Robotica (2024), 1–18
doi:10.1017/S0263574724001899

RESEARCH ARTICLE

Balanced Multi-UAV path planning for persistent monitoring

Xinru Zhan[1], Yang Chen[1], Xi Chen[1] and Wenhao Zhang[2]

[1]School of Information Science and Engineering, Wuhan University of Science and Technology, Wuhan, China
[2]Centre for Machine Vision, Bristol Robotics Laboratory, UWE Bristol, UK
Corresponding author: Yang Chen; Email: chenyag@wust.edu.cn

Received: 18 December 2023; **Revised:** 1 September 2024; **Accepted:** 3 October 2024

Keywords: Path planning; persistent monitoring; task balancing; K-means clustering

Abstract
In scenarios such as environmental data collection and traffic monitoring, timely responses to real-time situations are facilitated by persistently accessing nodes with revisiting constraints using unmanned aerial vehicles (UAVs). However, imbalanced task allocation may pose risks to the safety of UAVs and potentially lead to failures in monitoring tasks. For instance, continuous visits to nodes without replenishment may damage UAV batteries, while delays in recharging could result in missing task deadlines, ultimately causing task failures. Therefore, this study investigates the problem of achieving balanced multi-UAV path planning for persistent monitoring tasks, which has not been previously researched according to the authors' knowledge. The main contribution of this study is the proposal of two novel indicators to assist in balancing task allocation regarding multi-UAV path planning for persistent monitoring. One of the indicators is namely the waiting factor, which reflects the urgency of a task node waiting to be accessed, and the other is the difficulty level which is introduced to measure the difficulty of tasks undertaken by a UAV. By minimizing differences in difficulty level among UAVs, we can ensure equilibrium in task allocation. For a single UAV, the ant colony initialized genetic algorithm (ACIGA) has been proposed to plan its path and obtain its difficulty level. For multiple UAVs, the K-means clustering algorithm has been improved based on difficulty levels to achieve balanced task allocation. Simulation experiments demonstrated that the difficulty level could effectively reflect the difficulty of tasks and that the proposed algorithms could enable UAVs to achieve balanced task allocation.

1. Introduction
Persistent monitoring requires uninterrupted and continuous surveillance of the target environment to obtain real-time information. Common task scenarios include environmental data collection [1], power line inspections [2], wildfire tracking [3], and rescue operations [4]. Unmanned aerial vehicles (UAVs) are renowned for their high flexibility and ease of deployment, making them valuable assets across industries and various sectors of human life. For example, UAVs can serve as auxiliary devices for data collection, significantly reducing data collection delays [5]. Furthermore, equipped with cameras and diverse sensors, UAVs are ideal for monitoring tasks [6].

Multi-robot systems demonstrate exceptional robustness and scalability, making them highly effective for various applications in dynamic environments. Employing multiple UAVs in large-scale persistent monitoring helps to address the limitations of a single UAV, such as insufficient energy, limited resources, and singular performance, while enabling better task coordination [7]. However, imbalanced task allocation to UAVs often poses significant challenges to the execution of these tasks. For example, overloading poses a threat to the safety of UAVs and, in extreme cases, may result in tasks being unable to be completed at all. Meanwhile, timely replenishment of UAVs is also important in persistent monitoring. Broadly speaking, the recharging methods for UAVs can be classified into dynamic replenishment and static replenishment. The former method involves utilizing unmanned ground vehicles (UGVs) to transport batteries to UAVs to be replaced at rendezvous nodes [8]. However, the mobility of UGVs is limited by the road network, and failing to reach the rendezvous node may lead to task failure. The latter method relies on UAVs returning to a designated base station for energy replenishment at scheduled intervals which is deemed more reliable [9]. This paper focuses on the fairness of task allocation among multiple UAVs, specifically in the situation where UAVs need to return to a fixed base station for replenishment while visiting monitoring nodes with revisit constraints.

On the basics of coverage path planning, which typically involves determining routes to cover an entire area [10], path planning in some persistent monitoring scenarios also focuses on the optimization of monitoring frequency. Hari et al. [11] investigated the problem of UAVs with limited energy for persistent monitoring. It was defined that within one energy cycle, a UAV could only visit K nodes before returning to a depot for energy replenishment. They provided tight lower bounds that help in evaluating the quality of any feasible routes developed. Scott et al. [12] concentrated on path planning in persistent intelligence surveillance reconnaissance missions, treating it as a multi-depot multiple traveling salesman problem (MDTSP), which ensures that the revisit time of each node is the time required to complete the TSP task once. By the Lagrangian algorithm proposed, the coupling constraints in the primal problem were relaxed. Both studies utilized static charging stations for UAV replenishment, enabling persistent monitoring. With UGV serving as mobile charging for UAVs, Lin et al. [13] used multiple UAVs to periodically monitor large areas. However, the proposed method was not suitable for monitoring conditions with strict revisit requirements. Shu et al. [14] incorporated path entropy to enhance the safety of UAV flight routes in persistent monitoring tasks, making it difficult for intruders to detect patrol patterns. Nevertheless, the multi-UAV scenario was not taken into account. Besides, some authors have explored optimizing the number of UAVs for monitoring missions with specific revisit constraints. Feng et al. [15] aimed to determine the minimum number of UAVs for coverage task with dynamic priorities. By the proposed recursive algorithm of the minimum spanning tree, the upper bound of the minimum number of UAVs was determined. However, the energy limit of UAVs was not taken into account. Lien et al. [16] presented an algorithm that considers flight endurance, charging time, and latency constraints in a unified framework to find the minimum number of UAVs. Nonetheless, only cases with same monitor latencies across all points were considered. None of the aforementioned studies have addressed the impact of imbalanced task allocation on the persistent monitoring tasks of multiple UAVs. In fact, imbalanced task allocation could potentially affect the efficiency of monitoring tasks, leading to increased energy consumption and maintenance costs for UAVs.

Balanced task assignment can reduce competition and conflict among robot systems, lower the risk during task execution, and enhance the safety and reliability of overall task performance. Wang et al. [17] aimed to plan the energy-balanced path for multiple ferrying UAVs, effectively addressing the issue of task failure due to energy mismatch in search and rescue scenarios. Various studies aim to achieve task balance within robot groups by minimizing the maximum travel time of robots [18–21], focusing on enhancing algorithms to tackle these challenges. For instance, Huo et al. [18] enhanced the simulated annealing algorithm to address task allocation and path planning issues in 3D vehicle routing problem (VRP). Lee et al. [19] introduced a method that evenly distributes paths to each robot without generating redundant paths, effectively preventing collisions among robots. You et al. [20] devised a multi-angle K-means clustering algorithm for the initial partitioning of the different types of tasks and a two-staged strategy for further balanced task assignment. Guo et al. [21] put forward a novel $5\frac{1}{3}$-approximation algorithm, which is much closer to the optimal. Nevertheless, the differences in task load for each individual robot were ignored in these works. Tang et al. [22] aimed to balance the path length of robots in persistent collaborative coverage tasks. By the proposed feedback mechanism, the bias between the tasks of the robots was greatly reduced. Yu et al. [23] considered balancing the UAVs in scattered region coverage missions. Balanced time consumption of UAVs was achieved through the improved two-staged solution. However, the energy limit of UAVs was not considered. Furthermore, none of the approaches in the above researches are suitable for task scenarios with revisit constraints.

Table I. *Summary of the literature review.*

		Objective	Uniqueness	Drawback
Path planning considering optimizing the monitoring frequency	Ref. [11]	Minimizing the maximum revisit time of monitoring targets.	The dwell time of the UAV at targets, fuel capacity, and the servicing times of the UAV were considered comprehensively.	Multi-UAV situation was not taken into account.
	Ref. [12]	Minimizing the maximum revisit time of monitoring targets.	The presented method was able to produce quality solutions within a preset time limit.	The balance between UAVs was not considered.
	Ref. [13]	Minimizing the maximum revisit time of monitoring areas.	UGVs were used as mobile charging stations for UAVs considering obstacle avoidance.	Not suitable for the task with strict monitoring frequency.
	Ref. [14]	Planning a route considering the monitoring frequency and the randomness of the path.	The security of the monitoring path was improved.	Multi-UAV situation was not considered.
	Ref. [15]	Determining the minimum number of UAVs for coverage problem with dynamic priorities.	The mode could be adjusted to meet the desired monitoring frequency.	The energy limit of UAVs was not considered.
	Ref. [16]	Determining the minimum number of UAVs for persistent monitoring.	The endurance of UAVs, charging time, and latency constraints were first considered in a unified framework.	The different latencies of the monitoring nodes were not taken into account.

Table I. *Continued.*

		Objective	Uniqueness	Drawback
Path planning considering balance between robots	Ref. [17]	Minimizing the planned path length and the variance of all the ferry UAVs' energy factor to obtain balanced missions.	The different initial energy of ferrying UAVs was considered.	The choice of weight coefficients to solve multi-optimization was subjective.
	Ref. [18]	Minimizing the maximum flight distance of UAVs.	The proposed algorithm was more efficient compared with the exact algorithm CPLEX and three representative algorithms.	The difference between the UAVs' task load was ignored.
	Ref. [19]	Minimizing the maximum patrol path of robots.	Avoiding collisions between robots.	A smaller number of robots would result in a more balanced mission since there were no time limits for coverage missions.
	Ref. [20]	Minimizing the maximum cost of robots.	The variability of the execution cost was considered.	The proposed approach may not be able to cope with the limited energy of UAVs.
	Ref. [21]	Minimizing the maximum flight time of UAVs in search and rescue scenarios.	The limited battery of the UAV was taken into account.	The time the UAV spent returning to the depot to change its battery was not considered in the balanced allocations.
	Ref. [22]	Planning balanced paths for robots while ensuring obstacle avoidance and minimizing coverage cycles.	The environmental complexity and path length differences of each robot were taken into account.	The proposed algorithm could be improved to decrease the time spent on the entire problem.
	Ref. [23]	Planning balanced paths for UAVs in coverage tasks.	The scattered coverage regions were considered.	The energy limit of UAVs was not considered.

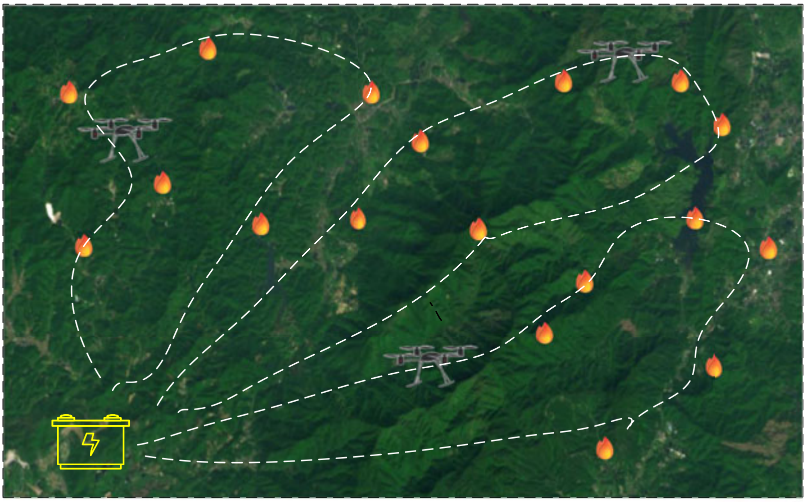


Figure 1. *Persistent monitoring scenario, where multiple UAVs are used to monitor wildfire situations and return to a fixed base station for replacing the battery at the end of their energy cycle.*

A comprehensive table (see Table I) has been made to summarize the features of the aforementioned studies. It can be concluded that few research has been conducted on the balance between multiple UAVs for persistent monitoring tasks with strict revisit constraints while considering the limited energy of UAVs. Therefore, this study investigates balanced multi-UAV path planning for persistent monitoring tasks, where UAVs must return to the base to replace their batteries after completing an energy cycle (see Figure 1). The key contributions are as follows:

1. An assessment method is proposed to balance the tasks among UAVs in persistent monitoring using two innovative indicators. The first indicator is called the ***waiting factor***, which is used to measure the urgency of a node waiting to be revisited after UAVs returned from the base, where a larger waiting factor signifies less urgency and a smaller waiting factor indicates that the node requires prompt visitation. The other one is called the ***difficulty level***, which is introduced to evaluate the task difficulty for each UAV by taking into account the mean and variance of the waiting factors, as well as the number of task nodes.
2. For a single UAV, the ant colony initialized genetic algorithm (ACIGA) algorithm has been proposed to plan its path and calculate its difficulty level. For multiple UAVs, improved K-means clustering algorithm based on difficulty levels has been proposed to achieve balanced task allocation.

The rest of this study is organized as follows: Section 2 introduces the problem background and defines the balance indicators. Section 3 presents the mathematical model used to study the problem. Section 4 introduces ACIGA for planning paths and the improved K-means clustering algorithm for task allocation. Section 5 analyzes the simulation experimental results and evaluates the effectiveness of the algorithms. Lastly, Section 6 concludes the study.

2. Problem descriptions and task balance indicators

2.1. Problem description

To formulate this problem, the set of task nodes that requires persistent monitoring is represented by G. Each task node, denoted as i, has a monitoring period T_i, indicating how often it should be revisited.

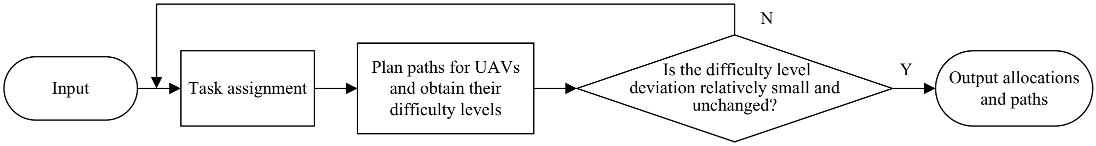


Figure 2. *Research methodology.*

The UAVs all depart from the base s_0 and are capable of visiting a specified set of nodes within one UAV energy cycle before returning to the base for energy replenishment. The algorithmic flow of the proposed method is shown in Figure 2, and this study mainly addresses the following two issues:

1. When there is only one UAV, plan a path that satisfies the monitoring constraints of each node and obtain its difficulty level;
2. When there are multiple UAVs, it is ensured that each node is assigned to only one UAV and task allocation is adjusted based on difficulty levels of UAVs to achieve balanced allocation of tasks.

Assuming that the maximum monitoring step size in one UAV energy cycle is K, Hari et al. [24] proposed a procedure to estimate it considering the relative positions of task nodes and the actual access sequence of UAVs. The difference is that in this paper, in order to ensure that each task node is visited at least once within one energy cycle of UAV, the monitoring step size of UAVs needs to be greater than the number of their assigned task nodes. Since the UAVs discussed in this paper share identical dynamic characteristics, the maximum monitoring step size in one UAV energy cycle (i.e., K) for each UAV remains consistent.

2.2. Task balance indicators

The waiting time for a task node is defined as the elapsed time since it was last visited by a UAV. Upon receiving a new visit by a UAV, the waiting time is reset to zero. The remaining lifespan of a node i is calculated by subtracting its waiting time from the monitoring period T_i. Assuming that the time required for a UAV to replace the battery at the base is fixed as Δ, unexpected situations such as maintenance or malfunction may arise during the process, resulting in the actual replenishing time to exceed Δ. Therefore, it is preferable to avoid particularly urgent task nodes when returning from the base to the task area, ensuring that UAV has enough buffer time for accidents and avoids violating the monitoring constraints of task nodes.

The ***waiting factor*** is introduced in this study to describe the urgency of a node waiting to be revisited when UAVs finish replenishment. And the waiting factor of node i assigned to UAV n is denoted by V_i^n, as shown in equation (1):

$$V_i^n = \frac{T_{is_0}^n}{t_{is_0}^n} \tag{1}$$

where $T_{is_0}^n$ is the remaining lifespan of node i when UAV n completes its energy replenishment at the base. $t_{is_0}^n$ represents the minimum flight time for UAV n from the base to node i. To ensure successful execution of missions, the waiting factor of node i needs to satisfy the following constraint:

$$V_i^n \geq 1 \tag{2}$$

This is because when $0 \leq V_i^n < 1$, the mission will fail even if the UAV n visits node i from the base immediately; when $V_i^n < 0$, UAV n has already violated the monitoring constraint before returning to the base. The larger the value V_i^n is, the less urgent it is for the node i to be accessed; on the contrary, the node should be accessed as soon as possible.

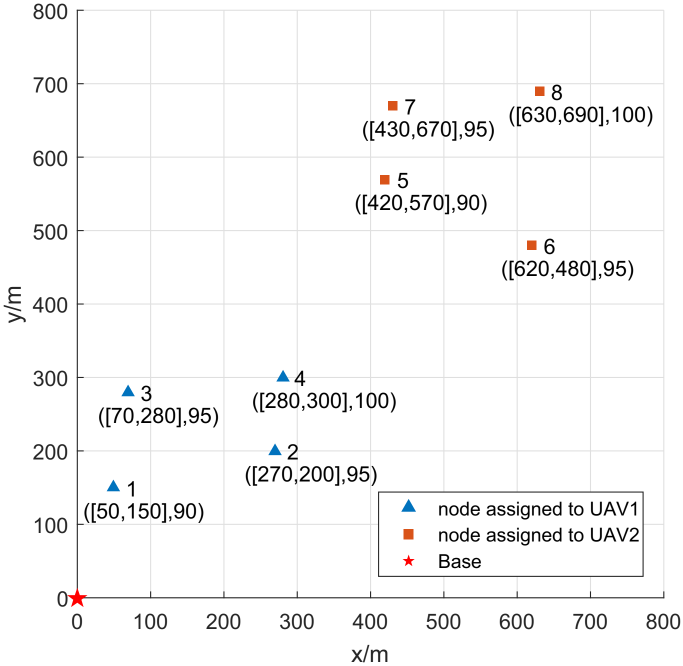


Figure 3. *Example.*

Choosing V_i^n instead of $T_{is_0}^n$ to reflect the urgency of node i will be more conducive to balancing task allocation. This is briefly demonstrated with the example in Figure 3, where the information in parentheses represents the node coordinates (in meters) as well as the remaining lifetime (in seconds) of the node, given that the eight nodes have been divided between two UAVs traveling at a speed of 10 m/s. The average remaining lifespan of the task nodes for both UAVs is the same. However, the average waiting factor for nodes assigned to UAV1 is 3.56, whereas the average waiting factor for nodes assigned to UAV2 is 1.19. In terms of average remaining lifespan, both of them bear similar task difficulty. From the perspective of average waiting factor, UAV2 faces a more difficult task. Intuitively, since the task nodes of UAV2 are further away from the base than UAV1, the actual execution difficulty of UAV2 is greater than that of UAV1. Therefore, it can be concluded that using waiting factors will better reflect UAV's task difficulty and further ensure balanced task allocation.

In addition, there are other factors that need to be considered when assessing the difficulty of UAV tasks. For instance, if all nodes require access simultaneously, the UAV will experience significant pressure. Similarly, a task becomes harder when there are more nodes assigned to the UAV. Therefore, the difficulty of the task could be defined by comprehensively considering the mean and variance of the waiting factors, as well as the number of the nodes assigned to UAV. In this study, the concept of ***difficulty level*** is proposed to quantify the task difficulty of UAVs, which is denoted by C. Therefore, the optimization of C can be regarded as the combination of optimization of the mean, variance, and number of the waiting factors. This optimization problem could be inspired by the mathematical approach commonly used in economics to allocate resources or assets over time while considering the trade-off between the expected return and the variance (or risk) of those returns. This is called dynamic mean variance optimization, which can be divided into three categories [25]. The first two methods require optimizing one variable subject to given before optimizing the other one. The third method simultaneously optimizes a combination of mean $E(\cdot)$ and variance $\sigma^2(\cdot)$, such as Sharpe ratio E/σ or weight combinations

Table II. *List of notations.*

Symbol	Description
N	the quantity of UAVs
G	the set of all nodes
G_n	the set of nodes assigned to the UAV n
K	the maximum steps in one UAV energy cycle
k	the current step of the UAV
$c^n_{k-1,k}$	the time taken by the UAV n from step $k-1$ to k
$f^n_{k,i}$	the waiting time of node i at the k-th step of the UAV n
T_i	the maximum monitoring period of node i
Δ	time for replacing the UAV battery at the base
$T^n_{is_0}$	the remaining lifespan of node i when the UAV n leaves the base
$t^n_{is_0}$	the minimum flight time for the UAV n from node i to the base
v	the speed of the UAV
l_{is_0}	the minimum distance from node i to the base
V^n_i	the waiting factor of node i assigned to the UAV n
$\overline{V_n}$	the mean of waiting factors for the nodes assigned to the UAV n
σ^2_n	the variance of waiting factors for the nodes assigned to the UAV n
C_n	the difficulty level of the UAV n
S_n	the flight time of the UAV n during one energy cycle
W_n	the penalty cost of the UAV n

$E(\cdot) - \beta\sigma^2(\cdot)$, where β is the coefficient. When certain constraints are violated, the negative waiting factor of the task node can lead to a large variance. Therefore, β should be as small as possible, $\beta > 0$. In summary, this paper defines the difficulty level of UAV n as follows:

$$C_n = \frac{|G_n|}{\overline{V_n} + \beta\sigma^2_n} \tag{3}$$

where $\overline{V_n}$ is the mean of waiting factors for all nodes assigned to the UAV n, σ^2_n is the variance, and $|G_n|$ represents the number of task nodes. The higher the difficulty level is (i.e., the larger the value C_n is), the more challenging tasks the UAV n will undertake. Conversely, when the value of C_n is smaller, the task is easier.

It is crucial to emphasize that although the indicators suggested in this study may not hold direct physical significance, the experimental results have demonstrated that these indicators can accurately represent their intended meanings.

3. Persistent monitoring model

Section 3.1 introduces the mathematical model for path planning when there is only one UAV. Section 3.2 presents the mathematical model for balanced task assignment to multiple UAVs. For ease of reference, Table II summarizes the notations introduced in this paper.

3.1. Path planning model

Use $\{1, 2, \ldots, K\}$ to represent the set of discrete steps a UAV takes within one energy cycle, assuming that the UAV starts and ends its trajectory at the base during each cycle. Moreover, it is required that each node is visited at least once, and only one node can be accessed per step. Furthermore, it is not allowed for UAVs to visit the same node during adjacent steps in order to avoid energy waste. The

Boolean variable $y^n_{k,i}$ indicates that UAV n has reached the node i at step k when its value is 1. The above conditions can be expressed as follows:

$$y^n_{1,s_0} = 1, y^n_{K,s_0} = 1 \tag{4}$$

$$\sum_{k=1}^{K} y^n_{k,i} \geq 1, \forall i \in G_n \tag{5}$$

$$\sum_{i=1}^{|G_n|} y^n_{k,i} = 1, k = 1, 2, \ldots, K \tag{6}$$

$$y^n_{k,i} + y^n_{k+1,i} \leq 1, \forall i \in G_n, k = 1, 2, \ldots, K-1 \tag{7}$$

where G_n represents the set of nodes assigned to the UAV n. The waiting time $f^n_{k,i}$ of node i ($i \in G_n$) at the k-th step of UAV n can be defined as:

$$f^n_{k,i} = \begin{cases} 0 & , k = 1 \\ \left(1 - y^n_{k,i}\right)\left(f^n_{k-1,i} + c^n_{k-1,k}\right) & , k = 2, 3, \ldots, 2K \end{cases} \tag{8}$$

where $c^n_{k-1,k}$ is the time taken by UAV n from step $k-1$ to k. To meet the monitoring requirements, the waiting time at each node should satisfy the following constraint:

$$f^n_{k,i} \leq T_i, \forall i \in G_n, k = 1, \ldots, 2K \tag{9}$$

where T_i is the maximum monitoring period of node i. The UAV needs to repeat the same path to achieve persistent monitoring. The upper bound of $2K$ for k in Equation 8 and Equation 9 is justified by the fact that as long as the path remains within the constraints during the UAV's second flight (i.e., from step $K+1$ to $2K$), the route will definitely satisfy the monitoring requirements. Besides, the remaining lifespan of node i when UAV n leaves base $T^n_{is_0}$ and the minimum flight time from node i to base $t^n_{is_0}$ in Equation (1) can be represented as follows:

$$T^n_{is_0} = T_i - f^n_{K,i} - \Delta, \forall i \in G_n \tag{10}$$

$$t^n_{is_0} = \frac{l_{is_0}}{v}, \forall i \in G_n \tag{11}$$

where l_{is_0} represents the linear distance from node i to the base and v is the speed of the UAV n.

For UAV n, its penalty cost W_n for violating constraints is constructed as the sum of overdue time for all monitoring nodes:

$$W_n = \sum_{i=1}^{|G_n|} \sum_{k=1}^{2K} \max\left\{0, f^n_{k,i} - T_i\right\} \tag{12}$$

In order to minimize the urgency of nodes and to realistically reflect tasks difficulty, the chosen route should ensure that the UAV's difficulty level is minimized. Although the flight time of a UAV is constrained by the maximum energy step size K, it is also necessary to take the flight time S into account to avoid UAVs exceeding their actual energy range. The mathematical model for path planning of UAV n can be expressed as follows:

$$\begin{aligned} \min \quad & C_n + \gamma_1 S_n + \gamma_2 W_n \\ \text{s.t.} \quad & \overline{V_n}|G_n| = \sum_{i \in G_n} V^n_i, \sigma^2_n\, |G_n| = \sum_{i \in G_n} \left(V^n_i - \overline{V_n}\right)^2, \\ & \text{(1)-(12)} \end{aligned} \tag{13}$$

where γ_1 ensures that the difficulty level and flight time of UAV are in the same order of magnitude, which depends on the environment and the flying speed of UAV, $\gamma_1 > 0$. In order to reflect the punishment for violating such constraints, the penalty will be increased by a factor of γ_2 times, $\gamma_2 > 0$.

3.2. Task assignment model

If there are N UAVs available, the task assignment serves to allocate unique task nodes to each UAV while ensuring balanced allocations of tasks among all the UAVs. The mathematical model is as follows:

$$\begin{aligned} \min \quad & C_{\max} - C_{\min} \\ \text{s.t.} \quad & G_i \cap G_j = \varnothing, i = 1, 2, \ldots N, j = 1, 2, \ldots N, i \neq j \\ & G_i \neq \varnothing, i = 1, 2, \ldots N \\ & \bigcup_{i=1}^{N} G_i = G \end{aligned} \tag{14}$$

where $C_{\max}$ is the highest difficulty level among UAVs, while $C_{\min}$ is the lowest. By minimizing the difference between the two (i.e., the maximum deviation of difficulty levels), we can achieve balanced task allocation. This approach prevents the situation where $C_{\max}$ is already at its minimum value, while the difference between $C_{\min}$ and $C_{\max}$ is still significant.

4. Algorithm design

To adjust the task allocation to UAVs, we need to plan their paths first. Section 4.1 introduces the ACIGA for planning a path for a UAV to obtain its difficulty level. Section 4.2 explains how to use difficulty levels to improve K-means clustering algorithm in order to obtain balanced task allocations.

4.1. Ant Colony Initialized Genetic Algorithm (ACIGA)

Heuristic-based methods are effective approaches for solving path planning problems [26]. Among these methods, intelligent optimization algorithms such as genetic algorithms (GAs) are commonly used. Furthermore, most of the existing intelligent optimization algorithms require constructing an initial population that meets certain requirements. The path to be sought in our method differs from a typical TSP path, as the length of a TSP path is determined by the number of nodes that UAV needs to visit, whereas the path length in this paper is fixed as K. For UAV n, when $K = |G_n| + 2$, a random arrangement of $|G_n|$ forms a path. When $K > |G_n| + 2$, a feasible approach for constructing a path is to first generate a random arrangement of $|G_n|$ to ensure that all task nodes are accessed. Then, fill the remaining positions with random numbers between 1 and $|G_n|$, satisfying that adjacent positions have different numbers.

However, the initial population generated in this encoding method often violates the monitoring constraints, affecting the search for optimal solutions. In contrast to other intelligent algorithms, the ant colony algorithm constructs paths by selecting the next node based on heuristic probability, eliminating the need for an initial population for optimization and improving the quality. In this study, we build upon the method introduced by [14] and enhance its heuristic function to generate an initial path. The improvement of heuristics is as follows:

$$\eta_{kj} = \frac{f_{k-1,i}}{d_{ij}/v}, k = 2, 3, \ldots, K \tag{15}$$

This encourages the ants to select nodes with longer waiting time and shorter distance from their current position. To ensure that all nodes are visited, the remaining path length and the number of unvisited

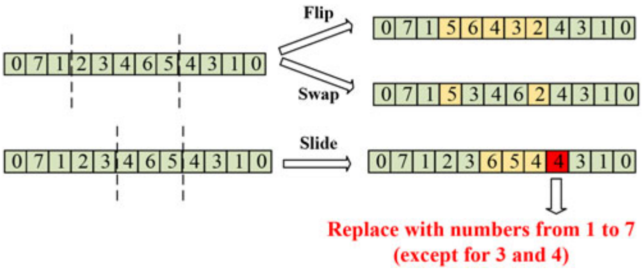


Figure 4. *Mutation methods (assume $K = 12$, $|G_n| = 7$ and 0 represents the base).*

Algorithm 1. ACIGA for planning a path for single UAV.

Input: Task node information assigned to the UAV n
Output: UAV's path and its difficulty level

```
Using the improved ant colony algorithm to obtain the initial population Chrom;
for t = 1 : Max_iter1 do
    Calculate the difficulty level and objective function for each chromosome, save the current
      optimal chromosome;
    Randomly sort the chromosomes in Chrom to form the population pop;
    for p = 4 : 4 : pop_size do
        t0 is the optimal chromosome in the (p–3)-th to the p-th chromosomes in pop;
        Flip t0 to obtain t1;
        Swap t0 to obtain  2;
        Slide t0 to obtain t3;
        Assign t0, t1, t2, and t3 in order to the (p–3)-th to the p-th chromosome of Chrom;
    end
end
```

---

nodes are considered for choosing the next node. If the number of unvisited nodes matches the remaining step length, the unvisited nodes are added directly to the target set. Initializing paths with an enhanced ant colony algorithm reduces the chances of violating constraints and improves the search for optimal solutions in subsequent steps.

While the 2_opt operator utilized in [14] proves to be a potent technique for local search, its efficiency diminishes with increased number of ants and nodes. To avoid this, we limit the use of the ant colony algorithm to the first generation for constructing the initial population. Subsequently, we integrate an enhanced GA characterized by rapid convergence to optimize this initial population. Figure 4 shows the mutation methods involved in GA. The entire ACIGA algorithm is presented in Algorithm 1.

4.2. Improved K-means clustering algorithm

K-means is a widely used clustering algorithm known for its versatility, fast processing, and simplicity. Assigning closer nodes to the same UAV helps to minimize its flight time. As the number of task nodes significantly impacts the UAV's task difficulty, we aim to achieve a balanced allocation by adjusting the number of task nodes assigned to each UAV. Inspired from the method for equalizing the length

Algorithm 2. Improved K-means clustering algorithm for multi-UAV task allocations.

Input: All task node information and the quantity of UAVs
Output: Task node allocation and the paths of each UAV

Generate initial clustering and invoke ACIGA for planning a path of each UAV;
while $C_{\max} - C_{\min} \geq \varepsilon$ **do**
  Update ω based on Equation (16), recluster, and plan a path of each UAV;
end
for $t = 1 : Max\_iter2$ **do**
  Assign a certain number of nodes from the cluster with the lowest difficulty level to the cluster with the highest difficulty level based on their distances from the cluster centers;
  Update the result if it is better than the previous one; otherwise, keep the previous best solution;
end

of robots' path in [22], we redefine the distance between nodes and cluster centers by incorporating difficulty levels in Equation 16, which aims to facilitate balanced task allocation:

$$d^*_{ij} = \frac{d_{ij}}{\omega_i}, i \in center, j \in G \& j \notin center \tag{16a}$$

$$C^*_i = 1/C_i, i = 1, \ldots N \tag{16b}$$

$$\omega_i = \omega_i + p\left(C^*_i - \overline{C^*}\right), i = 1, \ldots N \tag{16c}$$

where d_{ij} represents the actual Euclidean distance from a task node j to the cluster center i, ω_i is the scaling factor of the cluster center i, and d^*_{ij} represents the distance between a task node j and a cluster center i during clustering. p is an amplification factor greater than 0. $\overline{C^*}$ is the mean of C^*. The update method of ω_i indicates that when the UAV assigned to cluster i has a smaller difficulty level, ω_i should be increased accordingly, which means that this cluster will attract more task nodes. Otherwise, it will reduce the number of nodes in the cluster. This improved K-means clustering algorithm is presented in Algorithm 2.

5. Simulation experiment and result analysis

5.1. Single UAV path planning

This subsection presents the experimental results of using ACIGA to plan a path for a single UAV. The UAV's speed is fixed at 10 m/s, and $\Delta = 60$s, $\beta = 0.007$, $\gamma_1 = 0.001$, $\gamma_2 = 1000$. Figure 5 shows the distribution map of 15 task nodes, and Table III lists the monitoring period of each node. The parameter settings for ant colony algorithm refer to [14], $Max\_iter1 = 100$, $pop\_size = 1000$. All experiments in this paper were conducted in MATLAB R2016a.

When $K = 17$, the maximum waiting time of each task node is consistent and equals the sum of one energy cycle flight time S and Δ. Table IV compares the results obtained from different objective functions, which shows that both paths obtained can satisfy the monitoring period constraints. Although the flight time obtained under the proposed objective function is slightly longer, both the mean and variance of waiting factors are larger. This indicates that with a slightly higher cost in flight time, our proposed path ensures that the overall urgency level for visiting all task nodes is reduced, allowing more time for dealing with unexpected situations at the base.

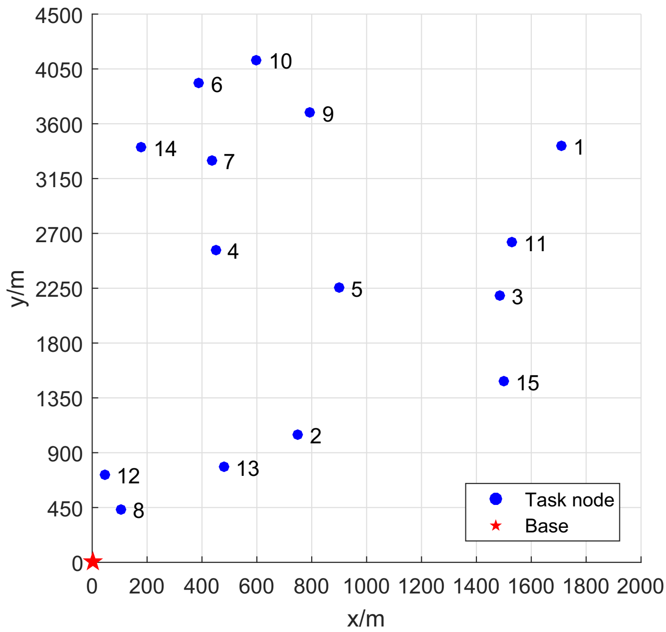


Figure 5. *The locations of nodes.*

The experimental results in Figure 6 demonstrate the effectiveness of ACIGA in solving path problem with $K = 22$. Figure 6(a) illustrates that all nodes have been visited, while Figure 6(c) shows the fast convergence speed of ACIGA. Since the UAV achieves persistent monitoring through a looping path, the maximum waiting time for each task node will be their maximum waiting time during UAV's second flight. Figure 6(b) displays the maximum waiting time of each task node during the UAV's first two flights, confirming the path's compliance with the requirements. For task nodes visited only once within UAV's one energy cycle, their maximum waiting time equals the sum of S and Δ.

To further demonstrate the algorithm's performances, ACIGA was compared to three other algorithms, namely GA without ant colony algorithm initialization, the improved Firefly Algorithm (FA), and the Overdue-aware Ant Colony Optimization (OACO) algorithm [14]. The comparison was based on the results of objective function and time cost of the solution (measured in seconds). Among them, the number of fireflies in FA is 1000, the number of ants in OACO is 100 (the original paper used only 10 ants), and both have an iteration count of 100. The experimental conditions are consistent with Figure 5 and Table II, $K = 22$. Table V displays the comparison between the results of 20 solutions. It can be seen that the proposed ACIGA not only has a relatively faster solving speed but also has better results than other algorithms. These benefits are critical because fast pathfinding is crucial for solving the whole task allocation problem. Furthermore, ensuring minimal variation in objective results for UAVs with the same task allocation helps avoid deception on assignment results and reduce computation time.

5.2. Multi-UAV balanced task allocation

This section will verify the effectiveness of the improved K-means clustering algorithm based on UAVs' difficulty levels and demonstrate that the proposed indicator can represent the difficulty of UAVs' tasks. Assuming that there are 3 UAVs available, 43 nodes need to be monitored, with a step size $K = 30$ for one energy cycle of each UAV. The weight factors ω of the initial cluster centers are all set to 10, and $p = 5$, $\varepsilon = 0.5$, $Max\_iter2 = 10$.

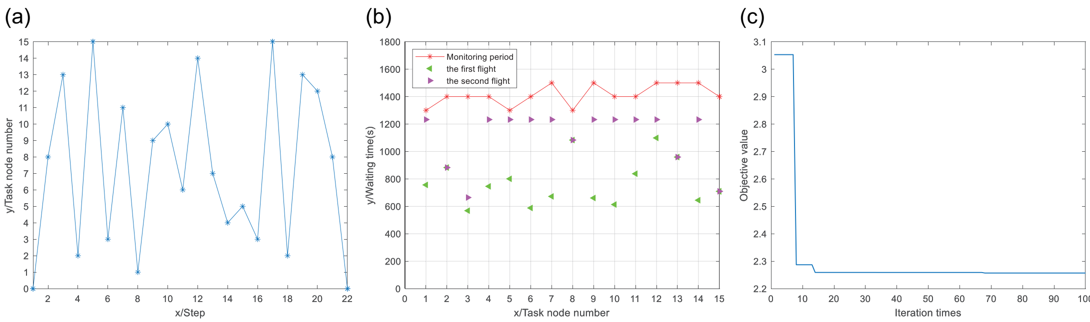
According to Figure 8(a) and Figure 8(b), task nodes assigned to UAV3 with numbers 31, 34, 38, 39, 40, 41, 42, and 43 violate monitoring constraints. It is due to imbalanced task allocation that UAV3

Table III. *The monitoring periods of nodes.*

Monitoring period (s)	Task node
1300	1,5,8
1400	2,3,4,6,10,11,15
1500	7,9,12,13,14

Table IV. *Results of different objective functions.*

Objective function	min $\gamma_{1S} + \gamma_{2W}$	min $C + \gamma_{1S} + \gamma_{2W}$
Flight time S(s)	**1046**	1065
Mean of waiting factor $\overline{V}$	5.56	**6.54**
Variance of waiting factor σ^2	48	**54**



The flight path of UAV.

The maximum waiting time of each node in the first two flights of UAV.

Objective function iteration curve.

Figure 6. *Results of ACIGA with* $K = 22$.

violates the constraints, as we have demonstrated the effectiveness of the path solving algorithm in Section 5.1. Hence, task allocation needs adjustment. The difficulty levels of the three UAVs are 1.6925, 2.8341, and 8.3252, with UAV3 having the highest difficulty level. This demonstrates that the proposed indicator effectively represents the task difficulty of UAVs.

Figure 9 shows the results of the improved K-means algorithm, while Figure 10 depicts variation in the maximum difficulty level deviation with the algorithm. When the green nodes in Figure 9(b) do not overlap with the purple nodes, it indicates that the maximum waiting time for that node occurs when the UAV visits the node, returns to the base for replacing the battery, and then visits the node again. Figure 9(b) and Figure 9(c) demonstrate that as UAVs return from the base to the task area and are required to promptly visit all task nodes first, each UAV encounters some task nodes that have a maximum waiting time close to its monitoring period. This suggests that assigning any additional task node to any UAV at this time would increase the risk of violating the constraints. Thus, the task allocation is balanced. Furthermore, the difficulty levels of the three UAVs are very similar, 3.6316, 3.7745, and 3.7802, respectively. It is important to note that assigning No.3 node in Figure 9(a) to UAV2 may seem more energy efficient. However, this would actually result in a larger maximum difficulty level deviation (i.e., the objective value of Equation 14). Because the maximum deviation still exists between UAV1 and UAV3, with the difficulty level of UAV1 decreasing while the difficulty level of UAV3 remains unchanged. To promote the balance of task allocation among UAVs in this paper, it is reasonable to assign No. 3 node to UAV1.

Table V. Algorithm performance comparison.

Algorithm	ACIGA	GA	FA	OACO
Maximum obj	**2.260**	2.331	3.053	3.053
Minimum obj	**2.257**	2.257	2.259	2.259
Average time cost	21.4	**20.9**	60.6	922.2

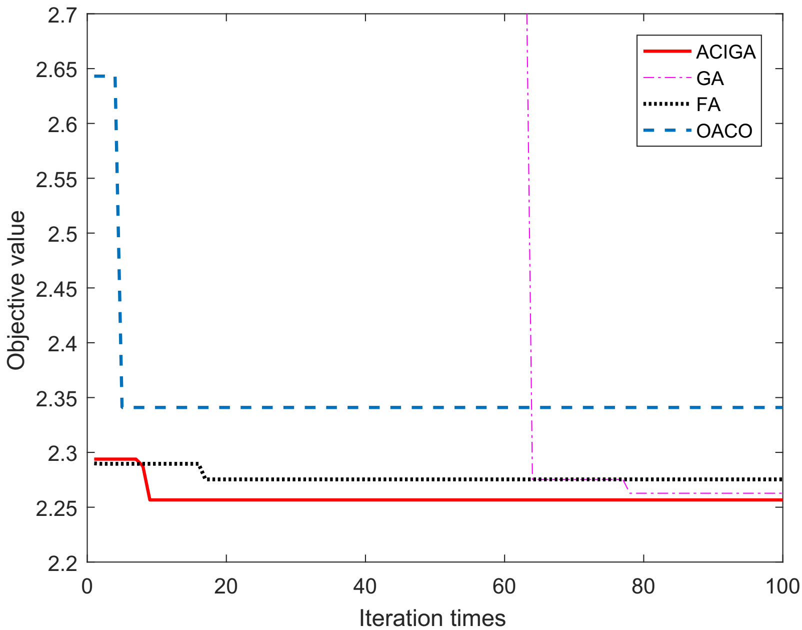


Figure 7. *Comparison of iteration curves from different algorithms.*

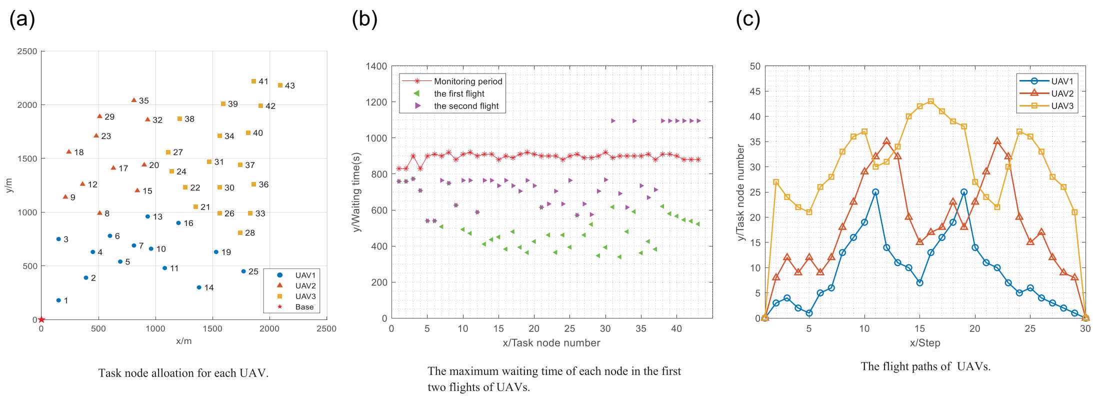
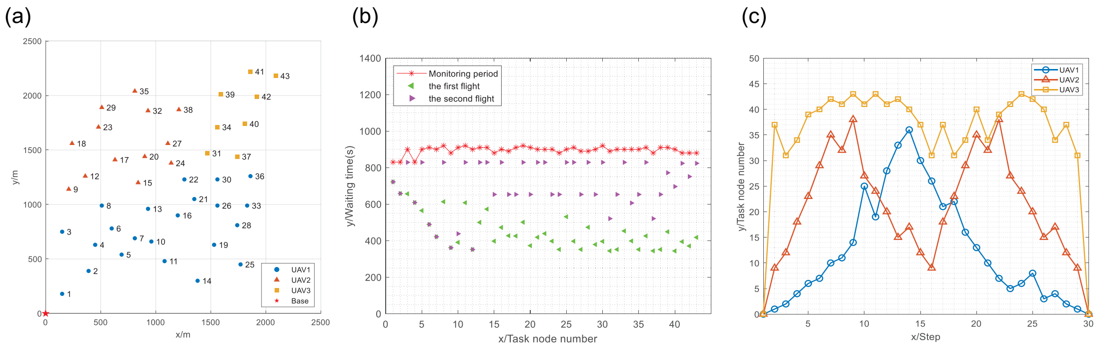


Figure 8. *Results from the ordinary K-means.*

Table VI shows the comparison of the maximum difficulty level deviation under different quantities of UAVs. Due to different initial conditions, the final deviation may vary. However, it can be seen that the algorithm proposed in this paper can effectively reduce the difference to ensure balanced task allocation.



Task node alloation for each UAV.

The maximum waiting time of each node in the first two flights of UAVs.

The flight paths of UAVs.

Figure 9. *Results from the improved K-means.*

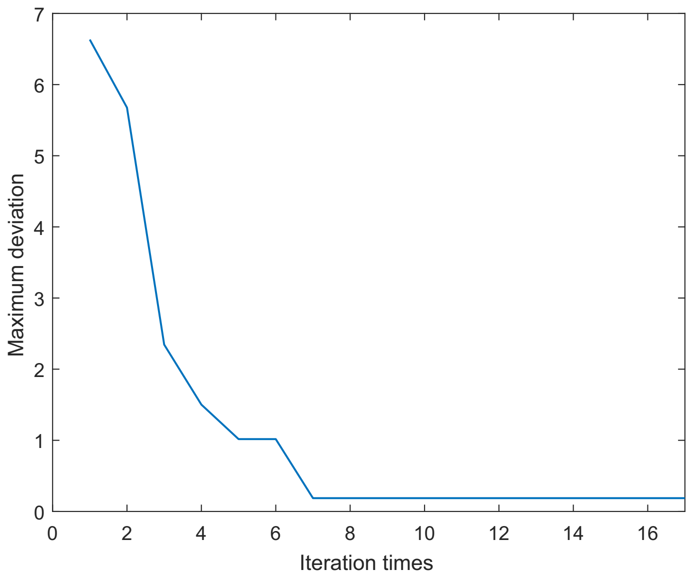


Figure 10. *The variation of maximum deviation.*

6. Conclusion

The utilization of multiple UAVs for conducting persistent monitoring tasks with revisit constraints has been widely adopted. However, UAVs have limited energy and require periodic returns to the base for replacing the battery, resulting in time wastage and the potential for mission delays. Therefore, balanced task assignment for multiple UAVs is crucial for guaranteeing UAV safety and improving task success rates. This study introduces two novel indicators to achieve balanced path planning for persistent monitoring. By considering the distance between the task node and the base along with the remaining lifespan of the node, the waiting factor can more accurately be used to indicate the urgency of visiting an individual task node after UAVs have returned from the base. Then, by taking into account the number of task nodes and the variance and mean of waiting factors, the task difficulty of the UAV can be evaluated. The objective for a UAV is to ensure a minimum difficulty level, allowing sufficient time for accessing task nodes during the return from the base to the mission area. By minimizing the deviation of difficulty levels among UAVs, balanced task allocation is achieved. To address these issues, this study additionally proposes ACIGA for UAV's path planning and an improved K-means clustering method for balanced

Table VI. *Maximum deviation of difficulty level under different quantities of UAVs.*

UAV quantity	3	6	10
Ordinary K-means	6.6327	3.4628	4.2047
Improved K-means	**0.1486**	**0.0750**	**0.3674**

task allocation of multiple UAVs. The experimental results have demonstrated the effectiveness of the proposed algorithms and confirmed that the introduced indicators could assist in achieving balanced multi-UAV path planning for persistent monitoring tasks. However, the dynamics of UAVs and the communications among UAVs, as well as between UAVs and the base, are ignored in this paper. When a UAV malfunctions during the mission, it is crucial to reallocate the remaining UAVs promptly to complete the tasks assigned to the faulty UAV. How to balance the workload of these UAVs is also an important issue to consider.

Author contributions. Xinru Zhan and Yang Chen designed the innovative indicators and the mathematical model. Xi Chen and Wenhao Zhang further developed this design. All four authors contributed to writing this article.

Financial support. The work was financially supported by the National Natural Science Foundation of China (62173262).

Competing interests. The authors declare no conflicts of interest exist.

Ethical approval. Not applicable.

References

[1] K. Messaoudi, O. S. Oubbati, A. Rachedi and T. Bendouma, "UAV-UGV-based system for AOI minimization in IOT networks," *ICC 2023-IEEE International Conference on Communications*, Rome, Italy, IEEE (2023), 2023-May, pp. 4743–4748.

[2] C. Xu, Q. Li, Q. Zhou, S. Zhang, D. Yu and Y. Ma, "Power line-guided automatic electric transmission line inspection system," *IEEE Trans. Instrum. Meas.* **71**, 1–18 (2022).

[3] R. Bailon-Ruiz and S. Lacroix, "Wildfire remote sensing with UAVs: A review from the autonomy point of view," *2020 International Conference on Unmanned Aircraft Systems (ICUAS)*, IEEE (2020) pp. 412–420.

[4] C. Zhang, W. Zhou, W. Qin and W. Tang, "A novel UAV path planning approach: Heuristic crossing search and rescue optimization algorithm," *Expert Syst. Appl.* **215**, 119243 (2023).

[5] K. Messaoudi, O. S. Oubbati, A. Rachedi, A. Lakas, T. Bendouma and N. Chaib, "A survey of UAV-based data collection: Challenges, solutions and future perspectives," *J. Netw. Comput. Appl.* **216**, 103670 (2023).

[6] B. Alzahrani, O. S. Oubbati, A. Barnawi, M. Atiquzzaman and D. Alghazzawi, "UAV assistance paradigm: State-of-the-art in applications and challenges," *J. Netw. Comput. Appl.* **166**, 102706 (2020).

[7] B. Yu, S. Fan, W. Cui, K. Xia and L. Wang, "A multi-UAV cooperative mission planning method based on SA-WOA algorithm for three-dimensional space atmospheric environment detection," *Robotica* **42**(7), 2243–2280 (2024).

[8] Y. Chen, S. Ren, Z. Chen, M. Chen and H. Wu, "Path planning for vehicle-borne system consisting of multi air–ground robots," *Robotica* **38**(3), 493–511 (2020).

[9] A. B. Asghar, S. Sundaram and S. L. Smith, Multi-robot persistent monitoring: minimizing latency and number of robots with recharging constraints, arXiv preprint arXiv:2303.08935 (2023).

[10] M. Sharma and H. K. Voruganti, "Multi-objective optimization approach for coverage path planning of mobile robot," *Robotica* **42**(7), 2125–2149 (2024).

[11] S. K. K. Hari, S. Rathinam, S. Darbha, S. G. Manyam, K. Kalyanam and D. Casbeer, "Bounds on optimal revisit times in persistent monitoring missions with a distinct and remote service station," *IEEE T. Robot.* **39**(2), 1070–1086 (2022).

[12] D. Scott, S. G. Manyam, D. W. Casbeer and M. Kumar, "A lagrangian algorithm for multiple depot traveling salesman problem with revisit period constraints," *IEEE T. Autom. Sci. Eng.* **20**(1), 690–702 (2022).

[13] X. Lin, Y. Yazıcıoğlu and D. Aksaray, "Robust planning for persistent surveillance with energy-constrained UAVs and mobile charging stations," *IEEE Robot. Autom. Lett.* **7**(2), 4157–4164 (2022).

[14] Y. Shu, Y. Chen, M. Hu, H. Wu and X. Zhao, "UAV path planning based on simultaneous optimization of monitoring frequency and security," *2022 34th Chinese Control and Decision Conference (CCDC)*, Piscataway, NJ, IEEE (2022) pp. 3808–3814.

[15] L. Feng and J. Katupitiya, "UAV-based persistent full area coverage with dynamic priorities," *Robot. Auton. Syst.* **157**, 104244 (2022).

[16] J.-M. Lien, S. Rodriguez and M. Morales, "Persistent covering with latency and energy constraints," *IEEE Robot. Autom. Lett.* **6**(2), 998–1003 (2021).

[17] L. Wang, X. Zhang, P. Deng, J. Kang, Z. Gao, L. Liu, "An energy-balanced path planning algorithm for multiple ferrying UAVs based on GA," *Int. J. Aerospace Eng.* **2020**, 3516149 (2020).

[18] L. Huo, J. Zhu, G. Wu and Z. Li, "A novel simulated annealing based strategy for balanced UAV task assignment and path planning," *Sensors* **20**(17), 4769 (2020).

[19] S. Lee, "A multi-robot balanced coverage path planning strategy for patrol missions," *2021 21st International Conference on Control, Automation and Systems (ICCAS)*, Jeju, Korea, IEEE (2021) pp. 1567–1569.

[20] J. You, J. Jia, X. Pang, J. Wen, Y. Shi and J. Zeng, "A novel multi-robot task assignment scheme based on a multi-angle k-means clustering algorithm and a two-stage load-balancing strategy," *Electronics* **12**(18), 3842 (2023).

[21] Q. Guo, J. Peng, W. Xu, W. Liang, X. Jia, Z. Xu, Y. Yang and M. Wang, "Minimizing the longest tour time among a fleet of UAVs for disaster area surveillance," *IEEE T. Mobile Comput.* **21**(7), 2451–2465 (2020).

[22] Y. Tang, R. Zhou, G. Sun, B. Di and R. Xiong, "A novel cooperative path planning for multirobot persistent coverage in complex environments," *IEEE Sens. J.* **20**(8), 4485–4495 (2020).

[23] X. Yu, S. Jin, D. Shi, L. Li, Y. Kang and J. Zou, "Balanced multi-region coverage path planning for unmanned aerial vehicles," *IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Piscataway, NJ, IEEE (2020) pp. 3499–3506.

[24] S. Hari, S. Rathinam, S. Darbha, K. Kalyanam, S. Manyam and D. Casbeer, "Optimal UAV route planning for persistent monitoring missions," *IEEE T. Robot.* **37**(2), 550–566 (2020).

[25] L. Xia, "Risk-sensitive markov decision processes with combined metrics of mean and variance," *Prod. Oper. Manag.* **29**(12), 2808–2827 (2020).

[26] J. S. Tan, J. Gao, H. Wang, H. Zhang and Y. Zhang, "Multi-UAV path planning based on IB-ABC with restricted planned arrival sequence," *Robotica* **41**(4), 1244–1257 (2023).

Cite this article: X. Zhan, Y. Chen, X. Chen and W. Zhang, "Balanced Multi-UAV path planning for persistent monitoring", Robotica. https://doi.org/10.1017/S0263574724001899