

Article

A Study on Drone Logistics Delivery Based on Multi-Center Routing

Yong Yang ^{1,2,*} , Yujie Fu ^{1,2}, Bowen Wang ^{1,2}, Kaijun Xu ^{1,2} and Weiqi Feng ^{1,2} 

¹ School of Flight Technology, Civil Aviation Flight University of China, Guanghan 618307, China; fuyujie@cafuc.edu.cn (Y.F.); wangbowen@cafuc.edu.cn (B.W.); xukaijun@cafuc.edu.cn (K.X.); fengweiqi@cafuc.edu.cn (W.F.)

² Sichuan Provincial Engineering Research Center of Domestic Civil Aircraft Flight and Operation Support, Chengdu 610021, China

* Correspondence: yangyong@cafuc.edu.cn

Highlights

What are the main findings?

- For UAV logistics delivery in complex mountainous terrain, a two-stage collaborative optimization framework has been proposed, which integrates multi-site location selection, local trajectory planning, and task scheduling within a multi-site UAV logistics system.
- For local trajectory planning, a Modified Starfish Optimization Algorithm (MISFOA) incorporating a hybrid strategy of chaotic initialization and elite SA-DE local enhancement is proposed. On validations against benchmark functions and path-planning optimization, its performance shows a significant improvement over the original SFOA.

What are the implications of the main findings?

- The proposed framework provides an effective and scalable optimization method for multi-hub UAV logistics systems in mountainous terrain subject to constraints such as terrain, payload, time windows and capacity.
- This study extends the application of the Modified Starfish Optimization Algorithm to the field of UAV logistics planning and provides a practical foundation for future in-flight validation and real-world low-altitude logistics deployment.

Abstract

With the rapid growth in e-commerce demand, increasing pressure on same-day delivery, and rising last-mile logistics costs, UAV-based logistics systems have emerged as a promising solution for efficient transportation in complex environments. In mountainous regions, however, irregular terrain, limited infrastructure accessibility, and strict flight constraints significantly increase the difficulty of logistics planning. To address these challenges, this study proposes a two-layer collaborative optimization framework for multi-center UAV logistics delivery systems. At the lower level, a multi-center site selection model was developed to determine the optimal distribution center locations and assign task areas. A trajectory cost matrix was constructed by comprehensively considering multiple constraints. The model was solved using a hybrid strategy that combines chaotic initialization and local enhancement based on the elite saDE method to improve the Starfish Optimization Algorithm, called the Mixed-Strategy Improved Starfish Optimization Algorithm (MISFOA), thereby generating feasible three-dimensional flight trajectories between local nodes. At the upper level, an improved Adaptive Large Neighborhood Search (IALNS) algorithm



Academic Editor: Mumtaz Karatas

Received: 25 May 2026

Revised: 28 June 2026

Accepted: 28 June 2026

Published: 1 July 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

is applied to perform UAV mission assignment and route scheduling within each distribution center, based on the trajectory cost matrix pre-calculated at the lower level. The proposed framework achieves effective information exchange and hierarchical coupling between center selection and scheduling at the distribution level, thereby enabling unified optimization of the multi-center location and coordinated dispatch system. Simulation results demonstrate that the proposed method significantly improves delivery efficiency and solution quality in complex mountainous environments while ensuring trajectory feasibility and operational safety. This model provides a scalable and practical optimization framework for low-altitude logistics network planning under complex constraints.

Keywords: logistics; UAV; multi-center site; mixed strategy-improved starfish optimization algorithm; improved adaptive broad-domain search algorithm

1. Introduction

Urban logistics and delivery have long served as a vital foundation for the platform economy and internet-based consumption [1]; however, with the rapid development of the low-altitude economy and smart logistics, the demand for drone delivery in complex terrain has become increasingly prominent. Particularly in mountainous areas, river valleys and remote villages and towns, traditional ground transport modes are easily constrained by road conditions, undulating terrain and the adequacy of infrastructure, leading to issues such as significant detours, lengthy delivery times and high overall costs [2]. In contrast, drones possess advantages such as strong obstacle-crossing capabilities, flexible deployment and rapid response times, and are becoming a key technological enabler for last-mile logistics in complex environments. Recent review studies also indicate that logistics and delivery, emergency transport and autonomous navigation in complex environments have become key application areas in research on drone path planning and collaborative scheduling.

In recent years, research into the application of drones in logistics and delivery systems has yielded significant results, with the focus gradually shifting from simple last-mile delivery to key aspects such as hub layout, collaborative delivery and route scheduling. At the logistics network level, in 2023, Rave et al. developed a two-level site selection and routing model incorporating a hybrid delivery system of trucks and drones. They integrated the location of micro-distribution centers and fleet configuration into a unified optimization framework, and employed a problem-specific large-neighborhood search method to solve the problem [3]. In 2026, to address the issues of transportation times affected by traffic conditions and time-constrained truck–drone hybrid delivery systems, Shuaibu et al. proposed an optimization framework based on reinforcement learning, supported by an exact mixed-integer linear programming (MILP) model. Within this framework, drones operate in coordination with trucks under constraints related to flight range and synchronization, and the objective function penalizes delivery delays to ensure compliance with time-window requirements [4]. In 2024, Peng et al. further investigated the multi-distribution-center electric vehicle–drone collaborative delivery problem, incorporating time-varying traffic conditions, charging times, time windows, payload, and range constraints into a unified model, and solved it using an improved adaptive large-neighborhood search algorithm [5]. In 2024, Meng et al. addressed the multi-visit drone collaborative delivery problem with soft time windows and random truck travel times, constructing an optimization model aimed at minimizing total cost and designing corresponding iterative route construction and improved algorithms [6]. In 2025, Dai et al. highlighted, from the perspective of urban low-altitude delivery network layout, that if actual flight path distances are used in place

of traditional straight-line distances for the siting of distribution centers, the location and layout of the centers will undergo significant changes, indicating that actual path costs have a substantial impact on the layout of UAV distribution centers [7]. Lakhwani in 2025 proposed a Fifth-Party Logistics (5PL)-driven coordination framework to address the lack of integration between autonomous delivery systems and digital logistics platforms. This framework incorporates drones as intelligent execution agents into a cloud-driven coordination layer, supported by artificial intelligence, the Internet of Things, and blockchain technology to enable real-time coordination, adaptive route planning, and end-to-end supply chain optimization under dynamic operational conditions [8].

In terms of problem-solving methods, the current optimization approaches for UAV path planning and collaborative delivery models can broadly be categorized into deterministic graph search, meta-heuristic and swarm intelligence optimization algorithms, and artificial intelligence and machine learning methods [9]. Among these, deterministic graph search methods offer the advantages of clear theoretical foundations, high path interpretability and high solution accuracy; however, in high-dimensional, non-convex and complex constraint scenarios, their efficiency is reduced due to issues such as high computational complexity and insufficient real-time performance. In contrast, meta-heuristic and swarm intelligence optimization algorithms simulate biological behavior, physical processes or group collaboration mechanisms found in nature to progressively search the solution space and improve solution quality. As they do not rely on explicit gradients, demonstrate strong adaptability to complex non-convex problems and possess good global search capabilities, they are widely applied in research on UAV path planning and collaborative delivery. However, as the task environment has expanded from two-dimensional static scenarios to three-dimensional complex spaces and multi-UAV collaborative scenarios, path planning algorithms must not only accommodate range, energy consumption and safety constraints, but also balance solution accuracy, computational efficiency and real-time adaptability. Meta-heuristic and swarm intelligence optimization methods are widely applied in UAV path planning and collaborative delivery research due to their independence from explicit gradients, strong adaptability to complex non-convex problems, and superior global search capabilities [10]. In recent years, researchers have proposed many improved intelligent optimization algorithms focusing on path planning in continuous spaces, obstacle avoidance in complex environments, and multi-objective cost optimization. In 2025, Wang et al. addressed issues such as search efficiency, convergence accuracy, and the tendency to get stuck in local optima in multi-UAV path planning under complex environments by proposing a reinforcement learning-driven multi-strategy continuous ant colony optimization algorithm. By introducing strategy guidance and continuous space search mechanisms, they improved the quality and stability of path planning in complex environments [11]. In 2025, Cheng et al. addressed the issues of slow convergence and insufficient local search capabilities in traditional single-element heuristic algorithms for 3D UAV path planning in complex environments. They proposed a multi-stage hybrid meta-heuristic framework that, by sequentially integrating particle swarm optimization with other local improvement mechanisms, achieved a simultaneous improvement in path quality and solution robustness in complex scenarios [12]. AI/ML methods demonstrate significant potential in dynamic environment perception, autonomous decision-making and online adjustment, but still face challenges such as high training costs, insufficient engineering interpretability and limited generalization capabilities.

In summary, existing research has primarily focused on urban or flat terrain environments, where transportation modes typically involve either coordinated truck-drone operations or single-drone delivery systems under single-hub configurations. However, there remains a lack of comprehensive characterization of terrain undulations, no-fly zone

constraints, and three-dimensional flight path costs in low-altitude mountainous airspace. Furthermore, existing studies often treat hub location, flight path planning, and task scheduling as independent processes, making it difficult to capture the inherent interdependencies among these components at a system level; moreover, many models still use Euclidean distance or road distance as a proxy for the actual flight path cost, which can lead to discrepancies between site selection results and actual delivery costs in complex terrain scenarios. A comparison of existing studies is shown in Table 1.

Table 1. Comparative study.

Problem-Solving Tools	Environment	Constraint	Author
Dijkstra, A*, RRT* and improved versions	3D environmental mapping and obstacle avoidance	The path is highly interpretable and offers high local solution accuracy	Sushma, M. B. et al. [13]
Swarm intelligence, meta-heuristics and hybrid optimization algorithms	Complex three-dimensional environments, dynamic obstacles, multi-objective constraints	Does not rely on gradients, is suitable for complex non-convex problems, and has strong global search capabilities	Li, J et al. [14]
Reinforcement learning, deep learning, data-driven optimization	Dynamic environments, autonomous decision-making, online scheduling	Strong online perception and dynamic decision-making capabilities, suitable for real-time re-planning	Bacha, Anis Mahmoud [15]
Improved intelligent optimization algorithm	Mountainous terrain, moving obstacles, energy consumption and safety risks	Can handle complex terrain and multi-constraint path costs effectively	Wang, Wentao [16]

Considering the aforementioned issues, this paper proposes a two-tier optimization framework for unmanned aerial vehicle (UAV) logistics delivery systems operating in complex mountainous terrain. The framework comprises a multi-hub site selection and local trajectory planning layer, and an intra-hub task scheduling layer. At the lower level, multi-center site selection and local trajectory planning are performed concurrently. Task points are assigned to distribution centers based on spatial costs, and a 3D trajectory cost matrix is constructed using terrain elevation, safety constraints, and flight feasibility conditions. This matrix encodes the actual traversal costs between any two nodes within each center and serves as a bridge between the geometric space and operational scheduling. At the upper level, the task scheduling problem within each center is modeled as a combinatorial optimization problem, in which drones are assigned to task sequences based on the pre-computed trajectory cost matrix, subject to constraints such as capacity, time windows, and task uniqueness. Thus, the lower level provides a spatial cost structure, while the upper level performs discrete decision optimization, thereby forming a hierarchical, feedback-driven coupling mechanism, as shown in Figure 1.

This research builds upon and extends previous work, employing optimization algorithms to extend fixed-hub logistics delivery to a multi-hub logistics delivery system, thereby better accommodating practical requirements. The main innovations are as follows:

(1) A two-level collaborative optimization framework suitable for drone logistics delivery systems in complex mountainous terrain is proposed, achieving collaborative optimization of multi-hub site selection, task assignment, local trajectory planning, and intra-hub task allocation and scheduling;

(2) A hybrid strategy is proposed to improve the Starfish optimization algorithm by integrating chaotic initialization with the elite saDE local enhancement mechanism for local trajectory planning under complex constraints;

(3) A multi-center scheduling mechanism based on a local trajectory cost matrix is constructed, enabling the allocation of UAV tasks and path scheduling within each center to be aligned with the actual flight costs at the lower level.

Multi-Center Location and Coordinated Dispatch framework

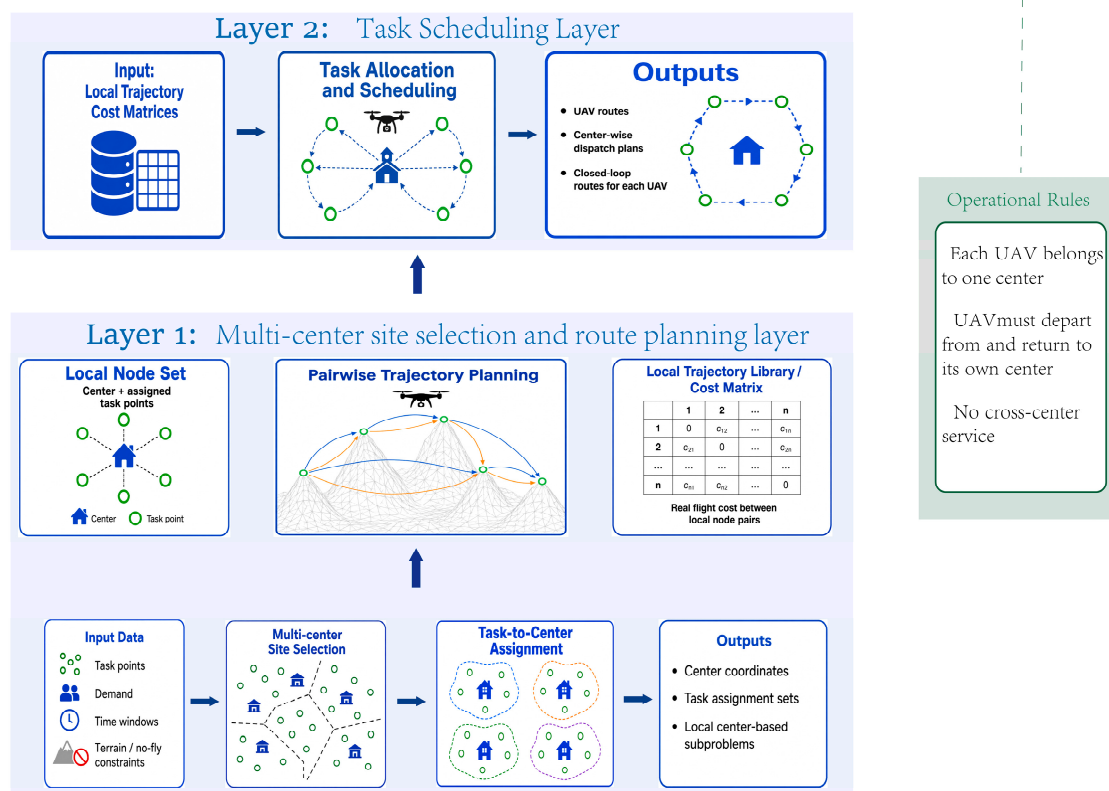


Figure 1. Schematic diagram of the research process.

2. Modeling

2.1. Model Assumptions

This study investigates a two-level model for the Multi-Center Location and Coordinated Dispatch Problem (MLCDP). To better characterize the relationship between site selection, flight paths and task scheduling within a logistics system, the following assumptions are made regarding the model:

- (1) A multi-center distribution model is adopted, whereby candidate distribution centers may be located at any point on a continuous terrain plane, provided they are not within no-fly zones or obstacles.
- (2) The optimal number of distribution centers is determined by back-calculation based on the total task demand.
- (3) Each task point is assigned to only one distribution center.
- (4) The location coordinates, task payload, and time window of each task point are known and remain constant throughout the planning period.
- (5) Each task point must be served exactly once by a single drone; each drone takes off from a distribution center, executes its assigned task sequence, and must ultimately return to the distribution center.

- (6) The upper and lower limits of construction costs for all distribution centers are identical and known; their maximum task capacity is also identical and known.
- (7) All drones are of the same type, i.e., they share identical parameters, such as payload, endurance, safety distance and pitch angle.
- (8) The dispatch cost of each drone is known.
- (9) There is a sufficient number of drones available; the optimal number of drones to be used is determined by working backwards from the task requirements.
- (10) Weather conditions such as wind speed, thunderstorms and smog are not taken into account.

2.2. Key to Symbols

The variables used in this chapter and their meanings are shown in Table 2.

Table 2. Variable symbols and their meanings.

Type	Variable Symbol	Meaning of Variables
Assembly	G	Distribution center hub, $G = \{1, 2, 3, \dots, u\}$ $g \in G$
	I	Gather at the task point, $I = \{1, 2, 3, \dots, n\}$
	I_g	The set of task points belonging to the center g
	L	Waypoint set, $L = \{1, 2, 3, \dots, m\}$
	V	The set of task points and center points, $V = I \cup G$
	K	Drones, assemble, $K = \{1, 2, 3, \dots, b\}$
Parameters	(x_g, y_g, z_g)	The coordinates of the distribution center requiring optimization, $g \in G$
	$i = (x_i, y_i, z_i)$	Task point coordinates
	d_{ig}	The straight-line distance between drop-off point i and distribution center g
	d_{ij}^g	The straight-line distance between task point i and task point j at distribution center g
	L_i	The Euclidean distance between waypoint l and waypoint $l + 1$
	q_i	Number of tasks at the task point i
	p_c	The maximum number of tasks a drone can perform
	g_c	The maximum workload that a single distribution center can handle
	w_i	Time taken to reach the task point i
	$w_{i,1}, w_{i,2}$	The left and right time windows of the task point i
	$x_{\max}, x_{\min}$	Upper and lower bounds of the candidate rectangle's x-axis
	$y_{\max}, y_{\min}$	Upper and lower limits of the y-boundary of the candidate rectangle
	(x_c, y_c)	Center of the candidate rectangle
	h_x, h_y	Half-width and half-length of a rectangular bounding box
	$z_{\max}, z_{\min}$	Upper and lower limits of terrain surface elevation
	$B_{\max}, B_{\min}$	Upper and lower limits on distribution center construction costs
	ρ_g	Offset between distribution center g and the center of the candidate rectangle
	δ	Weights of cost items in the site selection model
	α	Weights of cost terms in the flight path planning model
Decision variables	$x_{ij}^{kg} \in \{0, 1\}$	If drone k , belonging to center g , flies from waypoint i to waypoint j , then $x_{ij}^k = 1$; otherwise, $x_{ij}^k = 0$
	$y_g \in \{0, 1\}$	If candidate point g is selected as the distribution center, then $y_g = 1$; otherwise, $y_m = 0$
	$z_{ig} \in \{0, 1\}$	If task point i is assigned to center g , then $z_{ig} = 1$; otherwise $z_{ig} = 0$

2.3. Mathematical Model

To investigate the coordinated optimization relationships between distribution center location, task point assignment, route planning and task schedule in the Multi-Center Location and Coordinated Dispatch Problem (MLCDP), this paper constructs a two-level

optimization model comprising a multi-center location and local route planning layer, and an intra-center task dispatch layer. The overall objective function of the system is defined as shown in Equation (1):

$$J(G, L, T) = fit_{site}(G) + fit_{route}(G, L) + fit_{task}(G, T) \quad (1)$$

Here, J represents the overall objective function of the system; $fit_{site}(G)$ represents the site selection cost function corresponding to the layout of multiple distribution centers and the assignment of task points; $fit_{route}(G, L)$ represents the flight cost function generated by local trajectory planning under the distribution center layout G ; $fit_{task}(G, T)$ represents the task allocation and collaborative scheduling cost function at the task schedule layer, under the distribution center G layout and trajectory cost matrix conditions. This model utilizes the local trajectory cost matrix to facilitate data exchange between the multi-center site selection layer, the local trajectory planning layer, and the intra-center task scheduling layer, thereby achieving the overall optimization of multi-center layout, local trajectory optimization, and collaborative scheduling under a unified objective function.

2.3.1. Multi-Center Site Selection and Local Path Planning Layer

A single distribution center struggles to meet the demands of efficient delivery in logistics scenarios involving large coverage areas and high volumes of tasks; consequently, a multi-distribution-center layout has become a key strategy for reducing service distances, improving delivery efficiency and lowering system operating costs. This paper first constructs an outer bounding box based on the planar coordinates of all task points and then expands this by 5% to obtain candidate areas for distribution centers. Within this continuous candidate area, several sets of initial positions for distribution centers are generated, and small perturbations are applied to form multiple sets of initial solutions. Subsequently, a multi-starting-point local optimization strategy is employed to refine the continuous coordinates of the multi-center layout. For each set of candidate center solutions, the system first calculates the service distance from each task point to all distribution centers and assigns task points to centers according to the nearest-center principle. On this basis, each distribution center calculates the distance cost, elevation-based operational cost, construction cost, and transport cost for its assigned task points only. The costs for each center are then summed up to obtain the comprehensive objective value for the current candidate solution, as shown in Figure 2.

The mathematical model for site selection is shown in Equation (2):

$$fit_{site}(G) = \min \sum_{g \in G} \left(\delta_1 H_{Length}^g + \delta_2 H_{Elev}^g + \delta_3 H_{Build}^g + \delta_4 H_{Trans}^g \right) \quad (2)$$

Here, H represents the objective function for site selection, H_{Length}^g represents distance costs, H_{Elev}^g represents the operational and maintenance costs of the distribution center, H_{Build}^g represents the construction costs of the distribution center, and H_{Trans}^g represents transport costs, as shown in Equations (3)–(7). In this study, a weighted aggregation strategy is adopted to balance the influence of each cost component, where the weighting coefficients are set as $\delta_1 = 0.35$, $\delta_2 = 0.25$, $\delta_3 = 0.35$, and $\delta_4 = 0.05$:

$$g = \left\lceil \frac{\sum_{i \in I} q_i}{g_c} \right\rceil \quad (3)$$

$$H_{Length}^g = \frac{1}{\sum_{i \in I} z_{ig}} z_{ig} d_{ig} \quad i \in I, g \in G \quad (4)$$

$$d_{ig} = \sqrt{(x_g - x_i)^2 + (y_g - y_i)^2} \quad i \in I, g \in G \quad (5)$$

$$H_{Elev}^g = \frac{z_g - z_{min}}{z_{max} - z_{min}} \quad g \in G \quad (6)$$

$$H_{Build}^g = B_{min} + \rho_g (B_{max} - B_{min}) \quad g \in G \quad (7)$$

$$H_{Trans}^g = \sum_{i \in I} z_{ig} q_i d_{ig} \quad g \in G \quad (8)$$

$$h_x = \frac{x_{max} - x_{min}}{2} \quad (9)$$

$$h_y = \frac{y_{max} - y_{min}}{2} \quad (10)$$

$$\rho_g = \frac{1}{\sqrt{2}} \sqrt{\left(\frac{x_g - x_c}{h_x}\right)^2 + \left(\frac{y_g - y_c}{h_y}\right)^2} \quad g \in G \quad (11)$$

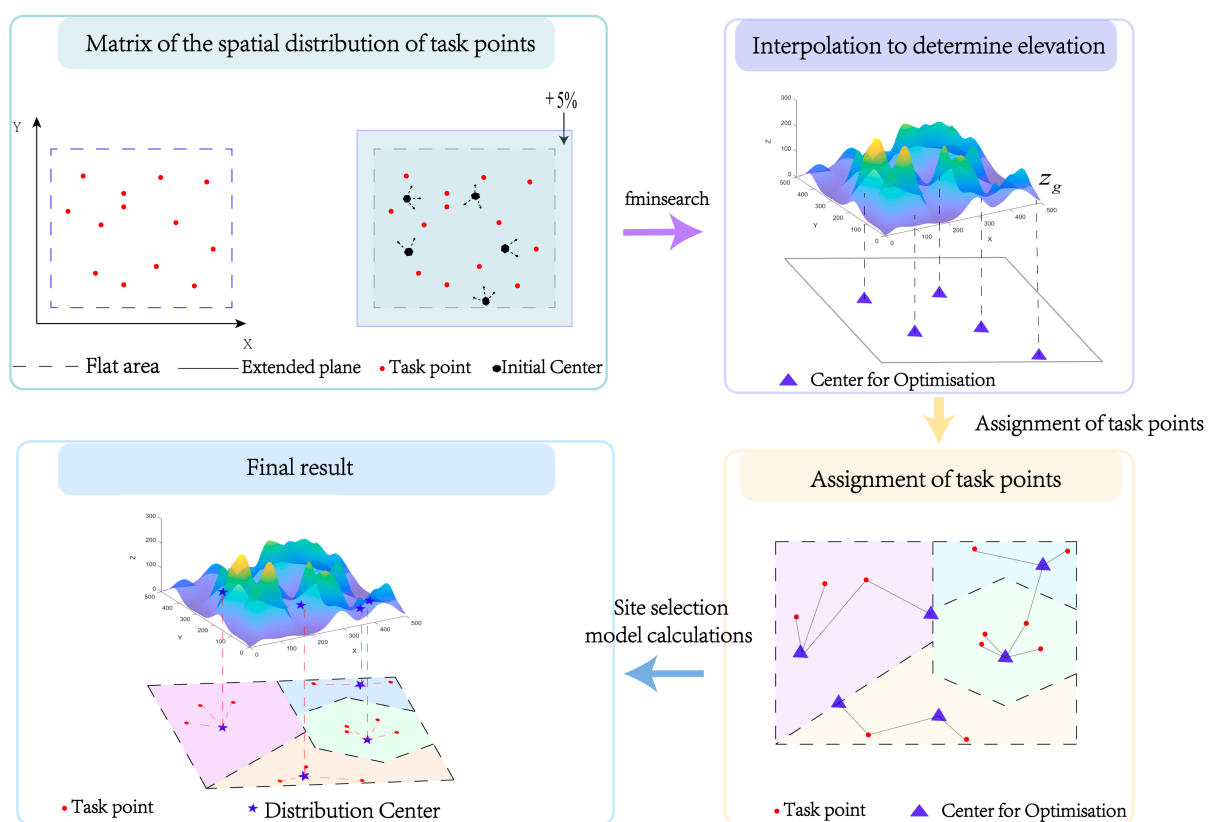


Figure 2. Hub selection schematic.

In this context, Equation (4) represents the distance cost, denoting the average service distance from center g to its assigned task points; the smaller this value, the closer the distribution center is to the overall center of gravity of the task area, and the shorter the average service distance. z_{ig} is the task point assignment variable. In mountainous terrain, higher distribution center elevations typically entail higher expenditure on equipment maintenance and site construction. By normalizing the elevation of the distribution center's location, the operational and maintenance costs are defined as shown in Equation (6). A higher value indicates higher operational and maintenance costs, allowing the optimization process to proactively avoid unnecessary high-altitude risks. Considering that both task volume and service distance jointly influence the distribution center's transport expenditure, transport costs are defined as shown in Formula (8). These represent the

operational costs required for drones to complete deliveries to all task points, influenced by the distribution center's location. Transport costs are correlated with both demand distribution and spatial distance distribution; they do not solely favor areas with dense task points but low task volume. Consequently, operational costs are calculated as the sum of the product of each task point's volume and its service distance to the distribution center [17]. The greater the offset of the distribution center from the geometric center of the candidate area, the higher the investment required for construction organization and infrastructure. When the distribution center is close to the geometric center of the task area, construction and material organization are relatively convenient, leading to lower costs; conversely, costs increase. Here, h_x and h_y denote the half-width and half-length of the rectangular boundary, i.e., the maximum distance that can be reached by moving from the center of the area along the E-axis or F-axis to the rectangular boundary; the offset ρ is shown in Equations (9)–(11).

Route planning involves generating feasible 3D flight paths between any two nodes based on sets of local nodes at each distribution center, taking into account terrain elevation, safe altitude, boundary constraints and flight performance limitations. The results of this local planning are then saved as a trajectory cost matrix for use by the subsequent task scheduling layer. The mathematical models for the site selection and route planning layers are shown in Equations (12) and (13):

$$F_{\text{router}}^g(G, L) = \left[F_{\text{Length}}^g(i, j), F_{\text{Space}}^g(i, j), F_{\text{height}}^g(i, j), F_{\text{Pitch}}^g(i, j), F_{\text{Threat}}^g(i, j) \right] \quad (12)$$

$$\text{fit}_{\text{router}}(G, L) = \min \sum_{i=1}^5 \alpha_i F + \text{restion}_1 \cdot 100 \quad (13)$$

In this context, F_{height} represents the cost of flight altitude; F_{Threat} represents the cost of threat zones, including sensitive areas L , F_{Space} restricted airspace and other high-risk areas; F_{Pitch} represents the cost of pitch angle; F_{Length} represents the flight distance constraint; restion_1 represents the sum of terrain constraints and boundary constraints; and $\text{fit}_{\text{route}}$ represents the overall objective function for the site selection and route optimization layer [18]. In this study, a weighted aggregation strategy is adopted to balance the influence of each cost component, where the weighting coefficients are set as $\alpha_1 = 0.5$, $\alpha_2 = 0.05$, $\alpha_3 = 0.25$, $\alpha_4 = 0.15$, and $\alpha_5 = 0.05$.

2.3.2. Task Scheduling Layer Model

Based on the local flight path cost matrices for each center generated by the multi-center site selection and local flight path planning layers, the task scheduling layer performs multi-drone task allocation and path scheduling within each distribution center. This layer employs a model that comprehensively considers both the cost of the drone payload and the task time cost, with the objective of minimizing the system's total transport cost, as shown in Equations (14) and (15):

$$F_{\text{task}}^g(G, L) = \left[F_{\text{timewin}}^g, F_{\text{load}}^g \right] \quad (14)$$

$$\text{fit}_{\text{task}}(G, T) = \sum_{i=1}^2 \text{restion}_2 F \quad (15)$$

Here, F_{timewin} represents the time constraint cost, F_{load} represents the load constraint cost, restion_2 represents the sum of the penalty terms for task uniqueness and traffic balancing constraints, and fit_{task} represents the overall performance of the task allocation and scheduling layer [19].

3. Algorithm Improvements

The lower layer employs a multi-source local optimization strategy, simultaneously determining task point assignments during the site selection process. It outputs the coordinates of the distribution centers, the task point allocation results, and the local node sets for each center. For each center's local node set, the MISFOA is used to perform 3D trajectory planning for any pair of nodes between the distribution center and the task points belonging to that center, generating local trajectories and trajectory cost matrices. The top level reads the local trajectory matrices corresponding to each center and, subject to constraints such as capacity, time windows and task uniqueness, uses IALNS to complete UAV task allocation and path scheduling. Information is transmitted between problem layers via the combination of distribution centers and trajectory matrices, thereby achieving layer-by-layer collaborative solutions from multi-center site selection and local trajectory generation to task scheduling within each center.

3.1. Starfish Optimization Algorithm (SFOA)

The Starfish Optimization Algorithm (SFOA) is a meta-heuristic intelligent optimization algorithm proposed in 2024 [20]. The algorithm is inspired by the foraging, predation, and regeneration behaviors of starfish. During the exploration phase, each starfish individual updates its position based on global random walk. In the exploitation phase, the starfish move toward promising regions guided by the current optimal solution to converge on the objective function. In the regeneration phase, underperforming individuals are replaced or reinitialized to maintain population diversity and avoid premature convergence. This process continues until the end of the iteration. The algorithm alternates between exploration and exploitation phases, where global search ensures coverage of the solution space, while local optimization improves the accuracy of the solution. The specific process is illustrated in Figure 3.

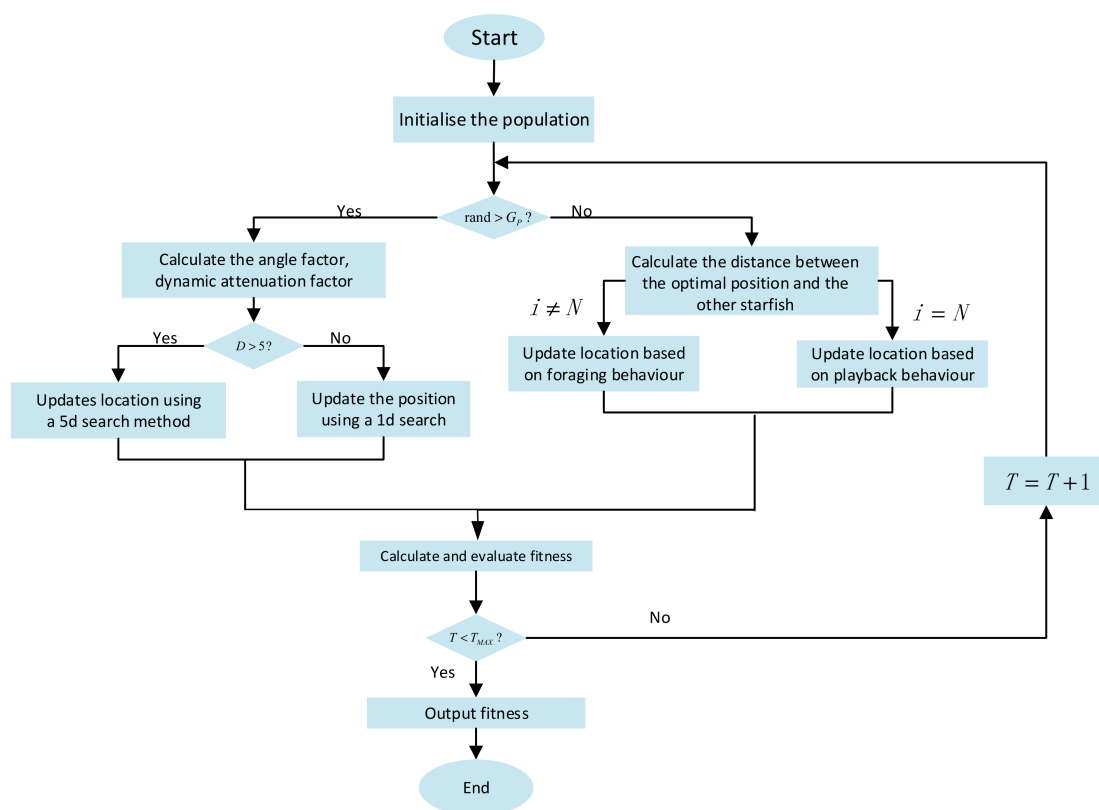


Figure 3. Flow chart of the Starfish Optimization Algorithm.

3.2. Mixed-Strategy Improved Starfish Optimization Algorithm, MISFOA

The Starfish Optimization Algorithm (SFOA) is a meta-heuristic algorithm that simulates the behavior of starfish, such as exploration, predation and regeneration. To address the limitations of the original SFOA in handling complex optimization problems, this paper proposes a Mixed-Strategy Improved Starfish Optimization Algorithm (MISFOA) for three-dimensional UAV flight path planning under complex terrain conditions. This is based on retaining the original global search framework of the SFOA, while addressing issues such as uneven initial population distribution in high-dimensional continuous spaces, insufficient local exploration capability, and slow improvement in convergence in optimization during later stages [21]. Firstly, during the population initialization stage, chaotic mappings are employed to generate initial individuals [22,23], thereby improving the uniformity and diversity of the initial population; secondly, following each iteration of the Starfish population update, a self-adaptive Differential Evolution (saDE) local enhancement mechanism is applied to the current optimal individual to further enhance the ability to perform fine-grained searches in the vicinity of high-quality solutions. Through these improvements, the MISFOA retains the global search characteristics of the original SFOA while enhancing its local exploration capabilities and convergence stability in complex constrained route planning problems. The process is illustrated in Figure 4.

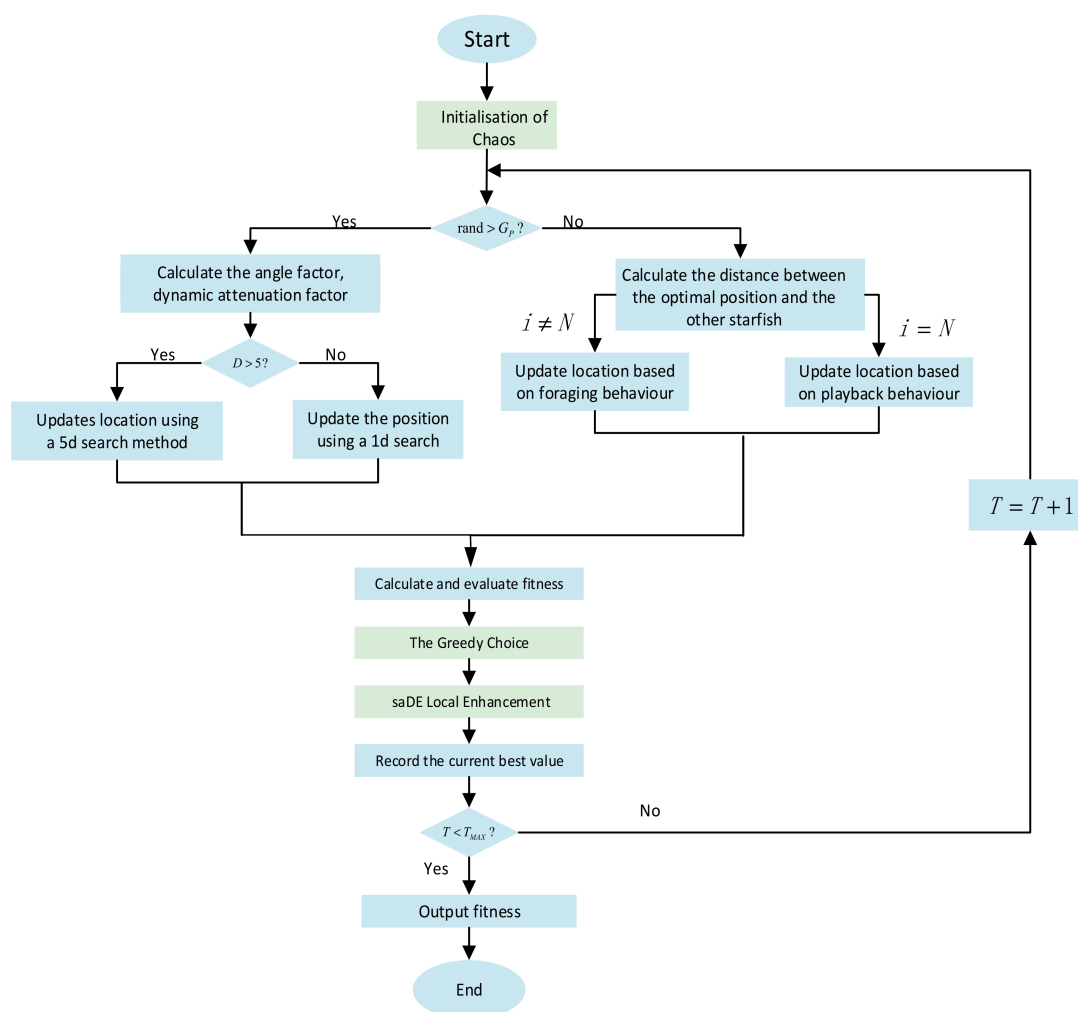


Figure 4. Flow chart of the Mixed-Strategy Improved Starfish Optimization Algorithm (the blue sections represent parts that are identical to the original algorithm, while the green sections indicate improvements to the algorithm).

3.2.1. Sine Chaos Initialization

In the field of optimization, chaotic mappings are often used as alternatives to random number generation mechanisms. In order to select the chaotic mechanism best suited to the SFOA, a total of six chaotic mappings were chosen to improve the initialization of the SFOA. After analyzing the results of multiple independent runs on the CEC2022 test function and recording metrics such as the optimal value, mean, standard deviation, median, worst-case value and average running time, a Friedman mean ranking analysis was performed. The results are shown in Figure 5.

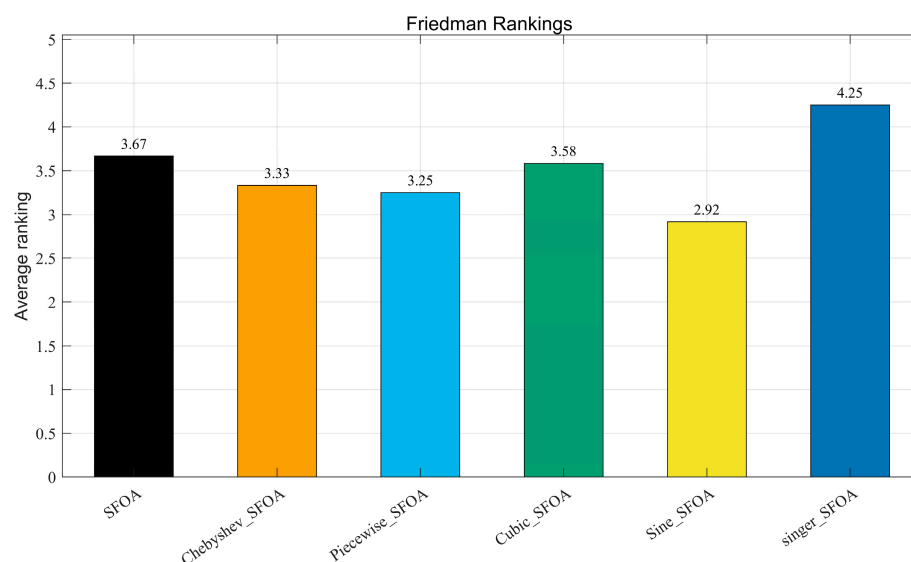


Figure 5. Friedman ranking results.

The lower the Friedman ranking value, the better the algorithm's overall performance across the test functions. As can be seen from the figure, Sine_SFOA has the lowest average Friedman ranking, at 2.92, indicating that it offers the best overall performance. Piecewise_SFOA and Chebyshev_SFOA have average rankings of 3.25 and 3.33, respectively, and their performance also exceeds that of the original SFOA, suggesting that chaotic initialization strategies can improve the search performance of the SFOA to a certain extent. The average ranking of the SFOA is 3.67, indicating relatively average overall performance. In summary, the Sine chaotic map can more effectively enhance the SFOA's global search capability and population initialization quality, enabling the algorithm to achieve superior statistical performance across multiple test functions. The SFOA employs a uniform random method to generate the initial population; however, in high-dimensional continuous optimization problems, this method is prone to issues such as uneven distribution of individuals, local clustering, and unstable quality of initial solutions, thereby affecting the search efficiency of the algorithm in its early stages. To improve the uniformity of the initial population distribution, this paper introduces a chaotic initialization strategy into the MISFOA, utilizing a sinusoidal chaotic map to generate the initial population [24,25], as shown in Equations (16) and (17):

$$s_{1d} = rand(0,1) \quad (16)$$

$$s_{(p+1)d} = \sin(\pi c_{pd}) \quad p = 1, 2, 3, \dots, N - 1 \quad (17)$$

Here, s_{1d} denotes the chaotic variable of the first individual in the population on the d th dimension, and $rand(0,1)$ denotes a random initial value within the interval $(0,1)$. By

mapping the chaotic variable onto the search space $[l_d, u_d]$, the initial position of the p th individual on the d th dimension is given by Equation (18):

$$x_{pd} = l_d + s_{pd}(u_d - l_d) \quad (18)$$

In this context, l_d and u_d represent the lower and upper bounds of the search space, respectively [26].

3.2.2. saDE Local Enhancement Mechanism

The original SFOA demonstrates good exploration capabilities in global search, but its local exploration capabilities in the later stages are relatively weak. To address this shortcoming, the saDE mechanism is introduced in the later stages of the algorithm to enhance local exploration [27]; specifically, after each round of population update, differential evolution-based local exploration is performed on a subset of elite individuals within the current population [28,29], as shown in Equation (16):

$$V_i = e_i + \text{Factor}_i(x_{best} - e_i) + \text{Factor}_i(e_{i_1} - e_{i_2}) \quad e_{i_1} \neq e_{i_2} \quad (19)$$

Here, e_i denotes the elite individuals. In each iteration, the population individuals are first sorted in ascending order according to their fitness values, and the top 20% of individuals in terms of fitness are selected to form the elite set. V_i denotes the mutation operator of x_{best} , the current global optimal individual. Factor_i denotes the adaptive scaling factor, used to control the first term $\text{Factor}_i(x_{best} - e_i)$ of the differential perturbation step, which guides elite individuals towards the current optimal solution, thereby enhancing local exploration capabilities; the second term $\text{Factor}_i(e_{i_1} - e_{i_2})$ utilizes the diversity information among population individuals to generate perturbations, maintaining search diversity and reducing the risk of getting stuck in local optima. To balance the perturbation intensity, experimental individuals are generated via binomial crossover, as shown in Equations (20)–(22):

$$\text{test}_{i,j} = \begin{cases} V_{i,j} & \text{rand} \leq \text{cross}_i \text{ or } j = j_{rand} \\ x_{i,j} & \text{otherwise} \end{cases} \quad (20)$$

$$\text{Factor}_i = \min\{0.9, \max(0.1, 0.5 + 0.3\mathcal{N}[0, 1])\} \quad (21)$$

$$\text{cross}_i = \min\{0.95, \max(0.05, 0.5 + 0.1\mathcal{N}[0, 1])\} \quad (22)$$

Here, $\text{test}_{i,j}$ denotes the experimental individual, cross_i denotes the adaptive crossover probability, and j_{rand} ensures that the experimental individual inherits at least one mutation dimension. Finally, the fitness of the experimental individual is evaluated; if it is superior to the original elite individual, a greedy selection mechanism is employed to affect the replacement, as shown in Equation (23):

$$e_i = \begin{cases} \text{test}_i & f(\text{test}_i) < f(e_i) \\ e_i & \text{otherwise} \end{cases} \quad (23)$$

As this mechanism applies only to a small number of elite individuals within the current population, it neither disrupts the SFOA's original global search framework nor provides stronger local refinement capabilities in the vicinity of high-quality solutions.

3.2.3. Improvements to Validation

The pseudocode for the improved algorithm is shown in Algorithm 1.

Algorithm 1: Pseudocode of Mixed-Strategy Improved Starfish Optimization Algorithm**Input:** Algorithm parameters**Output:** Global solution

```

1: Initialize the population using a sinusoidal chaotic map.
2: Calculate the fitness values for all starfish individuals.
3: Determine the current best individual  $X_{best}$  and its fitness value  $f_{best}$ .
4: While  $T \leq T_{max}$ 
5:     Calculate the angle factor and the dynamic energy factor.
6:     If  $rand > G_p$ 
7:         For each starfish Do
8:             If  $D > 5$  then
9:                 Select 5 dimensions at random.
10:                For  $j = 1 : 5$  Do
11:                    Generate random angle parameters; update the positions of the selected
dimensions in accordance with the high-dimensional exploration strategy.
12:                End For
13:                Perform boundary checks on the updated position.
14:            Else
15:                Select 1 dimension at random;
16:                Select two starfish at random.
17:                Update the positions of the selected dimensions in accordance with the
low-dimensional exploration strategy.
18:                Perform boundary checks on the updated position.
19:            End If
20:        End For
21:    Else
22:        Select 5 starfish individuals at random; calculate the distance vectors between the
current optimal position and the selected individuals.
23:        For each starfish Do
24:            Select two distance vectors at random; update the positions of the individuals in
accordance with the development strategy.
25:            If  $i = N$  then
26:                Update individual positions according to the regeneration strategy.
27:            End If
28:            Perform boundary checks on the updated position.
29:        End For
30:    End If
31:    Calculate the fitness values of the individuals after the update.
32:    Using a greedy selection strategy, if a new individual is superior to the original one, it is
substituted.
33:    Update the current global optimal solution  $X_{best}$ , execute the elite saDE local
enhancement strategy.
34:    Obtain the global best solution in iteration  $T$  step.
35:     $T = T + 1$ ;
36: End While
37: Obtain global solution.

```

To test the effectiveness of the improvement [30], experimental tests will be conducted using CEC2022 [31], and the MISFOA will be compared with six other algorithms—DBO [32], DBOPSO [33], PSO [34], DE [35], saDE [36] and the SFOA—under identical conditions. The population size was set to 50, with 500 iterations, and the experiment was run independently 30 times. The results are shown in Table 3.

Table 3. CEC2022 test results.

F	MISFOA		SFOA		DBO		DBOPSO		PSO		DE		saDE	
	avg	opt	avg	opt	avg	opt	avg	opt	avg	opt	avg	opt	avg	opt
F1	656.841	339.339	1256.278	434.994	29,375.602	17,356.845	8415.032	2437.119	31051.206	11,458.788	41,737.424	29,627.314	509.490	300.000
F2	446.783	406.537	448.353	408.842	927.447	594.089	495.104	437.741	738.780	591.716	449.739	449.112	450.681	444.895
F3	600.763	600.206	604.499	602.023	656.533	644.398	612.005	603.917	645.739	621.874	600.051	600.025	600.0059	600.000
F4	869.539	831.559	891.048	847.340	930.771	899.497	863.046	834.341	869.355	835.308	921.898	907.266	848.270	823.721
F5	922.530	901.883	1333.227	946.089	2724.822	1914.516	1110.825	912.173	2017.308	1316.107	1413.654	1118.831	900.638	900
F6	9504.620	2196.348	14,641.828	2640.865	38,590,860.18	7,906,609.969	11,647,760.440	25,718.168	154,990.569	1912.236	7,441,097.444	3,172,570.955	8031.004	1864.721
F7	2064.034	2040.089	2071.453	2043.295	2156.857	2091.464	2087.765	2042.655	2157.875	2056.233	2063.306	2047.763	2045.216	2032.455
F8	2229.783	2226.901	2234.122	2225.939	2325.754	2232.074	2254.277	2229.184	2325.639	2224.021	2231.288	2227.070	2226.369	2224.272
F9	2480.781	2480.781	2480.792	2480.782	2656.878	2555.583	2516.573	2480.868	2669.315	2576.998	2480.803	2480.789	2480.781	2480.781
F10	2917.786	2500.560	2961.604	2500.587	2723.325	2517.371	2502.896	2500.594	3923.208	2501.329	2524.569	2500.932	2532.974	2500.411
F11	2987.125	2908.074	3184.500	2848.426	6496.649	5682.451	3183.003	2980.878	12,357.416	6709.905	2901.531	2901.082	2900.000	2900
F12	2942.135	2933.162	2943.350	2934.761	3168.782	3073.430	2970.095	2939.049	3300.752	3096.575	2947.194	2942.621	2942.699	2935.243

As shown in the table above, the MISFOA achieved lower average fitness values than the SFOA across all 12 test functions, indicating that the introduced improvement strategy can consistently enhance the algorithm's performance. The optimization effects of MISFOA are particularly evident in functions such as F1, F5 and F6. In F1, the average fitness of the MISFOA decreased by approximately 47.72%; in F5, the average fitness decreased by approximately 30.80%; and in F6, the average fitness decreased by approximately 35.09%. This indicates that sinusoidal chaotic initialization improves the quality of the initial population and enhances the algorithm's early-stage global search capability, while the elite saDE local enhancement strategy further improves the exploration accuracy near the elite solutions, thereby effectively alleviating the issue of insufficient convergence accuracy in the original SFOA. The algorithms were ranked using Friedman. The results are shown in Figure 6.

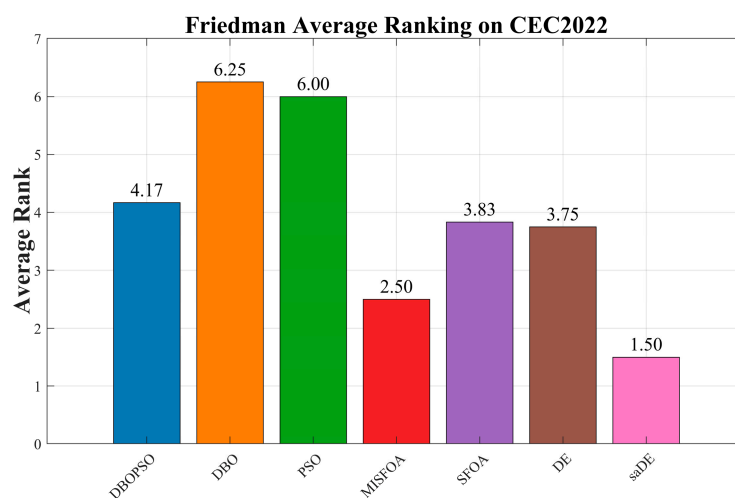


Figure 6. Rankings.

As shown in the figure, the MISFOA achieved an average ranking of 2.50, second only to saDE's 1.50, and significantly outperforming the original SFOA's 3.83, DE's 3.75, DBOPSO's 4.17, PSO's 6.00 and DBO's 6.25. Compared with the original SFOA, the average Friedman rank of the MISFOA improved by 1.33, indicating a marked improvement in its overall optimization performance. Pairwise comparisons between the MISFOA and the other six algorithms were conducted using the Wilcoxon test, with the results shown in Table 4.

Table 4. Results of the Wilcoxon test.

F	SFOA	DBO	DBOPSO	PSO	DE	saDE
	<i>p</i>	<i>p</i>	<i>p</i>	<i>p</i>	<i>p</i>	<i>p</i>
F1	4.4205×10^{-6}	3.01986×10^{-11}	3.01986×10^{-11}	3.01986×10^{-11}	3.01986×10^{-11}	2.37682×10^{-7}
F2	0.002499392	3.01986×10^{-11}	1.10772×10^{-6}	3.01986×10^{-11}	1.10234×10^{-8}	0.195790758
F3	3.01986×10^{-11}	3.01986×10^{-11}	3.01986×10^{-11}	3.01986×10^{-11}	3.01986×10^{-11}	3.01986×10^{-11}
F4	0.00185748	3.33839×10^{-11}	0.118817344	0.970516051	3.01986×10^{-11}	1.86817×10^{-5}
F5	4.97517×10^{-11}	3.01986×10^{-11}	4.99795×10^{-9}	3.01986×10^{-11}	3.01986×10^{-11}	4.97517×10^{-11}
F6	0.016954881	3.01986×10^{-11}	3.01986×10^{-11}	0.482516904	3.01986×10^{-11}	1.74791×10^{-5}
F7	0.043583548	3.01986×10^{-11}	0.000729511	4.19968×10^{-10}	0.958731491	7.08811×10^{-8}
F8	1.10772×10^{-6}	4.50432×10^{-11}	4.61591×10^{-10}	2.00229×10^{-6}	0.002265781	1.77691×10^{-10}
F9	3.01986×10^{-11}	3.01986×10^{-11}	3.01986×10^{-11}	3.01986×10^{-11}	3.01986×10^{-11}	6.79657×10^{-10}
F10	0.437641335	0.491782957	0.371077032	5.46203×10^{-6}	0.706171488	0.000158461
F11	2.49131×10^{-6}	3.01986×10^{-11}	7.73869×10^{-6}	3.01986×10^{-11}	3.01986×10^{-11}	3.01986×10^{-11}
F12	0.290472149	3.01986×10^{-11}	1.69795×10^{-8}	3.01986×10^{-11}	1.01877×10^{-5}	0.994101921
+/-/=	10/2/0	11/1/0	10/2/0	10/2/0	8/2/2	0/2/10

As shown in Table 4, MISFOA demonstrates a clear advantage over other algorithms. The results against DBO were 11/1/0, while the results against DBOPSO, PSO and the MISFOA were all 10/2/0, indicating that its overall performance is significantly superior to these two categories of comparison algorithms. Compared with DE, the statistical results for the MISFOA were 8/2/2, suggesting that the MISFOA still holds an advantage on most test functions, although DE performed better on a few functions. The comparison with saDE yielded a result of 0/2/10, indicating that, within this set of test functions, the standalone saDE algorithm achieved superior results on most functions, consistent with the conclusion that saDE ranked first and the MISFOA second in the Friedman average ranking. This is because saDE itself is a strong local search algorithm centered on differential variation and adaptive parameter tuning and possesses high optimization accuracy on certain continuous benchmark functions. The test functions, Friedman rankings and Wilcoxon tests all validate the effectiveness of the algorithmic improvements.

To further validate the effectiveness of the improved algorithm in UAV flight path planning, a terrain surface with 20 randomly generated peaks was created within a $100 \times 100 \times 300$ space. Under the same conditions, seven algorithms were used to plan flight paths for a single UAV. The planning results are shown in Figure 7.

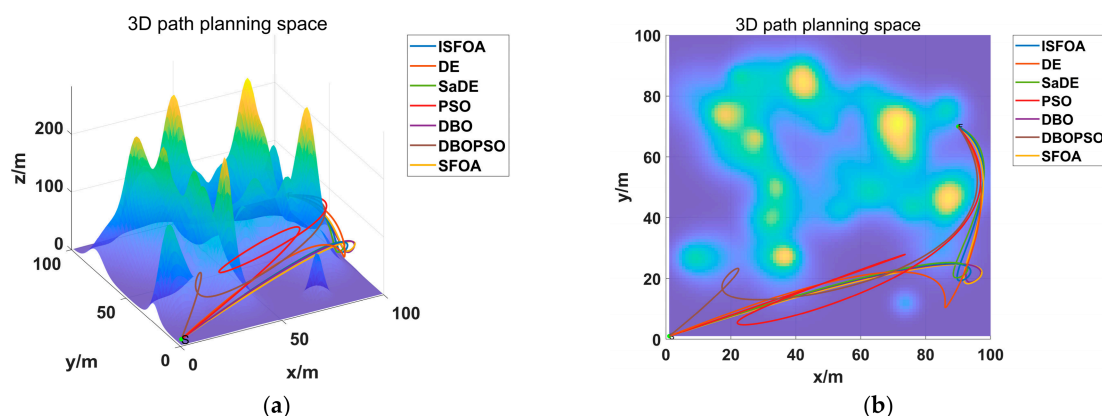


Figure 7. Planning track chart. (a) Side view. (b) Top view.

As can be seen from the figure, all algorithms successfully planned a trajectory that smoothly connected the starting point to the endpoint while avoiding the mountain peaks. The specific results are shown in Table 5.

Table 5. Statistics on algorithm results.

Indicators	MISFOA	SFOA	DBOPSO	DBO	PSO	DE	saDE
Best	3.623129×10^2	3.735570×10^2	3.634078×10^2	4.595924×10^2	4.550377×10^2	3.733853×10^2	3.617943×10^2
Mean	3.625864×10^2	3.751518×10^2	3.643088×10^2	6.818767×10^2	2.588752×10^4	3.743909×10^2	3.617943×10^2
Standard	4.085092×10^{-1}	1.629662×10^0	1.346548×10^0	3.650635×10^2	2.246375×10^4	1.397974×10^0	7.352082×10^{-3}

As shown in the table, the MISFOA had a clear advantage over most of the comparison algorithms. Its best value was very close to that of saDE and slightly better than those of the SFOA, DBOPSO, DE, and PSO, indicating that it possesses highly competitive global optimization capabilities. In terms of the average, it significantly outperformed the SFOA, DBOPSO, DBO, and PSO, with a 3.35% improvement over the SFOA. In terms of standard deviation, the MISFOA demonstrated strong stability, outperforming the SFOA by 74.93% and DBOPSO by 69.66%. Its overall performance was slightly inferior to that of saDE, consistent with the results of the Wilcoxon test and Friedman ranking. This confirms that chaotic initialization and the elite saDE enhancement measures effectively improve the algorithm's robustness.

4. Simulation Experiments

4.1. Experimental Parameters

Simulation experiments were conducted on the MATLAB R2024a platform to evaluate the effectiveness of the proposed two-stage collaborative optimization framework for multi-hub UAV logistics delivery in complex mountainous terrain. At this stage of the study, 30 mission points were generated within the mountainous environment to simulate real-world logistics and transportation demands, with their spatial coordinates, mission requirements, service times and time window constraints clearly defined. The elevation of the mission points was derived through interpolation of the terrain surface model. Concurrently, key UAV operational parameters—including maximum payload, safe altitude, cruising speed and hub service capacity—were explicitly incorporated into the model. The aim was to approximate real-world low-altitude logistics scenarios and provide a meaningful basis for evaluating the feasibility and effectiveness of the proposed method. The mission point information is shown in Table 6.

Table 6. Starting and ending point coordinate information.

Task Point Number	Coordinates	Time Window	Opening Hours	Workload
1	(50,50)	(600,630)	20	40
2	(380,50)	(570,600)	20	10
3	(50,450)	(60,150)	20	40
4	(450,220)	(540,570)	20	10
5	(250,250)	(20,60)	20	20
6	(100,100)	(480,520)	20	10
7	(200,360)	(160,200)	20	40
8	(350,400)	(220,270)	20	30
9	(300,100)	(460,500)	20	10
10	(300,120)	(300,330)	20	5
11	(72,88)	(80,130)	20	15
12	(410,65)	(520,560)	20	25
13	(58,420)	(140,190)	20	35
14	(438,248)	(560,600)	20	10
15	(235,278)	(40,90)	20	20
16	(118,135)	(460,500)	20	10
17	(188,338)	(180,230)	20	30
18	(332,392)	(260,310)	20	25
19	(286,96)	(420,460)	20	10
20	(318,148)	(320,360)	20	5
21	(100,280)	(540,580)	20	30
22	(450,450)	(480,520)	20	35
23	(150,450)	(90,140)	20	10
24	(200,450)	(510,550)	20	15
25	(410,320)	(30,80)	20	5
26	(202,30)	(430,470)	20	25
27	(205,117)	(170,220)	20	10
28	(350,250)	(250,300)	20	40
29	(50,320)	(390,430)	20	10
30	(35,280)	(310,350)	20	40

In this study, the drone parameters were set to a maximum payload of 110, a safe altitude of 5, and a cruising speed of 10. The maximum number of tasks that a single center can undertake is 200.

4.2. Experimental Results

Firstly, multi-center site selection and task assignment were achieved through a local optimization strategy, as shown in Figure 8.

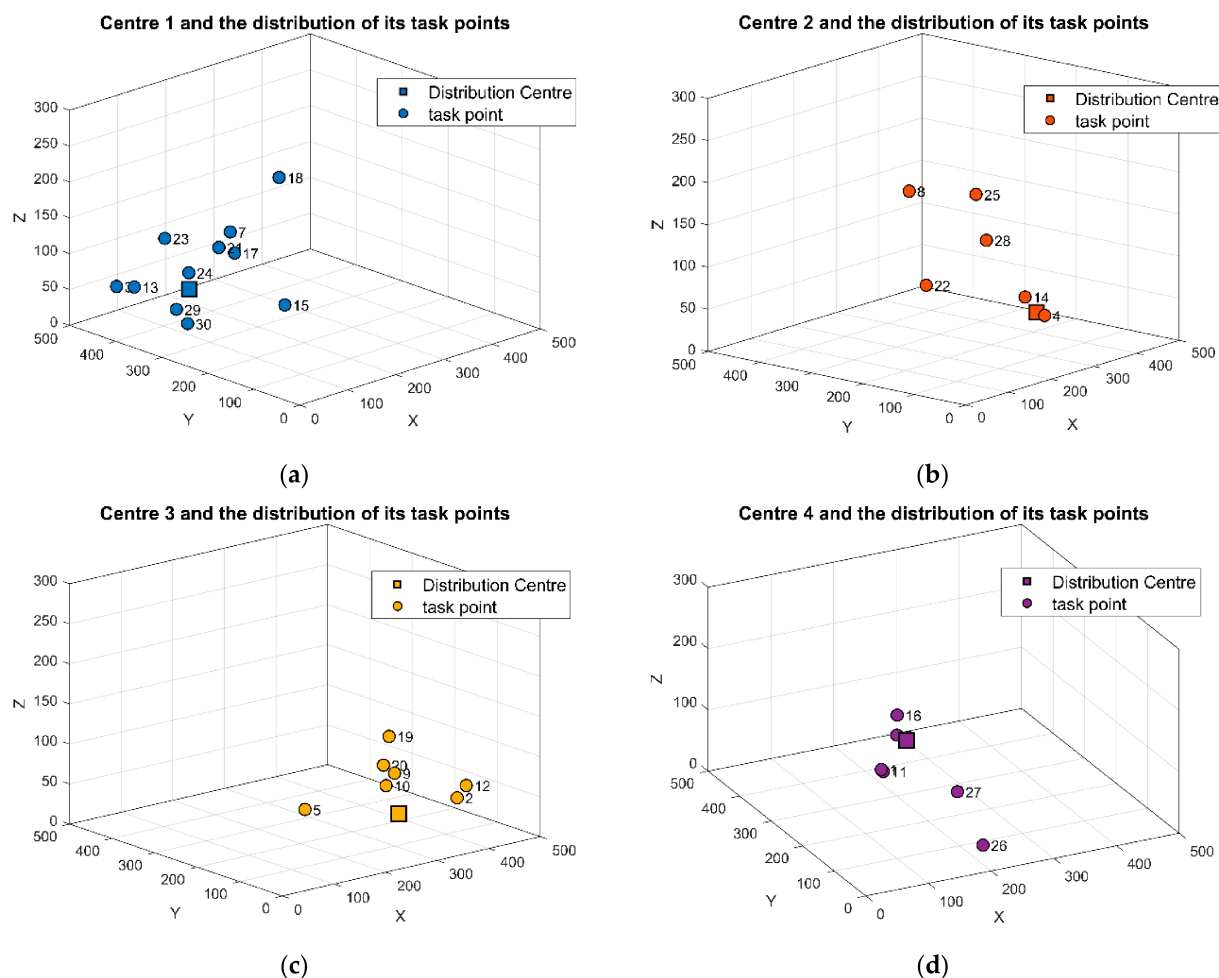


Figure 8. Side view of the distribution center and its delivery points. (a) Center 1 and its task points. (b) Center 2 and its task points. (c) Center 3 and its task points. (d) Center 4 and its task points.

As shown in Figure 5, the back-calculation of the task volume indicates that a total of four distribution centers was required. The coordinates of the first distribution center were (136.33, 382.27, 47.46), the second distribution center was located at (447.72, 233.91, 11.47), the third distribution center was located at (321.97, 117.19, 34.08), and the fourth distribution center was located at (107.59, 84.89, 196.73). The positioning of these centers effectively balanced the average service distance to all task points while avoiding high-risk terrain areas. This demonstrates the advantages of jointly modeling average service distance and terrain elevation penalties within the site selection model, providing an efficient starting point for subsequent route planning and task allocation. The overall layout is shown in Figure 9.

Once the locations of the distribution centers had been determined, the local flight path cost matrices were constructed. At the top level, using the local flight path cost matrices corresponding to each center as input, collaborative allocation and scheduling of multiple drones was performed for 30 task points, subject to constraints on capacity and time windows. The distribution of flight paths for drones from different centers is shown in Figure 10.

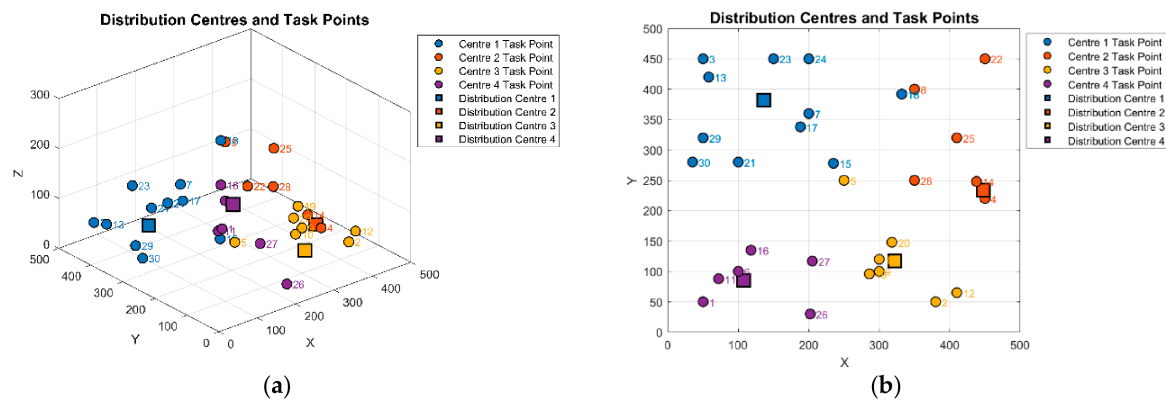


Figure 9. Schematic diagram of the overall layout. (a) Side view of spatial distribution. (b) Top view of spatial distribution.

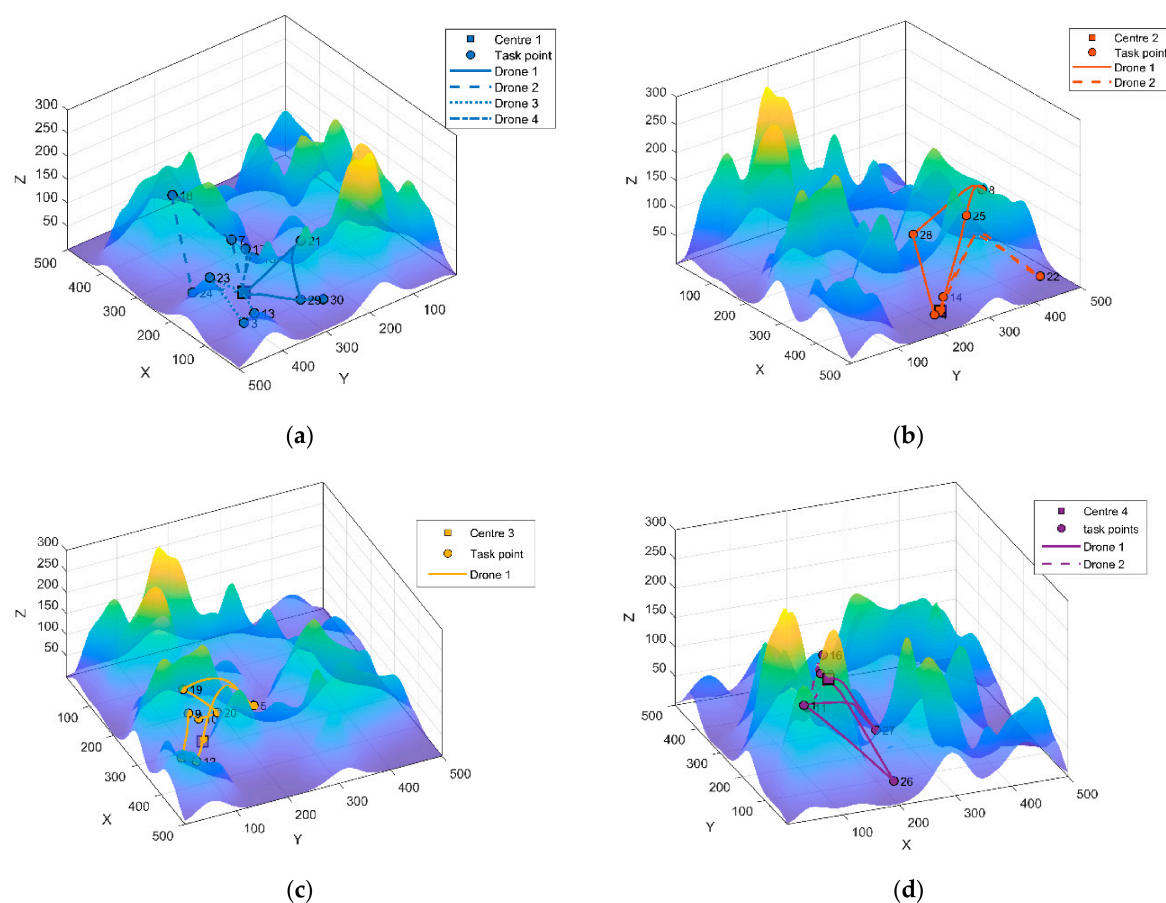


Figure 10. Side view of drone flight paths at different centers. (a) Center 1 flight path planning. (b) Center 2 flight path planning. (c) Center 3 flight path planning. (d) Center 4 flight path planning.

The flight paths of the drones corresponding to different distribution centers clearly illustrate the results of the three-dimensional path planning for the centers and their task points. Under the two-layer framework, the flight paths for each center were adjusted appropriately in response to variations in terrain. This ensured a closed-loop connection between the distribution center as the starting point and the destination, while maintaining effective coverage of the task points, demonstrating that the local path planning mechanism developed is capable of meeting flight requirements in complex mountainous terrain. Specific task scheduling information is shown in Table 7.

Table 7. Task scheduling statistics.

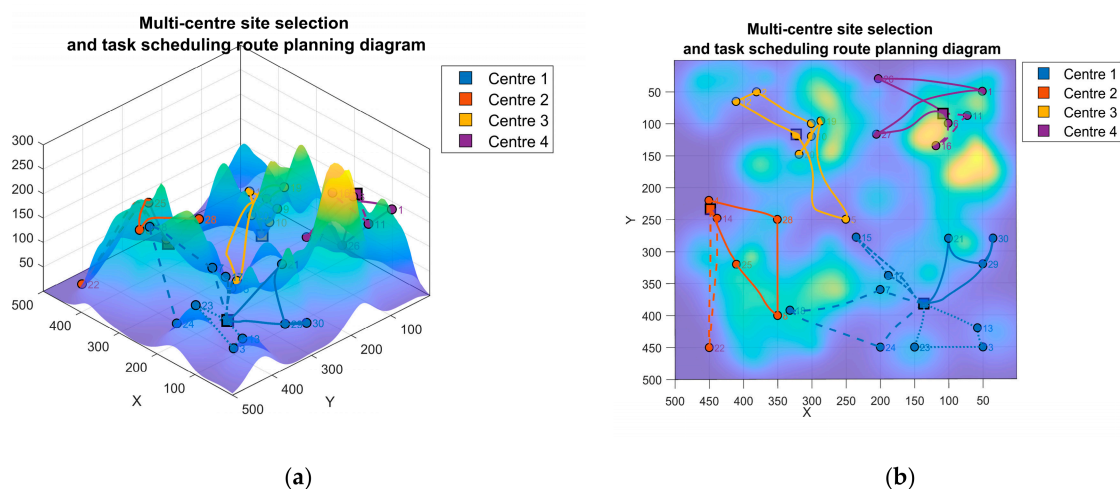
Center	Drone Number	Route Checkpoint Number	Workload	Flight Distance
C_1	1	C_1 -30-29-21- C_1	80	1204.71
	2	C_1 -7-18-24- C_1	80	1205.41
	3	C_1 -13-3-23- C_1	85	1203.31
	4	C_1 -15-17- C_1	50	903.86
C_2	1	C_2 -25-8-28-4- C_2	85	1506.15
	2	C_2 -22-14- C_2	45	906.36
C_3	1	C_3 -15-19-20-10-9-2-12- C_3	85	2414.79
C_4	1	C_3 -27-1-26- C_3	75	1207.84
	2	C_3 -11-16-6- C_3	35	1202.14

Table 7 shows the final schedule plan for the model: the workload across the four central drones was balanced without compromising load capacity or time windows. The overall cost parameters of the model are shown in Table 8.

Table 8. Model cost parameters.

Parameter	Value
H_{Length}^g	120.46230
H_{Elev}^g	448.21071
H_{Build}^g	159.20768
H_{Trans}^g	248,290.47192
F_{height}	5.68004
F_{Threat}	19,519.023966
F_{Space}	10,510.24367
F_{Length}	10,986.87236
F_{Pitch}	118.00000

All flight paths are shown in Figure 11.

**Figure 11.** Three-layer structural planning track chart. (a) Side view. (b) Top view.

As shown in the figure, analysis of the flight path patterns for each center revealed that, in some centers, the task points were relatively dispersed. Consequently, a greater number

of drones were deployed to form multiple, relatively independent local flight paths, thereby reducing the service range of individual drones and enhancing the efficiency of parallel delivery. In contrast, for centers where task points were more concentrated, the flight path structure was relatively compact. This indicates that the results of the multi-center division were being correctly utilized by the flight path planning and task scheduling layers. From a three-dimensional spatial perspective, none of the flight paths generated for each center intersected with terrain obstacles, indicating that the MISFOA effectively balances terrain constraints, safe altitude constraints and flight path smoothness constraints during the planning process. Furthermore, there was no conflict between the task points of flight paths assigned to different UAVs within the same center, demonstrating the algorithm's ability to coordinate and divide tasks within local task sets. These results validate the effectiveness of the three-layer framework proposed in this paper.

5. Limitations and Future Work

The two-stage collaborative optimization framework proposed in this study has demonstrated its effectiveness, but several limitations still exist.

Firstly, this study was validated primarily through numerical simulation rather than actual real-world UAV flight experiments. Although the simulation scenarios were designed to closely approximate real-world low-altitude logistics conditions in mountainous terrain, the validation framework inevitably relies on simplified assumptions. The modeled scenarios incorporate key operational factors such as mission locations, mission requirements, service times, time window constraints, UAV payload limitations, safe flight altitudes, cruising speeds, and central service capacity. However, despite this comprehensive configuration, simulation-based validation cannot fully capture the uncertainties and disturbances encountered in real operational environments. These include wind effects, communication delays, positioning errors, sensor noise, and flight control deviations, all of which may significantly influence UAV trajectory stability and system-level performance in practical deployments. Secondly, the current framework assumes relatively stable operating conditions and does not explicitly account for dynamic environmental changes, such as temporary no-fly zones, ad hoc missions, weather disruptions, or real-time traffic adjustments in low-altitude airspace. In actual UAV logistics systems, these dynamic factors may significantly impact local trajectory feasibility and the performance of internal scheduling within the hub.

In future research, we will first conduct small-scale real-world UAV flight experiments to further validate the engineering feasibility of the proposed framework. Secondly, we will introduce dynamic environment modelling and online rescheduling mechanisms to enhance the model's adaptability under time-varying and uncertain operational conditions. Finally, the research team aims to further explore multi-center collaboration mechanisms and combine intelligent learning methods with optimization algorithms, thereby enhancing the flexibility, scalability and real-time responsiveness of multi-center UAV logistics systems in complex terrain environments.

6. Conclusions

This paper proposes a two-level model for the Multi-Center Location and Coordinated Dispatch Problem (MLCDP) in the context of unmanned aerial vehicle (UAV) logistics networks within mountainous transport systems. The model aims to achieve unified decision-making regarding multi-center location, local three-dimensional UAV flight path planning, and task allocation and dispatching among multiple UAVs within each center, with the objective of minimizing total system transport costs and enhancing overall delivery efficiency. The lower layer focuses on multi-center site selection and local trajectory

planning. Following the determination of distribution center layouts and the assignment of task points, it generates three-dimensional trajectories and trajectory cost matrices between local nodes within each center. The upper layer, utilizing the local trajectory cost matrices generated by the lower layer, optimizes the task allocation and path scheduling of multiple UAVs within each distribution center. Information exchange and cost feedback between the two layers are achieved via the flight path cost matrix, thereby forming a unified decision-making mechanism that seamlessly integrates distribution center layout, local flight path generation, and intra-center task scheduling. Simulation results demonstrate that the proposed method can effectively accomplish multi-center distribution center layout and multi-drone collaborative operations under typical complex mountainous terrain, while improving the overall operational efficiency of the system and ensuring flight path feasibility and flight safety. This study not only validates the effectiveness of the improved SFOA but also demonstrates the effectiveness of the MLCDP two-layer collaborative optimization framework in complex constrained logistics delivery scenarios. The two-tier structure of the MLCDP provides an alternative, scalable optimization method for UAV logistics network planning within the context of the low-altitude economy. In the future, research may be further conducted in areas such as dynamic environment modeling, multi-center coordination mechanisms, integrated energy–payload coupling modeling, and the integration of intelligent learning algorithms. This will drive the development of UAV logistics scheduling systems towards greater intelligence, adaptability and real-time responsiveness, while supporting the digital and intelligent evolution of low-altitude transport systems.

Author Contributions: Writing—review and editing, Y.Y.; writing—original draft preparation, Y.F.; visualization, B.W.; investigation, K.X.; data curation, W.F. All authors have read and agreed to the published version of the manuscript.

Funding: This study has been supported by the Open Fund Support from the Sichuan Provincial Engineering Research Centre for Flight and Operational Support of Domestic Civil Aircraft (Grant No. MJCYZY202503); this work has also been supported by the Sichuan Provincial Civil Aviation Airport Smart Operation and Maintenance Engineering Research Centre Independent Research Project (Grant No. JCZX2023ZZ07 and JCZX2024ZZ25) and the Fundamental Research Funds for the Central Universities (Grant No. 26CAFUC01004, 25CAFUC04001, 26CAFUC03008).

Data Availability Statement: The original contributions presented in the manuscript are included in the article; any further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Wan, L.; Xu, S.; Hang, J.; Zhang, Q.; Yuan, Z. Fuzzy Maximal Covering Model for Drone Emergency Distribution Center Selection. *J. Transp. Syst. Eng. Inf. Technol.* **2026**, *26*, 166–175. [[CrossRef](#)]
2. Yağmur, E.; Yüce, M. A GIS-driven MCDM framework for parcel locker site selection in urban environments: The case of Istanbul. *Ain Shams Eng. J.* **2026**, *17*, 104065. [[CrossRef](#)]
3. Rave, A.; Fontaine, P.; Kuhn, H. Drone location and vehicle fleet planning with trucks and aerial drones. *Eur. J. Oper. Res.* **2023**, *308*, 113–130. [[CrossRef](#)]
4. Shuaibu, A.S.; Mahmoud, A.S.; Sheltami, T.R.; Elgabri, A. Reinforcement Learning and Exact Optimization for Traffic-Weighted Hybrid Truck-Drone Last-Mile Delivery with Deadlines. *IEEE Trans. Intell. Transp. Syst.* **2026**, 1–18. [[CrossRef](#)]
5. Peng, Y.; Zhu, W.J.; Yu, D.Z.; Liu, S.; Zhang, Y. Multi-Depot Electric Vehicle–Drone Collaborative-Delivery Routing Optimization with Time-Varying Vehicle Travel Time. *Vehicles* **2024**, *6*, 1812–1842. [[CrossRef](#)]
6. Meng, S.; Li, D.; Liu, J.; Chen, Y. The multi-visit drone-assisted routing problem with soft time windows and stochastic truck travel times. *Transp. Res. Part B Methodol.* **2024**, *190*, 103101. [[CrossRef](#)]
7. Dai, L.; Tian, Y.; Zhang, N.; Wan, L.; Hai, S. Research on Real-Path-Based UAV Distribution Center Layout in Urban Environments. *Aerospace* **2025**, *12*, 703. [[CrossRef](#)]

8. Lakhwani, T.S. Integrating 5PL Frameworks with Drone-Based Last-Mile Delivery: A Model for Future-Ready Logistics. *Transp. Dev. Res.* **2025**, *3*, 27–45. [\[CrossRef\]](#)
9. Yan, D.; Wang, X.; Suo, X.; Hu, S.; Song, J. A Review of Algorithms for Maximum Independent Set Problems. *Comput. Eng. Appl.* **2026**, 1–21. Available online: <https://link.cnki.net/urlid/11.2127.TP.20260305.1521.008> (accessed on 3 May 2026).
10. Wang, S.; Li, Y.; Zhao, Y.; Liu, C. A review of unmanned aerial vehicle trajectory planning algorithms. *J. Harbin Eng. Univ.* **2026**, *47*, 708–719.
11. Wang, Y.; Liu, J.; Qian, Y.; Yi, W. Path Planning for Multi-UAV in a Complex Environment Based on Reinforcement-Learning-Driven Continuous Ant Colony Optimization. *Drones* **2025**, *9*, 638. [\[CrossRef\]](#)
12. Cheng, Q.; Zhang, Z.; Du, Y.; Zhao, X. Hybrid metaheuristic-driven 3D path planning for UAVs in complex urban environments: A multi-objective fusion framework. *Geocarto Int.* **2025**, *40*, 2538099. [\[CrossRef\]](#)
13. Sushma, M.B.; Mahmoodi, B.; Tan, W.; Kang, L.; Xu, Q. Spatial drone path planning: A systematic review of parameters and algorithms. *J. Transp. Geogr.* **2025**, *125*, 104209. [\[CrossRef\]](#)
14. Li, J.; Li, J.; Zhang, J. A Comprehensive Review of Path-Planning Algorithms for Multi-UAV Swarms. *Drones* **2026**, *10*, 11. [\[CrossRef\]](#)
15. Bacha, A.M.; Zamoum, R.B.; Lachekhab, F. Machine learning paradigms for UAV path planning: Review and challenges. *J. Robot. Control (JRC)* **2025**, *6*, 215–233. [\[CrossRef\]](#)
16. Wang, W.T.; Tian, J. Dynamic UAV path planning in mountainous terrain utilizing an arithmetic optimization algorithm incorporating adaptive thermal conduction search and elite population genetic strategies. *Aerosp. Sci. Technol.* **2025**, *168*, 110950. [\[CrossRef\]](#)
17. Wan, M.; Ye, C. Location and Routing Optimization of Logistics Distribution Centre Based on Bi-level Programming. *Complex Syst. Complex. Sci.* **2025**, *22*, 118–124. [\[CrossRef\]](#)
18. Yang, Y.; Fu, Y.; Lu, D.; Xiang, H.; Xu, K. Three-Dimensional Unmanned Aerial Vehicle Trajectory Planning Based on the Improved Whale Optimization Algorithm. *Symmetry* **2024**, *16*, 1561. [\[CrossRef\]](#)
19. Yang, Y.; Fu, Y.; Xin, R.; Feng, W.; Xu, K. Multi-UAV Trajectory Planning Based on a Two-Layer Algorithm Under Four-Dimensional Constraints. *Drones* **2025**, *9*, 471. [\[CrossRef\]](#)
20. Zhong, C.; Li, G.; Meng, Z.; Li, H.; Yildiz, A.R.; Mirjalili, S. Starfish optimization algorithm (SFOA): A bio-inspired metaheuristic algorithm for global optimization compared with 100 optimizers. *Neural Comput. Appl.* **2024**, *37*, 2641–3683. [\[CrossRef\]](#)
21. Yang, Y.; Fu, Y.; Li, D.; Xu, K. Multiple UAVs trajectory planning based on starfish optimization algorithm. *Sci. Technol. Eng.* **2026**, *26*, 387–394. [\[CrossRef\]](#)
22. Cui, G.; Song, J.; Han, Y.; Wu, H.; Zhou, F.; Wang, Y. Chaos-Based Inertia Weight Particle Swarm Optimization for Path Planning of Unmanned Surface Vehicles. *J. Mar. Sci. Appl.* **2025**, 1–10. [\[CrossRef\]](#)
23. Eiben, A.E.; Smith, J.E. *Introduction to Evolutionary Computing*; Springer: Berlin/Heidelberg, Germany, 2015.
24. Zheng, Y.; Li, L.; Qian, L.; Cheng, B.; Hou, W.; Zhuang, Y. Sine-SSA-BP Ship Trajectory Prediction Based on Chaotic Mapping Improved Sparrow Search Algorithm. *Sensors* **2023**, *23*, 704. [\[CrossRef\]](#) [\[PubMed\]](#)
25. Lin, M.; Dai, Z.; Qiu, Z.; Chen, H.; Lin, H. Multi-strategy enhanced sand cat swarm optimization algorithm and its engineering applications. *Sci. Rep.* **2025**, *15*, 34391. [\[CrossRef\]](#) [\[PubMed\]](#)
26. Han, B.; Hua, Z.; Wang, N.; Zhu, L.; Chen, M.; Bao, B. Initials-Boosted Coexisting Chaos in a 2-D Sine Map and Its Hardware Implementation. *IEEE Trans. Ind. Inform.* **2021**, *17*, 1132–1140. [\[CrossRef\]](#)
27. Qin, A.K.; Huang, V.L.; Suganthan, P.N. Differential Evolution Algorithm with Strategy Adaptation for Global Numerical Optimization. *IEEE Trans. Evol. Comput.* **2009**, *13*, 398–417. [\[CrossRef\]](#)
28. Zhu, L.; Zhou, G.; Zhou, Y.; Luo, Q.; Huang, H.; Wei, X. Self-adaptive differential evolution-based coati optimization algorithm for multi-robot path planning. *Robotica* **2025**, *43*, 959–996. [\[CrossRef\]](#)
29. Pan, Q.; Tang, J.; Wang, H.; Li, H.; Chen, X.; Lao, S. SPSADE: An improved self-adaptive differential evolution algorithm with a shuffled frog-leaping strategy. *Artif. Intell. Rev.* **2022**, *55*, 3937–3978. [\[CrossRef\]](#)
30. Li, H.; Tang, J.; Pan, Q.; Zhan, J.; Lao, S. Ensemble of population-based metaheuristic algorithms. *Comput. Mater. Contin.* **2023**, *76*, 2835–2859. [\[CrossRef\]](#)
31. Abhishek, K.; Price Kenneth, V.; Suganthan Ponnuthurai, N.; Wagdy, M.A.; Handi Anas, A. *Problem Definitions and Evaluation Criteria for the CEC 2022 Special Session and Competition on Single Objective Bound Constrained Numerical Optimization*; Nanyang Technological University: Singapore, 2021.
32. Xue, J.; Shen, B. Dung beetle optimizer: A new meta-heuristic algorithm for global optimization. *J. Supercomput.* **2023**, *79*, 7305–7336. [\[CrossRef\]](#)
33. Yang, Y.; Sun, L.; Xu, K.; Xiang, H.-H.; Feng, W.-Q. UAV Three-Dimensional Path Planning Based on Improved Dung Beetle Optimizer Algorithm. *Appl. Sci.* **2026**, *16*, 5243. [\[CrossRef\]](#)
34. Kennedy, J.; Eberhart, R. Particle swarm optimization. In *Proceedings of the IEEE International Conference on Neural Networks, Perth, WA, Australia, 27 November–1 December 1995*; IEEE: New York, NY, USA, 2002; Volume 4, pp. 1942–1948. [\[CrossRef\]](#)

35. Opara, K.R.; Arabas, J. Differential evolution: A survey of theoretical analyses. *Swarm Evol. Comput.* **2019**, *44*, 546–558. [[CrossRef](#)]
36. Mahamed, G.; Salman, A.; Engelbrecht, A. Self-adaptive differential evolution. In *Computational Intelligence and Security, Xi'an, China, 15–19 December 2005*; Springer: Berlin/Heidelberg, Germany, 2005; Volume 3801, pp. 192–199. [[CrossRef](#)]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.