

Research Article

Open Access



# UAV path planning based on a dual-strategy ant colony optimization algorithm

Xiaoming Mai<sup>1</sup>, Na Dong<sup>1</sup>, Shuai Liu<sup>1</sup>, Hao Chen<sup>1</sup>

<sup>1</sup>School of Electrical and Information Engineering, Tianjin University, Tianjin 300110, China.

**Correspondence to:** Prof. Na Dong, School of Electrical and Information Engineering, Tianjin University, No. 92, Weijin Road, Nankai District, Tianjin 300110, China. E-mail: dongna@tju.edu.cn

**How to cite this article:** Mai X, Dong N, Liu S, Chen H. UAV path planning based on a dual-strategy ant colony optimization algorithm. *Intell Robot* 2023;3(4):466-84. <http://dx.doi.org/10.20517/ir.2023.37>

**Received:** 20 Oct 2023 **First Decision:** 16 Nov 2023 **Revised:** 28 Nov 2023 **Accepted:** 8 Dec 2023 **Published:** 21 Dec 2023

**Academic Editor:** Haibin Duan, Simon X. Yang **Copy Editor:** Pei-Yun Wang **Production Editor:** Pei-Yun Wang

## Abstract

With the rapid development of modern communication and automatic control technologies, unmanned aerial vehicles (UAVs) have increasingly gained importance in both military and civilian domains. Path planning, a critical aspect for achieving autonomous aerial navigation, has consistently been a focal point in UAV research. However, traditional ant colony algorithms need to be improved for the drawbacks of susceptibility to local optima and weak convergence capabilities. Consequently, a novel path planning methodology is proposed based on a dual-strategy ant colony algorithm. In detail, an improved state transition probability rule is introduced, redefining ant movement rules by integrating the state transition strategy of deterministic selection during the iterative process. Additionally, heuristic information on adjacent node distance and mountain height is added to further improve the search efficiency of the algorithm. Then, a new dynamically adjusted pheromone update strategy is proposed. The update strategy is continuously adjusted during the iteration process, which is beneficial to the algorithm's global search in the early stage and accelerated convergence in the later stage, preventing the algorithm from falling into local optimality and improving its convergence. Based on the above improvements, a new variation of ant colony optimization (ACO) called dual-strategy ACO algorithm is formed. Experimental results prove that dual-strategy ACO has superior global search capabilities and convergence characteristics from four key aspects: path length, fitness values, iteration number, and running time.

**Keywords:** Path planning, ant colony optimization algorithm, heuristic information, dynamic adjustment of pheromones



© The Author(s) 2023. **Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License (<https://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, sharing, adaptation, distribution and reproduction in any medium or format, for any purpose, even commercially, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.



## 1. INTRODUCTION

Due to their unique technical attributes, unmanned aerial vehicles (UAVs) can be equipped with diverse sensor devices to achieve real-time environmental monitoring, self-position determination, continuous flight attitude adjustments, and obstacle avoidance. Consequently, UAVs exhibit a remarkable capacity to accomplish their designated tasks efficiently<sup>[1]</sup>. The primary objective of UAV path planning is to find an optimal and feasible path within a known environment that is free of conflicts and meets the optimization criteria, given predefined start and end point locations<sup>[2]</sup>. In real-world settings, UAV flight missions are subject to many uncontrollable factors due to the inherent uncertainty and dynamics of the environment. Hence, research on UAV path planning holds profound practical significance.

Scholars have developed a plethora of algorithms to address the problem of two-dimensional path planning, such as the element decomposition method<sup>[3]</sup>, the potential field method<sup>[4]</sup>, and Dijkstra's algorithm<sup>[5]</sup>, among others. Nevertheless, these algorithms typically overlook height constraints and may not align with the practical flight requirements of real UAVs. Therefore, three-dimensional (3D) path planning, which systematically accounts for many constraints, has emerged as the central research focus in path planning. Wu *et al.* introduced a multi-step A\* search algorithm for offline and online path planning for UAVs in a four-dimensional context, encompassing three spatial and temporal dimensions<sup>[6]</sup>. Shorakaei *et al.* devised a path planning methodology that leverages probability graphs, integrating them with genetic algorithms and introducing novel genetic operators to select apt chromosomes for crossover operations<sup>[7]</sup>. Roberge *et al.* proposed using genetic algorithms and particle swarm algorithms to solve autonomous UAV path planning problems in complex 3D environments, taking into account the width of the UAV and the optimal trajectory criterion in 3D environments to reduce the execution time of the solution<sup>[8]</sup>. Abeywickrama *et al.* presented an artificial potential field model that demonstrates remarkable efficiency in reducing collisions among UAVs<sup>[9]</sup>. Vanegas *et al.* proposed a method for optimizing 3D UAV path planning using a non-holonomic constraint path planning approach<sup>[10]</sup>. Lastly, Jain *et al.* introduced an innovative algorithm based on the Multiverse Optimizer algorithm to enhance the time efficiency and precision of UAV path planning within a 3D environment<sup>[11]</sup>. This approach incorporates the Munkres algorithm into UAV path planning, further augmenting its effectiveness. Wang *et al.* introduced an optimized list-based simulated annealing (LBSA) algorithm tailored to address the challenges posed by the large-scale traveling salesman problem (TSP)<sup>[12]</sup>. Li presented a refined tabu search algorithm incorporating a greedy algorithm for addressing the random vehicle routing problem<sup>[13]</sup>. Kala *et al.* integrated a fuzzy inference system with an A\* algorithm to address challenges in robot path planning<sup>[14]</sup>. Since Dorigo *et al.* proposed ant colony optimization (ACO), it has gradually been applied in logistics and path planning<sup>[15]</sup>. The algorithm benefits from strong robustness and good information feedback by imitating the principle of ant colony foraging, which helps solve the challenge of complex path planning. In the 1990s, the prominent representatives of ACO algorithms were the Ant System (AS)<sup>[16]</sup> and the two most successful variants: MAX-MIN Ant System (MMAS)<sup>[17]</sup> and Ant Colony System<sup>[18]</sup>. ACO algorithms have constantly been modified and extensively developed up to this day. Li *et al.* used the geometric optimization method to guide the ants, accelerating the convergence speed<sup>[19]</sup>. However, there was an issue with individual ants becoming disoriented. Literature<sup>[20,21]</sup> combines self-adaptation and ACO algorithms to improve the algorithm's capability to find the global optimum through adaptive parallelism and information updating strategies. Despite these advancements, it is worth noting that the resulting path generated by this algorithm still exhibits non-smooth characteristics. Chen *et al.* incorporated Poisson distribution to simulate the influence of unknown factors and established a three-color raster map<sup>[22]</sup>. The improved algorithm can design the optimal route safely and effectively in the environment under the influence of unknown factors. Yi *et al.* introduced a multi-factor heuristic function strategy to improve ACO's global search capability and convergence<sup>[23]</sup>. Ning *et al.* designed an enhanced pheromone update mechanism based on ACO, strengthening pheromones on edges and enhancing global search capabilities and convergence<sup>[24]</sup>. Wang *et al.* transformed ACO into a Time-Sensitive Network (TSN), resulting in better convergence speed, optimization ability, and reduced susceptibility to local optima compared to traditional ACO<sup>[25]</sup>. Miao *et al.* proposed an improved Adaptive ACO (IAACO)

strategy for integrated global optimization in robot path planning<sup>[26]</sup>. Lyridis presented an enhanced fuzzy logic ACO method, demonstrating superior performance to traditional ACO<sup>[27]</sup>. Hou *et al.* introduced an enhanced ACO approach with a communication mechanism, accelerated convergence through an extended roulette wheel, and designed an adaptive sigmoid decay function to optimize heuristic information in different stages<sup>[28]</sup>. Although the research above has improved ACO and achieved preliminary results, they have not fully considered the maneuverability constraints of UAVs in real-world scenarios. To improve the algorithm's ability to search globally, speed up the convergence rate, and generate safe and smooth paths, which will lead to more efficient UAV path planning that meets practical requirements, it is necessary to optimize the existing research further.

In this paper, we propose a novel path planning method based on a dual-strategy ACO (DSACO) algorithm. Our approach centers on optimizing the state transition function and pheromone update rules to enhance the algorithm's performance. Firstly, we refine the heuristic factor of the state transition function by incorporating 3D characteristics, which include adding heuristic information regarding the distances between adjacent nodes and the heights of the mountains. Then, a path evaluation function is proposed based on distance, height, and turning cost. The dynamically adjusted pheromone update strategy helps ants to conduct a global search in the early stage of the algorithm, accelerates convergence in the later stage of the algorithm, and guides ants towards the path of the global optimal solution. Doing so effectively steers the ants towards the path leading to the global optimal solution. Based on the above improvements, a new variation of ACO called the DSACO algorithm is formed. Subsequently, it is compared with other algorithms based on different terrain environments. Experimental results prove that DSACO has superior global search capabilities and convergence characteristics from four aspects: path length, fitness values, iteration number, and running time.

## 2. PROBLEM STATEMENT

This paper primarily addresses the issue of static path planning. In this context, static path planning entails the establishment of an environment model for UAV path planning while simultaneously considering the performance constraints and a comprehensive assessment of the costs associated with the UAV. The ultimate objective is to pre-plan the path before the UAV embarks on its flight mission.

### 2.1 3D path planning environment modeling

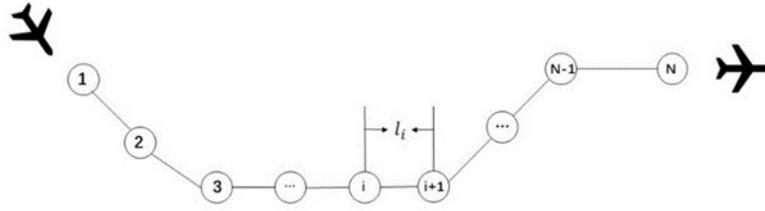
In static path planning, the UAV's flight environment can be ascertained before takeoff. Consequently, environment modeling is vital as it serves as the cornerstone upon which the UAV can base its search for the optimal path, ultimately facilitating the efficient execution of tasks.

#### 2.1.1 Mountain modeling

This paper studies the problem of UAV path planning in the 3D mountain environment. Given that mountains can be approximated as cones, the mountainous terrain is characterized by multiple cones with distinct positions and shapes. We employ a 3D figure described by a natural exponential function with the base number "e" to elucidate this concept. In this representation, the xOy plane serves as the horizontal reference, and a point on the mountain is denoted as (x, y, z). The terrain of the natural mountain is described through an exponential function, as illustrated in Equation (1):

$$Z(x, y) = \sum_{i=1}^n h_i e^{-\frac{(x-x_{oi})^2}{x_{si}^2} - \frac{(y-y_{oi})^2}{y_{si}^2}} \tag{1}$$

Among them,  $Z(x, y)$  represents the height value at the point (x, y),  $n$  represents the number of peaks in the mountain environment,  $(x_{oi}, y_{oi})$  represents the center coordinates of the  $i^{th}$  peak,  $h_i$  represents the maximum height of the  $i^{th}$  mountain, and  $(x_{si}, y_{si})$  represents the slope of the mountain. The advantage of simulating mountain peaks with a two-dimensional normal distribution function is that it allows for convenient simula-



**Figure 1.** Schematic diagram of UAV path.

tion of mountain topography with varying heights and numbers by adjusting parameters.

### 2.1.2 Search area rules

In the 3D coordinate system, the x-axis represents the longitude direction, the y-axis represents the latitude direction, and the z-axis represents the altitude dimension. When defining the operational space for the UAV's task execution, its path space within this 3D coordinate system is also established. By utilizing  $x_{Grid}$ ,  $y_{Grid}$ , and  $z_{Grid}$  as the step sizes, we partition the x, y, and z dimensions into equal intervals. This process yields a discretized set of 3D points within the path planning space. Consequently, the UAV's path planning task can be abstractly interpreted as selecting path points from these 3D coordinates, ultimately determining the optimal path from the initial point S ( $x_s, y_s, z_s$ ) to the destination point T ( $x_T, y_T, z_T$ ).

To streamline the path planning process, we establish the primary direction for ant movement as the longitude direction. This means that the UAV moves along the x-axis with a fixed step of  $x_{Grid}$ . Additionally, the ants are limited to a maximum allowable distance in the latitudinal and altitudinal directions,  $D_{y_{max}}$  and  $D_{z_{max}}$ , respectively. This restriction reduces the search space for the ants when selecting the next path point, resulting in improved algorithm efficiency.

## 2.2 Maneuverability constraints of UAV

To ensure the feasibility and practical relevance of UAV path planning, it is crucial to consider both environmental conditions and the performance constraints of the UAV itself. This paper considers several key performance constraints, including the following aspects:

### (1) Maximum path distance

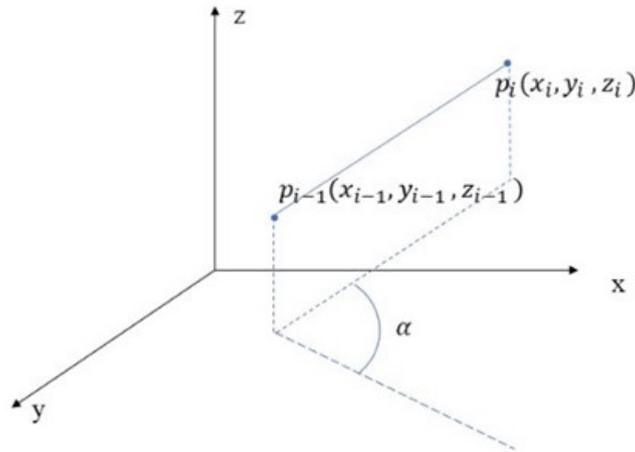
The maximum path distance is the farthest distance that the UAV can fly while utilizing its total energy capacity. The path obtained by planning generally refers to the total length of each node on the search path. Assuming that the number of nodes in a certain path is  $N$ ,  $l_i$  is the length of the  $i^{th}$  path,  $L$  is the total path distance, and  $L_{max}$  is the maximum path distance. The schematic diagram of the UAV path is shown in Figure 1, and the constraint expression that the path length must satisfy is shown in Equation (2).

$$L = \sum_{i=1}^N l_i \leq L_{max} \quad (2)$$

### (2) Minimum segment length

When the UAV changes its flight direction, it needs to maintain the original direction and continue flying for a certain distance due to the influence of inertia. The minimum segment length is the shortest distance to continue flying in the original direction before changing. Let  $L_{min}$  be the distance of the minimum segment, and  $l_i$  be the length of the  $i^{th}$  segment of the path. The length of each flight path should satisfy Equation (3):

$$L = l_i \geq L_{min} \quad (3)$$



**Figure 2.** Pitch angle diagram of UAV path point.

(3) Maximum and minimum flight altitudes

UAVs must fly at a specific altitude above the ground to ensure safety for both themselves and their operators. Since the flight altitude varies with the path, imposing a minimum altitude constraint is necessary. At the same time, to maintain regular communication with the ground, reduce energy consumption, and ensure its protection, the maximum altitude of the flight needs to be limited. Let  $H_{min}$  be the minimum flight altitude,  $H_{max}$  be the maximum flight altitude, and  $h_i$  be the altitude of the path point  $i$ . At this time, the UAV flight altitude constraint can be expressed by Equation (4):

$$H_{min} \leq h_i \leq H_{max} \tag{4}$$

(4) Maximum pitch angle

Due to the influence of the UAV’s physical performance, cargo weight, and obstacle avoidance ability, it is necessary to limit its pitch angle. If the pitch angle is too large, it can easily cause overturning and compromise safety. Let  $(x_{i-1}, y_{i-1}, z_{i-1})$  be the position coordinates of path point  $i-1$ ,  $(x_i, y_i, z_i)$  be the position coordinates of path point  $i$ , and  $\alpha_{max}$  is the maximum pitch angle of the UAV. The schematic diagram of the pitch angle of the UAV in flight is shown in Figure 2, and the mathematical expression of the constraint equation is shown in Equation (5):

$$\tan^{-1} \left[ \frac{|z_i - z_{i-1}|}{\sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2}} \right] \leq \alpha_{max} \tag{5}$$

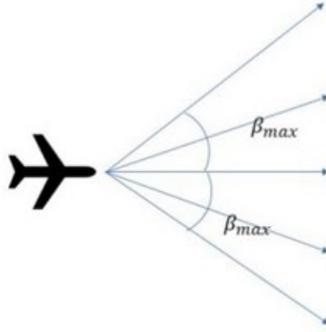
(5) Maximum horizontal turning angle

UAVs are limited by their mechanical properties and must adhere to a specified angle range when changing flight direction. The smaller the turning angle, the more stable the flight of the UAV. Let  $\beta_{max}$  be the maximum horizontal turning angle; the constraints to be satisfied are shown in Equation (6), and the schematic diagram is shown in Figure 3.

$$\beta \leq \beta_{max} \tag{6}$$

**3. BASELINE ANT COLONY OPTIMIZATION ALGORITHM**

The ACO algorithm draws inspiration from the foraging behavior observed in ant species. In this natural behavior, ants deposit pheromones on the ground to mark favorable paths that other colony members should



**Figure 3.** UAV maximum turning angle.

follow. The core principle of this algorithm involves representing potential solutions to a problem through paths selected by the simulated ants. These paths collectively constitute the space of feasible solutions, and within this space, the optimal solution corresponds to the shortest path.

### 3.1 State transition function

The ant starts its journey from the initial point, calculating the transition probabilities to various states. It selects the next node based on the pheromone level  $\tau_{ij}$  and the heuristic function  $\eta_{ij}$  until it reaches the target point. In the early stages of the ACO algorithm, pheromone accumulation is minimal, and path selection primarily relies on the heuristic function. As pheromone concentrations reach a certain threshold, the heuristic function and the pheromone level influence path selection. In the traditional ACO algorithm, the probability of a transition from node  $i$  to node  $j$  for ant  $k$  can be expressed using Equation (7):

$$P_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha [\eta_{ij}(t)]^\beta}{\sum_{s \in C} [\tau_{is}(t)]^\alpha [\eta_{is}(t)]^\beta} & j \in C \\ 0 & \text{others} \end{cases} \quad (7)$$

where  $C$  is the set of next nodes that the current node can choose,  $\alpha$  and  $\beta$  are the weight parameters of pheromone and heuristic function, respectively, reflecting the weight of the ant colony's prior knowledge and unexplored paths.  $\tau_{ij}$  represents the pheromone concentration,  $\eta_{ij}$  represents the heuristic function,  $i$  represents the current ant, and  $j$  represents the candidate node.

### 3.2 Pheromone update rules

Pheromone concentrations naturally diminish over time. The pheromone update process occurs once all ants have finished their path search from the starting point to the target point. This pheromone update involves two components: the residual pheromone and the freshly released pheromone. The update process is shown in Equations (8)-(10):

$$\tau_{ij}(t+1) = (1 - \rho)\tau_{ij}(t) + \Delta\tau_{ij}(t) \quad (8)$$

$$\Delta\tau_{ij} = \sum_{k=1}^m \Delta\tau_{ij}^k \quad (9)$$

$$\Delta\tau_{ij}^k = \begin{cases} Q/L_k & \text{Ant } k \text{ used edge}(i, j) \text{ in its tour} \\ 0 & \text{others} \end{cases} \quad (10)$$

In Equations (8)-(10),  $\tau_{ij}$  denotes the total amount of pheromones,  $\rho$  is the evaporation rate,  $\Delta\tau_{ij}$  represents the increment of pheromone,  $\Delta\tau_{ij}^k$  is the quantity of pheromone laid on edge  $(i, j)$  by ant  $k$ ,  $Q$  is a constant, and  $L_k$  is the length of the tour constructed by ant  $k$ .

#### 4. PATH PLANNING BASED ON DUAL STRATEGY ACO ALGORITHM

The traditional ACO algorithm often relies on a stochastic mechanism to choose the next node, which can lead to an indiscriminate search during the initial phase of the algorithm and result in slow convergence. While the algorithm does possess positive feedback traits, if it initially identifies a sub-optimal solution, this positive feedback can subsequently trap the algorithm in a local optimum. To address the aforementioned issues, enhancements are implemented in two critical aspects: (1) the method for selecting candidate nodes; and (2) the rules governing pheromone updates.

##### 4.1 State transition strategy of deterministic selection

While searching for the next feasible node, the heuristic values for candidate nodes are determined using the Equation  $Q_{search} = S * M * D$ . The specific equations for S, M, and D are provided in Equations (11)-(13), respectively.

$$S = \begin{cases} 1 & h_{t+1} \geq H + h_c \\ 0 & \text{others} \end{cases} \tag{11}$$

$$M = \frac{1}{1 + h_{t+1}} \tag{12}$$

$$D = \frac{1}{\sqrt{1 + \Delta y^2 + \Delta h^2} + \sqrt{1 + (x_B - x_{t+1})^2 + (y_B - y_{t+1})^2 + (h_B - h_{t+1})^2}} \tag{13}$$

Among them, S signifies the accessibility of the next node, taking the value 1 if the node is reachable and 0 otherwise.  $h_c$  is the safety threshold, and H represents the height of the mountain node. M represents the probability of selecting a candidate node,  $x_{t+1}$ ,  $y_{t+1}$ , and  $h_{t+1}$  represent the 3D coordinates of the candidate node,  $x_B$ ,  $y_B$ , and  $z_B$  are the 3D coordinates of the mountain node, D is the correlation coefficient between two nodes,  $\Delta y$  represents the interpolation of the y-axis of the two nodes, and  $\Delta z$  represents the interpolation of the z-axis of the two nodes.

The traditional ACO algorithm employs a probabilistic transition strategy where transition probabilities between nodes are calculated, and the roulette method is commonly used to select the next node. As the algorithm progresses into its later stages, paths with clear advantages become more evident, and a deterministic node transition strategy can be applied to expedite algorithm convergence. In light of this, we have designed a novel state transition function to choose the subsequent node at the later stage of the algorithm, as shown in Equation (14):

$$P_{ij}^k(t) = \begin{cases} \operatorname{argmax}_{s \in C} [\tau_{is}(t)]^\alpha [\eta_{is}(t)]^\beta & q \leq q_0 \\ \begin{cases} \frac{[\tau_{ij}(t)]^\alpha [\eta_{ij}(t)]^\beta}{\sum_{s \in C} [\tau_{is}(t)]^\alpha [\eta_{is}(t)]^\beta} & j \in C \\ 0 & \text{others} \end{cases} & q \geq q_0 \end{cases} \tag{14}$$

where q is a tunable parameter within the interval [0, 1] that represents the probability of using a deterministic node transition strategy.  $q_0$  determines the probability of selecting either the deterministic or random selection modes. Consequently,  $q_0$  plays a crucial role in influencing the convergence speed and the global search capability. A higher value of  $q_0$  favors the likelihood of choosing the deterministic mode for selecting the next point, resulting in an accelerated convergence speed. However, this comes at the cost of reduced global search capability. Conversely, a smaller  $q_0$  inclines the path shift towards the random mode, introducing more randomness in the selection process and enhancing the global search capability. A rule for setting the  $q_0$  value must be established to balance between deterministic and random modes.  $q_0$  is calculated as shown in

Equation (15):

$$\begin{cases} q_0 = \begin{cases} \frac{K-k}{K}q_{0\_initial} & k < k_0 \\ \frac{k-k_0}{K}q_{0\_initial} + \frac{q_{0\_initial}}{2} & k \geq k_0 \end{cases} \\ k_0 = 0.7K \end{cases} \quad (15)$$

where  $k$  is the current number of iterations,  $K$  is the maximum number of iterations, and  $q_{0\_initial}$  is the original value of  $q_0$ .

#### 4.2 Dynamically adjusted pheromone update strategy

In traditional ACO algorithms applied to path planning, there is a risk that if the pheromone level on a particular path becomes excessively high, it can lead to a higher probability of subsequent ants choosing that path. This can constrain exploring other potentially more feasible paths, causing the algorithm to stagnate prematurely. We have introduced improvements to the pheromone update strategy to address this concern. Specifically, we now update the pheromone solely on the path traversed by the optimal ant in the current iteration. Additionally, we enforce maximum and minimum limits on the pheromone values to prevent them from becoming overly dominant or negligible, as shown in Equations (16)-(18):

$$\tau_{ij}^{best}(t+1) = (1-\rho)\tau_{ij}^{best}(t) + \Delta\tau_{ij}^{best}(t) \quad (16)$$

$$\Delta\tau_{ij}^{best}(t) = \frac{Q}{(1-p)Fitness_{now} + pFitness_{global}} \quad (17)$$

$$\tau_{ij}^{best}(t+1) = \begin{cases} \tau_{max} & \tilde{\tau}_{ij}^{best}(t+1) \geq \tau_{max} \\ \tilde{\tau}_{ij}^{best}(t+1) & \tau_{min} < \tilde{\tau}_{ij}^{best}(t+1) < \tau_{max} \\ \tau_{min} & \tilde{\tau}_{ij}^{best}(t+1) \leq \tau_{min} \end{cases} \quad (18)$$

Among them,  $\Delta\tau_{ij}^{best}$  represents the change in pheromone concentration between two nodes under the current optimal path,  $\rho$  is the pheromone evaporation parameter, which determines the pheromone attenuation ratio at the current point after each update,  $p$  represents the weight of the global optimal fitness value as the basis for pheromone update,  $Q$  is the pheromone constant.  $Fitness_{now}$  represents the fitness value corresponding to the current path, and  $Fitness_{global}$  represents the global optimal fitness value.  $\tau_{max}$  represents the maximum pheromone concentration,  $\tau_{min}$  represents the minimum pheromone concentration, and  $\tau_{ij}^{best}(t+1)$  is the pheromone concentration at the  $t+1$  iteration.

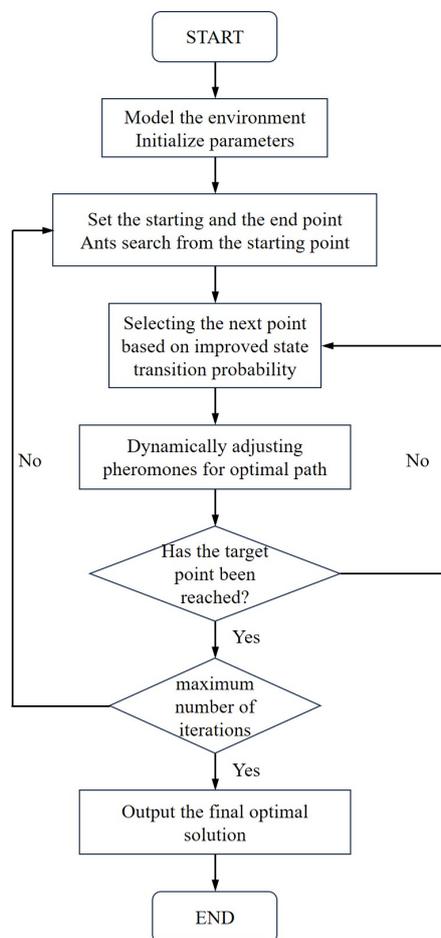
In order to ensure that ants have more opportunities to explore new paths at the beginning of the iteration and optimize the optimal path in the later stages of the iteration, the probability  $p$  decreases proportionally as the number of iterations increases. We set the attenuation ratio, represented as  $\gamma_p$ , to 0.9, as shown in Equation (19):

$$p(t+1) = \begin{cases} p(t) & mod(t, 100) \neq 0 \\ p(t) * \gamma_p & mod(t, 100) = 0 \end{cases} \quad (19)$$

$p \in [0, 1]$  represents the weight of the global optimal fitness value as the basis for pheromone update.

#### 4.3 Path evaluation function

The heuristic function is crucial in determining the search path, and its strengths and weaknesses directly influence the algorithm's convergence speed during iterations. Due to the traditional heuristic function having fewer constraints, the blind search phenomenon will occur in the early search for ants. Therefore, the algorithm exhibits slow convergence and requires many iterations to reach a solution. Considering the distinctive application environment of UAVs, this paper introduces an evaluation function grounded in distance, height,



**Figure 4.** UAV path planning flow chart based on dual-strategy ant colony algorithm.

and turning costs, as depicted in Equation (20). The calculation methods of distance, height, and number of turns are shown in Equations (21)-(23), respectively.

$$Fitness = f_1 D_0 + f_2 D_h + f_3 D_T \tag{20}$$

$$D_0 = \sum_{i=1}^N l_i \tag{21}$$

$$D_h = \sum_{i=1}^N h_i \tag{22}$$

$$D_T = \sum_{i=1}^N T_i \tag{23}$$

where  $l_i$  is the length of the  $i^{th}$  flight segment,  $N$  is the number of flight segments,  $h_i$  is the height of the  $i^{th}$  flight segment,  $T_i$  is the angle between the  $i^{th}$  flight segment and the previous one, and  $f_1, f_2,$  and  $f_3$  are the adjustment coefficients of each factor.

Based on the above improvements, a new variation of ACO called the DSACO algorithm is formed. This algorithm is summarized and detailed in Algorithm 1. Moreover, the overall flow chart of UAV path planning based on the DSACO algorithm is shown in Figure 4.

**Algorithm 1:** dual-strategy ant colony optimization (DSACO)

---

**Data:** Mountain Information, Set of ants  
**Result:** The optimal path

- 1 Initialization parameter information;
- 2 **for**  $i=1$  to the size of the map **do**
- 3 |   **if** the grid is free **then**
- 4 | |   Initialize pheromone concentration;
- 5 |   **end**
- 6 **end**
- 7 **for**  $k=1$  to population size **do**
- 8 |   Place all ants at the start point S;
- 9 |   **for**  $iter=1$  to the Maximum number of iterations **do**
- 10 | |   **for**  $p=-D_{ymax}$  to  $D_{ymax}$  **do**
- 11 | | |   **for**  $q=-D_{zmax}$  to  $D_{zmax}$  **do**
- 12 | | | |   **if** the position is in the map **then**
- 13 | | | | |   Calculate the heuristic function value according to Equation 19;
- 14 | | | | |   Calculate the state transition probability of the next node according to Equation 14;
- 15 | | | |   **end**
- 16 | | |   **end**
- 17 | |   **end**
- 18 |   Select the next point, update the current location of ants;
- 19 |   Dynamically updated pheromones according to Equation 15;
- 20 |   **if** pheromone concentration  $\geq \tau_{max}$  **then**
- 21 | |   pheromone concentration =  $\tau_{max}$ ;
- 22 |   **else**
- 23 | |   **if** pheromone concentration  $\leq \tau_{min}$  **then**
- 24 | | |   pheromone concentration =  $\tau_{min}$ ;
- 25 | |   **end**
- 26 |   **end**
- 27 **end**
- 28 **end**
- 29 Output the final optimal path.

---

**Table 1. Simple mountain environment parameters**

Serial number of peaks	Location(x,y)/km	Height/km	Slope/km
1	(30,9)	0.66	(4,3)
2	(16,17)	1.64	(7,8)
3	(35,32)	1.3	(5,8)

**Table 2. Medium complex mountain environment parameters**

Serial number of peaks	Location(x,y)/km	Height/km	Slope/km
1	(40,10)	1.8	(4,3)
2	(12,17)	0.66	(2,1)
3	(21,25)	1.86	(4,4)
4	(33,34)	1.64	(7,8)
5	(10,41)	0.3	(5,8)
6	(14,32)	0.96	(12,8)
7	(6,26)	1.3	(6,5)
8	(33,15)	2.3	(4,12)

**Table 3. Complex mountain environment parameters**

Serial number of peaks	Location(x,y)/km	Height/km	Slope/km
1	(40,10)	1.08	(4,3)
2	(35,45)	0.66	(2,2)
3	(36,25)	1.86	(3,5)
4	(15,40)	1.64	(2,3)
5	(10,25)	0.3	(8,5)
6	(33,44)	1.12	(10,8)
7	(47,30)	1.6	(6,6)
8	(20,24)	0.64	(7,8)
9	(10,41)	1.34	(5,8)
10	(14,32)	0.96	(12,8)
11	(6,26)	1.3	(6,5)
12	(33, 42)	1.5	(4, 8)

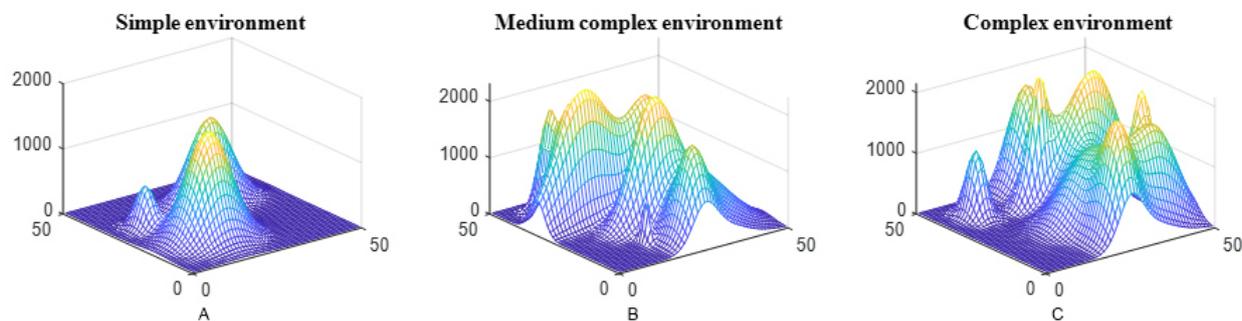
#### 4.4 Complexity analysis

In this section, the time complexity and space complexity of DSACO are analyzed. Simple instructions in the algorithm are omitted, which does not affect its computational complexity. The time complexity of the DSACO mainly depends on the problem size, the number of iterations, and the number of ants in each iteration. The time complexity of the DSACO algorithm is  $O(I \cdot M \cdot N)$ , where  $I$  is the number of iterations,  $M$  is the problem size, and  $N$  is the number of ants. The problem size is related to the search space. The space complexity of DSACO is mainly affected by the pheromone matrix. Typically, the size of the pheromone matrix is  $M \times M$ , where  $M$  is the problem size. Therefore, the space complexity of DSACO is  $O(M^2)$ .

## 5. SIMULATION RESULTS

### 5.1 Mountain modeling and parameters setting

The experiment adopts the grid method to simulate the 3D mountain environment. The size of the terrain is set to  $50 \text{ km} \times 50 \text{ km} \times 2.4 \text{ km}$ , the length and width of each grid in the horizontal plane are 1 km, and each grid in the vertical direction is 0.2 km. To enhance applicability, simulations are conducted in environments of varying complexity, including simple, moderately complex, and highly complex mountainous environments. According to the parameters in Tables 1-3 and Equation (1), the mountain environment is modeled based on MATLAB R2018a software, as shown in Figure 5.



**Figure 5.** We present mountain terrain in different environments in Figure 5A-C. (A) Simple environment; (B) Medium complex environment; (C) Complex environment.

## 5.2 Parameter optimization of DSACO

The parameter selection of ACO directly influences its performance. Currently, no well-established theoretical analysis method can decisively determine the optimal parameter combination. Therefore, to identify suitable DSACO parameters, we conducted a statistical analysis of the critical parameters. Specific test parameters include the pheromone evaporation rate  $\rho$  in Equation (16) and  $\tau_{max}$  and  $\tau_{min}$  in Equation (17). In each experiment, only one parameter was modified while keeping the other parameters constant. Ten simulations were conducted for each selected parameter combination to minimize the impact of random errors. The experiments use a complex mountainous terrain model, as shown in Figure 5C, with the parameter configurations detailed in Table 3.

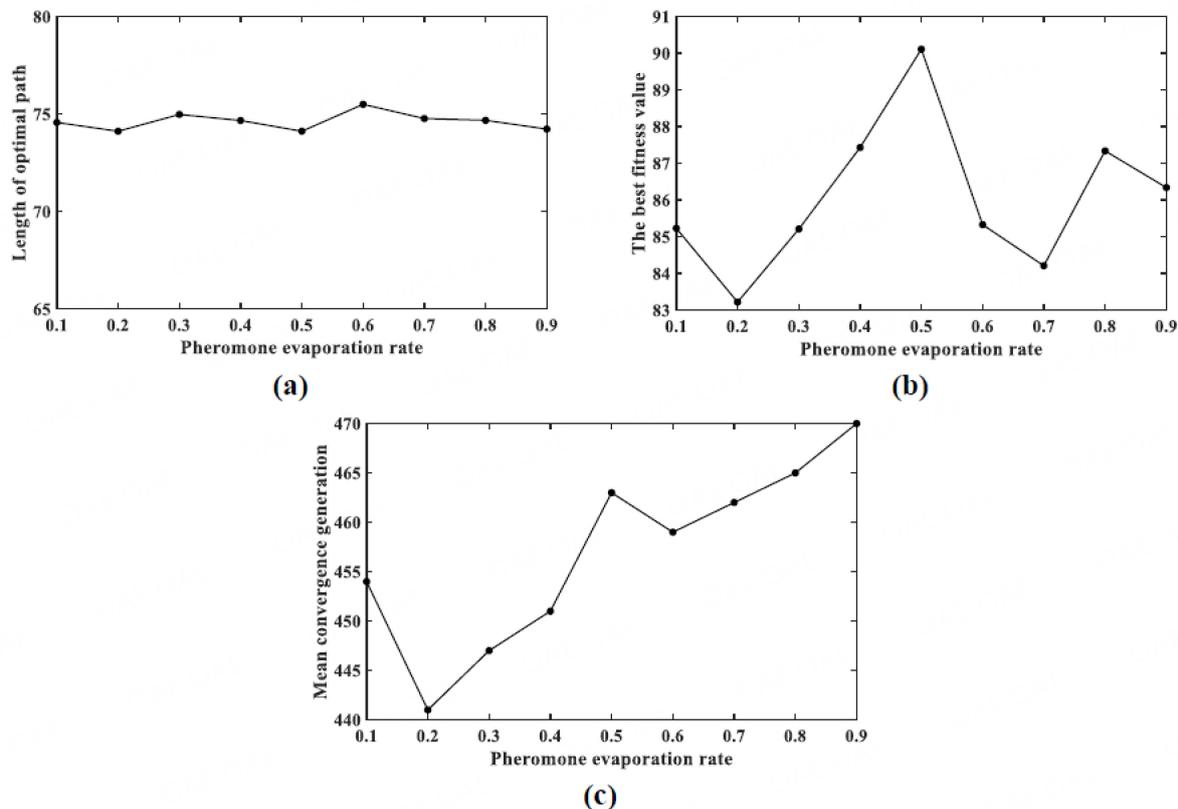
The first parameter is pheromone evaporation rate  $\rho$ . Value for  $\rho$  is varied from 0.1 to 0.9 in increments of 0.1. Simultaneously, other critical parameters of DSACO are held constant:  $m = 10$ ,  $n = 500$ ,  $Q = 100$ ,  $\alpha = 1$ ,  $\beta = 8$ ,  $\tau_{max} = 0.9$ ,  $\tau_{min} = 0.03$ ,  $p = 1$ . The experimental results, including optimal path length, optimal fitness value, and average convergence generation, are illustrated in Figure 6. Despite the negligible impact of  $\rho$  on the optimal path length, as shown in Figure 6A, the optimal fitness value and average convergence generation reach their minimum when  $\rho = 0.2$ . Therefore, it is recommended to set  $\rho$  to 0.2.

The second parameter is  $\tau_{max}$ . Values for  $\tau_{max}$  are varied from 0.5 to 0.9 in increments of 0.05, and other critical parameters of DSACO are held constant:  $m = 10$ ,  $n = 500$ ,  $Q = 100$ ,  $\alpha = 1$ ,  $\beta = 8$ ,  $\rho = 0.2$ ,  $\tau_{min} = 0.03$ ,  $p = 1$ . Experimental results for different values are illustrated in Figure 7. Figure 7A shows that the choice of  $\tau_{max}$  has minimal impact on the optimal path length. Ignoring the fluctuations depicted in Figure 7B, a larger value of  $\tau_{max}$  is more likely to yield a smaller fitness value. Figure 7C indicates that a smaller value of  $\tau_{max}$  increases the average convergence generation, suggesting the need for a larger  $\tau_{max}$  to enhance convergence speed. Overall, the value of  $\tau_{max}$  will be set to 0.9.

The third parameter is  $\tau_{min}$ . Values for  $\tau_{min}$  are varied from 0.01 to 0.09 in increments of 0.01, and other critical parameters of DSACO are held constant:  $m = 10$ ,  $n = 500$ ,  $Q = 100$ ,  $\alpha = 1$ ,  $\beta = 8$ ,  $\rho = 0.2$ ,  $\tau_{max} = 0.9$ ,  $p = 1$ . The experimental results for different  $\tau_{min}$  values are illustrated in Figure 8. Concerning the optimal path length and fitness value, the influence of  $\tau_{min}$  is not particularly pronounced, as depicted in Figure 8A and B. Regarding the average convergence generation, a peak is observed when  $\tau_{min}$  is set to 0.03, as shown in Figure 8C. Therefore, the value of  $\tau_{min}$  will be set to 0.03 to attain improved DSACO performance.

## 5.3 Algorithm comparison imitation

After analyzing the influences of the main parameters of DSACO, an optimal combination of main parameters is obtained, as shown in Table 4. Set the starting point A coordinates (1, 17, 0.6) and the target point B coordi-



**Figure 6.** The influences of  $\rho$  on path length, fitness values, and mean convergence generation.

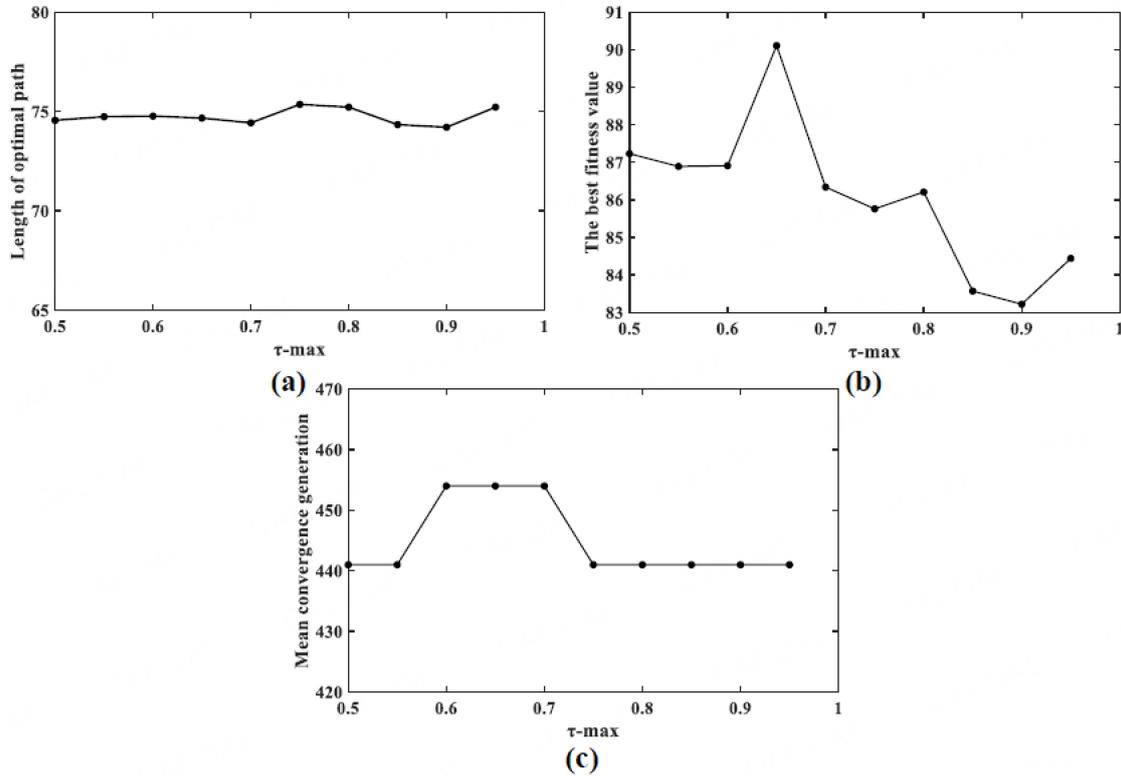
**Table 4.** The values of the experimental parameters

Symbol	Description	Value
$m$	Number of ants	10
$n$	Number of iterations	500
$Q$	Pheromone intensity	100
$\alpha$	Pheromone incentive factor	1
$\beta$	Expected heuristic factor	8
$\rho$	Pheromone evaporation rate	0.2
$\tau_{max}$	The maximum value of pheromone	0.9
$\tau_{min}$	The minimum value of pheromone	0.03
$\rho$	The weight of the global optimal fitness value	1

nates (50, 42, 0.6). The maximum single moving distance of the UAV in the horizontal plane is two grids, and the maximum moving distance in the vertical direction is one grid. To ensure that the height of the UAV is within the safe range, the height constraint is set to  $50 \text{ m} < h < 2 \text{ km}$ .

In order to illustrate the superiority of the method in this paper, the well-known improved ACO approaches: AS and MMAS, the improved ACO by Chen *et al.*, and the DSACO in this paper are introduced for simulation and comparison research.

Figures 9-11 show the path planning simulation diagrams of four different algorithms in simple, medium, complex, and complex environments, respectively. Through comparison, it is found that despite no apparent difference between the paths obtained by the four algorithms in simple mountain environments, DSACO seeks the shortest path compared to other approaches in medium complex and complex mountain environments due



**Figure 7.** The influences of  $\tau_{max}$  on path length, fitness values, and mean convergence generation.

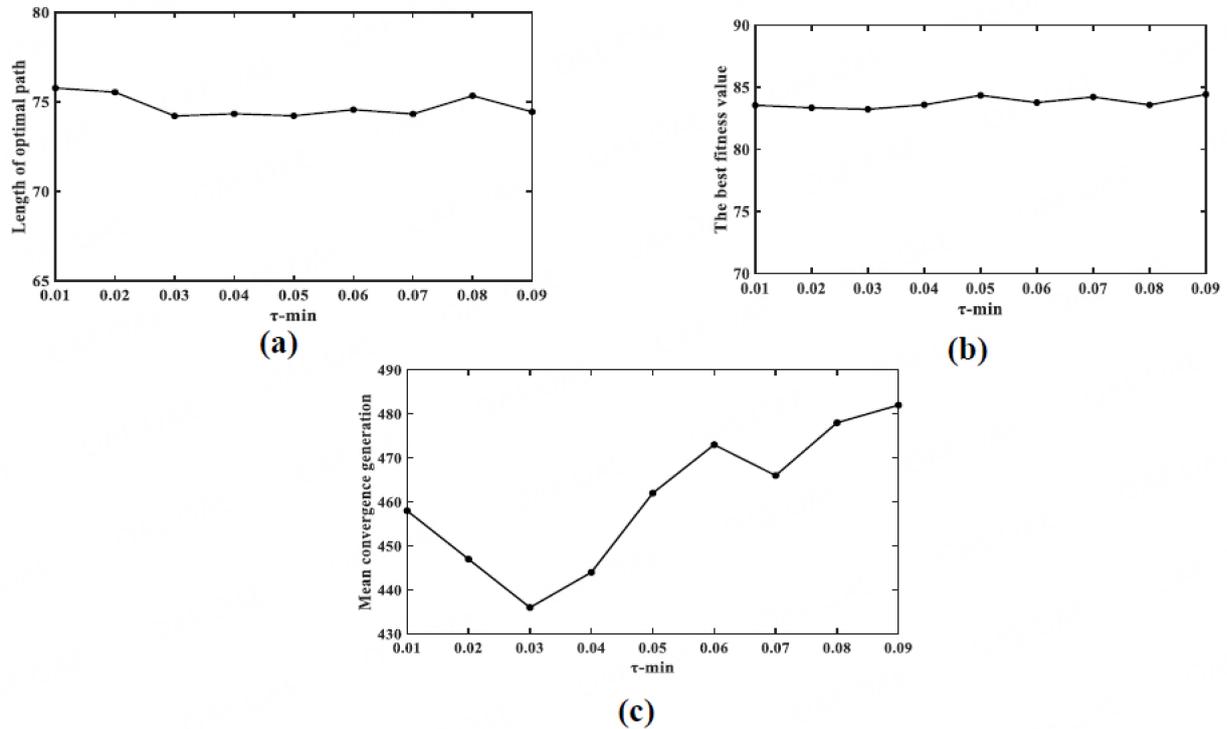
**Table 5. Simulation results in complex terrain environment**

	AS	MMAS	Improved ACO by Chen <i>et al.</i>	DSACO
Optimal path length	79.9710	79.2324	77.4436	74.1204
Optimal fitness value	90.7962	89.5247	84.5741	83.0249
Number of iterations	421	407	398	383
Running time/s	3.3321	3.7479	4.2152	3.9541

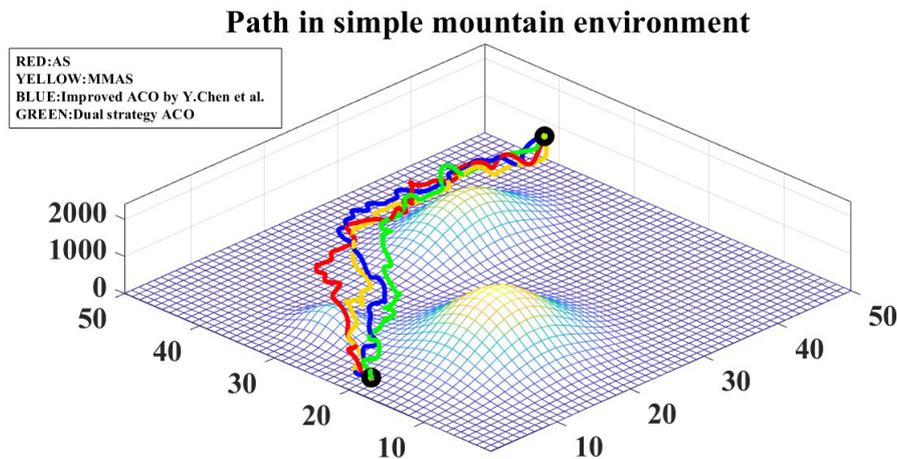
to optimizing the pheromone update mechanism and limiting the pheromone value, the approximate global optimal path is found.

Figures 12-14 show the best individual fitness trends of the four different algorithms under simple, medium, complex, and complex environments, respectively. By comparing the simulation results, we found that the traditional ACO algorithm has a good convergence speed in a simple environment. By comparing the simulation results, we found that AS and MMAS have an excellent convergence speed in a simple environment. However, they are prone to get stuck in local optima in a complex environment. Compared with the improved ACO algorithm by Chen *et al.*, the fitness of DSACO can be reduced to a lower level, indicating that the algorithm has a better path search capability. At the same time, DSACO can reach a smaller fitness with fewer iterations, which indicates that the algorithm has a faster convergence speed.

In order to minimize errors, we conducted 30 experiments and calculated the average values of the optimal path length, the optimal fitness value, the number of iterations required to reach the optimal fitness value, and the running time for the four different algorithms in complex mountain environments, as shown in Table 5.



**Figure 8.** The influences of  $\tau_{min}$  on path length, fitness values, and mean convergence generation.

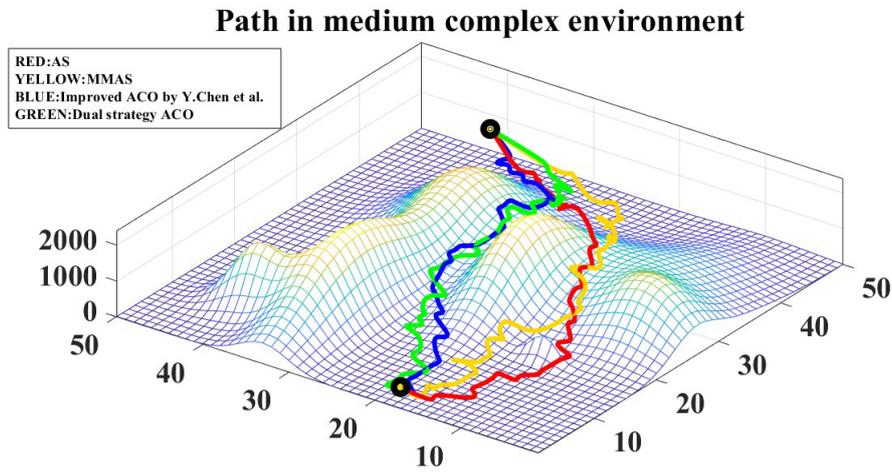


**Figure 9.** The path of four algorithms in simple mountain environments.

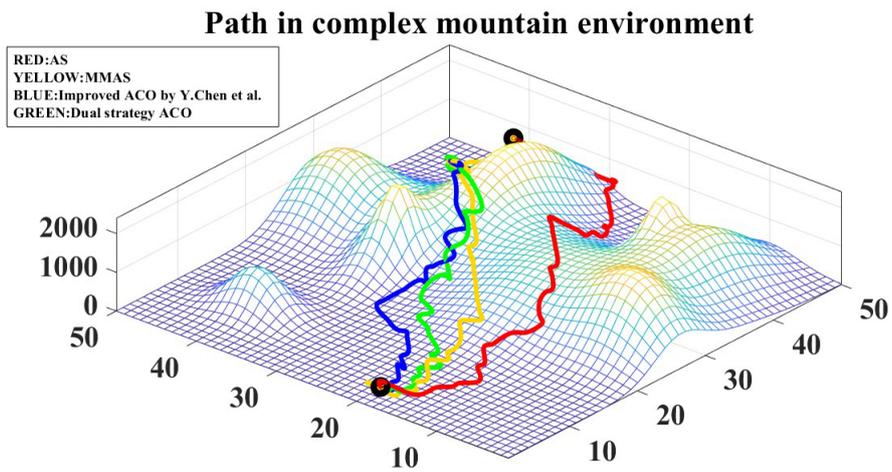
To sum up, DSACO has advantages in path length, fitness values, and number of iterations. At the same time, due to the increase in heuristic function constraints, the running time is better than the improved ACO and slightly worse than AS and MMAS, but it is also within our acceptable range.

## 6. CONCLUSIONS

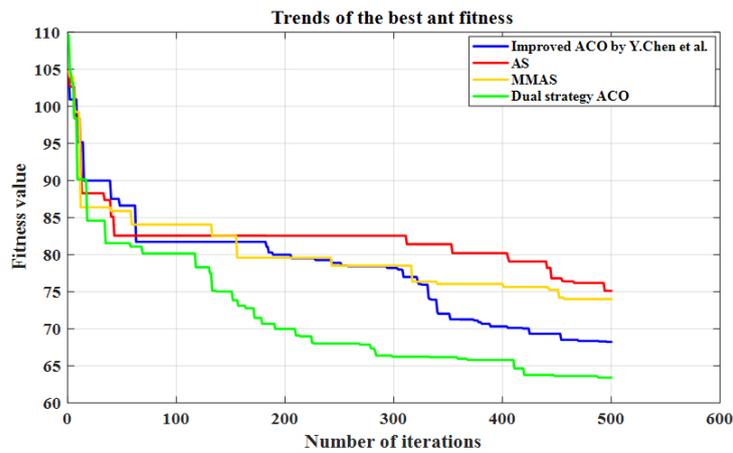
In this paper, we optimized and improved the traditional ACO algorithm, which is prone to getting stuck in local optima and has a slow convergence speed. Employing the deterministic state transition strategy to re-define the movement rules of the ants and implementing a dynamically adjusted pheromone update strategy



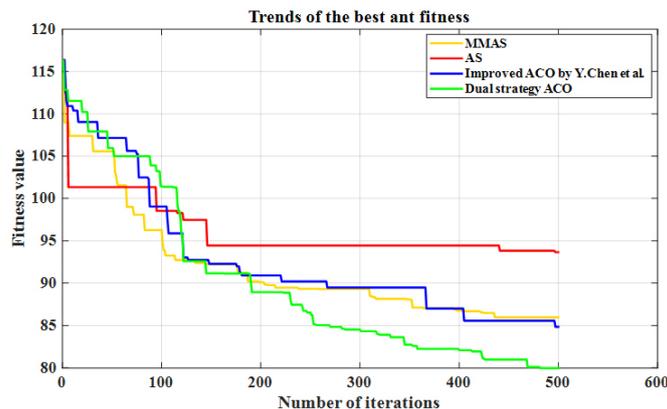
**Figure 10.** The path of four algorithms in moderately complex mountain environments.



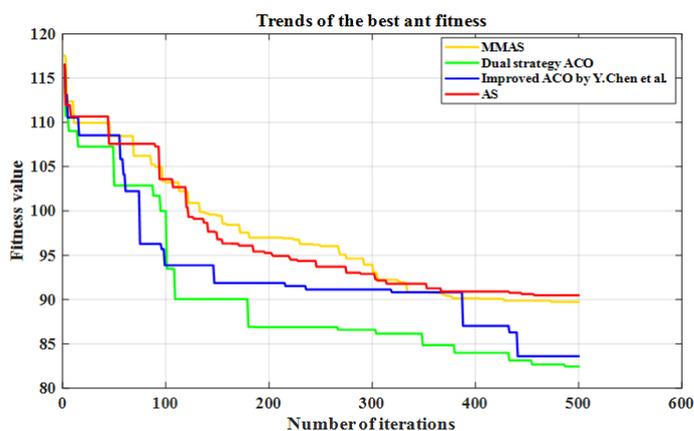
**Figure 11.** The path of four algorithms in complex mountain environments.



**Figure 12.** Convergence trend in simple environments.



**Figure 13.** Convergence trend in medium complex environments.



**Figure 14.** Convergence trend in complex environments.

enhances the performance of path optimization, search efficiency, and convergence speed. This approach prevents the occurrence of local optima and improves the path planning performance of the UAV. The simulation results show that the path length of DSACO is reduced by 7.3%, 6.5%, and 4.3%, respectively, compared with AS, MMAS, and improved AS. Compared with AS, MMAS, and improved AS, the fitness value of DSACO decreased by 8.6%, 7.2%, and 2%, respectively. The algorithm reduces the number of iterations from 421, 407, and 398 to 383, and the running time also increases slightly within the acceptable range.

However, the DSACO algorithm still has some shortcomings, such as not considering the influence of dynamic obstacles when constructing the mountain model and not considering the complex kinematics and dynamics constraints of UAVs, which limits its application. In the future, the path planning method can be improved in the following aspects: (1) the current path planning method mainly deals with problems in static environments, and in the future, it should be extended to dynamic environments and consider the interaction effects of the aircraft and other moving objects to realize smarter path planning; (2) machine learning technology can be studied to be applied to path planning to improve the intelligence and adaptability of the path planning algorithm by learning a large amount of historical path data and flight experience.

## DECLARATIONS

### Authors' contributions

Made substantial contributions to the conception and design of the study and performed the analysis of the results: Dong N

Carries out algorithm design and improvement and conducted theoretical analysis: Mai X, Liu S, Chen H

### Availability of data and materials

Not applicable.

### Financial support and sponsorship

This work was supported by the National Natural Science Foundation of China (No. 62273253), the Tianjin Natural Science Foundation Key Project (No. 22JCZDJC00330), and the funding of Joint Laboratory for Electric Power Robots of China Southern Power Grid Co., Ltd. and Electric Power Research Institute of Guangdong Power Grid Co., Ltd (No. GDDKY2022KF06).

### Conflicts of interest

All authors declared that there are no conflicts of interest.

### Ethical approval and consent to participate

Not applicable.

### Consent for publication

Not applicable.

### Copyright

© The Author(s) 2023.

## REFERENCES

1. Cerotti D, Distefano S, Merlino G, Puliafito A. A crowd-cooperative approach for intelligent transportation systems. *IEEE Trans Intell Transp Syst* 2017;18:1529-39. [DOI](#)
2. Zhao Y, Zheng Z, Liu Y. Survey on computational-intelligence-based UAV path planning. *Knowl Based Syst* 2018;158:54-64. [DOI](#)
3. Duchoň F, Babinec A, Kajan M, et al. Path planning with modified star algorithm for a mobile robot. *Procedia Eng* 2014;96:59-69. [DOI](#)
4. Zhang T, Zhu Y, Song J. Real-time motion planning for mobile robots by means of artificial potential field method in unknown environment. *Ind Rob* 2010;37:384-400. [DOI](#)
5. Wang H, Yu Y, Yuan Q. Application of Dijkstra algorithm in robot path-planning. In: 2011 Second International Conference on Mechanic Automation and Control Engineering; 2011 Jul 15-17; Hohhot. IEEE; 2011. pp. 1067-97. [DOI](#)
6. Wu PPY, Campbell D, Merz T. Multi-objective four-dimensional vehicle motion planning in large dynamic environments. *IEEE Trans Syst Man Cybern B* 2011;41:621-34. [DOI](#)
7. Shorakaei H, Vahdani M, Imani B, Gholami A. Optimal cooperative path planning of unmanned aerial vehicles by a parallel genetic algorithm. *Robotica* 2016;34:823-36. [DOI](#)
8. Roberge V, Tarbouchi M, Labonte G. Comparison of parallel genetic algorithm and particle swarm optimization for real-time UAV path planning. *IEEE Trans Industr Inform* 2013;9:132-41. [DOI](#)
9. Abeywickrama HV, Jayawickrama BA, He Y, Dutkiewicz E. Potential field based inter-UAV collision avoidance using virtual target relocation. In: 2018 IEEE 87th Vehicular Technology Conference (VTC Spring); 2018 Jun 03-06; Porto, Portugal. IEEE; 2018. p. 1-5. [DOI](#)
10. Vanegas G, Samaniego F, Girbes V, Armesto L, Garcia-Nieto S. Smooth 3D path planning for non-holonomic UAVs. In: 2018 7th International Conference on Systems and Control (ICSC); 2018 Oct 24-26; Valencia, Spain. IEEE; 2018. p. 1-6. [DOI](#)
11. Jain G, Yadav G, Prakash D, Shukla A, Tiwari R. MVO-based path planning scheme with coordination of UAVs in 3-D environment. *J Comput Sci* 2019;37:101060. [DOI](#)
12. Wang L, Cai R, Lin M, Zhong Y. Enhanced list-based simulated annealing algorithm for large-scale traveling salesman problem. *IEEE Access* 2019;7:144366-80. [DOI](#)
13. Li G, Li J. An improved tabu search algorithm for the stochastic vehicle routing problem with soft time windows. *IEEE Access* 2020;8:158115-24. [DOI](#)

14. Kala R, Shukla A, Tiwari R. Fusion of probabilistic A\* algorithm and fuzzy inference system for robotic path planning. *Artif Intell Rev* 2010;33:307-27. [DOI](#)
15. Dorigo M, Maniezzo V, Colomi A. Positive feedback as a search strategy. *Tech Rep* 1991;91-016. Available from: <https://api.semanticscholar.org/CorpusID:16027138>. [Last accessed on 15 Dec 2023]
16. Dorigo M, Maniezzo V, Colomi A. Ant system: optimization by a colony of cooperating agents. *IEEE Trans Syst Man Cybern B* 1996;26:29-41. [DOI](#)
17. Sttzle T, Hoos H. Improving the ant system: a detailed report on the MAX-MIN ant system. Available from: <https://api.semanticscholar.org/CorpusID:14922469>. [Last accessed on 15 Dec 2023]
18. Dorigo M, Gambardella LM. Ant colony system: a cooperative learning approach to the traveling salesman problem. *IEEE Trans Evol Comput* 1997;1:53-66. [DOI](#)
19. Li P, Wang H, Li X. Improved ant colony algorithm for global path planning. *AIP Conf Proc* 2017;1820:080013. [DOI](#)
20. Huang M, Ding P, Huan JX. Global path planning for mobile robot based on improved ant colony algorithms. *Appl Mech Mater* 2013;418:15-9. [DOI](#)
21. Luo Q, Wang H, Zheng Y, He J. Research on path planning of mobile robot based on improved ant colony algorithm. *Neural Comput Appl* 2020;32:1555-66. [DOI](#)
22. Chen Y, Wu J, He C, Zhang S. Intelligent warehouse robot path planning based on improved ant colony algorithm. *IEEE Access* 2023;11:12360-7. [DOI](#)
23. Yi N, Xu J, Yan L, Huang L. Task optimization and scheduling of distributed cyber-physical system based on improved ant colony algorithm. *Future Gener Comput Syst* 2020;109:134-48. [DOI](#)
24. Ning J, Zhang Q, Zhang C, Zhang B. A best-path-updating information-guided ant colony optimization algorithm. *Inf Sci* 2018;433:142-62. [DOI](#)
25. Wang Y, Chen J, Ning W, et al. A time-sensitive network scheduling algorithm based on improved ant colony optimization. *Ale Eng J* 2021;60:107-14. [DOI](#)
26. Miao C, Chen G, Yan C, Wu Y. Path planning optimization of indoor mobile robot based on adaptive ant colony algorithm. *Comput Ind Eng* 2021;156:107230. [DOI](#)
27. Lyridis DV. An improved ant colony optimization algorithm for unmanned surface vehicle local path planning with multi-modality constraints. *Ocean Eng* 2021;241:109890. [DOI](#)
28. Hou W, Xiong Z, Wang C, Chen H. Enhanced ant colony algorithm with communication mechanism for mobile robot path planning. *Robot Auton Syst* 2022;148:103949. [DOI](#)