Hindawi
International Journal of Aerospace Engineering
Volume 2019, Article ID 3504248, 12 pages
https://doi.org/10.1155/2019/3504248

Research Article

A Two-Layer Task Assignment Algorithm for UAV Swarm Based on Feature Weight Clustering

Xiaowei Fu,[1,2] Peng Feng,[1] Bin Li,[2,3] and Xiaoguang Gao[1]

[1] *School of Electronics and Information, Northwestern Polytechnical University, Xi'an 710129, China*
[2] *Shaanxi Key Laboratory of Integrated and Intelligent Navigation, Xi'an 710068, China*
[3] *The 20th Research Institute of CETC, Xi'an 710068, China*

Correspondence should be addressed to Xiaowei Fu; fxw@nwpu.edu.cn

Received 13 March 2019; Accepted 3 September 2019; Published 26 November 2019

Academic Editor: Jeremy Straub

For the large-scale operations of unmanned aerial vehicle (UAV) swarm and the large number of UAVs, this paper proposes a two-layer task and resource assignment algorithm based on feature weight clustering. According to the numbers and types of task resources of each UAV and the distances between different UAVs, the UAV swarm is divided into multiple UAV clusters, and the large-scale allocation problem is transformed into several related small-scale problems. A two-layer task assignment algorithm based on the consensus-based bundle algorithm (CBBA) is proposed, and this algorithm uses different consensus rules between clusters and within clusters, which ensures that the UAV swarm gets a conflict-free task assignment solution in real time. The simulation results show that the algorithm can assign tasks effectively and efficiently when the number of UAVs and targets is large.

1. Introduction

UAV swarm consists of a large number of small UAVs [1], and the cooperative task and resource assignment of UAV swarm is to real-time coordinate the UAV swarm in order to achieve an overall mission objective. A mission can be divided into different tasks, and a number of specialized UAVs are then assigned to solve each task cooperatively [2–4].

The CBBA algorithm is a kind of the distributed auction-based algorithms to resolve multiple agent task assignment problem [5], and it is a multiassignment decentralized auction approach with a consensus protocol that guarantees a conflict-free solution despite possible inconsistencies in situational awareness. An extension to CBBA [6] has enabled incorporation of heterogeneity in the UAV capabilities and task time windows, which significantly extends the mission characteristics that can be handled. Coupled CBBA is designed to create feasible assignments for a network of autonomous UAVs in the presence of the temporal coupling constraints [7], and temporal constraints include several specified relationships between the chosen visit times for a subset of tasks. The consensus phase of the CBBA algorithm relies on coordinated communication between all UAVs, which is achieved by propagating UAVs' bid information through the communication links. As the number of UAVs in the network increases, this consensus approach may overflow the network bandwidth. But in these works, the communication links between all UAVs have high bandwidth, low latency, low cost, and high reliability. However, the real communication links between UAVs do not possess all of these characteristics. Asynchronous CBBA extends CBBA to account for more realistic asynchronous communication protocols by minimizing communication load while preserving the convergence properties [8], and it produces consistent task assignments using relatively little bandwidth and without requiring artificial time delays. The comparisons between global and local convergence in asynchronous consensus algorithms are discussed in [9]. Bid warped consensus-based bundle algorithm deals with the task assignment problems that global information consistency assumptions are difficult to enforce [10], and it relies only on a local best estimate of the global information state, which is referred to as local information consistency. CBBA with partial replanning (CBBA-PR) extends the CBBA algorithm to allocate new appeared tasks quickly without a full reallocation of existing tasks [11], and it enables multi-UAV team to trade-off between convergence time and increased coordination by resetting a portion of their previous allocation at every round of bidding on tasks.

For the task assignment problem of the UAV swarm, the existing CBBA algorithm will need more communication times as the number of UAVs increases; moreover, each UAV of the swarm may have different capacity and number of resources, which are not considered in the existing CBBA algorithms.

Although the aforementioned works have made some improvements of the CBBA algorithm, they still cannot adapt to the characteristics of large number of UAVs in the UAV swarm. The computational complexity of task assignment is still a crucial problem in the use of the UAV swarm. The hierarchical method is a feasible approach to reduce the computation cost of complex optimization problems, which divides the problem into several levels of subproblems. Each level of subproblem has its own objectives and constraints. The output of one level becomes the input of the next level. By solving the subproblems on different levels in order, the original problem can be solved. Although this approach may miss the best solution, it can produce satisfactory solutions in much less time than other methods.

This paper proposes a two-layer task assignment algorithm based on feature weight clustering, which could decompose the large-scale task assignment problem of the UAV swarm effectively, and the efficiency of task assignment is greatly improved.

2. Task Assignment Model of the UAV Swarm

In a prior work [12], the task resource, task reward, and task assignment models are presented as follows.

2.1. Task Resource Model. Given a swarm of N heterogeneous UAVs $U=\{U_1, U_2, \cdots, U_N\}$ and a set of M targets $T=\{T_1, T_2, \cdots, T_M\}$, the UAV swarm is divided into two subsets $U=\{U_I, U_A\}$ according to the task resource type of each UAV; U_I is the set of electronic interfering UAVs and U_A is the set of attack UAVs. Each UAV belongs to only one of the two subsets, and it carries n kinds of task resources. The task resource vector of the attack UAV U_i is represented by $\mathbf{resSu}_i^A=\{\text{resSu}_{i,1}^A, \text{resSu}_{i,2}^A, \cdots, \text{resSu}_{i,n}^A\}$, where $\text{resSu}_{i,q}^A$, $q=1, \cdots, n$, indicates the quantity of q^{th} type of weapons carried by UAV U_i; and the task resource vector of the electronic interfering UAV U_j is represented by $\mathbf{resSu}_j^I=\left\{\text{resSu}_{j,1}^I, \quad \text{resSu}_{j,2}^I, \quad \cdots, \quad \text{resSu}_{j,m}^I\right\}$, where $\text{resSu}_{j,q}^I$, $q=1, \cdots, m$, indicates the quantity of q^{th} type of electronic interfering resources carried by UAV U_j.

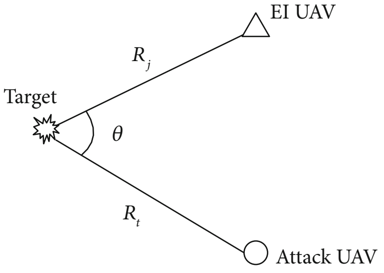


FIGURE 1: Schematic diagram of electronic interference process [12].

To attack target T_j, the required type and quantity of weapons are $\mathbf{resRe}_j^A=\left\{\text{res Re}_{j,1}^A, \quad \text{res Re}_{j,2}^A, \quad \cdots, \quad \text{res Re}_{j,n}^A\right\}$, where $\text{res Re}_{j,q}^A$, $q=1, \cdots, n$, indicates the required quantity of q^{th} type of weapons to attack target T_j. To interfere target T_j, the required type and quantity of electronic interfering resources are $\mathbf{resRe}_j^I=\left\{\text{res Re}_{j,1}^I, \quad \text{res Re}_{j,2}^I, \quad \cdots, \quad \text{res Re}_{j,m}^I\right\}$, where $\text{res Re}_{j,q}^I$, $q=1, \cdots, m$, indicates the required quantity of q^{th} type of resources to interfere target T_j.

To interfere and attack target T_x, the total amount of task resources carried by UAVs must meet the task requirements.

$$\begin{aligned} \sum_{U_i \in I_A} x_{i,k}^A \cdot \text{resSu}_{i,k}^A \geq \text{res Re}_{x,k}^A, \quad \forall k \in \{1, 2, \cdots, n\}, \\ \sum_{U_i \in I_A} x_{i,k}^I \cdot \text{resSu}_{i,k}^I \geq \text{res Re}_{x,k}^I, \quad \forall k \in \{1, 2, \cdots, n\}, \end{aligned} \tag{1}$$

where $x_{i,k}^A$ and $x_{i,k}^I$, respectively, indicate whether the kth weapon resource or interference payload of U_i is used to the tasks on the target; $\text{resSu}_{i,k}^A$ and $\text{resSu}_{i,k}^I$ indicate the number of the kth weapon resources or interference payload of U_i, respectively.

2.2. Task Reward Model

Definition 1 (the initial reward of attack task). Suppose the damage probability of U_i to one target is $p_{i,a}$, the initial reward of attack task is defined as

$$G_{i,j}^A = V_j p_{i,a} - D_j, \tag{2}$$

where V_j and D_j are the value and threat level of target T_j.

The attack task reduces the threat level of the target; therefore, the threat level of target T_j that has been attacked is

$$D_j^* = (1 - p_{i,a}) D_j. \tag{3}$$

Definition 2 (the initial reward of electronic interference task). As shown in Figure 1, the electronic interference task is related to the attack task, and it should be performed before attack task a certain period of time. The UAV swarm assigns an attack task to a target at first and then estimates the start time of electronic interference task according to the time of the attack task.

The maximum interference distance R_t is a function of R_j and the angle θ, which is

$$R_t = f\left(R_j, \theta\right). \tag{4}$$

The reward of the electronic interference task is defined as

$$G_{i,j}^I = \begin{cases} 0, & R^a < R_t, \\ \delta G_{i,j}^A, & R^a \geq R_t, \end{cases} \tag{5}$$

where R_t indicates the maximum interference distance of UAV U_i and R^a represents the distance between the UAV U_i and the target T_j.

2.3. Task Assignment Model. The task and resource assignment model of the UAV swarm is described as

$$\begin{aligned} \max \quad & \left\{ J = \sum_{i=1}^{N} \frac{\sum_{j=1}^{M} X_{i,j} \cdot C_{i,j}(\mathbf{p}_i)}{\sum_{i=1}^{N} X_{i,j} \cdot \mathrm{Len}_i(\mathbf{p}_i)} \right\} \\ \text{s.t.} \quad & \sum_{j=1}^{M} X_{i,j} \leq L, \quad \forall U_i \in U \\ & X_{i,j}^k \cdot \mathrm{resSu}_{i,j}^k \leq \mathrm{resSu}_i^k \\ & \sum_{i=1}^{N} X_{i,j}^k \cdot \mathrm{resSu}_{i,j}^k \geq \mathrm{res\ Re}_j^k \\ & X_{i,j} \in \{0, 1\}, \quad \forall (i,j) \in U \times T, \end{aligned} \tag{6}$$

where $X_{i,j}$ denotes whether UAV U_i is assigned to target j. L represents the maximum number of tasks of each UAV. Vector $\mathbf{p}_i \in (T \cup \{\varnothing\})^L$ represents an ordered sequence of task path of U_i. $\mathrm{Len}_i(\mathbf{p}_i)$ indicates the length of the path that the UAV performs the current task sequence. The fractional function $C_{i,j}(\mathbf{p}_i)$ represents the total task reward that is calculated as described in equations (2) and (5). $X_{i,j}^k$ denotes whether UAV U_i with kth resource is assigned to target j. $\mathrm{resSu}_{i,j}^k$ indicates the number of the kth resources of U_i that is assigned to target j, and resSu_i^k indicates the number of the kth resources of U_i. $\mathrm{res\ Re}_j^k$ indicates the required number of the kth resources for the tasks on target j.

3. UAV Clustering Based on Distance and Task Resources

3.1. Mathematical Model of Cluster Analysis. Cluster analysis is the process of partitioning a set of data objects into subsets. Each subset is a cluster, such that objects in a cluster are similar to one another yet dissimilar to objects in other clusters [13].

Given a set of data points $S = \{X_1, X_2, \cdots, X_n\}$ in R^m space, the position with the smallest distance from point X_j $(j = 1, 2, \cdots, n)$ to the k positions $(V_1, V_2, \cdots, V_k)$ is called the nearest position of the point X_j. The distance measure to the nearest position is recorded as $d_j = \min \sum_{i=1,\cdots,k} d_{ji}$, where d_{ji} is called as the distance measure from point X_j to position V_i. The sum of the minimum distance metrics for n points is $\sum_{j=1}^{n} d_j$.

According to the above description, the clustering problem can be converted into an optimization problem, which is described as

Input. n data points in R^m;

Output. k positions in R^m so that the sum of the minimum distance measures of n points is minimized.

3.2. k-Medoids Clustering Algorithm. Generally, compared to k-means clustering algorithm, k-medoids clustering algorithm showed its superiority in execution time, sensitivity towards outlier data, and reduction of noise since it employs the method of minimization of the sum of dissimilarities of datasets [14]. Although there are many enhanced k-means algorithms [15, 16], we still choose the k-medoids algorithm for clustering from the aspects of computational complexity and algorithm efficiency.

The optimization objective function of the k-medoids clustering algorithm can be generally defined as follows:

$$E = \sum_{i=1}^{k} \sum_{X \in C_i} \|X - V_i\|_p, \tag{7}$$

where E represents the sum of the deviations between each data point in the data point set and its cluster center point; X represents the points in R^m; V_i represents the center point of cluster C_i (both X and V_i are m-dimensional); $\|X - V_i\|_p$ represents the p-order metric between X and V_i, typically the square of the Euclidean distance in the distance space $(p = 2)$ is used.

The k-medoids clustering algorithm is described as [17]

Input. The number of clusters k, a set of data points containing n points

Output. k clusters and the subset of data points they contain.

Step 1. From the data point set, k data points are randomly selected to form a current cluster center point set. Each of the k data point represents the initial center point of a cluster.

Step 2. Calculate the objective function value according to equation (10) and assign all data points to the cluster represented by the nearest center point.

Step 3. For each point $X_j (j = 1, 2, \cdots, n)$ of the data point set, the following procedure is performed. Try to replace each existing center point $V_i (i = 1, 2, \cdots, k)$ with the current point X_j, and calculate the objective function value according to equation (10). Compare the objective function values of all candidate alternatives, and the center point of the cluster with the minimum value is replaced by X_j.

Step 4. Finally, the k clusters and their center points are obtained; the optimized clustering results of the data points contained therein are collected.

3.3. Clustering Model Based on Feature Weight. In the cluster analysis, in order to reflect the different effects of each attribute pair on forming a good structural clustering subset and to describe the difference between point X_i and point X_j, a weighted "distance" metric can be defined (Norm), i.e., [18]

$$D(X_i, X_j) = \left(\sum_{k=1}^{m} \left(w_k \cdot d_k^2 (x_{ik}, x_{jk}) \right) \right)^{1/2}, \tag{8}$$

where the specific definition of kth attribute is determined according to its characteristics. If the kth attribute is an unordered category attribute, it is defined as

$$d_k(x_{ik}, x_{jk}) = \begin{cases} 0, & x_{ik} = x_{jk}, \\ 1, & x_{ik} \neq x_{jk}. \end{cases} \tag{9}$$

If the kth attribute is an ordered attribute, it is defined as

$$d_k(x_{ik}, x_{jk}) = |x_{ik} - x_{jk}|. \tag{10}$$

For the feature weight assignment problem, the optimal assignment of feature weight parameters $W = (w_1, w_2, \cdots, w_m)^T$ should be determined by using the distribution of data point sets $S = \{X_1, X_2, \cdots, X_n\}$ and class attribution. Feature weights must meet the following constraints:

$$\begin{cases} \sum_{j=1}^{m} w_j = m, \\ 0 \leq w_j \leq m, \quad j = 1, 2, \cdots, m. \end{cases} \tag{11}$$

The optimization objective function of the algorithm is

$$P(L, V, W) = \sum_{i=1}^{n} \sum_{j=1}^{k} \sum_{s=1}^{m} \left(l_{ij} \cdot w_s \cdot d_s^2 (x_{\text{is}}, v_{js}) \right), \tag{12}$$

where L represents a hard-divided membership matrix; V represents a cluster center point set; W represents feature weight parameters. When the data point X_i belongs to a cluster with a cluster center V_j, $l_{ij} = 1$; otherwise, $l_{ij} = 0$.

3.4. UAV Clustering Based on Distance and Task Resources. To measure the similarity between UAVs, two attributes are used to build the objective function model. In addition to the distance attribute, the balance of task resource of each UAV cluster is also taken as the optimization objective. The UAV swarm has n types of different task resources, and the UAV resource vector is expressed as

$$\mathbf{resSu} = \{\text{resSu}_1, \text{resSu}_2, \cdots, \text{resSu}_n\}. \tag{13}$$

All task resources in UAV cluster C_j is described as

$$\mathbf{resSu}^j = \left\{ \text{resSu}_1^j, \text{resSu}_2^j, \cdots, \text{resSu}_n^j \right\}. \tag{14}$$

Then, the balance of task resource of UAV cluster C_j is evaluated by the variance of various task resources:

$$\text{Balance\_resSu}^j = \sqrt{\frac{\sum_{k=1}^{n} \left(\text{resSu}_k^j - \overline{\text{resSu}^j} \right)}{n}}, \tag{15}$$

where

$$\overline{\text{resSu}^j} = \frac{\sum_{k=1}^{n} \text{resSu}_k^j}{n}. \tag{16}$$

In addition, the distance measurement of UAVs is

$$d_{i,j} = \sqrt{\left(x_i - x_v^j \right)^2 + \left(y_i - y_v^j \right)^2 + \left(z_i - z_v^j \right)^2}, \tag{17}$$

where (x_i, y_i, z_i) is the position of U_i and (x_v^j, y_v^j, z_v^j) is the position of center point V_j of UAV cluster C_j.

In summary, the final objective function is

$$P = w_1 \cdot \sum_{i=1}^{n} \sum_{j=1}^{k} \left(u_{ij} \cdot d_{i,j} \right) + w_2 \cdot \sum_{j=1}^{k} \text{Balance\_resSu}^j, \tag{18}$$

where w_1 is the feature weight of the distance metric, and w_2 is the feature weight of the balance of task resource. Choosing a reasonable feature weight matrix can ensure that the task resources in the UAV cluster are relatively balanced on the basis of the smallest distance.

4. Two-Layer Task Assignment Algorithm Based on Feature Weight Clustering

4.1. Algorithm Description. According to the objective function (equation (18)), the UAV swarm is partitioned into p UAV clusters represented by $M = \{M_1, M_2, \cdots, M_p\}$. For UAV cluster M_t, $U_t = \{U_{t,1}, U_{t,2}, \cdots, U_{t,n}\} \subseteq U$.

Each UAV cluster initially assigns tasks of the target set T based on the CBBA algorithm, and $\mathbf{B}_t$, $\mathbf{Z}_t$, and $\mathbf{Y}_t$ represent the task bundle, the list of winning UAVs, and the list of winning scores of UAV cluster M_t, respectively. The consensus rule between UAV clusters is shown as Table 1.

In Table 1, z_{kj}^c and z_{ij}^c represent the winning cluster that should be assigned to target T_j from the view of the sender cluster M_k and the receiver cluster M_i, respectively.

According to the results of the consensus between UAV clusters, UAVs in the same cluster will negotiate to get a conflict-free task assignment solution. In this process, the successful bidders of some tasks may come from other clusters, so the consensus rules need to be modified, as shown in Table 2, z_{kj} and z_{ij} represent the winning UAV that should

Table 1: The consensus rules between UAV clusters.

	Cluster M_k (sender) thinks z^c_{kj} is	Cluster M_i (receiver) thinks z^c_{ij} is	Receiver's action (default: leave)
1	k	i	IF $y_{kj} > y_{ij}$, **Update**
2		k	**Update**
3		$m \notin \{i, k\}$	IF $y_{kj} > y_{ij}$ OR $s_{km} > s_{im}$, **Update**
4		$\varnothing$	**Update**
5	i	i	**Leave**
6		k	**Reset**
7		$m \notin \{i, k\}$	IF $s_{km} > s_{im}$, **Reset**
8		$\varnothing$	**Leave**
9	$m \notin \{i, k\}$	i	IF $y_{kj} > y_{ij}$ AND $s_{km} > s_{im}$, **Update**
10		k	IF $s_{km} > s_{im}$, **Update**
11			**Reset**
12		$m \notin \{i, k\}$	IF $s_{km} > s_{im}$, **Update**
13		$n \notin \{i, k, p\}$	IF $s_{km} > s_{im}$ AND $s_{kn} > s_{\text{in}}$, **Update**
14			IF $s_{km} > s_{im}$ AND $y_{kj} > y_{ij}$, **Update**
15			IF $s_{kn} > s_{\text{in}}$ AND $s_{km} > s_{im}$, **Reset**
16		$\varnothing$	IF $s_{km} > s_{im}$, **Update**
17	$\varnothing$	i	**Leave**
18		k	**Update**
19		$m \notin \{i, k\}$	IF $s_{km} > s_{im}$, **Update**
20		$\varnothing$	**Leave**

Table 2: The consensus rules between UAVs within a UAV cluster.

	UAV U_k (sender) thinks z_{kj} is	UAV U_i (receiver) thinks z_{ij} is	Receiver's action (default: leave)
1	k	i	IF $y_{kj} > y_{ij}$, **Update**
2		k	**Update**
3		$m \notin \{i, k\}$	IF $y_{kj} > y_{ij}$ OR $s_{km} > s_{im}$, **Update**
4		Other clusters	IF $y_{kj} > y_{ij}$, **Update**
5		$\varnothing$	**Update**
6	i	i	**Leave**
7		k	**Reset**
8		$m \notin \{i, k\}$	IF $s_{km} > s_{im}$, **Reset**
9		Other clusters	**Reset**
10		$\varnothing$	**Leave**
11	$m \notin \{i, k\}$	i	IF $y_{kj} > y_{ij}$ AND $s_{km} > s_{im}$, **Update**
12		k	IF $s_{km} > s_{im}$, **Update**
13			**Reset**
14		$m \notin \{i, k\}$	IF $s_{km} > s_{im}$, **Update**
15		$n \notin \{i, k, p\}$	IF $s_{km} > s_{im}$ AND $s_{kn} > s_{\text{in}}$, **Update**
17			IF $s_{km} > s_{im}$ AND $y_{kj} > y_{ij}$, **Update**
18			IF $s_{kn} > s_{\text{in}}$ AND $s_{km} > s_{im}$, **Reset**
19		Other clusters	IF $s_{km} > s_{im}$ AND $y_{kj} > y_{ij}$, **Update**
20		$\varnothing$	IF $s_{km} > s_{im}$, **Update**
21	Other clusters	i	IF $y_{kj} > y_{ij}$, **Update**
22		k	**Reset**
23		$m \notin \{i, k\}$	IF $s_{km} > s_{im}$, **Update**
24		Other clusters	IF $s_{km} > s_{im}$ AND $s_{kn} > s_{\text{in}}$, **Update**
25			IF $s_{km} > s_{im}$ AND $y_{kj} > y_{ij}$, **Update**
26			IF $s_{kn} > s_{\text{in}}$ AND $s_{km} > s_{im}$, **Reset**
27	$\varnothing$	i	**Leave**
28		k	**Update**
29		$m \notin \{i, k\}$	IF $s_{km} > s_{im}$, **Update**
30		Other clusters	**Update**
31		$\varnothing$	**Leave**

be assigned to target T_j from the view of the sender UAV U_k and the receiver UAV U_i, respectively.

4.2. Fault Tolerance Analysis. One of the advantages of the UAV swarm is its high fault tolerance. This algorithm can successfully assign tasks for a certain UAV that cannot communicate with others at a certain time.

It is assumed that a UAV U_i cannot communicate with others at a certain time. If the task assignment procedure has not been triggered, the tasks will be assigned between UAVs that can communicate with each other after the procedure triggered. The proposed algorithm does not need to know the number of UAVs prior to the procedure. If U_i has been assigned some tasks, maybe it will affect the effectiveness of the UAV swarm. The UAV swarm will redistribute the remaining tasks as needed after tasks are finished.

4.3. Performance Analysis. Compared with the baseline CBBA algorithm, the proposed two-layer algorithm can significantly reduce the number of communication times and the corresponding iteration round required to achieve task consensus, thus improving the computational efficiency. This performance improvement is mainly due to the communication mode in consensus process of the two-layer structure.

In the consensus of the baseline CBBA algorithm, each UAV must send local task assignment information to all other UAVs while receiving task assignment information from them. Figure 2 shows the communication mode of U_1

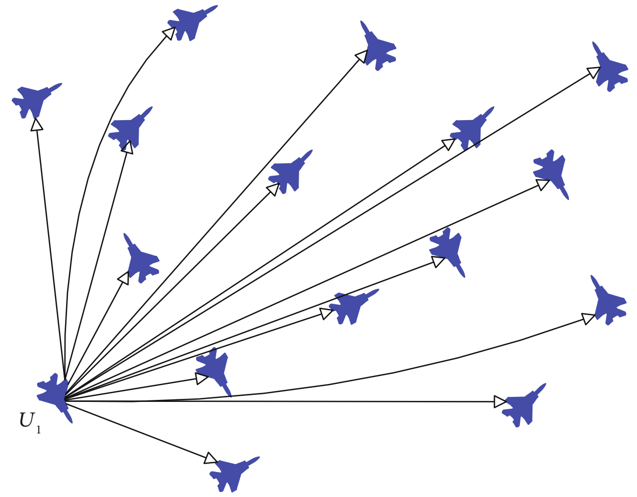


Figure 2: The communication mode in the consensus process in the baseline CBBA algorithm.

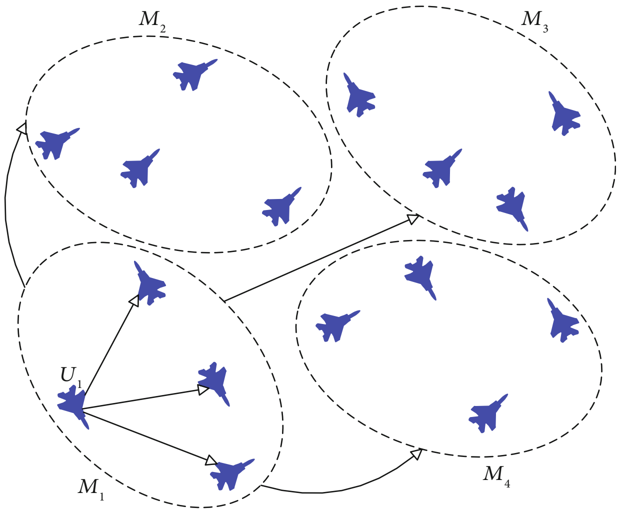


Figure 3: The communication mode in the consensus process in the two-layer algorithm.

in the consensus process in the baseline CBBA algorithm. U_1 sends its local task assignment information to other drones in the UAV swarm and updates the local information according to the consensus rules after receiving the information from other UAVs.

The proposed two-layer algorithm divides the UAV swarm into a two-layer structure, which separates the consensus process into the consensus process between clusters and within each cluster, and reduces the complexity of consensus. As shown in Figure 3, U_1 that belongs to cluster M_1 only needs to transmit the local task assignment information to the other 3 UAVs in cluster M_1 to achieve local consensus within the cluster. Cluster M_1 then communicates with other UAV clusters, which in turn achieves global task assignment consensus.

Extending the above comparison to all UAVs among the UAV swarm, it can be seen from this comparison that the number of communication times to achieve task consensus is significantly reduced. Through the two-layer structure, the complexity of the consensus process is significantly reduced, thereby improving the computational efficiency.

Table 3: The initial state of attack UAVs.

U_i	Initial position $(x_i, y_i)/m$	Vector of weapons resSu_i^A
U_1	(3063, 6368)	(3, 0, 5)
U_2	(5983, 8653)	(4, 2, 0)
U_3	(6345, 2777)	(3, 1, 0)
U_4	(890, 8834)	(0, 1, 3)
U_5	(1498, 3374)	(3, 3, 0)
U_6	(5404, 1473)	(0, 2, 3)

Table 4: The initial state of electronic interference UAVs.

U_i	Initial position $(x_i, y_i)/m$	Vector of interference resSu_i^I
U_7	(3577, 5476)	(3, 2, 0)
U_8	(3280, 2221)	(2, 0, 3)
U_9	(4873, 5475)	(0, 2, 4)
U_{10}	(8436, 5592)	(3, 4, 0)
U_{11}	(4903, 8890)	(3, 3, 0)
U_{12}	(6826, 8614)	(0, 1, 4)

5. Simulation

In order to prove the validity and efficiency of the two-layer task assignment algorithm for the UAV swarm with feature weight clustering, several sets of simulation experiments were designed and compared with the CBBA algorithm in [5].

In this study, the energy consumption is not considered, after the mission objective is achieved, and the operator in the ground control station will send return to base command to the swarm. We assumed that each UAV flies autonomous and can communicate information with others of the UAV swarm as necessary.

5.1. Algorithm Validation. Six attack UAVs are marked U_1-U_6, 6 electronic interference UAVs are marked U_7-U_{12}, and 6 targets are in a $10\,\text{km}^*10\,\text{km}$ rectangular area. The speed of each UAV is 50 m/s, and the maximum detection distance is 300 m. Assume that the UAV swarm is set to form four equal-sized UAV clusters.

The initial positions and task resource vectors of all UAVs and targets are generated in a random manner. Each UAV has three kinds of task resources, that is, attack UAVs have three kinds of weapons, and electronic interference UAVs have three kinds of interference payloads. Correspondingly, each target's attack task and electronic interference task also require three kinds of task resources. The initial states of the attack UAVs and the electronic interference UAVs are shown in Tables 3 and 4, respectively. The initial state of the target is shown in Table 5.

Table 5: The initial state of targets.

T_j	Initial position $\left(x_j, y_j\right)/m$	Requirement for attacking res Re_j^A	Requirement for interfering res Re_j^I
T_1	(4191, 6167)	(1, 3, 2)	(1, 1, 1)
T_2	(5675, 7621)	(1, 0, 2)	(1, 1, 2)
T_3	(3084, 3843)	(1, 2, 2)	(1, 2, 1)
T_4	(7194, 5519)	(2, 1, 1)	(2, 1, 0)
T_5	(3358, 8378)	(2, 1, 1)	(1, 1, 2)
T_6	(5138, 3959)	(1, 1, 1)	(1, 2, 1)

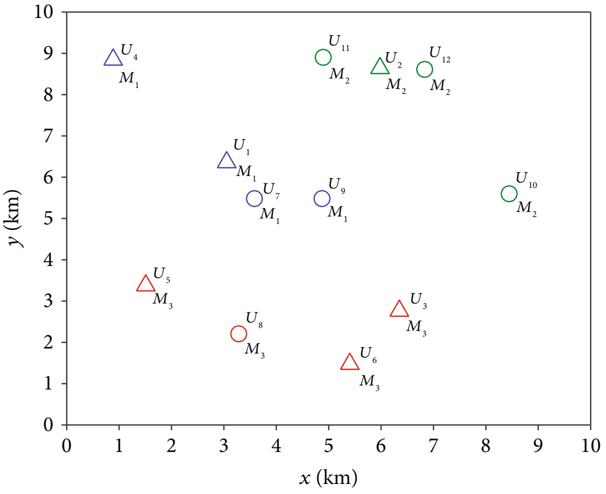


Figure 4: The clustering result obtained by normal cluster.

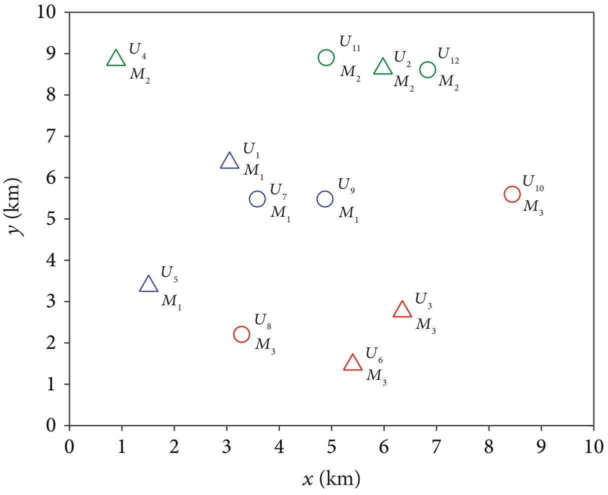


Figure 5: The clustering result obtained by the feature weight cluster.

5.1.1. Feature Weight Clustering Validity Verification. If the UAV swarm is clustered according to the distance measure, the clustering results obtained are shown in Figure 4. The clustering results based on the distance and task resource feature weights proposed in this paper are shown in Figure 5. In these figures, U_i denotes UAV ID, and M_i indicates the cluster ID to which the UAV belongs. The attack UAVs are denoted by a triangle, and the electronic interference UAVs are denoted by a circle. Each UAV cluster is denoted by a different color. Table 6 shows the comparison of cluster members and task resources between the two clustering results.

It can be seen from Figure 4 and Table 6 that the task resources of the clusters that are partitioned according to distance are completely unbalanced. For example, the weapon resource of cluster M_1 is seriously unbalanced; cluster M_2 contains only one attack UAV and there is no third type of weapon resource; M_3 contains only one interference UAV and there is no second type of interference payload. This unbalance will seriously affect the task resource assignment process and greatly increase the difficulty of achieving consensus distribution results.

It can be seen from Figure 5 and Table 6 that the clusters that are partitioned according to the feature weight clustering not only maintain the relatively close spatial distance between the UAVs among clusters but also achieve the balance of task resources. All three clusters have all types of task resources, and the number of various task resources is relatively balanced.

5.1.2. Algorithm Effectiveness Verification. To verify the effectiveness of the proposed algorithm, the task and resource assignment results obtained by the basic CBBA algorithm of [5] and the two-layer task assignment algorithm proposed in this paper are compared in the same scenario and initial state (as described in Tables 3–5). Figures 6 and 7 show the task sequence of the UAV swarm obtained by the CBBA algorithm of [5] and the two-layer assignment algorithm proposed in this paper, respectively. In these figures, the horizontal axis represents the timeline, and 12 rows on the vertical axis represent the task sequence of 12 UAVs. On the time axis, different color cylinders are used to represent the time intervals of UAV's tasks.

From the comparison of the two figures, the following can be seen. (1) The basic CBBA algorithm cannot fully utilize the resources of all the UAVs in the swarm, and the task assignment is unbalanced. For example, U_1 and U_9 are assigned 4 tasks, while U_2 only has one task, and U_{12} is not assigned any task. The two-layer assignment algorithm makes full use of the resources of each UAV, and the task assignment solution is balanced.

(2) Due to the unbalanced results of CBBA task allocation, the task completion time of the UAV swarm will increase correspondingly. Because the two-layer assignment algorithm can make full use of the performance and resources of each UAV, the UAV swarm can complete the task earlier. Under the scenario, the task completion time of the CBBA algorithm is 106 s, while the two-layer assignment algorithm is only 84 s.

Task assignment results for the UAV swarm was shown in Table 7.

5.2. Performance Analysis of the Two-Layer Task Assignment Algorithm. In order to comprehensively explore and compare the performance of the two-layer task assignment algorithm

TABLE 6: The cluster member and task resource obtained by two kinds of clustering.

M_i	Normal cluster				Feature weight cluster			
	Attack UAVs	Weapons	EI UAVs	Interference	Attack UAVs	Weapons	EI UAVs	Interference
M_1	$\|U_1, U_4\|$	(3, 1, 8)	$\|U_7, U_9\|$	(3, 4, 4)	$\|U_1, U_5\|$	(6, 3, 5)	$\|U_7, U_9\|$	(3, 4, 4)
M_2	$\|U_2\|$	(4, 2, 0)	$\|U_{10}, U_{11}, U_{12}\|$	(6, 8, 4)	$\|U_2, U_4\|$	(4, 3, 3)	$\|U_{11}, U_{12}\|$	(3, 4, 4)
M_3	$\|U_3, U_5, U_6\|$	(6, 6, 3)	$\|U_8\|$	(2, 0, 3)	$\|U_3, U_6\|$	(3, 3, 3)	$\|U_8, U_{10}\|$	(5, 4, 3)

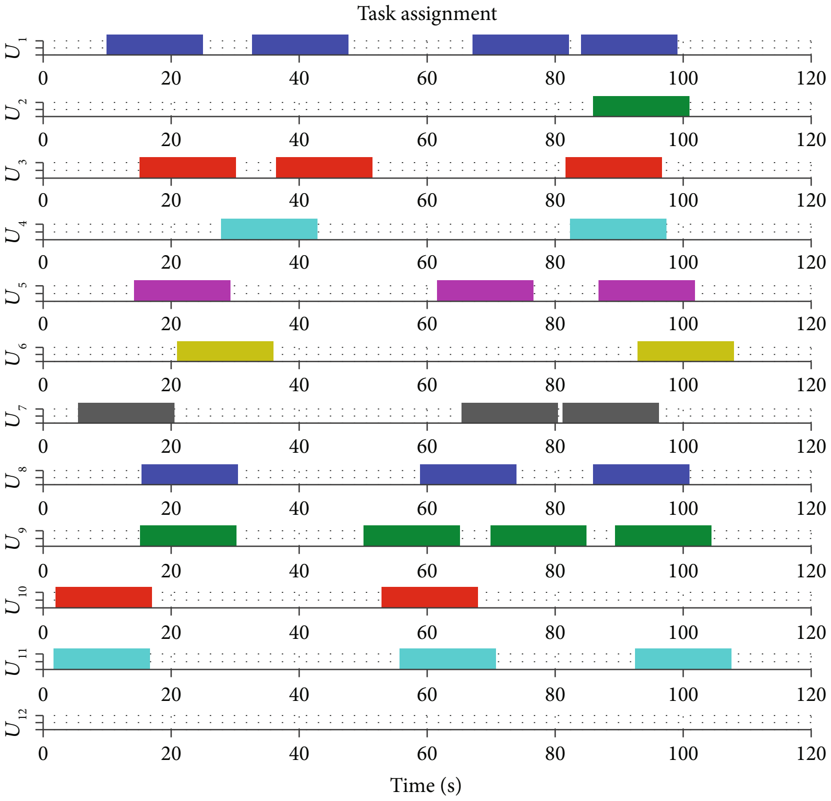


FIGURE 6: The task assignment obtained by the CBBA algorithm of [5].

and the CBBA algorithm, this section takes the number of targets and UAVs as independent variables and compares the algorithm time, total task score, task completion rate, task completion time, and the number of communication times. The initial positions and task resource vectors for all UAVs and targets are generated in a random manner.

5.2.1. Number of UAVs. In order to explore the adaptability of the algorithm to the number of UAVs, the number of targets is set to 12, and the number of UAVs in the swarm is set to 5, 10, 15, 20, and 25, respectively. Among the swarm, the number of attack UAVs and electronic interference UAVs is equal (or approximately equal). The UAV swarm is partitioned into different number of clusters, shown in Table 8.

The statistical results of task completion rate, algorithm calculation time, and communication times are shown in Figures 8–10. From the statistical results presented, the following can be seen:

(1) When the number of UAVs is small (for example, 5 UAVs perform tasks on 12 targets), the task completion rates of the two algorithms are lower. The reason is that the resources of the UAV swarm are too insufficient to meet the need of tasks for all targets. Even in this situation, the task completion rate of the two-layer assignment algorithm is still higher than that of the CBBA algorithm. This is due to fact that the two-layer assignment algorithm can make full use of the resources of each UAV to ensure the balance of task assignment

(2) With the increase of the number of UAVs, the calculation time of the two algorithms increases correspondingly. However, the CBBA algorithm increases exponentially, while the two-layer assignment algorithm grows slowly. The reason is that the two-layer assignment algorithm divides large-scale cooperative problems into several small-scale problems

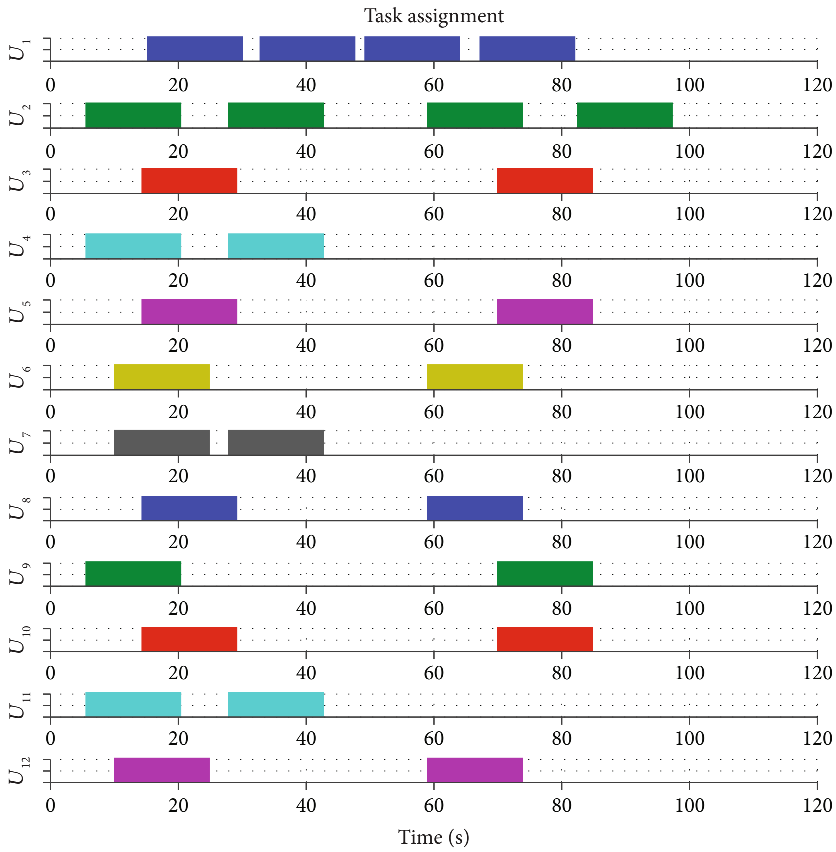


FIGURE 7: The task sequence obtained by the two-layer algorithm.

TABLE 7: The task assignment result obtained by the two-layer algorithm.

T_j	Task time	Task squad
T_1	(32, 42)	$\lvert U_2, U_4, U_7, U_{11}\rvert$
T_2	(5, 15)	$\lvert U_2, U_4, U_9, U_{12}\rvert$
T_3	(59, 69)	$\lvert U_1, U_2, U_6, U_8 U_{12}\rvert$
T_4	(74, 84)	$\lvert U_1, U_3, U_5, U_9 U_{10}\rvert$
T_5	(30, 40)	$\lvert U_1, U_9, U_{12}\rvert$
T_6	(16, 26)	$\lvert U_1, U_3, U_5, U_8, U_{10}\rvert$

TABLE 8: The clustering form of the two-layer algorithm.

Number of UAVs	Clustering form
5	2 UAV clusters, 2 or 3 UAVs each cluster
10	2 UAV clusters, 5 UAVs each cluster
15	3 UAV clusters, 5 UAVs each cluster
20	4 UAV clusters, 5 UAVs each cluster
25	5 UAV clusters, 5 UAVs each cluster

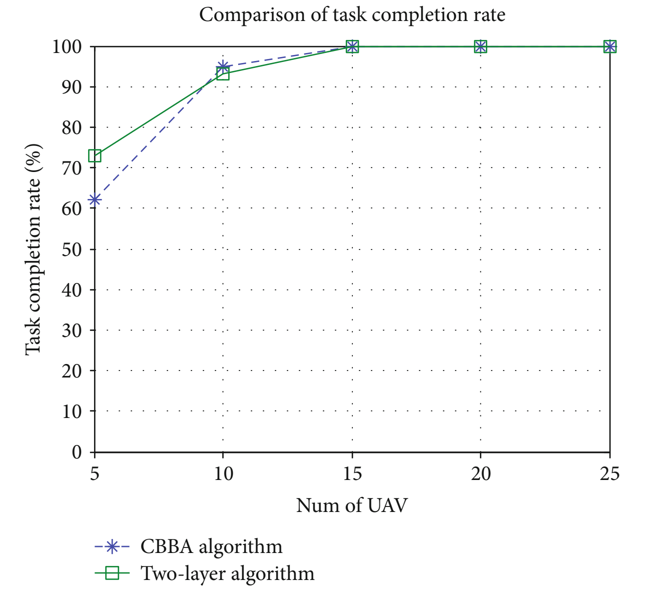


FIGURE 8: The comparison of task completion rate obtained by two types of algorithm.

(3) With the increase of the number of UAVs, the communication times increase correspondingly. The two-layer assignment algorithm increases slowly due to the fact that each UAV only needs to communicate with other UAVs in the same cluster in most situations, while in the CBBA algorithm, each UAV theoretically needs to communicate with all the UAVs in the whole swarm

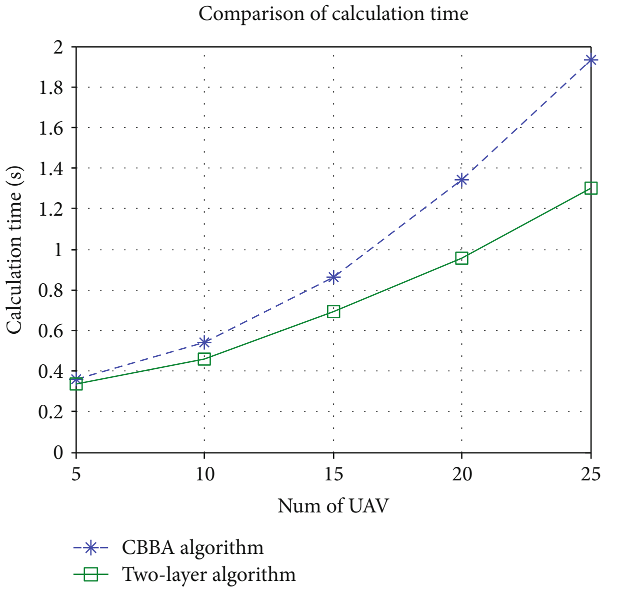


Figure 9: The comparison of calculation time obtained by two types of algorithm.

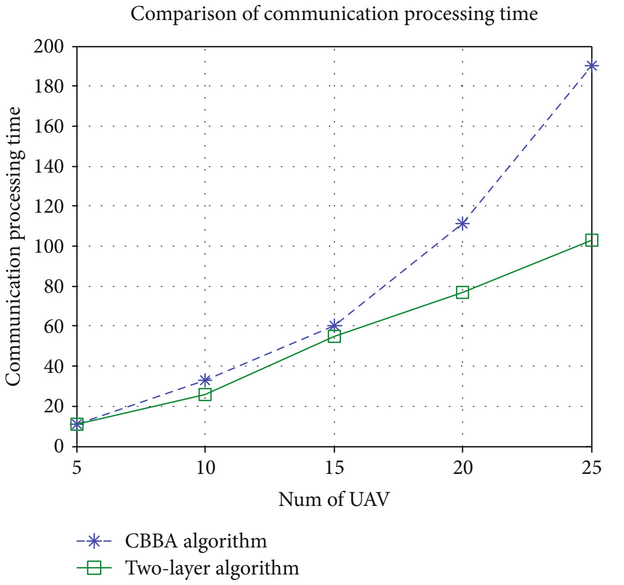


Figure 10: The comparison of communication processing times obtained by two types of algorithm.

5.2.2. Number of Targets. In order to explore the adaptability of the algorithm to the number of targets, the number of targets is set to 5, 10, 15, 20, and 25, respectively. The UAV swarm consists of 12 UAVs, including 6 attack UAVs and 6 electronic interference UAVs. Four sets of simulation examples are set up, as shown in Table 9. For each set of examples, the algorithm calculation time, total task score, and task completion rate are compared and analyzed.

The statistical results of task completion rate, algorithm calculation time, and total task score for are shown in Figures 11–13, respectively. From the statistical results presented, the following can be seen:

Table 9: Simulation examples.

Examples	UAV swarm task assignment algorithm
1	CBBA algorithm, 12 UAVs, no cluster
2	Two-layer algorithm, 2 UAV clusters, 6 UAVs each cluster
3	Two-layer algorithm, 3 UAV clusters, 3 UAVs each cluster
4	Two-layer algorithm, 4 UAV clusters, 3 UAVs each cluster

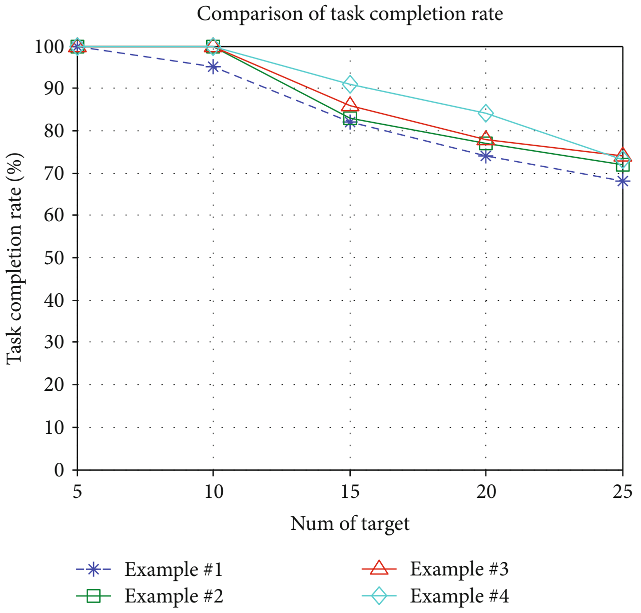


Figure 11: The comparison of total completion rate obtained by two types of algorithm.

(1) As the number of targets increases, the task completion rates of both algorithms decrease accordingly. Overall, the two-layer assignment algorithm has a higher task completion rate than the CBBA algorithm. As mentioned above, the two-layer assignment algorithm can make full use of the resources of each UAV to ensure the balance of task assignment. From the three examples of the two-layer task assignment algorithm, it can be seen that when the number of targets is not particularly large (within 20), the third case (4 UAV clusters) has the best effect; but when the number of targets increases to 25, the difference in task completion rates of the three samples is small. This is due to the fact that the number of tasks exceeds the upper limit of what the UAV swarm can accomplish

(2) With the increase of the number of targets, the calculation time of the CBBA algorithm increases exponentially, while that of the two-layer assignment algorithm increases relatively slowly. From the three examples of the two-layer task assignment algorithm, it can be seen that when the swarm is grouped

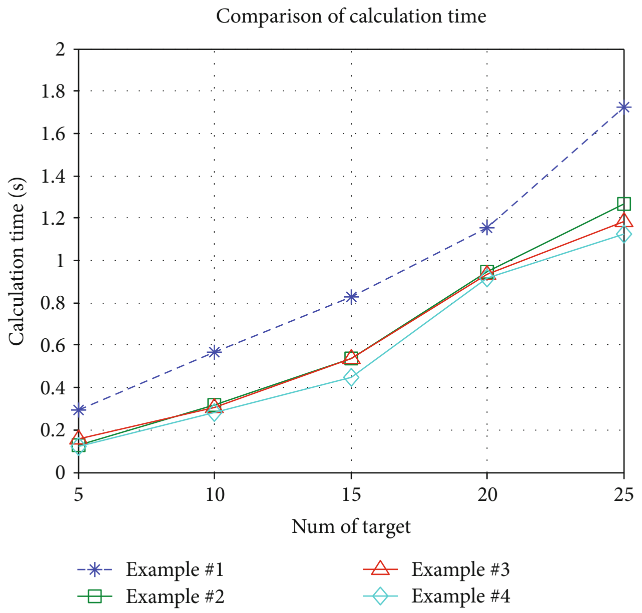


Figure 12: The comparison of calculation time obtained by two types of algorithm.

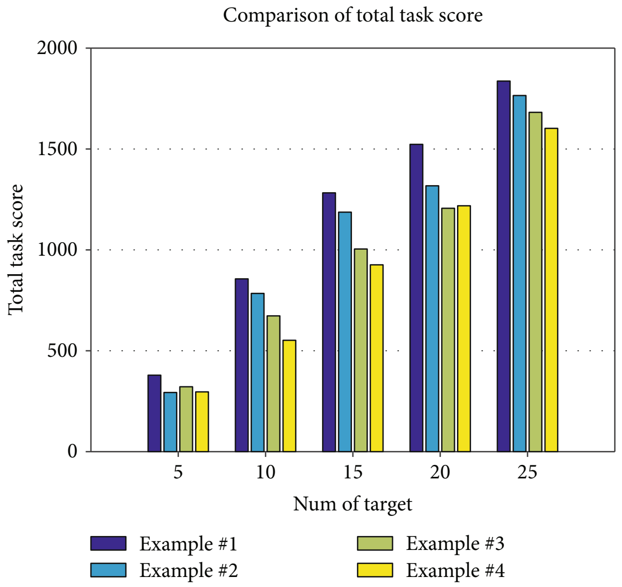


Figure 13: The comparison of total task score obtained by two types of algorithm.

into four UAV clusters, the calculation time of the algorithm is relatively short, but the advantage is not obvious

(3) As the number of targets increases, the total task scores of the two algorithms increase accordingly. This is due to the fact that the more the number of targets, the more tasks are assigned and the higher the total task score obtained. Overall, the total task score obtained by the two-layer assignment algorithm is smaller than the CBBA algorithm. This is due to the fact that the CBBA algorithm assigns tasks from a global perspective at the expense of longer computation time and more complex consensus processes and achieves an approximate optimal solution for the entire UAV swarm. The two-layer algorithm sacrifices some task rewards and achieves the real-time and high efficiency of task assignment when the number of targets is large. It can be seen from the three examples of the two-layer task assignment algorithm that the smaller the number of UAV clusters, the greater is the task reward

In summary, the proposed two-layer task assignment algorithm can make full use of task resources and has better real-time performance with the sacrifice of the global task reward. The two-layer task assignment algorithm has better adaptability to the number of UAVs and targets and is more suitable for the high real-time requirements of the UAV swarm task assignment problem. Different numbers of UAV clusters have a certain impact on the performance of the two-layer task assignment algorithm. The smaller number of UAV clusters does not reflect the advantages of the two-layer structure. The larger number of clusters leads to too many iterations of intercluster consensus. It can be seen from the simulation verification that when the number of UAV clusters is approximately equal to the number of UAVs among each cluster, the best comprehensive performance can be obtained.

6. Conclusion

In this paper, the feature weight clustering algorithm is applied to UAV swarm clustering, and a two-layer task assignment algorithm based on the basic CBBA algorithm is proposed. The feature weight clustering algorithm can make the two-layer task assignment algorithm more effectively. The algorithm uses different consistency rules between clusters and within clusters and can achieve task assignment consensus quickly and efficiently. The simulation results show that compared with the basic CBBA algorithm, the proposed algorithm can assign tasks effectively in real time when the number of UAVs and targets is large.

Data Availability

The numerical data used to support the findings of this study is included within the article.

Conflicts of Interest

The authors declare no conflict of interest.

Acknowledgments

This research was funded by the foundation of Shaanxi Key Laboratory of Integrated and Intelligent Navigation (grant number SKLIIN-20180104) and the Natural Science Foundation of Shaanxi Province (grant number 2019JQ-936).

References

[1] S. Cambone, *Unmanned Aircraft Systems Roadmap 2005-2030*, United States dept. of Defense office of the Secretary of Defense, 2005.

[2] A. K. Whitten, H. L. Choi, L. B. Johnson, and J. P. How, "Decentralized task assignment with coupled constraints in complex tasks," in *Proceedings of the American Control Conference*, pp. 1642–1649, San Francisco, CA, USA, 2011.

[3] L. F. Bertuccelli and M. L. Cummings, "Scenario-based robust scheduling for cooperative human-UAV visual search tasks," in *IEEE Conference on Decision and Control and European Control Conference*, pp. 5702–5707, Orlando, FL, USA, 2011.

[4] J. George, P. B. Sujit, and J. B. Sousa, "Search strategies for multiple UAV search and destroy tasks," *Journal of Intelligent & Robotic Systems*, vol. 61, no. 1-4, pp. 355–367, 2011.

[5] H. L. Choi, L. Brunet, and J. P. How, "Consensus-based decentralized auctions for robust task assignment," *IEEE Transactions on Robotics*, vol. 25, no. 4, pp. 912–926, 2009.

[6] S. Ponda, J. Redding, H.-L. Choi, J. P. How, M. A. Vavrina, and J. Vian, "Decentralized planning for complex missions with dynamic communication constraints," in *Proceedings of the 2010 American Control Conference*, Baltimore, MD, USA, 2010.

[7] A. K. Whitten, H. L. Choi, L. B. Johnson, and J. P. How, "Decentralized task assignment with coupled constraints in complex missions," in *Proceedings of the 2011 American Control Conference*, pp. 1642–1649, San Francisco, CA, USA, 2011.

[8] L. B. Johnson, S. Ponda, H.-L. Choi, and J. P. How, "Improving the efficiency of a decentralized tasking algorithm for UAV teams with asynchronous communications," in *AIAA Guidance, Navigation, and Control Conference (GNC)*, Toronto, Ontario, Canada, 2010.

[9] L. B. Johnson, S. S. Ponday, H.-L. Choi, and J. P. How, "Asynchronous decentralized task allocation for dynamic environments," in *Infotech@Aerospace 2011*, American Institute of Aeronautics and Astronautics, St. Louis, Missouri, 2011.

[10] L. B. Johnson, H. L. Choi, S. S. Ponda, and J. P. How, "Decentralized task allocation using local information consistency assumptions," *Journal of Aerospace Information Systems*, vol. 14, no. 2, pp. 103–122, 2017.

[11] N. Buckman, H. L. Choi, and J. P. How, "Partial replanning for decentralized dynamic task allocation," in *AIAA Scitech 2019 Forum*, San Diego, California, 2019.

[12] X. Fu, P. Feng, and X. Gao, "Swarm UAVs task and resource dynamic assignment algorithm based on task sequence mechanism," *IEEE Access*, vol. 7, no. 1, pp. 41090–41100, 2019.

[13] J. Han, *Data Mining: Concepts and Techniques*, Elsevier, 2005.

[14] K. G. Soni and A. Patel, "Comparative analysis of K-means and K-medoids algorithm on IRIS data," *International Journal of Computational Intelligence Research*, vol. 13, no. 5, pp. 899–906, 2017.

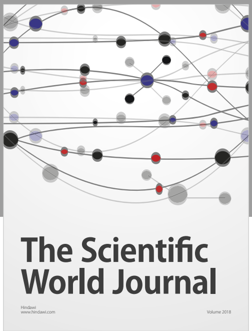
[15] S. Visalakshi and V. Radha, "An enhanced K-means clustering based outlier detection techniques to improve water contamination detection and classification," *Computational Intelligence in Data Mining - Volume 1*, vol. 31, pp. 303–313, 2015.

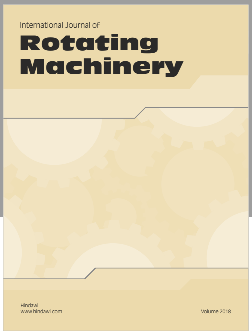
[16] S. Chen, "Nonlinear time series modelling and prediction using Gaussian RBF networks with enhanced clustering and RLS learning," *Electronics Letters*, vol. 31, no. 2, pp. 117-118, 1995.

[17] A. K. Jain and R. C. Dubes, "Algorithms for clustering data," *Technometrics*, vol. 32, no. 2, pp. 227–229, 1988.

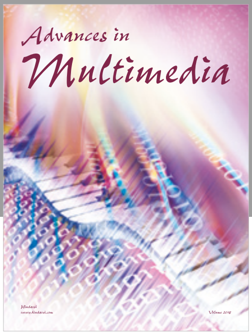
[18] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, En Broeck the Statistical Mechanics of Learning Rsity, 2nd edition, 2000.

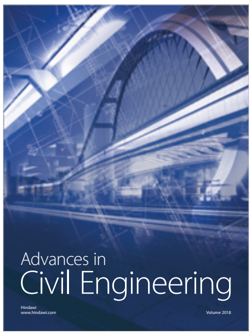
Journal of
Engineering

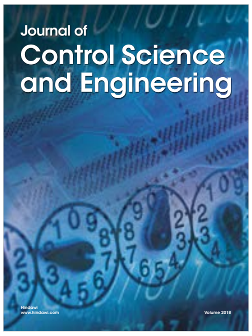

The Scientific
World Journal


International Journal of
Rotating
Machinery

Journal of
Sensors

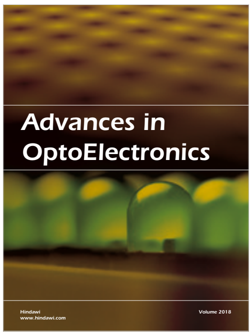

Advances in
Multimedia


Advances in
Civil Engineering


Journal of
Control Science
and Engineering

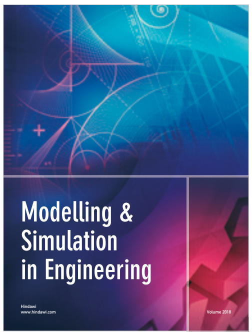
Journal of
Robotics

Journal of
Electrical and Computer
Engineering

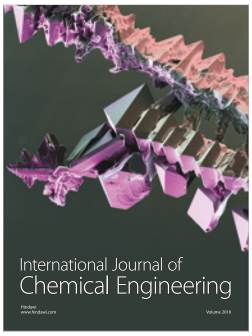

Advances in
OptoElectronics

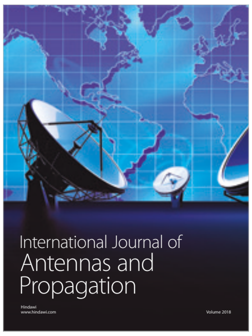
VLSI Design

International Journal of
Navigation and
Observation

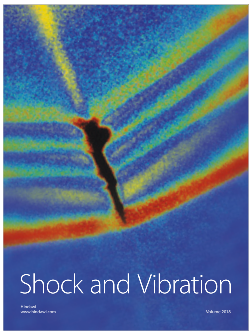

Modelling &
Simulation
in Engineering

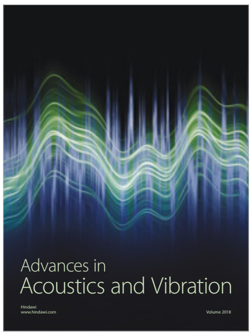
International Journal of
Aerospace
Engineering


International Journal of
Chemical Engineering


International Journal of
Antennas and
Propagation

Active and Passive
Electronic Components


Shock and Vibration


Advances in
Acoustics and Vibration