ICCK Transactions on Intelligent Systematics
http://dx.doi.org/10.62762/TIS.2025.409447

RESEARCH ARTICLE

# Multi-UAV Cooperative Task Allocation Based on Multi-strategy Clustering Ant Colony Optimization Algorithm

**Rui Wang**[1], **Yiqing Shan**[1], **Lianwei Sun**[1] and **Hui Sun**[1,*]

[1] School of Electronic Information and Automation, Civil Aviation University of China, Tianjin 300100, China

## Abstract

**To address the issues of low solving efficiency and susceptibility to local optima in multi-unmanned aerial vehicle (multi-UAV) task allocation algorithms within urban areas, this study constructs a task allocation model aiming to minimize economic costs for material delivery and reduce the urgency of rescue task demands. A multi-strategy clustering ant colony optimization algorithm (KMACO) is proposed for solution. Specifically, the K-means clustering method is utilized to partition the number of rescue tasks assigned to each UAV. In the ant colony optimization algorithm, a pheromone update strategy and a random evolution strategy are introduced to guide population search directions, thereby enhancing solving efficiency and avoiding local optima. Experimental results demonstrate that the proposed algorithm effectively reduces algorithm running time and operational economic costs while satisfying rescue task urgency requirements. Compared with conventional methods, KMACO shows superior performance in convergence speed and solution quality, thus providing an optimized decision-making framework for emergency rescue operations in complex urban environments.**

**Keywords**: UAV, task allocation, ant colony algorithm, K-means clustering.

**Academic Editor:**
Muhammad Attique Khan

**Submitted:** 27 March 2025
**Accepted:** 25 April 2025
**Published:** 11 July 2025

**Vol.** 2, **No.** 3, 2025.
10.62762/TIS.2025.409447

***Corresponding author:**
✉ Hui Sun
h-sun@cauc.edu.cn

## 1 Introduction

Urban disasters such as earthquakes, fires, and floods are often sudden, complex, and destructive, posing a serious threat to the lives and property safety of urban residents. UAV plays an increasingly important role in urban rescue due to its advantages of strong mobility, rapid deployment and access to dangerous areas. However, the ability of a single UAV is limited, and multi-UAV cooperative operation has become the key to improving the efficiency of urban rescue [1]. In the field of urban UAV emergency rescue, whether UAV can reasonably allocate material distribution tasks is the key to ensure rescue efficiency. Considering the urgency of rescue missions, multiple UAVs are usually required to perform multiple rescue missions in coordination to improve the overall material distribution efficiency [2]. By constructing a UAV cluster, according to the performance parameters and task requirements

**Citation**
Wang, R., Shan, Y., Sun, L., & Sun, H. (2025). Multi-UAV Cooperative Task Allocation Based on Multi-strategy Clustering Ant Colony Optimization Algorithm. *ICCK Transactions on Intelligent Systematics*, 2(3), 149–159.

of the UAV, the task allocation algorithm is adopted to realize the reasonable allocation of material distribution tasks, so as to ensure that each UAV can maximize its effectiveness in the cluster and avoid idle or excessive load of resources, thus significantly improving the overall rescue efficiency [3]. At present, UAV task allocation methods are mainly divided into two categories: mathematical programming methods and heuristic methods [4]. Mathematical programming methods ensure the global optimality of task allocation by accurately solving the mathematical model, which is suitable for simple and small-scale task scenarios, including linear programming method, mixed integer programming method, dynamic programming and so on [5–7]. The heuristic method quickly finds a feasible solution in a complex problem through an approximation algorithm, which is suitable for large-scale or dynamic task allocation scenarios. It mainly includes greedy algorithm, genetic algorithm, ant colony optimization algorithm, particle swarm optimization algorithm, etc. Considering the requirements of urban emergency rescue tasks, it is suitable for heuristic methods to solve. Han et al. [8] established a multi-UAV cooperative task allocation model in logistics transportation by considering the constraints of UAV operation reliability and flight performance and taking the minimum safety risk and logistics cost as the objective function. Zhou et al. [9] proposed a hybrid particle swarm optimization algorithm for multi-UAV cooperative task allocation. Through the improvement of the particle swarm optimization algorithm, a variety of strategies are introduced to enhance the search ability of the algorithm, and the speed and quality of task allocation are improved. Devi et al. [10] realized the discretization of UAV through binary matrix coding and applied particle swarm optimization to solve the multi-UAV task allocation problem. Jiang et al. [11] developed a multi-constraint model incorporating distribution time windows to address task allocation issues in logistics UAV applications. The model was solved using an improved PSO algorithm. Liu et al. [12] established a multi-objective function for the path coordination of a multi-unmanned aerial vehicle cruise system. Aiming to minimize both the number of UAVs and the total flight distance, a multi-objective evolutionary algorithm was employed to address this model. Although extensive research has been conducted on task allocation algorithms, the ant colony algorithm remains a prominent solution due to its strong robustness, maintaining its status as a classic approach in this field. Although extensive research has been conducted on task allocation algorithms, the ant colony algorithm remains a prominent solution in this domain due to its strong robustness, maintaining its status as a classic approach for addressing such problems [13]. Ning et al. [14] proposes an improved pheromone update mechanism based on the ant colony algorithm, as well as a novel pheromone smoothing mechanism to address the task allocation problem. By dynamically tracking changes in the optimal path's pheromones during the iterative process of the ant colony algorithm, this approach accelerates convergence speed. Additionally, the smoothing mechanism reinitializes the pheromone matrix to enhance population diversity and strengthen the global search capability of the algorithm. Tian et al. [15] addresses the issues of low utilization and uneven load distribution in task scheduling strategies. It proposes a high-performance scheduling algorithm based on a genetic-ant colony approach, aimed at reducing node load rates and significantly enhancing algorithm efficiency. Wu et al. [16] proposed a fusion-enhanced ant colony algorithm to address the issue of low task allocation efficiency for multiple UAVs in dense urban environments. By incorporating an adaptive pheromone mechanism and an extended heuristic strategy, the algorithm guides the population search direction and enhances task allocation performance in complex urban scenarios.

Liu et al. [17] proposed an enhanced ant colony algorithm with adaptive variable step size. This algorithm employs a novel pheromone concentration modification strategy, which is intended to address the limitations of the traditional ant colony algorithm, including its suboptimal convergence speed and vulnerability towards local optima.

To address these challenges, this paper proposes a task allocation model incorporating UAV operation costs and rescue mission urgency, along with a multi-strategy-enhanced clustering ant colony optimization algorithm. First, K-means clustering is applied to improve computational efficiency. Subsequently, a stochastic evolution strategy combined with a pheromone concentration update mechanism is implemented to guide population search directions. This framework enables the determination of optimal allocation schemes for multi-UAV systems. Finally, simulation experiments validate the algorithm's optimization performance and operational stability.

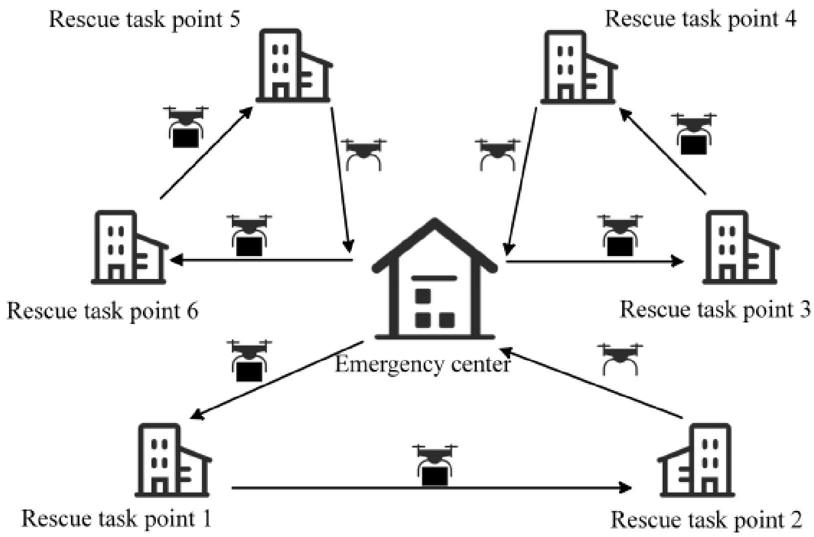



**Figure 1.** Diagram of UAV task allocation for emergency rescue.

## 2 Multi-UAV task allocation modeling

After an urban disaster occurs, the rescue center receives material requests from multiple rescue points. Given the known locations of these rescue points, emergency rescue drones are deployed for material distribution. This study aims to optimize the spatiotemporal allocation of emergency material distribution tasks and assign them efficiently to multiple drones, thereby improving rescue efficiency.

The problem is described as follows: There are N drones in the emergency rescue material reserve center, each with different models and performance parameters. These drones must deliver supplies to M rescue mission points, which vary in geographical location, material demand, and urgency. Each UAV must depart from the reserve center, follow a pre-planned task allocation route to deliver supplies to assigned mission points, and return to the reserve center upon completion. No new tasks are accepted after a drone departs. Through multi-UAV task allocation, rescue resources can be reasonably distributed, and rescue efficiency can be enhanced. The schematic diagram of UAV task allocation is shown in Figure 1.

### 2.1 Decision Variable

The main purpose of using multiple UAVs to carry out rescue missions is to make full use of rescue resources, improve rescue efficiency and ensure the safety of affected people by reasonably planning the rescue mission objectives and rescue sequence for each UAV. Assuming that the set of material rescue points is $X = X_1, X_2, \cdots, X_M$, the set of emergency rescue drones is $Y = Y_1, Y_2, \cdots, Y_N$, after satisfying that all rescue tasks can be performed by the drone to ensure that the rescue tasks can be successfully performed, the rescue task allocation decision variable $Z$ is set to represent the execution relationship between the drone $Y$ and the material rescue point $X$.

$$Z = \{Z_{n,m} \mid Z_{n,m} \in \{0,1\}\}_{N\times M} \tag{1}$$

where $Z_{n,m}$ is the UAV decision variable. When the value is 1, the rescue task $X_m$ is assigned to the UAV $Y_n$. When the value is 0, the rescue task $X_m$ is not assigned to the UAV $Y_n$.

### 2.2 Objective Function

#### 2.2.1 *Economic Cost*

Economic cost is one of the key factors that must be considered in the emergency rescue mission of UAV. The economic cost of UAV emergency rescue mainly involves the cost of UAV material distribution and transportation, that is, the costs incurred by UAV in the transportation process, including battery energy consumption, depreciation maintenance, etc. As the length of the transportation distance of the UAV has a direct impact on the economic cost, it can be deducted that the farther the distance, the higher the economic cost will be. Therefore, in the context of emergency rescue missions, it is essential to optimize the flight path of drones and minimize unnecessary travel distances. This approach aims to reduce the overall economic costs associated with such operations.

$$G_1 = \sum_{n=1}^{N}\sum_{m=1}^{M+1} \delta_n L_{n,m} Z_{n,m} \tag{2}$$

where $\delta_n$ represents the transportation cost per unit distance of the UAV $Y_n$; $L_{n,m}$ is the Euclidean distance from the current position of the UAV $Y_n$ to the rescue task $X_m$.

#### 2.2.2 *Demand Urgency Degree*

As a new type of transportation for material transportation, drones play an increasingly critical role in emergency rescue. Considering the urgency of the UAV's emergency rescue mission, it is necessary to conduct a comprehensive and in-depth analysis from the perspective of the overall material distribution route when performing its rescue mission. Demand urgency is the core index to determine the allocation of rescue resources and the order of task execution. Accurate assessment of demand urgency can ensure that limited rescue resources can be used efficiently, improve rescue efficiency, and minimize casualties and property losses. To scientifically determine the priority of UAV rescue mission location priorities, the evaluation metric of demand urgency has been introduced.

$$G_2 = \sum_{n=1}^{N}\sum_{m=1}^{M} o(pri_{n,m}) \times q(pri_{n,m}) \tag{3}$$

$$o(pri_{n,m}) = k_1 \times \mid pri_{n,m+1} - pri_{n,m} \mid \tag{4}$$

$$q(pri_{n,m}) = \begin{cases} 1 & if(pri_{n,m} < pri_{n,m+1}) \\ 0 & if(pri_{n,m} \geq pri_{n,m+1}) \end{cases} \tag{5}$$

where $o(pri_{n,m})$ is the demand urgency penalty coefficient function of UAV $Y_n$ to perform rescue tasks; $q(pri_{n,m})$ is an indicator function, which means that when the emergency coefficient $pri_{n,m}$ of the current rescue mission point $X_m$ is lower than that of the next rescue mission point $X_{m+1}$ in the task allocation order, the value is 1, otherwise it is 0.

Considering that there is a large difference in the value range of the above two sub-objective functions, the value of the sub-objective function is between [0,1] by using the min-max standardization method.

$$G' = \frac{G - G_{\min}}{G_{\max} - G_{\min}} \tag{6}$$

In summary, the task allocation objective function of assigning M tasks to N drones is:

$$\min G = o_1 G_1^{'} + o_2 G_2^{'} \tag{7}$$

where $o_1$ is the economic cost weight coefficient; $o_2$ is the weight coefficient of demand urgency. $G_1'$ and $G_2'$ are the economic cost and demand urgency after standardization, respectively.

### 2.3 Constraint Condition

#### 2.3.1 *Rescue mission constraints*

Considering the nature of the rescue task, it is necessary for the emergency rescue drone to complete all material distribution tasks. At the same time, each logistics distribution task can only be performed by a certain drone, and all tasks can only be performed once.

$$\sum_{n=1}^{N}\sum_{m=1}^{M} Z_{n,m} = M \tag{8}$$

$$\sum_{n=1}^{N} Z_{n,m} = 1, \forall m = 1, 2, \cdots, M \tag{9}$$

#### 2.3.2 *UAV flight range constraints*

Limited by the physical condition of the drone itself, when each drone performs a task, its flight range cannot exceed the rated range, otherwise the drone will fail to return successfully, and the departure position of each drone is used as the return point of the drone.

$$\sum_{m=1}^{M+1} L_{n,m} Z_{n,m} \leq L_{n(\max)} \tag{10}$$

where $L_{n(max)}$ is the maximum flight range of UAV $Y_n$; $L_{n,m+1}$ is the voyage of UAV $Y_n$ to return to the rescue center after completing all tasks.

#### 2.3.3 *UAV load constraints*

Due to the limitation of the UAV's own conditions, it is required that the total weight of the material requirements of each UAV's rescue point cannot exceed the load limit.

$$q_n = \sum_{m=1}^{M} q_m Z_{n,m} \leq Q_{n(\max)} \tag{11}$$

where $q_m$ is the amount of relief materials needed for the rescue mission point; $Q_{n(max)}$ is the maximum material weight that UAV $Y_n$ can carry.

## 3 Multi-strategy K-means Ant Colony Optimization Algorithm

The traditional ant colony algorithm is prone to problems such as low efficiency, slow convergence speed and easy to fall into local optimum when solving the problem of multi-UAV task allocation. In view of the above problems, the K-means clustering idea is used to allocate the number of rescue tasks for each UAV in advance [18]. The multi-strategy K-means

Ant Colony Optimization Algorithm (KMACO) is obtained by using the random evolution strategy and the pheromone concentration update strategy and improving the ant colony path node selection probability and the pheromone concentration update method respectively.

### 3.1 K-means Clustering

K-means clustering is an unsupervised machine learning algorithm. The fundamental principle underlying this algorithm is to divide the data set into K clusters and to continuously update the centroid position of each cluster through iteration until there is no change in its position, thereby achieving the division of the data set [19]. For solving the problem of multi-UAV multi-task point allocation, the K-means clustering idea is introduced. The rescue point $X = X_1, X_2, \cdots, X_M$ that needs to perform material delivery can be clustered into $Y = Y_1, Y_2, \cdots, Y_N$ clusters $C_k$ by the ratio of the total rescue task demand and the maximum load of the UAV. According to the given number of clusters, each UAV is divided into a corresponding rescue task point set. Considering that the ant colony algorithm has the disadvantages of slow operation speed and unstable solution results in dealing with task allocation problems, especially in dealing with large-scale allocation problems, the introduction of clustering ideas can effectively solve the above problems.

The flowchart of the K-means clustering algorithm is shown in Figure 2. The steps for applying K-means clustering to solve multi-UAV task allocation are as follows:

Step 1: Based on the data set $X = X_1, X_2, \cdots, X_M$ of rescue task points, determine the total amount of rescue materials. The number of clusters is determined by calculating the ratio of the total rescue materials to the maximum load capacity of the drones.

Step 2: Randomly select $N$ rescue point positions as the initial cluster centers. Calculate the distance between each rescue task point and the cluster centers. Assess whether the residual capacity of each cluster center meets the specified requirements. Assign the rescue point to the cluster center with the smallest distance, ensuring that the material quantity assigned to the cluster center does not exceed the maximum UAV load. If this condition is not met, assign the rescue point to the next closest cluster center.

Step 3: Recalculate the cluster centers based on the newly formed clusters by taking the mean position of all rescue points assigned to each cluster.

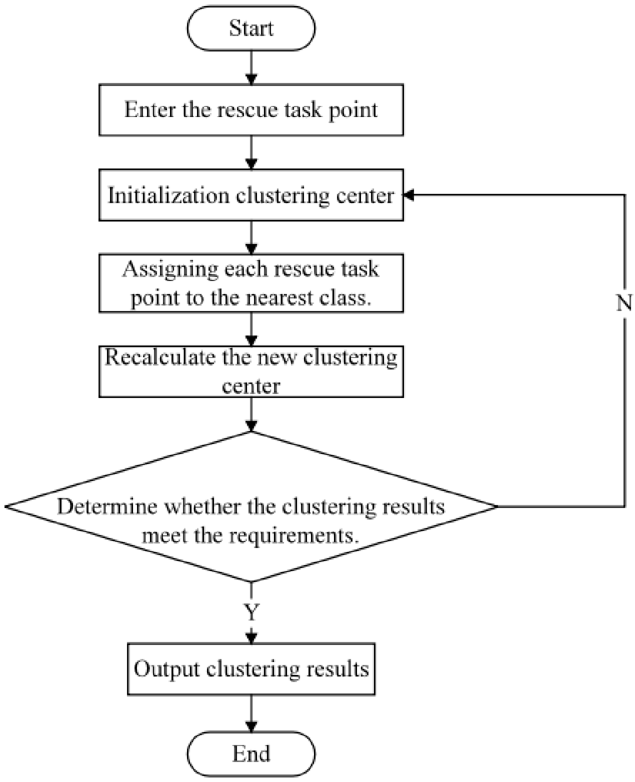



**Figure 2.** K-means clustering flow chart.

Step 4: Check whether the difference between the coordinates of the new cluster centers and the original cluster centers exceeds a predefined threshold. If the difference is greater than the threshold, return to Step 2. If the difference is within the threshold, save the clustering results and terminate the algorithm.

### 3.2 Random Evolutionary Strategy

In the ant colony algorithm for multi-UAV task allocation, the path selection probability mutation critical threshold $REF$ and the node selection probability random adjustment factor $EDT$ are introduced to change the path selection probability $q_{ij}^k(t)$ of the ant colony, which introduces more randomness and evolutionary ability to the ant colony algorithm [20]. This approach is designed to circumvent the algorithm's tendency to fall into the local optimal solution. It enhances the algorithm's global search capabilities and facilitates exploration of potential complex task allocations.

$$q_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha \cdot [\omega_{ij}]^\beta}{\sum_{s \in \text{allowed}_k} [\tau_{is}(t)]^\alpha \cdot [\omega_{is}]^\beta} \cdot rand & if \quad REF > EDT \\ \\ \frac{[\tau_{ij}(t)]^\alpha \cdot [\omega_{ij}]^\beta}{\sum_{s \in \text{allowed}_k} [\tau_{is}(t)]^\alpha \cdot [\omega_{is}]^\beta} & if \quad REF \leq EDT \end{cases} \tag{12}$$

where $\alpha$ and $\beta$ are pheromone weight factor and heuristic function weight factor respectively; $allowed_k$ is the set of all feasible nodes of the current ant; the critical threshold $EDT$ of path selection probability mutation is used to determine whether the probability of the ant colony needs to mutate when selecting the next path node. The random adjustment factor $REF$ of the node selection probability is a parameter with a range between values, which is used to adjust the probability of ants selecting the next path node.

### 3.3 Pheromone update strategy

The pheromone update in ant colony algorithm mainly includes pheromone evaporation update and pheromone concentration enhancement. Pheromone evaporation can prevent the algorithm from converging too quickly and getting stuck in a local optimal solution. Pheromone concentration enhancement rewards ants that find better paths, making subsequent ants more inclined to choose paths with stronger pheromone concentrations. However, the above pheromone update methods have problems of imbalance between global exploration and local exploitation as well as loss of diversity. Therefore, this paper proposes dynamic evaporation factor adjustment and elite population strategy to improve the above problems.

#### 3.3.1 Dynamic volatile factor regulation mechanism

When the traditional ant colony algorithm updates the pheromone, in order to avoid the supersaturation phenomenon caused by the continuous accumulation of pheromone, the pheromone volatilization mechanism is usually introduced, in which the volatilization factor is often set to a fixed constant [21]. However, this fixed volatilization factor setting method has certain limitations in practical applications, and cannot effectively meet the exploration needs of ant colonies at different stages. Therefore, the dynamic volatilization factor adjustment mechanism is introduced, and the ant colony algorithm is used to find the optimal task allocation process. In the different iteration stages, the demand for the size of the volatilization factor is different, and the pheromone volatilization factor is dynamically adjusted.

$$\rho = \begin{cases} \rho_{\min} & \rho < \rho_{\min} \\ \rho_0 \cdot e^{\frac{r(i)}{r(i-1)}-1} \cdot k_2 \cdot \frac{t}{T} & \rho_{\min} < \rho < \rho_{\max} \\ \rho_{\max} & \rho > \rho_{\max} \end{cases} \tag{13}$$

where $\rho_{min}$ and $\rho_{max}$ are the minimum and maximum pheromone volatilization coefficients; $\rho_0$ is the initial pheromone volatilization coefficient; $r(i)$ and $r(i-1)$ are the $i$ iteration and the $i-1$ iteration respectively to solve the optimal allocation results; $k_2$ is the pheromone adjustment coefficient; $t$ is the current iteration number of ant colony algorithm; $T$ is the maximum iteration number of the algorithm.

#### 3.3.2 Elite ant colony strategy

In the ant colony algorithm used in multi-UAV task allocation, the pheromone update phase is crucial. At this stage, the pheromone concentration of the path through which the elite ant colony performs the task will be strengthened [22]. After the pheromone concentration is increased, based on the guiding role of the pheromone, the subsequent ant colony will obviously be more inclined to choose the path of the elite ant colony when performing the search task. Through this mechanism, the subsequent ant colony increases its preference for these paths in the search process and then accelerates the aggregation of high-quality task allocation results, which is conducive to the algorithm to jump out of the local optimal solution and improve the overall performance and search efficiency of multi-UAV task allocation.

$$\tau_{ij}(t+1) = (1-\rho) \cdot \tau_{ij}(t) + \Delta\tau_{ij}^{\text{elite}}(t) \tag{14}$$

$$\Delta\tau_{ij}^{k}(t) = \begin{cases} \frac{Q}{L_k} + \frac{e}{L_{ses}} & \text{Through the best path} \\ \frac{Q}{L_k} & \text{otherwise} \end{cases} \tag{15}$$

where $\Delta\tau_{ij}^{k}(t)$ denotes the pheromone of the $k$th ant from the path node to the node; $Q$ is a constant, which is the amount of pheromone left by the ant colony under the unit path length; $L_k$ represents the length of the path taken by the $k$ ant in this cycle; $L_{best}$ represents the best path of the $k$ ant in this cycle; $e$ is the elite ant colony weight parameter.

### 3.4 KMACO algorithm flow

The flow chart of KMACO algorithm is shown in Figure 3. The main steps of KMACO algorithm to solve multi-UAV task allocation are as follows:

Step1: To accomplish the mission, mission-relevant information must first be acquired. The K-means clustering algorithm is subsequently employed to determine cluster quantities, with a critical constraint: the total material demand of all task points within each cluster must not exceed the UAV's maximum payload capacity. This approach ensures that every cluster comprehensively contains the mission-specific data required for UAV operations.

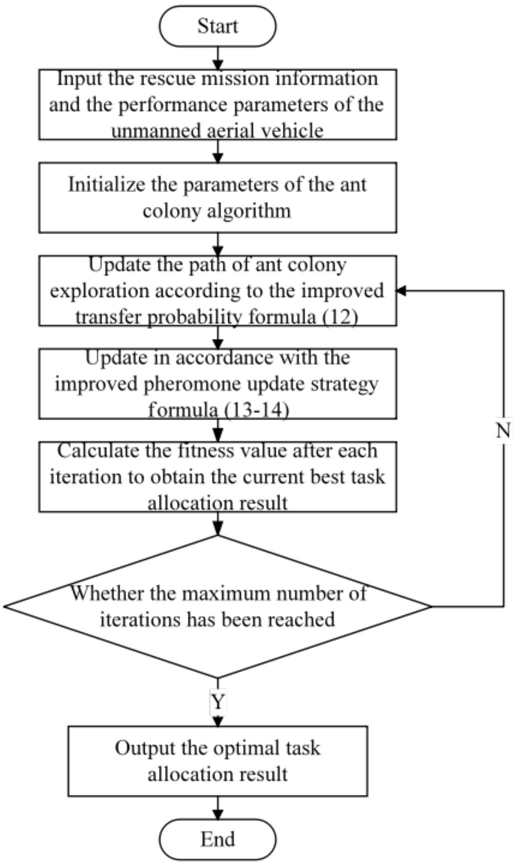



**Figure 3.** Flow chart of KMACO algorithm.

Step2: Initialize the parameters such as the number of ant populations, the number of iterations, and the pheromone volatilization factor to construct an initial pheromone matrix, which contains the initial pheromone concentration assigned to each task point by each drone.

Step3: Determine the paths that ants will continue to search based on the improved transition probability formula defined in Formula (12).

Step4: Update the pheromone levels in Ant colony paths based on the improved pheromone update strategy defined in Formulas (13-15).

Step5: All ants complete the path selection in turn, in which each ant constructs a set of task allocation schemes. By calculating the fitness value of each group of schemes, the optimal task allocation results of each iteration are compared to determine the current optimal allocation scheme.

Step6: Reiterate steps 3-5 until the termination condition is fulfilled, subsequently producing the optimal task allocation result.

## 4 Simulation Experiment and Analysis

### 4.1 Simulation Parameter Setting

To validate the mathematical model for multi-UAV task allocation and evaluate the effectiveness of the K-means Multi-strategy Ant Colony Optimization (KMACO) algorithm, simulation experiments were conducted using MATLAB R2022a on a computer equipped with 8 GB RAM and an Intel Core i7 processor. The experimental setup assumes a rescue center deploying three DJI Fly Cart 30 drones, with performance parameters provided in Table 1, to execute 15 rescue missions, where mission parameters are detailed in Table 2. The configuration parameters of the multi-strategy clustered Ant colony optimization algorithm are listed in Table 3.

**Table 1.** UAV parameter setting.

| DJI Fly Cart 30 | Parameter value |
|---|---|
| Airplane weight | 42.5kg |
| Maximum flight distance | 8km |
| Protective level of whole machine | IP55 |
| Maximum wind speed | 12m/s |
| Maximum load capacity | 30/kg |

**Table 2.** Parameters setting for rescue mission points.

| Serial number | X /m | Y / m | Z / m | demand / kg | urgency |
|---|---|---|---|---|---|
| T0 | 25 | 25 | 0 | - | - |
| T1 | 25 | 425 | 10 | 5 | 3.5 |
| T2 | 850 | 700 | 50 | 4 | 5.0 |
| T3 | 275 | 450 | 10 | 5 | 1.6 |
| T4 | 900 | 900 | 40 | 7 | 3.2 |
| T5 | 550 | 900 | 40 | 8 | 3.9 |
| T6 | 200 | 800 | 70 | 3 | 3.4 |
| T7 | 950 | 100 | 10 | 7 | 2.5 |
| T8 | 50 | 900 | 10 | 8 | 3.6 |
| T9 | 400 | 975 | 80 | 5 | 1.7 |
| T10 | 400 | 100 | 40 | 5 | 3.3 |
| T11 | 925 | 450 | 50 | 5 | 4.0 |
| T12 | 700 | 50 | 10 | 3 | 4.0 |
| T13 | 425 | 575 | 40 | 4 | 7.0 |
| T14 | 50 | 600 | 10 | 5 | 7.0 |
| T15 | 450 | 950 | 10 | 4 | 3.0 |

### 4.2 Algorithm effectiveness analysis

To validate the effectiveness of the proposed approach, we conducted comparative experiments using three algorithmic implementations: KACO (only clustering but not improved ant colony algorithm), KMACO

**Table 3.** Ant colony optimization algorithm parameter setting.

| Parameter name | Parameter value |
|---|---|
| Pheromone heuristic factor | 1 |
| Heuristic function factor | 3 |
| Pheromone volatilization coefficient | 0.2 |
| Pheromone concentration enhancement coefficient | 100 |
| Number of ants | 30 |
| The maximum number of iterations of the algorithm | 200 |

(clustering and improved ant colony algorithm) and traditional ant colony algorithm ACO are used to compare and verify the task allocation model. Considering the contingency of the algorithm, each algorithm runs 50 times under the condition that the number of all ant colonies and the number of iterations are the same, and the box line diagram of the task allocation results obtained by the three algorithms is drawn as shown in Figure 4.

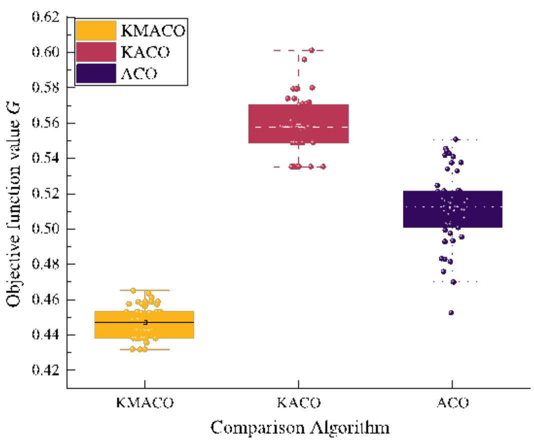



**Figure 4.** Task allocation result box plot.

To more effectively assess the performance advantages of the KMACO algorithm and examine the distribution characteristics and variability across different datasets, a boxplot comparing the task allocation results of the three algorithms is presented in Figure 4. As illustrated, the box corresponding to the KMACO algorithm is relatively compact, with data points densely clustered around it. This suggests that KMACO achieves low variance in objective function values, indicating high data consistency, low dispersion, and strong robustness. In comparison, the KACO algorithm exhibits a moderately sized box with a slightly wider spread of data points, reflecting greater variability than KMACO but still maintaining a more stable distribution than the ACO algorithm.

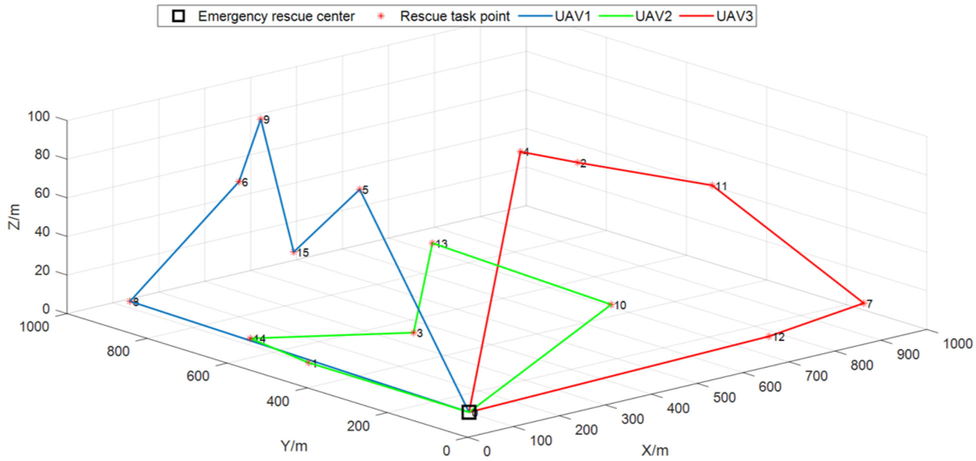
The ACO algorithm, however, shows a longer box length, with scatter points widely dispersed and some outliers far from the main cluster. This suggests that the objective function values of the ACO algorithm are highly discrete, leading to relatively poor stability. Overall, the boxplot analysis demonstrates that the task allocation results obtained by the KMACO algorithm are more densely distributed, yielding higher-quality solutions. This further highlights the superiority of the KMACO algorithm in addressing multi-UAV task allocation problems.

The multi-UAV cooperative task allocation results, solved by different algorithms, are presented in Figure 5, while the allocation sequence for three emergency rescue drones across 15 mission scenarios is shown in Table 4. By combining the information from both sources, it is clear that the allocation scheme derived from the KMACO algorithm is the most optimal. In urban disaster scenarios requiring multi-UAV task allocation, this scheme ensures that each drone operates within its physical constraints while effectively optimizing the objective function. It strikes a balance between economic efficiency and the urgency of rescue tasks, highlighting the rationality and feasibility of the KMACO algorithm in addressing multi-objective constraint problems.

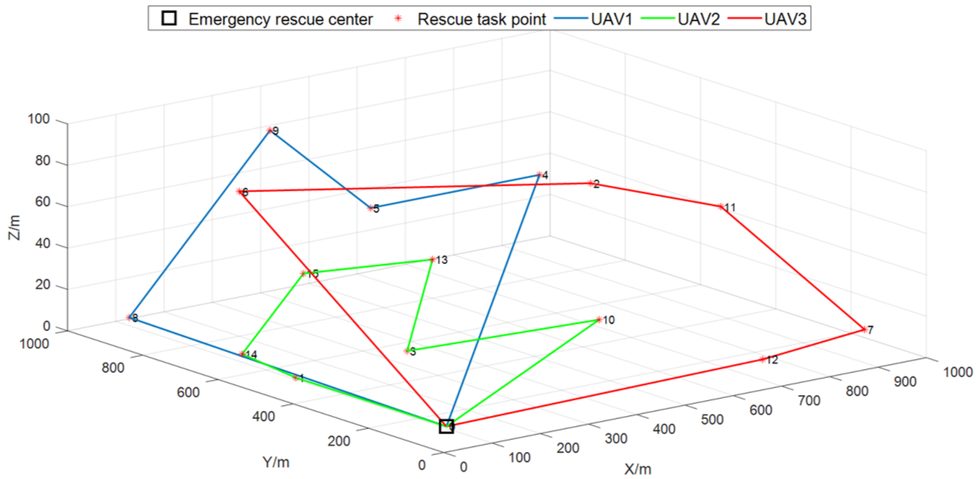
### 4.3 Simulation experiments and analysis under different data scales

To further demonstrate the applicability of the proposed algorithm in large-scale data scenarios, 30, 40, and 50 rescue task points were randomly selected in an $1km \times 1km$ area for simulation experiments. Meanwhile, the three comparison algorithms were independently executed 20 times to ensure the stability and reliability of the results. The relevant experimental data and statistical results are shown in Table 5.

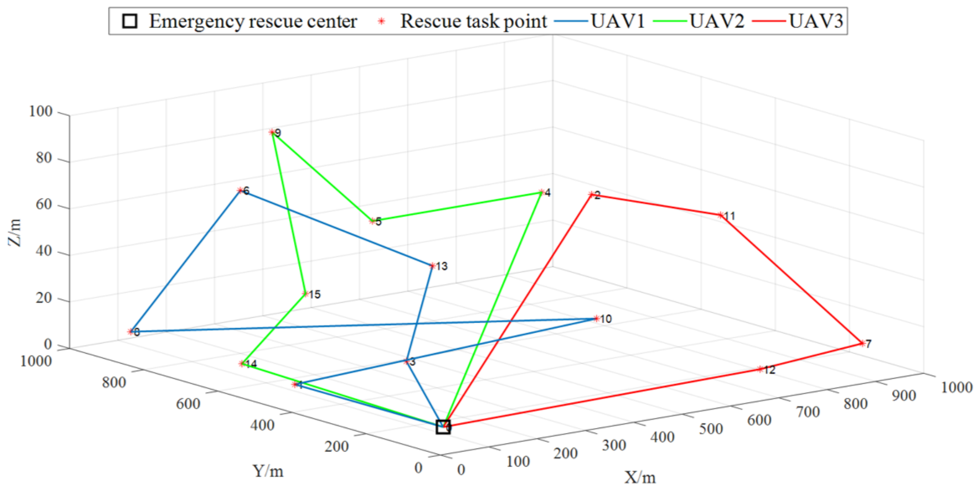
Experimental data demonstrate that the KMACO algorithm exhibits significant advantages in both task allocation effectiveness and computational efficiency. In terms of the objective function G-values, KMACO maintains the lowest optimal and average values across all problem scales. Particularly at n=50, its optimal G-value shows reductions of 13.8% and 28.7% compared to KACO and ACO respectively, highlighting its stability. Regarding computational efficiency, the runtime of the ACO algorithm increases significantly as the problem scale




(a)KMACO algorithm task allocation scheme.




(b)KACO algorithm task allocation scheme.




(c)ACO algorithm task allocation scheme.

**Figure 5.** Multi-UAV task allocation scheme.

expands, demonstrating a 14-fold speed disadvantage against KMACO at n=50. By implementing clustering strategies, KMACO optimizes time complexity from O($(n^2)$) to O(kn) while maintaining solution variance 60% lower than KACO. These findings conclusively validate the effectiveness of KMACO's integrated clustering and enhanced pheromone mechanisms in balancing global exploration with local

**Table 4.** Optimal delivery sequence of multi-UAV task allocation.

| UAV number | KMACO | KACO | ACO |
|---|---|---|---|
| U1 | T0-T1-T14-T3-T13-T10-T0 | T0-T4-T5-T9-T8-T0 | T0-T3-T13-T6-T8-T10-T1-T0 |
| U2 | T0-T12-T7-T11-T2-T4-T0 | T0-T6-T2-T11-T7-T12-T0 | T0-T14-T15-T9-T5-T4-T0 |
| U3 | T0-T5-T15-T9-T6-T8-T0 | T0-T1-T14-T15-T13-T3-T10-T0 | T0-T2-T11-T7-T12-T0 |

**Table 5.** Statistical data of each algorithm under different data scales.

| Algorithm | Indicators | n=30 | n=40 | n=50 |
|---|---|---|---|---|
| KMACO | optimal value $G$ | 0.88 | 1.16 | 1.12 |
| | average value $G$ | 0.95 | 1.21 | 1.21 |
| | running time/s | 8.65 | 11.89 | 15.15 |
| KACO | optimal value $G$ | 0.88 | 1.19 | 1.30 |
| | average value $G$ | 0.97 | 1.28 | 1.43 |
| | running time/s | 8.93 | 12.16 | 13.75 |
| ACO | optimal value $G$ | 1.03 | 1.25 | 1.57 |
| | average value $G$ | 1.09 | 1.35 | 1.64 |
| | running time/s | 115.81 | 209.90 | 211.97 |

exploitation, providing an efficient and stable solution for large-scale multi-UAV task allocation problems.

## 5 Conclusion

The present paper principally studies the multi-UAV task allocation problem based on the KMACO algorithm. Firstly, a mathematical model of multi-UAV task allocation considering the urgency and economy of rescue mission requirements is constructed. Subsequently, the traditional ant colony algorithm is enhanced via the integration of K-means clustering, pheromone concentration update, and random evolution strategy. The incorporation of K-means clustering contributes to a substantial reduction in the algorithm's execution time, while the pheromone concentration update and random evolution strategy serve to direct the ant colony's exploratory efforts, augment its global search capability, and expedite the convergence of the algorithm. The superiority of the KMACO algorithm is substantiated by simulation experiments, and based on this algorithm, a multi-rescue task point UAV material rescue task allocation scheme under multi-constraint conditions is generated.

## Data Availability Statement

Data will be made available on request.

## Funding

This work was supported by Key Research and Development Program of Tianjin, China of funding agency under 22YFZCSN00210.

## Conflicts of Interest

The authors declare no conflicts of interest.

## Ethical Approval and Consent to Participate

Not applicable.

## References

[1] Xie, S., Zhang, A., Bi, W., & Tang, Y. (2019). Multi-UAV mission allocation under constraint. *Applied Sciences*, *9*(11), 2184. [CrossRef]

[2] Yang, W. Z., & Xin, Y. (2021, March). Multi-UAV task assignment based on quantum genetic algorithm. In *Journal of Physics: Conference Series* (Vol. 1824, No. 1, p. 012010). IOP Publishing. [CrossRef]

[3] Miao, Y., Zhong, L., Yin, Y., Zou, C., & Luo, Z. (2017). Research on dynamic task allocation for multiple unmanned aerial vehicles. *Transactions of the Institute of Measurement and Control*, *39*(4), 466-474. [CrossRef]

[4] Poudel, S., & Moh, S. (2022). Task assignment algorithms for unmanned aerial vehicle networks: A comprehensive survey. *Vehicular Communications*, *35*, 100469. [CrossRef]

[5] Mangasarian, O. L. (2004). A Newton method for linear programming. *Journal of Optimization Theory and Applications*, *121*, 1-18. [CrossRef]

[6] Smith, J. C., & Taskin, Z. C. (2008). A tutorial guide to mixed-integer programming models and solution techniques. In *Optimization in medicine and biology* (pp. 521-548).

[7] Bellman, R. (1966). Dynamic programming. *Science*, *153*(3731), 34-37. [CrossRef]

[8] Han, L. I., Honghai, Z. H. A. N. G., Liandong, Z. H. A. N. G., & Hao, L. I. U. (2021). Multiple logistics unmanned aerial vehicle collaborative task allocation in urban areas. *Systems Engineering & Electronics*, *43*(12). [CrossRef]

[9] Zhou, X., & Yang, K. (2024). Cooperative multi-task assignment modeling of UAV based on particle swarm optimization. *Intelligent Decision Technologies*, *18*(2), 919-934.

[10] Devi, J., & Kruis, F. E. (2017, April). A fast Monte Carlo GPU based algorithm for particle breakage. In *2017 4th International Conference on Control, Decision and Information Technologies (CoDIT)* (pp. 0784-0789). IEEE. [CrossRef]

[11] Jiang, X., Zhou, Q., & Ye, Y. (2017, March). Method of task assignment for UAV based on particle swarm optimization in logistics. In *Proceedings of the 2017 International Conference on Intelligent Systems, Metaheuristics & Swarm Intelligence* (pp. 113-117). [CrossRef]

[12] Liu, X. F., Peng, Z. R., Chang, Y. T., & Zhang, L. Y. (2012). Multi-objective evolutionary approach for UAV cruise route planning to collect traffic information. *Journal of Central South University*, *19*(12), 3614-3621. [CrossRef]

[13] Liu, S., Leng, H., & Han, L. (2017). Pheromone model selection in ant colony optimization for the travelling salesman problem. *Chinese Journal of Electronics*, *26*(2), 223-229. [CrossRef]

[14] Ning, J., Zhang, Q., Zhang, C., & Zhang, B. (2018). A best-path-updating information-guided ant colony optimization algorithm. *Information Sciences*, *433*, 142-162. [CrossRef]

[15] Tian, Z., Zhang, S., & Gao, X. (2024). Application of Genetic-ant Colony Algorithm in High-performance Computing Task Scheduling. *Computer Applications and Software*, *41*(3), 253-257.

[16] Wu, X., Yin, Y., Xu, L., Wu, X., Meng, F., & Zhen, R. (2021). Multi-UAV task allocation based on improved genetic algorithm. *IEEE Access*, *9*, 100369-100379. [CrossRef]

[17] Liu, Y., & Mao, J. (2020). Research on path planning based on adaptive variable step size ant colony algorithm. *Electronic Measurement Technology*, *43*(7), 76-81.

[18] Blum, C. (2005). Ant colony optimization: Introduction and recent trends. *Physics of Life Reviews*, *2*(4), 353-373. [CrossRef]

[19] Ikotun, A. M., Ezugwu, A. E., Abualigah, L., Abuhaija, B., & Heming, J. (2023). K-means clustering algorithms: A comprehensive review, variants analysis, and advances in the era of big data. *Information Sciences*, *622*, 178-210. [CrossRef]

[20] Dorigo, M., & Socha, K. (2018). An introduction to ant colony optimization. In *Handbook of Approximation Algorithms and Metaheuristics* (pp. 395-408). Chapman and Hall/CRC.

[21] Sivanandam, S. N., Deepa, S. N., Sivanandam, S. N., & Deepa, S. N. (2008). Introduction to particle swarm optimization and ant colony optimization. In *Introduction to Genetic Algorithms* (pp. 403-424). [CrossRef]

[22] Yan, F., Chu, J., Hu, J., & Zhu, X. (2024). Cooperative task allocation with simultaneous arrival and resource constraint for multi-UAV using a genetic algorithm. *Expert Systems with Applications*, *245*, 123023. [CrossRef]



**Rui Wang**, born in 1980, Ph.D, professor, master's advisor. Her main research interests include aerospace material management strategies, complex system modeling, nonlinear system modeling. (Email: ruiwang@cauc.edu.cn)



**Yiqing Shan**, a 2022 graduate student majoring in control science and engineering from Civil Aviation University of China. The research direction is multi-UAV task allocation and path planning. (Email: 1126377420@qq.com)



**Lianwei Sun**, a 2023 graduate student majoring in control science and engineering from Civil Aviation University of China. The research direction is multi-UAV task allocation. (E-mail: tajuom@gmail.com)



**Hui Sun**, born in 1979, M.S., lecturer. His main research interests include wireless sensor networks, multi-agent system control, UAV swarm networking and coordination, and communications. (Email: h-sun@cauc.edu.cn)