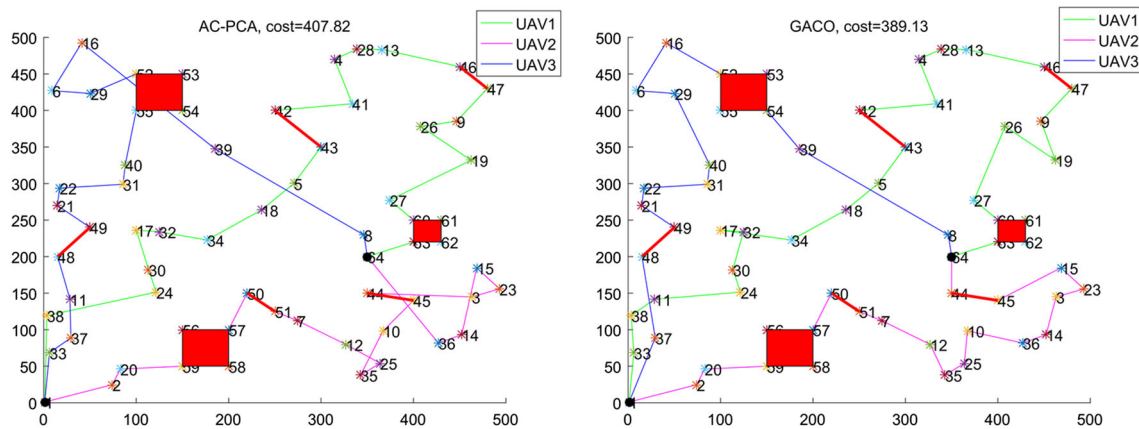
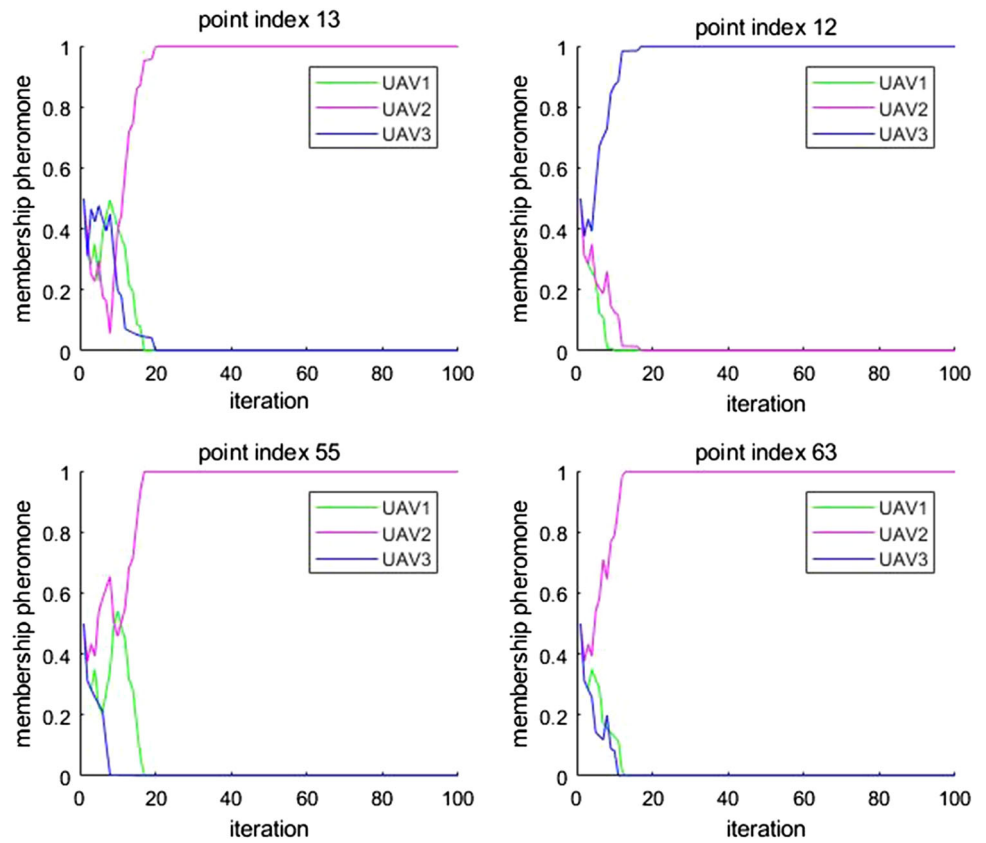
**Fig. 7** Convergence curve of sum cost

best result of each algorithm for comparison. The test solution expressed as path maps is given in Fig. 9. Paths of the UAVs obtained by the AC-PGA are more likely to cross together and cause collision than paths solved by the

GACO. Thus, the better solution can be achieved by GACO. In Fig. 10, the convergence ability of AC-PGA and GACO is shown. Although the initial cost of GACO is larger than that of AC-PGA, it converges to a lower value in less iterations. Thus, the new algorithm has better searching ability for solving this problem.

We compare the GACO with AC-PGA and OGA-DEMMO under different scenarios, and the OGA-DEMMO is the state-of-the-art evolutionary algorithm used for solving multiple UAV task allocation problem (Zhu et al. 2018). The initial parameters of OGA-DEMMO are set as: probabilities of crossover and mutation are 0.9 and 0.1; population size is 50. The initial parameters of GACO and AC-PGA are the same as the former part. All algorithms terminate after 100 iterations. Table 5 lists the information of these scenarios in detail. The scale of the problem is increasing because of not only the increase of UAV numbers but also the increase of targets. In this simulation, each algorithm runs 20 times for each scenario to get statistical results.

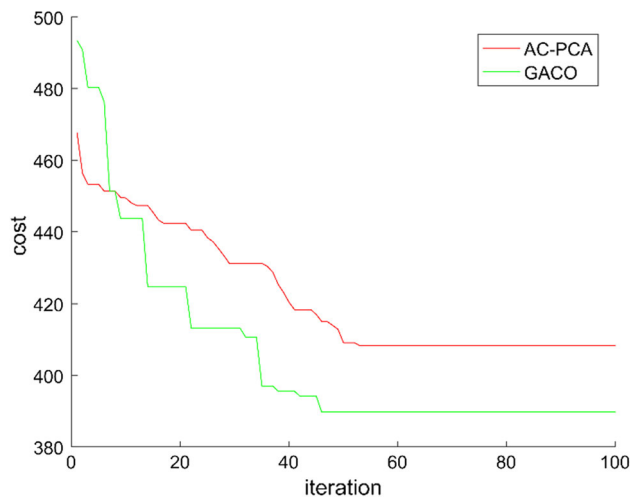
**Fig. 8** Convergence curve of membership pheromone



**Fig. 9** Solutions of AC-PCA and GACO

Figure 11 shows the result of 20 tests for different algorithms in different scenarios. The histogram is drawn based on the average cost value of 20 tests, and the error bar shows the maximum and minimum cost value in 20 tests. It is noted that the number in the histogram is the minimum cost value in 20 tests of each algorithm. According to the minimum cost values in 20 tests, with the increasing of problem scale, GACO can get best cost value for all scenarios among all these algorithms. While the

OGA-DEMMO can get best cost value in scenario I and scenario II, AC-PCA can only get best cost value in scenario I, but its error bar is always the shortest, which means that AC-PCA is more stable. Also, GACO can get best average cost value in all scenarios, and the gap continues to widen with the expansion of the scale of the problem. Thus, GACO is better than AC-PCA and OGA-DEMMO from the optimality of the results, especially for large-scale allocation problems.



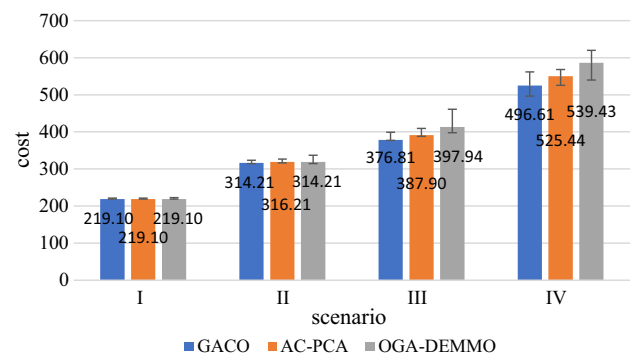
**Fig. 10** Convergence curve of AC-PCA and GACO

**Table 5** Scenarios information

Scenario ID	$N_u$	Number of point targets	Number of line targets	Number of area targets
I	2	5	2	1
II	3	10	3	2
III	3	50	3	2
IV	4	60	8	5

## 6 Conclusion

In this study, a novel multi-UAV reconnaissance task allocation model is proposed, targets are classified into point targets, line targets and area targets according to their geometric characteristics, the line targets and area targets are represented as two symbiotic point targets and four mutually exclusive point targets, respectively. The optimization objective is to minimize the weighted sum of the total UAV consumption and the task execution time. The GACO algorithm is proposed in order to solve the problem. To improve the optimality and convergence efficiency, it introduces the membership pheromone to learn correlation between UAVs and targets, and the negative feedback mechanism is proposed in the pheromone update process. The GACO algorithm has been evaluated and compared with ACO and OGA-DEMMO algorithm by numerical experiments of different scale scenarios. Results demonstrate that the GACO algorithm outperforms ACO and OGA-DEMMO algorithm in terms of global exploration capability performance, especially for large-scale task allocation scenarios.



**Fig. 11** Comparison results in different scenarios

**Author contributions** All authors contributed to the study conception and design, and all authors read and approved the final manuscript.

**Funding** No funding was received for conducting this study.

**Availability of data and material** All data used during the study are available from the corresponding author by request.

**Code availability** The raw code cannot be shared at this time as the code also forms part of an ongoing study.

## Compliance with ethical standards

**Conflict of interest** The authors, Sheng GAO, Jiazheng WU, and Jianliang AI, declare that they have no conflict of interest.

**Ethical approval** This article does not contain any studies with human participants or animals performed by any of the authors.

## References

- Albornoz C, Giraldo LF (2017) Trajectory design for efficient crop irrigation with a UAV. In: 2017 IEEE 3rd Colombian conference on automatic control (CCAC). IEEE, pp 1–6
- Alighanbari M (2004) Task assignment algorithms for teams of UAVs in dynamic environments. Massachusetts Institute of Technology
- Deb K (2014) Multi-objective optimization. In: Search methodologies. Springer, Boston, MA, pp 403–449
- Deng Q, Yu J, Wang N (2013) Cooperative task assignment of multiple heterogeneous unmanned aerial vehicles using a modified genetic algorithm with multi-type genes. Chin J Aeronaut 26(5):1238–1250
- Dokeroglu T, Sevinc E, Cosar A (2019) Artificial bee colony optimization for the quadratic assignment problem. Appl Soft Comput 76:595–606
- Fei S, Yan C, Lin-Cheng S (2008) UAV cooperative multi-task assignment based on ant colony algorithm. Acta Aeronaut Astronaut Sin 29:188–199
- Fu Z, Mao Y, He D et al (2019) Secure multi-UAV collaborative task allocation. IEEE Access 7:35579–35587
- Hadi GS, Varianto R, Trilaksono B et al (2014) Autonomous UAV system development for payload dropping mission. J Instrum Autom Syst 1(2):72–22
- Hoffmann G, Huang H, Waslander S et al (2007) Quadrotor helicopter flight dynamics and control: theory and experiment.

- In: AIAA guidance, navigation and control conference and exhibit. pp 6461–6481
- Jiang C, Wan Z, Peng Z (2020) A new efficient hybrid algorithm for large scale multiple traveling salesman problems. *Expert Syst Appl* 139:112867
- Junjie P, Dingwei W (2006) An ant colony optimization algorithm for multiple travelling salesman problem. In: First international conference on innovative computing, information and control-volume I (ICICIC'06), vol 1. IEEE, pp 210–213
- Kona H, Burde A, Zanwar DR (2015) A review of traveling salesman problem with time window constraint. *IJRST Int J Innov Res Sci Technol* 2:253–256
- Luo C, Miao W, Ullah H et al (2019) Unmanned aerial vehicles for disaster management. Geological disaster monitoring based on sensor networks. Springer, Singapore, pp 83–107
- Mafarja MM, Mirjalili S (2017) Hybrid whale optimization algorithm with simulated annealing for feature selection. *Neurocomputing* 260:302–312
- Merino L, Caballero F, Martínez-de Dios JR et al (2006) A cooperative perception system for multiple UAVs: application to automatic detection of forest fires. *J Field Robot* 23(3–4):165–184
- Nygaard KE, Chandler PR, Pachter M (2001) Dynamic network flow optimization models for air vehicle resource allocation. In: Proceedings of the 2001 American control conference (Cat. No. 01CH37148), vol 3. IEEE, pp 1853–1858
- O'Rourke KP, Carlton WB, Bailey TG et al (2001) Dynamic routing of unmanned aerial vehicles using reactive tabu search. *Mil Oper Res* 6:5–30
- Pandiri V, Singh A (2018) A hyper-heuristic based artificial bee colony algorithm for k-Interconnected multi-depot multi-traveling salesman problem. *Inf Sci* 463:261–281
- Pendharkar PC (2015) An ant colony optimization heuristic for constrained task allocation problem. *J Comput Sci* 7:37–47
- Ramirez-Atencia C, Bello-Orgaz G, R-Moreno MD et al (2017) Solving complex multi-UAV mission planning problems using multi-objective genetic algorithms. *Soft Comput* 21(17):4883–4900
- Somhom S, Modares A, Enkawa T (1999) Competition-based neural network for the multiple travelling salesmen problem with minmax objective. *Comput Oper Res* 26(4):395–407
- Srikanth GU, Geetha R (2018) Task scheduling using ant colony optimization in multicore architectures: a survey. *Soft Comput* 22(15):5179–5196
- Vincent T, Seipp F, Ruzika S et al (2013) Multiple objective branch and bound for mixed 0–1 linear programming: corrections and improvements for the biobjective case. *Comput Oper Res* 40(1):498–509
- Wang J, Jiang C, Han Z et al (2017) Taking drones to the next level: cooperative distributed unmanned-aerial-vehicular networks for small and mini drones. *IEEE Veh Technol Mag* 12(3):73–82
- Wang J, Jiang C, Zhang H et al (2020) Thirty years of machine learning: the road to pareto-optimal wireless networks. *IEEE Commun Surv Tutor* 22:1472–1514
- Wolpert DH, Macready WG (1997) No free lunch theorems for optimization. *IEEE Trans Evol Comput* 1(1):67–82
- Yang T, Jiang Z, Sun R et al (2020) Maritime search and rescue based on group mobile computing for UAVs and USVs. In: IEEE transactions on industrial informatics, pp 1
- Yousefikhoshbakht M, Didehvar F, Rahmati F (2013) Modification of the ant colony optimization for solving the multiple traveling salesman problem. *Rom J Inf Sci Technol* 16(1):65–80
- Yuan S, Skinner B, Huang S et al (2013) A new crossover approach for solving the multiple travelling salesmen problem using genetic algorithms. *Eur J Oper Res* 228(1):72–82
- Zhang W, Hu Y, He H et al (2019) Linear and dynamic programming algorithms for real-time task scheduling with task duplication. *J Supercomput* 75(2):494–509
- Zhu W, Li LIU, Teng L et al (2018) Multi-UAV reconnaissance task allocation for heterogeneous targets using an opposition-based genetic algorithm with double-chromosome encoding. *Chin J Aeronaut* 31(2):339–350

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.