Advances in Differential Equations and Control Processes 2026, 33(3), 4530.
doi: https://doi.org/10.59400/adecp4530

Article

Enhanced particle swarm optimization with adaptive weights and Gaussian mutation based 3D path planning for unmanned aerial vehicles

Gaofeng Che

School of Artificial Intelligence and Computer Science, Liaocheng University, Liaocheng 252000, China; chexuping@163.com

CITATION

Che G. Enhanced particle swarm optimization with adaptive weights and Gaussian mutation based 3D path planning for unmanned aerial vehicles. Advances in Differential Equations and Control Processes. 2026; 33(3): 4530. https://doi.org/10.59400/adecp4530

ARTICLE INFO

Received: 18 June 2026
Revised: 21 July 2026
Accepted: 27 July 2026
Available online: 14 August 2026

Abstract: Path planning remains a pivotal metric for evaluating unmanned aerial vehicles (UAVs) autonomy, where intelligent algorithms serve as the cornerstone for generating high-quality trajectories. Path planning is an indispensable part of the autonomous control system for UAVs. In addition, UAV mission planning has been proven to be an NP-hard problem. This paper systematically analyzes the state-of-the-art intelligent optimization algorithms for UAVs' path planning. Considering that traditional particle swarm optimization (PSO) and adaptive weight particle swarm optimization (AWPSO) easily converge to local optima, this work develops an enhanced PSO with adaptive weight and Gaussian mutation (EPSO-AWGM). The introduction of adaptive weight factors and Gaussian mutation operators helps the algorithm escape local optimum solutions. Moreover, the cubic spline method is utilized to smooth the UAVs' flight paths. To verify the performance of the proposed EPSO-AWGM algorithm and the quality of UAV path planning in a 3D environment, this paper sets the traditional PSO and AWPSO as control groups, conducting comparative simulation experiments on 3D terrain path planning. The simulations uniformly set initial parameters, with a particle population size of 20 and a maximum iteration count of 50. All algorithms operate in a 3D environment containing five mountain obstacles, with a unified starting coordinate at (1,1,1) and an ending coordinate at (29,29,17). To ensure objective experimental results and eliminate random error interference, all three algorithms are independently run 20 times. Simulation results show that the proposed EPSO-AWGM can obtain shorter and higher-quality flight paths. It therefore has great application potential in practical UAV path planning scenarios.

Keywords: path planning; unmanned aerial vehicles (UAVs); particle swarm optimization (PSO)

1. Introduction

With the rapid advancement of artificial intelligence, unmanned aerial vehicles (UAVs) are increasingly deployed in civil fields such as environmental mapping, remote sensing, and logistics. However, the complexity of urban and rural operational environments necessitates superior autonomous planning and control capabilities. Path planning remains a pivotal metric for evaluating UAV autonomy, where intelligent algorithms serve as the cornerstone for generating high-quality trajectories.

The path planning algorithm is the core component of the unmanned aerial vehicle path planning system, and its fundamental task is to search for the optimal or approximately optimal flight path from the starting point to the target point for the unmanned aerial vehicle in complex terrain space. In the past few decades, with the continuous expansion of drone application scenarios, more and more path planning methods have emerged to meet the increasingly complex planning needs. These methods have their own characteristics, but also have their own limitations. The current related research can be mainly divided into two categories: precise algorithms and intelligent biomimetic algorithms.

In terms of precise algorithms, such as Dijkstra's algorithm [1, 2], A* algorithm [3–6], simulated annealing algorithm [7], and artificial potential field method [8] are widely used due to their clear principles, relatively simple implementation, and controllable solution quality. However, such methods generally suffer from low computational efficiency and are mostly limited to path search in two-dimensional (2D) plane space. In practical applications, drones must strictly adhere to their kinematic and dynamic constraints during flight, including maximum turning angle, maximum climb/dive rate, minimum turning radius, and speed limits, which are often not fully considered in the traditional methods mentioned above. In addition, when faced with dynamic obstacle avoidance requirements in complex three-dimensional (3D) terrain, such precise algorithms often require a significant amount of computation time and storage space, making it difficult to meet real-time requirements. For example, although the artificial potential field method can quickly generate smooth paths in simple environments, when the target point is located near obstacles or there are symmetrical dead points in the potential field distribution, the combined forces of attraction and repulsion are easily cancelled out, causing the drone to fall into local minima and unable to reach the target point, and even in extreme cases unable to obtain a feasible path. Therefore, traditional precise algorithms often struggle to meet the timeliness requirements of engineering applications when dealing with high-dimensional, dynamic, and multi-constrained real-world drone task scenarios.

In 3D path planning, the requirement to avoid terrain collisions introduces significantly higher computational complexity compared to its two-dimensional counterpart. Consequently, enhancing the convergence speed and solution quality of planners in complex 3D environments remains a primary research impetus [9]. Furthermore, the increasing complexity of planning tasks and environmental constraints exacerbates the challenges faced by intelligent optimization algorithms in addressing high-dimensional problems.

Compared with precise algorithms, intelligent biomimetic algorithms exhibit superior flexibility in complex environments. This type of algorithm simulates the collective behavior patterns of natural biological populations, guiding the evolution of search directions through collaboration and information sharing among individuals. Typical representatives include the wolf pack algorithm [10, 11], particle swarm optimization algorithm (PSO) [12], and ant colony optimization algorithm (ACO) [13]. These algorithms have significant advantages such as relatively simple parameter settings, high search efficiency, strong environmental adaptability, and low engineering implementation threshold, which have attracted increasing academic attention in the field of drone path planning. Specifically, the wolf pack algorithm establishes an effective balance mechanism between global exploration and local exploitation by simulating the hunting and wandering behavior of wolf packs; The particle swarm optimization algorithm utilizes information transfer and velocity update rules between particles to quickly converge to high-quality solutions with fewer iterations; The ant colony optimization algorithm utilizes the accumulation and volatilization mechanism of pheromones to gradually select the optimal path in a discrete search space. It is precisely because of these unique advantages that intelligent biomimetic algorithms can often provide more practical and efficient solutions than traditional precise algorithms when dealing with high-dimensional, nonlinear, and multi-constraint unmanned aerial vehicle 3D path planning problems.

A hybrid algorithm has been proposed to address the task allocation and path planning problems of UAV swarms at sea, which embeds the core principles of the genetic algorithm and crossover mutation operator into the PSO framework [14]. This hybrid method utilizes the solution space recombination ability provided by crossover operations in genetic algorithms and the population diversity disturbance introduced by mutation operations, effectively compensating for the structural defects of traditional PSO algorithms such as population diversity attenuation and susceptibility to local optima in the later stages of search. The simulation experiment results confirm that compared with traditional algorithms, this method achieves significant improvement in task resource allocation efficiency and can generate better quality drone flight paths. On the other hand, to address the path planning challenges of drone swarms in complex environments, researchers have designed a distributed wolf pack algorithm that utilizes the dynamic transformation characteristics of individual roles in the pack, such as the functional switching mechanism between reconnaissance wolves, fierce attack wolves, and besieging wolves, to optimize task allocation processes and improve overall search efficiency. However, this method incurs high computational overhead when performing coverage search tasks and ignores key constraints such as obstacle avoidance in practical applications [15]. In addition, similar to the limitations of existing literature, the above methods are still limited to 2D environments, while the operational space of unmanned aerial vehicles in the real world is essentially three-dimensional, which not only includes terrain undulations and three-dimensional obstacles, but also involves complex factors such as energy consumption differences in the height dimension and meteorological stratification. This makes it difficult for the 2D planning results to be directly applicable in practical deployment.

In order to improve the performance of ACO in UAV applications, researchers have introduced a coefficient matrix and transition probability matrix to improve convergence [16]. Building on this, researchers have refined the pheromone update mechanism within UAVs task assignment model, enabling simultaneous consideration of task prioritization alongside path planning [17]. For three-dimensional multi-obstacle environments, the bat algorithm has been enhanced by integrating the local search capabilities of the bee colony algorithm, demonstrating a more efficient acquisition of optimal flight paths compared to standard single algorithms [18].

Addressing the tendency of traditional algorithms to fall into local optima, scholars have modified the sparrow search algorithm for 3D environments by introducing adaptive weight factors to accelerate convergence and employing a Cauchy-Gaussian strategy to overcome stagnation and bolster exploration capabilities [19]. Similarly, the grey wolf optimizer has been initialized using A* algorithms to generate the lead wolf, effectively reducing blind searches, while mutation operators were utilized to accelerate path optimization [20]. In dynamic scenarios, the firefly algorithm has been employed to navigate moving obstacle constraints, successfully generating collision-free safe paths [21].

Further improvements include the adoption of Cauchy mutation strategies and adaptive factors to prevent premature convergence in gravity algorithms [22]. For 3D static environments, novel modeling methods have been proposed to reduce computational complexity, integrating vibration functions into particle swarm optimization to enhance collision avoidance and overall performance [23]. Additionally, a divide-and-conquer strategy has been implemented to decompose high-dimensional path planning problems into manageable sub-problems, compensating for the deficiencies of standard particle swarm optimization in complex environments and improving global search ability through hybridization with the A* algorithm [24]. Finally, the wolf pack algorithm has seen performance gains through the design of an adaptive convergence factor based on wolf centrifugal distance, optimizing both convergence speed and accuracy [25].

Many improved PSO variants combining adaptive weight and Gaussian mutation have been proposed in recent years to solve premature convergence, as listed in **Table 1**.

Table 1. Comparison between EPSO-AWGM and closely related hybrid PSO variants.

Algorithms	Adaptive weight logic	Mutation trigger	Mutated object	Elite retention	Collision check after smoothing
PSOGM [12]	Fixed linear decay	Fixed probability	All particles	Unconditional retain	No smoothing module
AWPSO [14]	Iteration-independent linear	No Mutation	___	___	No smoothing module
AWPSO-CauGM [24]	Fixed linear decay	Fixed probability	All particles	Unconditional retain	No secondary collision check
IAWPSO [23]	Fixed linear decay	Fixed probability	Random particles	Unconditional retain	No secondary collision check
SSA-AW [19]	Adaptive weight without mutation	Cauchy mutation	All sparrows	Unconditional retain	No smoothing module
EPSO-AWGM	Staged decay coupled with iteration progress	Triggered by population aggregation degree	50% elite P_{best} particles	Retain only if fitness improves	Spline smoothing + dense point collision recheck

From **Table 1**, existing hybrid PSO variants have three common defects:

(1) Inertia weight follows fixed linear rules without matching the iteration-stage search demand.

(2) Mutation acts on all particles unconditionally, which may destroy high-quality paths.

(3) Trajectory smoothing and post-collision detection are ignored, leading to unfeasible smooth paths. To fill these gaps, this paper proposes EPSO-AWGM with three coupled improvement mechanisms.

Sampling-based planning methods are one of the mainstream paradigms for solving motion planning problems in high-dimensional spaces, with the Rapidly-exploring Random Tree (RRT) being the most representative. The core idea of RRT is to perform random sampling in the configuration space and iteratively expand the search tree by connecting sampled points to the nearest existing nodes in the tree, starting from the initial state as the root node [26]. This sample-based incremental construction approach enables RRT to perform exceptionally well in high-dimensional spaces or scenarios with highly complex geometric structures, effectively avoiding the computational infeasibility caused by the “curse of dimensionality” in grid-based discretization methods. However, the inherent randomness of the RRT algorithm makes it a probabilistically complete but suboptimal planning method within a finite number of iterations, and it typically returns a feasible path rather than an optimal one. Extensive research on sampling-based planning indicates that the quality of paths generated by RRT is jointly constrained by several critical parameters: the maximum number of iterations determines the upper limit of search resources the algorithm can invest. And the step size parameter creates a core trade-off between search accuracy and time cost, sets the step size too small results in slow tree expansion, and significantly increases search time or even makes it difficult to complete planning within the specified time, while setting the step size too large may cause collisions by crossing obstacle boundaries or generate overly rough, frequently turning, and non-smooth paths in obstacle-free regions, thereby affecting the execution performance of subsequent trajectory tracking [27].

Based on the analysis of the above literature, it can be concluded that classical intelligent biomimetic optimization algorithms commonly face common challenges in unmanned aerial vehicle path planning applications, such as slow convergence speed and susceptibility to local optima. Although these algorithms have strong global exploration potential due to their group collaboration mechanism, there is currently no universal algorithm that can achieve ideal planning results in all scenarios due to significant differences in spatial structure, threat distribution, and computational constraints in drone flight environments. Among numerous biomimetic algorithms, the Particle Swarm Optimization (PSO) algorithm has attracted much attention due to its advantages of simple structure, few parameters, easy implementation, and high single-iteration computational efficiency, and occupies an important position in practical engineering applications. However, the PSO algorithm commonly suffers from the problem of rapid decline in population diversity in the later stages of search, leading to premature aggregation of particles in a local area and loss of the ability to escape local optima. This structural flaw seriously restricts its performance in complex 3D path planning tasks. In view of this, this article focuses on the path planning problem of unmanned aerial vehicles in threat environments. Firstly, a refined specific threat environment model is constructed, which comprehensively considers multi-source threat factors such as terrain obstacles, radar detection, and air defense firepower. Based on this, the PSO algorithm is specifically improved to effectively suppress premature convergence, improve the global optimization ability and search stability of the algorithm, and ultimately generate an optimal flight path for unmanned aerial vehicles with the lowest comprehensive cost function and the highest safety under the premise of satisfying collision-free constraints.

The main contributions are as follows.

(1) Staged iteration-aware adaptive inertia weight mechanism

Traditional adaptive weight particle swarm optimization (AWPSO) relies on fixed linear inertia weight decay throughout the optimization process, which cannot adapt to the constantly changing search requirements during the iteration stage. In contrast, our staged adaptive weights decay monotonically with the number of iterations: high weight values are deployed in the early stages to expand the global search range on mountainous terrain, while low weight values are used in the later stages to refine local obstacle removal. This design achieves a dynamic balance between global exploration and local mining, eliminating the limitations of static constant weight configuration.

(2) Aggregation-aware elite Gaussian mutation strategy

In order to quantitatively characterize the convergence state of particle swarm optimization, the population aggregation degree S is calculated at each iteration as a feedback signal, and the mutation probability and variance of Gaussian perturbation are dynamically adjusted. We did not randomly perturb all particles, but only selected 50% of the elite pbest individuals for Gaussian random migration. The global optimal solution is only updated when the abrupt trajectory has a lower total flight cost. This tailored mutation scheme avoids disrupting high-quality candidate paths and preserves sufficient population diversity, enabling the population to overcome local optimal valleys formed by multi-peak mountain obstacles.

(3) Cubic spline trajectory post-processing module

The original optimized output of PSO is a sequence of discrete waypoints connected by dashed lines, which have discontinuous curvatures and violate the kinematic limitations of unmanned aerial vehicles. To address this issue, cubic spline interpolation was introduced to generate a global second-order continuous smooth trajectory that conforms to the constraints of drone velocity and acceleration. After curve fitting, dense equidistant sampling is performed for secondary collision detection. If any sampling point falls within the danger zone of terrain obstacles, the smooth curve is discarded and the original discrete waypoint path is restored, ultimately ensuring the geometric safety and feasibility of the output flight trajectory.

The remainder of this paper is organized as follows. Section 2 presents the 3D environment model and cost and objective functions of UAVs. Section 3 describes the proposed path planning strategy. Section 4 discusses the simulation results. Finally, Section 5 concludes the paper.

2. Problem description

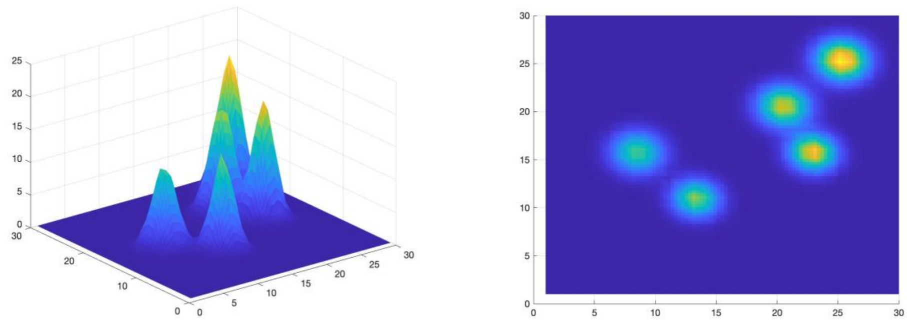
2.1. Environment model

To enhance fidelity to real-world flight conditions and accurately evaluate path quality, conventional 2D models prove inadequate. Consequently, this paper models mountainous terrain as obstacles to construct a 3D environment specific to UAV operations in forest regions. The following is a map of the flight environment in the simulated mountain. First, assuming that the height of the horizontal plane is used to construct a 3D space map, the horizontal coordinates of each mountain obstacle and UAV can be described by (x, y), so the mountain threat model can be described as follows:

In order to make the path planning results closer to real flight conditions and achieve accurate evaluation of path quality, traditional 2D plane models can no longer meet practical needs. In the real forest area drone operation scenario, factors such as terrain undulations, mountain obstacles, and vegetation coverage constitute complex three-dimensional spatial constraints. Relying solely on 2D projection will lose key vertical dimension information, resulting in a lack of feasibility guarantee for the planned path in the height direction. Therefore, this article abstractly models mountainous terrain as three-dimensional obstacles and constructs a refined 3D simulation environment specifically designed for unmanned aerial vehicle operations in forest areas. The 3D environment map is based on the horizontal plane as the reference plane, dividing the entire flight space into continuous voxel grids, where the spatial positions of each mountain obstacle and drone can be accurately described by 3D coordinates (x, y, z). Specifically, on the horizontal projection plane, the position of each mountain obstacle is determined by its plane coordinates (x, y), while its vertical threat range is characterized by the altitude distribution function z = f (x, y) of the mountain. This function reflects the undulating shape of the terrain surface, and any area where the drone flies below this curvature is considered an impassable collision threat zone. Based on this modeling approach, drones not only need to avoid horizontal mountain contours during flight, but also must maintain sufficient safe clearance height in the vertical dimension to ensure that the planned path truly has floatability and safety in three-dimensional space. The following text will provide a three-dimensional spatial map and corresponding mathematical model description of the simulated mountain flight environment, so the mountain threat model can be described as follows:

$$z_m = \sum_{j=0}^{M} h_j \exp\left[-\left(\frac{x - x_j}{\sigma_x}\right)^2 - \left(\frac{y - y_j}{\sigma_y}\right)^2\right] \tag{1}$$

where h_0 is the height of the horizontal plane with an altitude of 0, h_j (j = 1, 2, ..., M) represents the height of the mountain j, (x_j, y_j) represents the center coordinates of the mountain, and σ_x and σ_y denote the standard deviations in the x-axis and y-axis directions, respectively. The actual mountain is an irregular geometric structure, so it refers to the slope of the side of the mountain, which is used to calculate the slope of the mountain, as shown in **Figure 1**. The environmental parameters are shown in **Table 2**.



(a) 3D visualization of mountain **(b)** 2D horizontal projection of mountain

Figure 1. 3D environment model.

Table 2. Environment parameter settings.

Obstacle numbers	Obstacle point
1	(8, 15, 12)
2	(13, 11, 15)
3	(23, 16, 18)

Table 2. *Cont.*

Obstacle numbers	Obstacle point
4	(21, 20, 15)
5	(26, 25, 19)

2.2. Cost and objective function

Evaluating path quality solely based on collision avoidance is insufficient for comprehensive performance assessment. Therefore, to quantitatively measure path optimality and verify the effectiveness of the proposed algorithm, a multi-objective cost function tailored for mountainous and forested environments is established. The overall flight cost is categorized into internal and external components. External cost arises from potential collisions between the UAVs and terrain obstacles, imposing severe penalties for unsafe trajectories. Internal cost primarily reflects fuel consumption, which is directly proportional to the flight distance. Consequently, the total cost objective function integrates both safety constraints and energy efficiency, and is formulated as follows:

$$\min \mathrm{C} = \sum_{\mathrm{i}=1}^{\mathrm{H}+1} \left(\delta \mathrm{C_p} + (1-\delta)\mathrm{C_c}\right) \tag{2}$$

where H represents the number of discrete waypoints, if the path contains a starting point and an ending point, usually the number of nodes is H + 1. C_p and C_c represent path cost and collision cost respectively. δ is the weight coefficient.

The position of UAVi is represented as (x_i, y_i, z_i) and i = 1, 2, ..., N. The path cost and the collision cost are given as follows:

$$\mathrm{C_p} = \sum_{\mathrm{i}=1}^{\mathrm{H}+1} \sqrt{(\mathrm{x_i} - \mathrm{x_{H-1}})^2 + (\mathrm{y_i} - \mathrm{y_{H-1}})^2 + (\mathrm{z_i} - \mathrm{z_{H-1}})^2} \tag{3}$$

$$\mathrm{C_c} = \begin{cases} \infty, & (\mathrm{z_i} < \mathrm{h_j}) \cup (\mathrm{H_{ij}} \leq \mathrm{r_j}) \\ 0, & (\mathrm{z_i} \geq \mathrm{h_j}) \cap (\mathrm{H_{ij}} \neq \mathrm{r_j}) \end{cases} \tag{4}$$

$$\mathrm{H_{ij}} = \mathrm{r_i} + \mathrm{r_j}, \mathrm{i} \in (1, 2, ..., \mathrm{N}), \quad \mathrm{j} \in (1, 2, ..., \mathrm{M}) \tag{5}$$

In Equation (3), the Euclidean distance is used as the path cost, and Equation (4) represents the collision cost. z_i represents the flight height of the UAVi. As shown in **Figure 2**, r_i and r_j represent the flight safety radius of the UAVi and the radius of the danger source, respectively, and H_{ij} is the distance from the UAVi to obstacle j.

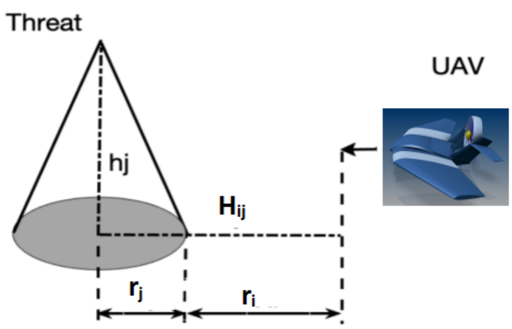


Figure 2. Threat distance between UAV and mountain.

3. Algorithm design

3.1. Basic PSO algorithm

Compared with numerous high-precision and heuristic optimization algorithms, PSO offers distinct advantages, including rapid convergence and minimal parameter tuning, making it highly valuable for path planning applications. The algorithm is inspired by the social behavior of bird flocks foraging in nature. In this analogy, each bird is abstracted as a particle, while the food source represents the global optimum. Initially, a population of particles with random positions and velocities is generated. During the iterative process, each particle updates its state based on fitness evaluations, tracking both its personal best position and the global best position. Through continuous interaction and information exchange among particles, the swarm converges toward the global optimal solution. The mathematical equations governing the velocity and position updates are presented as follows:

$$v_i^{t+1} = \lambda v_i^t + \alpha_1\eta_1\left(pbest_i - x_i^t\right) + \alpha_2\eta_2\left(Gbest_i - x_i^t\right) \tag{6}$$

$$x_i^{t+1} = x_i^t + v_i^{t+1} \tag{7}$$

where v is the velocity, λ is the inertia weight of the particle imitating the flight of birds, and α_1 and α_2 are the acceleration constants of the particle, and η_1 and η_2 are random values, which refer to the position vector of the (t + 1)-generation particle i.

$$\lambda = \frac{2}{\left|2-\theta-\sqrt{\theta^2-4\theta}\right|} \tag{8}$$

$$\theta = \alpha_1 + \alpha_2 \tag{9}$$

3.2. EPSO-AWGM algorithm

In addition, in order to improve the diversity of the algorithm, a Gaussian mutation strategy with a certain probability is inserted in the iterative process. When a certain conditional probability P occurs, 1/2 of the current local optimal particles are randomly selected to undergo the mutation operation. The particles that are better are replaced. This can make the algorithm jump out of the local optimum.

3.2.1. Staged adaptive inertia weight

Different from the fixed linear weight in AWPSO [14], the weight decays linearly with iteration progress:

$$w(t) = w_{max} - \frac{t}{T_{max}}\left(w_{max} - w_{min}\right),\quad w_{max} = 0.9 \text{ and } w_{min} = 0.4 \tag{10}$$

Early iterations adopt large weight for wide-range mountain global search; late iterations adopt small weight for precise vertical obstacle clearance adjustment.

3.2.2. Aggregation-aware adaptive elite Gaussian mutation

Step 1: Calculate population aggregation degree S each iteration to reflect particle clustering level:

$$S = \frac{1}{D}\sum_{d=1}^{D} \mathrm{std}(x_{1d}, x_{2d}, \dots, x_{Npd})/30 \tag{11}$$

where D is the total path dimension, std is the standard deviation of particle coordinate, $S \in [0, 1]$ and a smaller value means particles converge more severely.

Step 2: Dynamic mutation probability and Gaussian variance adjusted by S:

$$P = 0.3 \times (1 - S), \quad \sigma = 1.2/(S + 0.2) \tag{12}$$

When $S < 0.15$, mutation probability and perturbation variance are amplified to escape local optima.

Step 3: Elite particle Gaussian perturbation formula

$$\widetilde{X}_i = p_{best,i} * (1 + \mathrm{Gaussion}(\mu, \sigma^2), \text{ if } P \geq \mathrm{rand}(0, 1) \tag{13}$$

Step 4: Global optimum update rule

$$G_{best} = \begin{cases} p_{best,i}, & C(p_{best,i}) < C(\widetilde{X}_i)\ , i \in (1, 2, ...N) \\ \widetilde{X}_i, & C(p_{best,i}) \geq C(\widetilde{X}_i)\ , i \in (1, 2, ...N) \end{cases} \tag{14}$$

Remark 1. *After generating mutated particle $\widetilde{X_i}$, compare the total cost of original $p_{best,i}$ and mutated trajectory. Only update individual and global optimum if mutated path has lower cost, avoiding performance degradation caused by invalid random disturbance. We select 50% elite p_{best} particles for mutation, and simulation results verify this ratio achieves optimal balance between diversity and stability.*

3.2.3. Cubic spline smoothing

Among them, it represents the position of the mutated particle that has undergone Gaussian mutation. If the probability P satisfies the condition, the Gaussian mutation operation is performed, so that the particle can jump out of the local optimum. Finally, in order to make the drone path smoother, cubic spline interpolation is used between the start and end points. It makes the path smoother without mutation, which is in line with the kinematics of the drone. The effect before and after smoothing is shown in **Figure 3**.

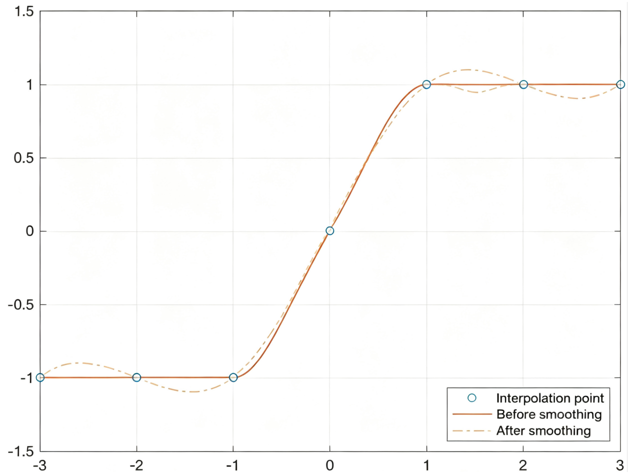


Figure 3. The effect of cubic spline interpolation.

3.3. Algorithmic steps of the EPSO-AWGM

The algorithm flow of using the Gaussian mutation compression factor particle swarm algorithm to solve the UAV three-dimensional path is as follows (see **Algorithm 1**).

Algorithm 1 EPSO-AWGM for 3D UAV path planning

1. Initialize 3D mountain environment, start (1,1,1), target (29,29,17), all hyperparameters.
2. Randomly initialize particle position & velocity within [0,30] space, execute boundary clipping.
3. Calculate initial cost C for all particles, initialize $p_{best,i}$ and global $G_{best.}$
4. For t = 1 to T_{max}:
   a. Compute staged adaptive inertia weight w(t) via Equation (10).
   b. Calculate population aggregation degree S via Equation (11).
   c. Update particle velocity & position via contraction-factor PSO Equations (6) and (7), clip out-of-bound coordinates.
   d. Compute current flight cost of each particle, update $p_{best,i}$ if current cost is lower.
   e. Calculate dynamic mutation P,σ via Equation (12).
   f. Randomly select 50% elite p_{best} particles to execute Gaussian mutation Equation (13).
   g. Compare mutated trajectory cost, update global G_{best} via Equation (14).
5. After iteration termination: extract discrete optimal waypoints from $G_{best.}$
6. Execute cubic spline interpolation to generate a smooth continuous trajectory.
7. Dense sampling collision detection on smooth curve; roll back discrete path if collision exists.
8. Output final collision-free smooth UAV flight trajectory.

3.4. Theoretical analysis of EPSO-AWGM

3.4.1. Convergence stability analysis

The staged weight w(t) monotonically decreases within [0.4,0.9], satisfying the contraction-factor PSO convergence sufficient condition. All particle coordinates are constrained in [0,30] via boundary clipping, so the particle search space is bounded without divergence during iterations.

3.4.2. Theoretical properties of adaptive Gaussian mutation

Gaussian distribution has high probability density near zero for local fine-tuning of UAVs' waypoints, and low-probability large jumps to escape local optimal mountain basins. The elite fitness retention rule ensures the global optimal cost sequence is monotonically non-increasing, guaranteeing algorithm optimization monotonicity. Dynamic mutation intensity controlled by aggregation degree avoids over-disturbance of dispersed elite populations.

3.4.3. Big-O computational complexity derivation

(1) PSO/AWPSO: $\mathrm{O(Np} * \mathrm{T_{max}} * \mathrm{D)}$

(2) EPSO-AWGM: $\mathrm{O(Np} * \mathrm{T_{max}} * \mathrm{D} + 0.5 * \mathrm{Np} * \mathrm{D} + \mathrm{N_{node}^3})$

Remark 2. *Extra overhead comes from elite Gaussian mutation and cubic spline interpolation. The cubic spline complexity N_{node}^3 is negligible under waypoint number D = 30, which will not affect offline UAVs pre-mission planning efficiency.*

3.4.4. UAVs real-time performance trade-off analysis

For offline pre-mission forest reconnaissance planning, the marginal extra computation time of EPSO-AWGM is fully acceptable. For online dynamic obstacle avoidance tasks, future work will optimize lightweight mutation and reduce path node quantity to lower time complexity and meet real-time response requirements.

4. Experimental simulation

4.1. Simulation parameters

In order to verify EPSO-AWGM performance, we set the 3D mountain benchmark environment: 5 multi-peak mountain obstacles, start (1,1,1), target (29,29,17). Five comparative algorithms are selected: PSO, AWPSO, SSA-AW [19], A*-GWO [20], EPSO-AWGM. Unified hyperparameters for all algorithms: Np = 20, T_{max} = 50. Each algorithm runs independently 20 repeated trials in the environment to eliminate random error interference. The simulation indicators include average computation time, optimal/average fitness, fitness standard deviation, shortest/average path length, and trajectory curvature variance. Core parameter settings are listed in **Table 3**.

Table 3. Algorithm parameter settings.

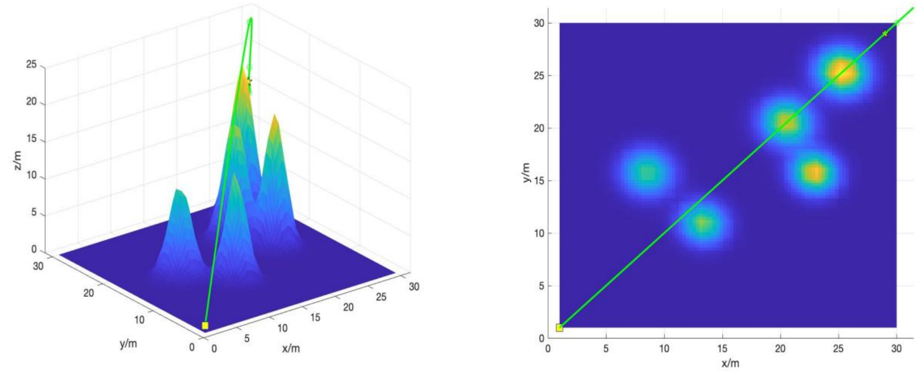
Parameters	Value	Parameter selection basis
Particle population Np	20	Mainstream UAVs PSO planning literature setting
Maximum iteration T_{max}	50	Balances convergence effect and computation cost
Inertia weight bounds w_{max}/w_{min}	0.9/0.4	Grid search pre-experiment optimal group
Acceleration coefficient α_1,α_2	2	Standard contraction-factor PSO setting
Elite mutation ratio	50%	Pre-experiment verified optimal diversity-stability balance
Weight coefficient δ	0.7	Ablation contrast: δ > 0.7 leads to collision paths; δ < 0.7 causes redundant long trajectories
UAV safety radius r_i	0.5 m	Small fixed-wing UAVs actual size parameter
Mountain obstacle base radius r_j	1.2 m	Terrain safety buffer threshold
Start	(1,1,1)	Unified predefined flight origin
Target	(29,29,17)	Unified predefined mission endpoint

4.2. Quantitative simulation results

PSO exhibits the worst performance with maximum average fitness, longest path length and largest fitness standard deviation, proving severe premature convergence in multi-peak terrain. AWPSO, SSA-AW and A*-GWO achieve moderate optimization effects via single adaptive weight or mutation improvement, yet their average path length and trajectory smoothness (curvature variance) are inferior to EPSO-AWGM. EPSO-AWGM obtains the minimum optimal/average fitness, shortest average flight path and far lower curvature variance due to cubic spline smoothing. Although its average computation time is slightly higher than AWPSO, the extra overhead comes from elite Gaussian mutation and spline post-processing modules, which bring significant trajectory quality improvement.

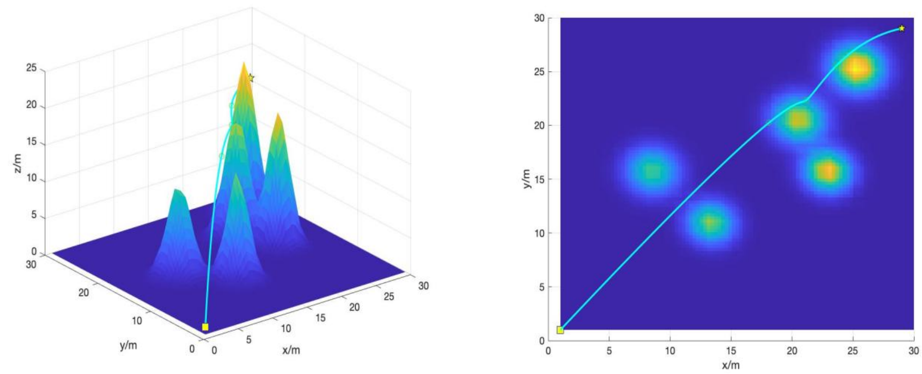
4.3. Path visualization and convergence curve analysis

The 3D spatial trajectory and 2D horizontal projection of PSO, AWPSO and EPSO-AWGM are shown in **Figures 4–6** respectively. PSO outputs an approximate straight-line path passing over the highest mountain peak, generating excessive climbing altitude and redundant flight distance.AWPSO avoids the highest peak but falls into a local optimum near the last mountain obstacle, producing redundant detour segments. EPSO-AWGM searches a smooth low-altitude channel between mountain peaks without redundant climbing or detour, with continuous curvature and no sharp turning points consistent with UAV kinematic constraints.



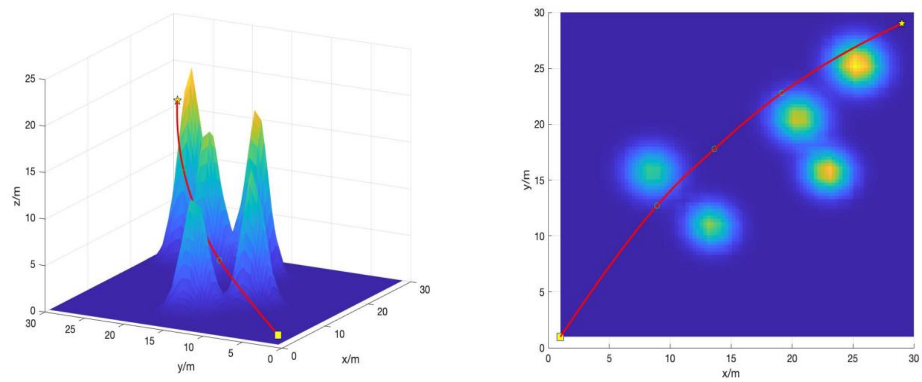
(a) 3D visualization of planned path **(b)** 2D horizontal projection of planned path

Figure 4. Path diagram of the PSO algorithm.



(a) 3D visualization of planned path **(b)** 2D horizontal projection of planned path

Figure 5. Path diagram of AWPSO algorithm.



(a) 3D visualization of planned path **(b)** 2D horizontal projection of planned path

Figure 6. Path diagram of EPSO-AWGM algorithm.

In order to more clearly see the flight path under the three algorithms, **Figures 4–6** show the flight path in the 3D environment and the corresponding 2D top-down view. From the flight paths under the three algorithms, it can be seen that the basic PSO algorithm in **Figure 4** leaps over the mountain. Although there is no mountain collision during the flight, the actual process often increases a lot of fuel consumption, causing unnecessary flight costs. The AWPSO in **Figure 5** significantly reduces the redundant flight path relative to the PSO. However, there is an extra path at the last mountain obstacle, which is stuck in a local optimum. Compared with the above two algorithms, the proposed EPSO-AWGM algorithm in **Figure 6** has better smoothness on the path, and reduces the redundant flight path. The flight space is relatively stable and excellent. The calculation results and iteration diagrams of the above three algorithms are shown in **Table 4** and **Figure 7**, respectively.

Table 4. Calculation results of UAV paths under three algorithms.

Algorithms	Calculation time mean (s)	Fitness optimum	Average fitness	Fitness std.	Path optimal value (m)	Average path length (m)	Curvature variance
PSO	380.41	51.48	63.83	8.72	70.83	70.83	12.67
AWPSO	287.15	42.94	43.83	2.15	44.23	48.11	8.32
SSA-AW	301.68	42.81	43.65	1.98	43.76	47.27	7.95
A*-GWO	326.94	42.76	43.52	1.84	4.341	46.89	7.61
EPSO-AWGM	314.52	42.70	43.40	1.63	42.93	45.58	2.14

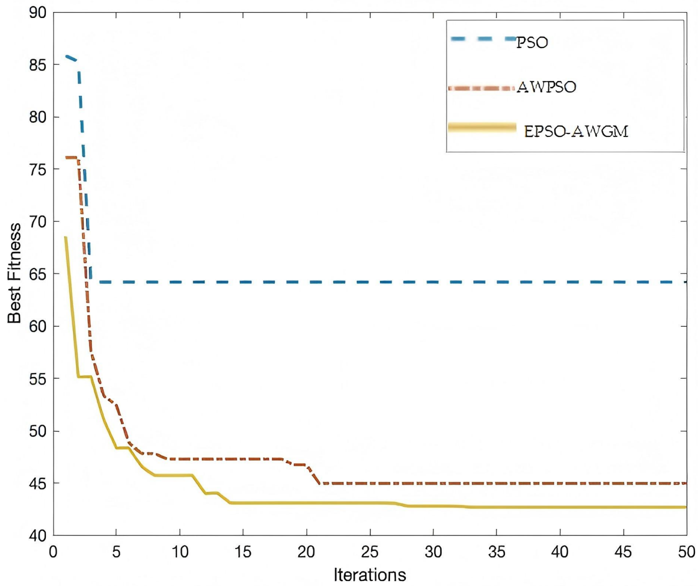


Figure 7. Iterative diagram of three algorithms.

Figure 7 illustrates the fitness iteration curves of the three algorithms over 50 iterations, where fitness is defined as the path distance cost in this simulation. As shown in **Figure 7**, the standard PSO algorithm tends to become trapped in local optima during the later stages of optimization due to its strong dependence on initial parameter settings. In contrast, the AWPSO algorithm demonstrates improved performance, successfully escaping local optima around the fourth iteration. Notably, the proposed EPSO-AWGM algorithm exhibits even more pronounced global search capability. This improvement is primarily attributed to the introduction of a contraction factor, which balances particle optimization behavior throughout the iterative process, and the incorporation of a Gaussian mutation operator in the later stages. These mechanisms collectively enhance the stability and robustness of the search process, enabling the algorithm to escape local optima more effectively. The results further validate the effectiveness of the proposed approach.

5. Conclusion

This paper proposes an improved particle swarm optimization algorithm (EPSO-AWGM) that integrates adaptive inertia weights and a Gaussian mutation mechanism, aiming to solve the inherent defect of the traditional PSO algorithm in premature convergence in 3D UAV path planning. Considering that the optimization performance of PSO algorithm largely depends on the initial setting of control parameters and their adjustment strategies during the iteration process, this paper introduces an adaptive inertia weight mechanism, which enables the inertia weight to dynamically adjust according to the current population's evolutionary state and search progress—assigning larger weight values in the early stage of search to maintain strong global exploration ability, and gradually reducing weights in the later stage of search to promote local fine development, thus achieving an effective dynamic balance between exploration and utilization. In addition, to further enhance the algorithm's ability to escape from local extremum traps, a Gaussian mutation operation with preset probabilities was introduced in the middle and later stages of optimization: when the convergence trend of the population tends to stagnate, a random disturbance that conforms to a Gaussian distribution is applied to the position component of the current optimal particle with a certain probability, generating a mutation solution and comparing it with the original solution, only retaining individuals with better fitness to enter the next generation population. This mechanism effectively enriches population diversity and significantly increases the probability of the algorithm escaping from local optima. At the same time, considering that the path node sequence directly output by the original PSO algorithm often has unnecessary path oscillations and heading mutations, which may affect the actual flight quality of the drone, this paper adopts the cubic spline interpolation method to perform curve fitting and smoothing processing on the generated discrete waypoints, effectively eliminating redundant fluctuations in the trajectory while ensuring no collision constraints, making the final drone flight path more continuous, smooth, and in line with aircraft dynamics constraints. The comparative experimental results in complex terrain environments show that compared with the standard PSO algorithm and adaptive weighted PSO (AWPSO), the EPSO-AWGM proposed in this paper exhibits significant advantages in multiple evaluation indicators such as solution quality, path length, safety, and trajectory smoothness.

Future research can be advanced in three key directions: firstly, we will extend the algorithm to high-dimensional real-world scenarios by combining coupled multi-physics field constraints, including wind interference, flight energy consumption, and communication link limitations, to improve its practicality under real flight conditions. Secondly, we will explore collaborative path planning for multiple drone swarms, achieving coordinated formation flying and reasonable task allocation under the constraints of collision avoidance, task priority, and communication topology among drones. Thirdly, for large-scale multi-objective task scenarios, we will combine the proposed EPSO-AWGM with a distributed computing architecture to further improve computational efficiency and online response speed, meeting the demand for real-time path replanning in dynamic and uncertain environments.

Funding: This research was funded by the Foundation of Henan Educational Committee, grant number 24A120006.

Institutional review board statement: Not applicable.

Informed consent statement: Not applicable.

Data availability statement: No data are associated with the manuscript.

Conflict of interest: The author declares no conflict of interest.

AI use statement: During the preparation of this work, the author used Doubao for language polishing and manuscript refinement. The author subsequently conducted the necessary review and revision of the content and takes full responsibility for the final version of the published article.

References

1. Haam S, Kim H, Yoo M, et al. Optimal additional evacuation route analysis in deep underground station structures using Dijkstra's algorithm. Developments in the Built Environment. 2025; 24: 100784. doi: 10.1016/j.dibe.2025.100784
2. Akay R, Yildirim MY. SBA*: An efficient method for 3D path planning of unmanned vehicles. Mathematics and Computers in Simulation. 2025; 231: 294–317. doi: 10.1016/j.matcom.2024.12.015
3. Zhang B, Cai X, Li G, et al. Path planning in radioactive environment of nuclear facilities based on modified A-star algorithm and search neighborhood optimization. Nuclear Engineering and Technology. 2025; 57(10): 103711. doi: 10.1016/j.net.2025.103711
4. Eltamaly AM, Rabie AH. A star-nosed mole optimizer (SNMO): A novel optimization algorithm. Computers and Electrical Engineering. 2026; 135: 111179. doi: 10.1016/j.compeleceng.2026.111179
5. Yang TY, Zou L, Wu ZX, et al. An improved A-star algorithm coupled with graph division and AIS data for ship path planning. Ocean Engineering. 2025; 330: 121234. doi: 10.1016/j.oceaneng.2025.121234
6. Veisi O, Moradi MA, Gharaei B, et al. Sustainable forestry logistics: Using modified A-star algorithm for efficient timber transportation route optimization. Forest Policy and Economics. 2025; 173: 103456. doi: 10.1016/j.forpol.2025.103456
7. Khalid K, Ayaz Y, Ali S, et al. Fast and efficient motion planning: using hybrid sampling and simulated annealing. Expert Systems with Applications. 2026; 331: 133173. doi: 10.1016/j.eswa.2026.133173
8. Yang Z, Deng L, Mao X, et al. A differential evolutionary algorithm improvement framework based on artificial potential fields. Applied Soft Computing. 2026; 192: 114714. doi: 10.1016/j.asoc.2026.114714
9. Jia Y, Wang S, Liu Z, et al. DPI-RRT*: a fast RRT* algorithm for mobile robot based on dynamic probability adjustment and bi-directional insertion. Engineering Computations. 2026; 43(6): 2365–2396. doi: 10.1108/EC-12-2024-1148
10. Lv L, Wu L, Xiao R, et al. Constrained multi-objective wolf pack algorithm based on diversity guidance and feasibility enhancement. Applied Soft Computing. 2026; 200: 115387. doi: 10.1016/j.asoc.2026.115387
11. Xiao W, Dong N, He K. A hybrid global wolf pack algorithm-based incremental conductance method under partial shading conditions. Solar Energy. 2025; 291: 113388. doi: 10.1016/j.solener.2025.113388
12. Yang X, Liu Y, Wang Y. Target quality variable prediction from multivariate process trajectories using kernel entropy component analysis and an adaptive neuro-fuzzy inference system optimized by improved Gaussian quantum-behaved particle swarm optimization. Expert Systems with Applications. 2026; 327: 132885. doi:

10.1016/j.eswa.2026.132885
13. Che G, Liu L, Yu Z. An improved ant colony optimization algorithm based on particle swarm optimization algorithm for path planning of autonomous underwater vehicle. Journal of Ambient Intelligence and Humanized Computing. 2020; 11(8): 3349–3354. doi: 10.1007/s12652-019-01531-8
14. Yan M, Yuan H, Xu J, et al. Task allocation and route planning of multiple UAVs in a marine environment based on an improved particle swarm optimization algorithm. EURASIP Journal on Advances in Signal Processing. 2021; 2021(1): 94. doi: 10.1186/s13634-021-00804-9
15. Hu J, Wu H, Zhan R, et al. Self-organized search-attack mission planning for UAV swarm based on wolf pack hunting behavior. Journal of Systems Engineering and Electronics. 2021; 32(6): 1463–1476. doi: 10.23919/JSEE.2021.000124
16. Liu W, Zheng X. Three-Dimensional Multi-Mission Planning of UAV Using Improved Ant Colony Optimization Algorithm Based on the Finite-Time Constraints. International Journal of Computational Intelligence Systems. 2020; 14(1): 79. doi: 10.2991/ijcis.d.201021.001
17. Su M, Cheng Y, Hu J, et al. Combined Optimization of Swarm Task Allocation and Path Planning Based on Improved Ant Colony Algorithm. Unmanned Systems Technology. 2021; 4: 40–50. (in Chinese)
18. Zhou X, Gao F, Fang X, et al. Improved Bat Algorithm for UAV Path Planning in Three-Dimensional Space. IEEE Access. 2021; 9: 20100–20116. doi: 10.1109/ACCESS.2021.3054179
19. Liu G, Shu C, Liang Z, et al. A Modified Sparrow Search Algorithm with Application in 3d Route Planning for UAV. Sensors. 2021; 21(4): 1224. doi: 10.3390/s21041224
20. Cao J, Zhang G, Xu P. A* initialized mutable gray wolf optimizer for UAV path planning. Computer Engineering and Applications. 2022; (4): 275–282. (in Chinese)
21. Goel U, Varshney S, Jain A, et al. Three Dimensional Path Planning for UAVs in Dynamic Environment using Glow-worm Swarm Optimization. Procedia Computer Science. 2018; 133: 230–239. doi: 10.1016/j.procs.2018.07.028
22. Xu H, Jiang S, Zhang A. Path Planning for Unmanned Aerial Vehicle Using a Mix-Strategy-Based Gravitational Search Algorithm. IEEE Access. 2021; 9: 57033–57045. doi: 10.1109/ACCESS.2021.3072796
23. Xu Z, Zhang E, Chen Q. Rotary unmanned aerial vehicles path planning in rough terrain based on multi-objective particle swarm optimization. Journal of Systems Engineering and Electronics. 2020; 31(1): 130–141. doi: 10.21629/JSEE.2020.01.14
24. Huang C. A Novel Three-Dimensional Path Planning Method for Fixed-Wing UAV Using Improved Particle Swarm Optimization Algorithm. International Journal of Aerospace Engineering. 2021; 2021(1): 7667173. doi: 10.1155/2021/7667173
25. Zhang W, Zhang S, Wu F, et al. Path Planning of UAV Based on Improved Adaptive Grey Wolf Optimization Algorithm. IEEE Access. 2021; 9: 89400–89411. doi: 10.1109/ACCESS.2021.3090776
26. Mai Z, Xin Z, Zhang X. Research on Robot Path Planning Based on Improved Traditional RRT Algorithm. Machine Building & Automation. 2022; (6): 177–179. (in Chinese)
27. Kang B, Huang J. Variable step size rapidly-exploring random tree (RRT) path planning algorithms and simulation of a mobile robot based on environment complexity. Journal of Beijing University of Chemical Technology (Natural Science). 2023; 50(4): 87–93. doi: 10.13543/j.bhxbzr.2023.04.011 (in Chinese)