

Article

A Wind Field–Perception Hybrid Algorithm for UAV Path Planning in Strong Wind Conditions

Hongping Pu ^{1,2,3,*}, Xinshuai Liu ^{2,3,*}, Shiyong Yang ¹, Chunlan Luo ⁴, Yuanyuan He ⁵, Mingju Chen ^{2,3}  and Xiaoxia Zheng ¹

¹ UAV Industry College, Chengdu Aeronautic Polytechnic, Chengdu 610100, China

² Sichuan Key Laboratory of Intelligent Perception and Control, Sichuan University of Science & Engineering, Yibin 644000, China

³ College of Automation and Information Engineering, Sichuan University of Science & Engineering, Yibin 644000, China

⁴ College of Electronic Information and Engineering, Yibin University, Yibin 644000, China

⁵ College of Art and Design, Pingdingshan University, Pingdingshan 467000, China

* Correspondence: redapplep@cap.edu.cn (H.P.); 324085404431@stu.suse.edu.cn (X.L.)

Abstract

As unmanned aerial vehicles (UAVs) are increasingly utilized in urban inspection and emergency rescue missions, path planning under strong wind conditions persists as a critical challenge. Traditional algorithms frequently exhibit deficiencies in environmental adaptability or encounter difficulties in balancing exploration and exploitation. This paper presents a dynamic-proportion Bat–Cuckoo Search (BA–CS) Hybrid Algorithm enhanced with wind field perception to tackle the challenges of UAV path planning in urban environments with strong winds, specifically addressing the issues of insufficient environmental adaptation and the exploration–exploitation imbalance. The algorithm integrates a dual-feedback mechanism that dynamically modifies the ratio of the BA/CS subpopulations in accordance with real-time iteration progress and population diversity. By incorporating wind field perception into population initialization, interpopulation information exchange, and wind resistance perturbation strategies, it attains efficient path optimization under multiple constraints. Experimental results under strong winds with speeds ranging from 10.8 to 13.8 m/s indicate that the proposed algorithm generates paths that are smooth, continuous, and entirely collision-free. It achieves a superior average wind resistance cost of 0.92, which is 9.8%, 17.1%, and 52.6% lower than those of the A*, RRT, and PSO algorithms, respectively. With a planning time of 3.95 s, it satisfies the path wind resistance stability requirements stipulated in the GB/T 38930-2020 standard, providing an effective solution for UAV inspection and emergency rescue operations in urban wind scenarios.

Keywords: unmanned aerial vehicle (UAV); path planning; strong wind environment; BA–CS hybrid algorithm; wind field perception



Academic Editor: Eugene Levner

Received: 2 December 2025

Revised: 16 January 2026

Accepted: 20 January 2026

Published: 26 January 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

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

1. Introduction

With the rapid development of modern technology, unmanned aerial vehicles (UAVs) have demonstrated significant application potential in disaster emergency rescue, urban inspection, and material delivery, owing to their low cost and high maneuverability. They have emerged as essential equipment for assisting or replacing manual operations in extreme environments [1]. However, as application scenarios extend to severe weather conditions such as strong winds, coupled with urban environments characterized by dense

buildings and intertwined dynamic obstacles, the challenges faced by UAV autonomous path planning have become increasingly prominent: global climate change has led to frequent strong winds, and strong wind disturbances (10.8–13.8 m/s) tend to cause path deviations [2]. This is further compounded by the complex urban environment with 20 static buildings and three dynamic obstacles, requiring UAV path planning to simultaneously meet three core requirements: wind resistance and energy efficiency, obstacle avoidance and safety, and optimal path [3].

To further clarify the complexity of these challenges in strong wind urban scenarios, three key aspects are elaborated as follows: First is the spatiotemporal heterogeneity of wind fields. Urban buildings cause significant shielding and turbulence effects—wind speed attenuates nonlinearly with the distance to buildings (Gaussian attenuation model), and eddies form behind high-rises, leading to random perturbations in wind direction and speed. Such unsteady wind fields easily induce path deviations in UAVs. Second is the coupling of multi-objective constraints. The planning must balance three core requirements: wind resistance and energy efficiency, obstacle avoidance safety, and engineering feasibility. Traditional algorithms often prioritize one aspect while neglecting others—for example, PSO is prone to premature convergence and fails to balance wind resistance and obstacle avoidance, as reflected by its excessive path length of 2670.96 m in comparative experiments. Third is the dynamic adaptation to wind field changes. Existing hybrid algorithms (e.g., CS-GA) mostly adopt fixed subpopulation proportions, which cannot respond to real-time wind speed fluctuations in urban scenarios, resulting in exploration–exploitation imbalance and reduced planning accuracy.

Path planning algorithms can be classified into traditional heuristic algorithms and swarm intelligence optimization algorithms based on their adapted scenarios [4]. Traditional heuristic algorithms (e.g., A* [5,6], RRT [7], DFS [8], BFS [9]) are well-suited for static and known environments; however, their heuristic functions are prone to failure under strong wind conditions, resulting in insufficient wind field adaptability. Swarm intelligence algorithms (e.g., PSO [10], BA [11], CS [12]), which balance exploration and exploitation through group-based optimization, have inherent limitations: BA possesses strong local exploitation capability but lacks sufficient global search, while the converse holds for CS; PSO is susceptible to premature convergence, resulting in a high obstacle collision risk during path planning in strong wind scenarios [13].

In recent years, the development of hybrid algorithms has gradually deepened. For instance, Hybrid Artificial Bee Colony–Cuckoo Search Optimization (ABC-CSO) [14] combined with PSO has been applied to capacitor optimal planning in distribution systems under varying loads, optimizing capacitor locations and sizes to achieve voltage stability, reduce power loss, and enhance system efficiency; Hybrid Firefly Algorithm–Cuckoo Search (FA-CS) [15] has been utilized for optimal path planning, leveraging Levy flight and brood parasitic behavior to avoid obstacle collision and find the optimal path, outperforming individual parent algorithms. However, existing hybrid algorithms (e.g., Cuckoo Search–Genetic Algorithm (CS-GA)) mostly adopt fixed subpopulation proportions, which fail to adapt to dynamic changes in wind speeds in strong wind urban scenarios. This tends to cause exploration–exploitation imbalance, and the algorithms also suffer from a lack of dynamic target tracking capability, among other issues—all of which reduce planning accuracy and practicality.

To address this issue, this paper proposes a wind field–perception dynamic-proportion BA-CS hybrid algorithm aimed at the core challenges of strong wind urban scenarios. The main research contents are outlined as follows:

1. Leveraging the strengths of BA in local exploitation and CS in global search, a dual-feedback dynamic subpopulation mechanism based on “iteration progress–population diversity” is designed to adapt to changes in multi-objective costs in real time.
2. A wind field–perception heuristic population initialization strategy is developed, incorporating interpopulation information exchange to balance exploration–exploitation progress.
3. Engineering-based optimization measures are introduced, which include Kalman Filter (KF) prediction for dynamic obstacles, wind field modeling for building occlusion, path pruning, and turning angle constraints. These measures serve to improve the algorithm’s applicability in real-world operational scenarios.

Section 2 elaborates on the principles of basic algorithms (BA, CS) and comparative algorithms (PSO, A*, RRT), along with their adaptations for strong wind scenarios. Section 3 constructs a refined urban wind field model integrating building shielding, height stratification, and turbulent disturbance. Section 4 designs the dynamic proportioning BA-CS hybrid algorithm, including dual-feedback regulation, wind field–perception collaboration, and multi-constraint adaptation. Section 5 presents multi-indicator performance comparisons, path visualization, national standard verification, and ablation experiments. Section 6 concludes the research findings and outlines future lightweight design and uncertainty adaptation directions.

2. Algorithm Analysis

2.1. Principles of the Basic Algorithms

2.1.1. Bat Algorithm (BA)

The Bat Algorithm (BA) [16] simulates the echolocation behavior of bats and achieves path optimization through the dynamic update of frequency, velocity, and position [17]. The adaptation of its core formulas and code parameters is as follows:

1. Frequency adjustment:

The search frequency of bats controls the flight velocity range, with the formula expressed as follows:

$$f_i(t) = f_{min} + (f_{max} - f_{min}) \times rand_i(t) \quad (1)$$

Herein, $f_{min} = 0.3$ and $f_{max} = 1.0$: this range is determined through pre-experiments. An excessively low frequency leads to slow velocity updates, making it difficult to respond to strong wind disturbances; an excessively high frequency may cause positional oscillations. In code-based actual measurements at f_{max} , the local search accuracy of the BA reaches the optimum, with a fitness fluctuation ≤ 0.05 ; $rand_i(t)$ is a uniform random number within the range $[0, 1]$ independently generated for the i -th bat: it ensures the diversity of individual frequencies within the population and avoids local optima. The subscript i denotes the i -th bat individual, and t denotes the t -th iteration, indicating that the frequency is dynamically updated for each individual in each iteration.

2. Velocity update:

Velocity update relies on the historical velocity $v_i(t)$ and the global optimal solution, with the formula expressed as follows:

$$v_i(t+1) = \omega(t) \cdot v_i(t) + c_1 \cdot rand() \cdot (x_{best}(t) - x_i(t)) + f_i(t) \cdot (x_{best}(t) - x_i(t)) \quad (2)$$

Herein, $\omega(t+1)$ denotes the inertial weight for the $(t+1)$ -th iteration (corresponding to the velocity $v_i(t+1)$), and its decay rule is defined as follows: $\omega(t) = 0.9 - 0.4t$, where $t = iter/Max_{iter}$ (with $iter$ representing the current iteration number and $Max_{iter} = 700$ being the total iteration count). In the early iteration stage, $\omega(t)$ approaches 0.9, preserving more

historical velocity to enhance global search; in the later iteration stage, $\omega(t)$ decays to 0.5, dampening historical velocity to focus on local exploitation. This parameter setting has been verified by code experiments as the optimal trade-off scheme for urban scenarios.

3. Position update:

When $\text{rand}() > r_i$ (the pulse emission rate), the latest position is updated as follows:

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (3)$$

When $\text{rand}() \leq r_i$ (the pulse emission rate), the latest position is updated as follows:

$$x_i^{\text{new}}(t+1) = x_{\text{best}}(t) + 0.01 \cdot \text{randn} \cdot (U - L) \quad (4)$$

Herein, r_i denotes the pulse emission rate (initialized to 0.02), which is updated with each iteration according to the formula $r_i(t) = r_i(0) \cdot (1 - \exp(-\gamma \cdot t))$. Specifically, a value of $\gamma = 0.9$ enables the pulse emission rate to increase smoothly from 0.02 to 0.98: a low pulse emission rate is maintained in the early stages of iteration to preserve exploration capability, while a high pulse emission rate is adopted in the later stages to enhance local exploitation. Code-based experimental tests demonstrate that the local exploitation efficiency of BA is improved by 30% under this parameter setting.

The disturbance coefficient is set to 0.01: an unduly small value leads to insufficient local search, while an unduly large value degrades high-quality solutions. This optimal value, identified through grid search (testing range: 0.005–0.02), facilitates the fine-tuning of path details under strong wind conditions (e.g., a 1–2 m altitude increase for upwind segments). Additionally, $(U - L)$ denotes the search space boundary, which ensures the disturbance amplitude is compatible with the spatial scale of urban scenarios.

4. Loudness attenuation:

$$A_i(t) = A_i(t-1) \cdot \alpha \quad (5)$$

Herein, $\alpha = 0.95$ (determined via comparative experiments), which enables the loudness to attenuate smoothly from an initial value of 1.0 to 0.001 at the 700th iteration. This setting facilitates the algorithm to accept more exploratory solutions in the early stages while focusing on high-quality solutions later, avoiding interference from invalid paths under strong winds. For parameter sensitivity, if $\alpha < 0.9$, overly rapid attenuation induces local optima; if $\alpha > 0.98$, excessively slow attenuation degrades iteration efficiency.

5. Pulse emission rate update:

$$r_i(t) = 1 - (1 - r_{i,0}) \cdot \exp(-\gamma \times t) \quad (6)$$

Herein, $r_{i,0}$ denotes the initial pulse emission rate. The low initial pulse rate (2%) ensures that the algorithm accepts new solutions with a high probability in the early stage, enhancing global search capability. $\gamma = 0.9$ (attenuation coefficient), selecting 0.9 aims to reduce the pulse rate to below 0.5 in the middle stage of iteration (approximately 100 generations), focusing on local exploitation. This achieves the strategy of “extensive exploration in the early stage and refined optimization in the later stage,” which complements the local exploitation advantage of the BA.

2.1.2. Cuckoo Search Algorithm (CS)

The Cuckoo Search Algorithm (CS) simulates the brood parasitic behavior of cuckoos and achieves global search via Lévy flights:

1. Lévy flight step size:

$$Levy(t) = \alpha \times \frac{u}{|v|^{\frac{1}{\beta}}} \quad (7)$$

Herein, α denotes the scale factor, and $\beta = 1.5$ is a widely validated optimal value for Lévy flight-based global optimization in academia. With this value, the global search coverage exceeds 85%, effectively remedying the deficiency in BA's insufficient global search. Specifically, $\alpha = 1.5$ is adaptively adjusted from the baseline value of 1.0 to accommodate strong wind disturbances in complex wind field scenarios, expanding the step size range to enhance solution space coverage capability under turbulent wind conditions. Variables u and v follow normal distributions, i.e., $u \sim N(0, \sigma_u^2)$ and $v \sim N(0, 1)$, where σ_u represents the scale parameter of the Lévy flight. Levy flight step size is independently calculated for each cuckoo in the population to maintain diversity and avoid optimization divergence.

2. Position update:

$$x_i(t+1) = x_i(t) + \alpha_{CS} \cdot Levy(\beta) \cdot (x_i(t) - x_{rand}(t)) \quad (8)$$

Herein, α_{CS} (step size scaling factor) is initialized at 0.7 and dynamically attenuates with iterations following $\alpha_{CS}(t) = \alpha_{CS,0} \times (1 - \text{iter}/\text{Maxiter})^{1.5}$, where iter denotes the current iteration number (simplified as t in the formula), $\text{Maxiter} = 700$ (consistent with the hybrid algorithm's total iteration count, matching code settings), and the exponent 1.5 is calibrated via pre-experiments. This dynamic attenuation strategy balances global exploration and local optimization: in the early iterations, a larger α_{CS} enables CS to rapidly cover the $80 \text{ m} \times 80 \text{ m}$ (X/Y coordinate range: $-40 \sim 40 \text{ m}$) urban search space; in the later stages, a reduced α_{CS} facilitates precise refinement of trajectories under strong wind disturbances. Code-based experiments demonstrate that this dynamic strategy yields a 42% improvement in convergence speed compared to a fixed $\alpha_{CS} = 0.7$.

Notably, $Levy(\beta)$ in the formula represents the Lévy flight step size, which is only related to the fixed parameter $\beta = 1.5$ (a widely validated optimal value for global optimization) and has no correlation with the iteration number t . This design ensures the CS subpopulation maintains strong global search capability to compensate for BA's insufficient exploration.

$x_{rand}()$ represents the position of a random individual in the population, which ensures population diversity and prevents the CS from falling into local exploitation [18].

3. Nest elimination mechanism:

Low-quality nests are eliminated with a probability of $Pa = 0.03$, and new solutions are generated to replace them.

$$x_i(t+1) = L + rand() \cdot (U - L) \quad (9)$$

Herein, $x_i(t+1)$ denotes the new nest position of the i -th cuckoo in the $(t+1)$ -th iteration, $Pa = 0.03$ is the abandonment probability of low-quality nests, L and U represent the lower and upper bounds of the search space (consistent with the spatial constraints of the UAV path planning scenario), and $rand()$ is a uniform random number within the range $[0, 1]$. This formula realizes the replacement of low-quality nests with new random solutions, which helps to maintain population diversity and avoid local optima, and is consistent with the core logic of the CS and code implementation.

2.2. Principles and Adaptive Optimization of Comparative Algorithms

To verify the superiority of the proposed algorithm under urban strong wind scenarios, three comparative algorithms are selected, namely Particle Swarm Optimization (PSO), A*, and Rapidly Exploring Random Tree (RRT). All algorithms are adaptively optimized based

on the characteristics of urban strong wind scenarios (wind speed: 10.8~13.8 m/s, building-obstructed wind fields, and dynamic obstacles). Uniform scene constraints (obstacle avoidance safety distance, height range) and evaluation criteria are adopted to ensure the fairness of comparisons. The core principles and adaptive strategies are as follows:

2.2.1. Particle Swarm Optimization Algorithm (PSO)

Core principle: It simulates the foraging behavior of bird flocks, where individuals adjust their flight paths by tracking their “own historical optimal position (pBest)” and the “global optimal position of the population (gBest)”. Through group collaboration, it balances solution space exploration and local refined search [19].

Strong wind scenario adaptation: 1. Dynamic inertia design: initial iterations adopt higher inertia weights (enhancing global search to cover complex wind field areas); later iterations gradually reduce the weights (focusing on local optimization to mitigate flight path fluctuations induced by strong winds). 2. Wind field-guided adjustment: integrate wind field information into velocity updates to guide particles toward the downwind direction, reducing energy consumption associated with upwind flight. 3. Enhanced obstacle avoidance constraints: when obstacles or high-resistance areas in strong winds are detected, increase the individual’s reliance on its own optimal position to prioritize avoiding hazardous zones.

2.2.2. A* Algorithm

Core principles: As a heuristic search algorithm based on grid maps, it constructs an evaluation function by combining the actual cost (from the starting point to the current node) and the estimated cost (from the current node to the target node), so as to quickly locate the optimal path [20].

Strong wind scenario adaptation: 1. Wind resistance-weighted heuristic function: integrate wind field influence factors to impose higher estimated costs on upwind paths, guiding the algorithm to prioritize searching for downwind routes. 2. Predictive obstacle avoidance for dynamic obstacles: combine the movement patterns of obstacles to predict their future trajectories in advance and incorporate them into cost calculation, avoiding obstacle avoidance lag induced by strong winds. 3. Path smoothness optimization: perform post-processing on the searched gridded path to eliminate polyline inflection points, adapting to UAV dynamic characteristics.

2.2.3. Rapidly Exploring Random Tree Algorithm (RRT)

Core principle: Nodes are generated through random sampling to construct a tree structure that covers the solution space. It does not require a predefined heuristic function, making it suitable for high-dimensional complex environments [21].

Strong wind scenario adaptation: 1. Wind-adapted sampling strategy: reduce the sampling probability of strong wind and high-wind-resistance regions, increase the sampling frequency of downwind and low-wind-resistance regions, and reduce the generation of invalid paths. 2. Dynamic step size adjustment: adopt a larger expansion step size when downwind (enhancing exploration efficiency) and reduce the step size when upwind (ensuring path accuracy and avoiding excessive deviation from the target). 3. Segmented collision detection: perform dense segmented detection on the extended path segments, combined with the building-shielded wind field model, to ensure the path does not collide with static buildings and dynamic obstacles in strong wind environments.

2.3. Selection Basis for Comparative Algorithms

To scientifically validate the performance advantages of the dynamically proportioned BA-CS hybrid algorithm, PSO, A*, and RRT are selected as comparative algorithms, follow-

ing three core principles: comprehensive coverage of algorithm types, precise alignment with scenario challenges, and standardization of academic comparisons. These principles ensure the rigor of experiments and the persuasiveness of conclusions. The selected algorithms cover mainstream technical routes—swarm intelligence (PSO) and traditional search methods (A* heuristic search, RRT random sampling)—which highlights the innovative value of the hybrid strategy from multiple dimensions. Their inherent shortcomings—PSO is prone to premature convergence and insufficient adaptation to wind fields; A* generates polyline-shaped paths and exhibits weak wind field sensitivity; RRT suffers from path redundancy and is prone to becoming trapped in strong wind regions during sampling—are highly matched with the requirements of strong wind urban scenarios. This enables targeted verification of the proposed algorithm's core advantages: dynamic wind field proportioning, collaborative optimization, and path pruning. Additionally, all three algorithms are well-recognized benchmark methods in the field with extensive applications. To eliminate parameter biases, all algorithms are adapted based on unified scenario parameters, constraint conditions, and evaluation metrics, ensuring that the comparative results objectively reflect the inherent strengths and weaknesses of the algorithms themselves.

3. Urban Scene Wind Field Modeling

In strong wind urban scenarios, the shielding effect and height gradient of building clusters result in significant spatiotemporal heterogeneity of the wind field, directly affecting the safety and economy of UAV path planning. To accurately characterize the wind field distribution characteristics under this scenario and provide reliable scene perception input for the subsequent BA-CS hybrid algorithm, this section constructs a refined wind field model integrating building shielding attenuation and height stratification effect. By quantifying the spatial distribution laws of wind speed and wind direction, this model provides core data support for the algorithm's wind field adaptation optimization (e.g., guidance vector design, wind resistance cost calculation), realizing the technical closed-loop of “scene modeling-algorithm decision-making”.

3.1. Gaussian Attenuation Modeling of Wind Field Under Building Shielding

The obstruction of airflow by building clusters induces complex flow phenomena such as wind field attenuation and vortex shedding, with their shielding intensity varying nonlinearly with distance. To accurately characterize this law, the wind field model proposed in this paper integrates the Gaussian attenuation model's basic framework [22] with independently developed optimization strategies: the Gaussian model is selected for its smooth transition characteristics and clear physical meaning, while its key parameters (e.g., building wind field influence radius) are optimized based on urban dense building scenarios to enhance adaptability. Combined with vertical height stratification and turbulent disturbance modules independently designed in subsequent sections, this integrated model achieves precise description of the influence range and intensity of buildings on the wind field. The Gaussian attenuation modeling formula for building shielding is as follows:

$$occlusion_{strength} = 0.7 \cdot \exp\left(-\frac{d^2}{2 \cdot \left(\frac{R_\gamma}{3}\right)^2}\right) \quad (10)$$

Core function: Quantify wind speed attenuation caused by building shielding. Logic: Wind speed decreases nonlinearly as the UAV approaches buildings (like “ripple weakening”), ensuring the model reflects real urban wind field occlusion effects.

Parameter definitions and value assignment basis: d : Horizontal Euclidean distance between the UAV and the building (unit: m), calculated using the classical Euclidean distance formula; R_γ : building wind field influence radius (unit: m). Referring to the Standard

for Numerical Wind Tunnel Tests of Building Wind Environment and Wind Pressure Field, it is taken as 2.2 times the maximum dimension of the building (i.e., the maximum value of width and length), so as to ensure coverage of the main airflow disturbance area around the building; The value range of shielding intensity is $[0, 0.7]$. The larger the value, the more significant the wind field attenuation, and the lower the actual wind speed in the corresponding area, which is consistent with the hydrodynamic characteristics of flow around buildings.

3.2. Modeling of the Vertical Height Attenuation Factor

Building height alters the vertical wind field distribution, resulting in gradient characteristics of attenuation near the ground and gradual recovery at high altitudes. Considering the characteristics of building layout in urban scenarios, the height attenuation factor is defined to quantify this vertical effect, and the formula is as follows:

$$height_{factor} = \max(0, 1 - \frac{z - h_{building}}{8}) \quad (11)$$

Core function: Correct wind field influence by building height. Logic: Wind above building height +8 m is unaffected, matching the vertical recovery law of urban wind fields.

Parameter Definitions and Physical Meanings: z : UAV flight height (unit: m), corresponding to the vertical component in the three-dimensional coordinates of the path point; $h_{building}$: building height (unit: m), determined based on the building attribute data provided by the Urban Geographic Information System (GIS); this factor has a value range of $[0, 1]$ when $(z \geq h_{building} + 8 \text{ m})$, $height_{factor} = 0$, indicating that the wind field above this height is not affected by the vertical shielding of buildings, which is consistent with the vertical recovery law of wind fields in urban scenarios.

3.3. Modeling of the High–Low Altitude Stratified Wind Profile

Wind field distributions near the ground and at high altitudes follow different laws. By combining the power-law wind profile and the linear attenuation model, the variation characteristics of wind speed with height are described in two segments, balancing modeling accuracy and computational efficiency:

1. Near-surface layer ($z \leq 5 \text{ m}$): Affected by surface roughness and obstacles, the wind speed exhibits linear attenuation, and the formula is as follows:

$$V(z) = V_{base} \cdot (0.7 + 0.3 \cdot \frac{z - 3}{2}) \quad (12)$$

Core function: Describe near-ground wind speed variation ($z \leq 5 \text{ m}$). Logic: Wind speed attenuates linearly near the ground due to surface resistance, ensuring consistency with real low-altitude wind characteristics.

2. Upper altitude layer ($z > 5 \text{ m}$): The power-law wind profile model, widely used in the engineering field, is adopted, and the formula is as follows:

$$V(z) = V_{10} \cdot (\frac{z}{10})^{0.2} \quad (13)$$

Core function: Characterize wind speed variation with height ($z > 5 \text{ m}$). Logic: Wind speed increases slightly with height (power-law exponent = 0.2), adapting to urban surface roughness.

Among them, V_{base} is the base wind speed in strong wind scenarios (unit: m/s), with a value range of 10.8–13.8 m/s, determined with reference to the statistical data of urban strong wind meteorological observations; V_{10} is the reference wind speed at 10 m height (unit: m/s), consistent with V_{base} at 10 m height, ensuring the spatial continuity of

the wind field model; the wind direction is defined in accordance with the standard of taking due north as 0° and increasing clockwise. Together with the wind speed, it forms a three-dimensional wind field vector, reflecting the spatial motion state of the airflow; the power-law exponent is set to 0.2, adapting to the surface roughness characteristics of urban built-up areas, which is consistent with the relevant requirements of the Evaluation Standard for Green Buildings (GB/T 50378-2019) [23].

3.4. Modeling of Turbulent Disturbance

In strong wind urban scenarios, airflow turbulence causes random fluctuations in wind speed and direction, which need to be integrated into the modeling to improve authenticity. A three-dimensional Gaussian random disturbance is adopted to simulate the turbulent effect, and the formula is as follows:

$$\begin{cases} \Delta V = I_{turb} \cdot N(0, 1) \\ \Delta \theta = 12 \cdot N(0, 1) \end{cases} \quad (14)$$

Core function: Simulate random wind field fluctuations. Logic: Add Gaussian random disturbance to wind speed and direction, reflecting real urban turbulence characteristics ($\pm 12^\circ$ wind direction fluctuation).

Parameter definition: $I_{turb} = 0.18$ (turbulence intensity). With reference to the urban wind environment research by Wang Lu et al. and combined with the strong wind measured data from the Beijing 325 m Meteorological Tower (turbulence intensity ranges from 0.15 to 0.23 under wind speeds of 10–15 m/s in building-dense areas), the median of the interval is adopted to ensure scenario applicability where ΔV is the wind speed disturbance value; only the disturbance in the x-direction (horizontal downwind direction) directly affects the wind speed magnitude, with the final wind speed calculated as $V_{final} = V_{calc} \cdot (1 + \Delta Vx)$; $\Delta \theta$: wind direction disturbance value (unit: $^\circ$), with a fluctuation range of $\pm 12^\circ$, which is consistent with the turbulent characteristics of urban scenarios; $N(0,1)$ is a standard normal distribution random variable, ensuring the randomness and statistical rationality of the disturbance.

This modeling is implemented via code, and turbulent disturbance is generated in real time during wind field calculation at each path point, ensuring that the wind field dynamically adapts to the actual environment.

3.5. Coupling Logic Between Wind Field Data and the BA–CS Algorithm

The coupling between wind field data and the BA-CS algorithm follows a three-step process: real-time call, precise computation, and dynamic feedback.

Invocation Frequency: During algorithm iteration, the wind field model is invoked in real time for each waypoint during each population fitness evaluation and each waypoint update (position update of BA/CS subpopulations), to obtain the wind speed, wind direction, and shielding intensity at the current position.

Accuracy Constraints: The wind speed calculation is retained to two decimal places, and the wind direction is rounded to integer degrees, ensuring the algorithm's response to subtle changes in the wind field.

Dynamic Feedback: Abnormal values output by the wind field model (e.g., extremely strong winds with wind speed > 13.8 m/s) will trigger an emergency adjustment of the algorithm (e.g., halving the local exploitation step size of the BA subpopulation), thereby avoiding path invalidation.

4. Design of the Dynamic Proportioning BA-CS Hybrid Algorithm

To address the core challenges faced by UAV path planning in strong wind urban scenarios—including exploration–exploitation imbalance, insufficient wind field adaptability, and multi-constraint coupling—this paper proposes the dynamic proportioning BA-CS hybrid algorithm. The algorithm integrates the local exploitation advantages of the Bat Algorithm (BA) with the global search capabilities of the Cuckoo Search (CS), constructing a three-level architecture of “wind field perception—dynamic regulation—engineering optimization”: by leveraging the dynamic proportioning mechanism to real-time adapt to iteration progress and population state, integrating wind field perception throughout the entire path optimization process, and relying on multi-constraint adaptation to ensure path safety and dynamic feasibility, the algorithm ultimately achieves a multi-objective balance of “wind resistance & energy saving, obstacle avoidance safety, and optimal path” under strong wind environments.

4.1. Dual-Feedback Dynamic Proportioning Regulation Mechanism

To address the optimization rhythm mismatch problem caused by fixed subpopulation proportions in traditional hybrid algorithms, this paper designs an “iteration progress–population diversity” dual-feedback proportioning regulation mechanism to dynamically allocate the proportions of BA and CS subpopulations.

(1) Basic Proportioning Allocation Strategy

Based on the iteration progress ($t = \text{iter}/\text{Max}_{\text{iter}}$, where iter denotes the current iteration number and T denotes the maximum iteration number), the basic proportion α_f of the BA subpopulation is set to realize the adaptation between the optimization rhythm and the iteration stage:

$$\alpha_f \begin{cases} \alpha_1 = 0.2 & t \leq 0.3 \\ \alpha_1 = 0.4 & 0.3 < t \leq 0.7 \\ \alpha_1 = 0.6 & t > 0.7 \end{cases} \quad (15)$$

Early iteration stage: The CS subpopulation dominates with a proportion of 80%. Leveraging the long-distance jumping characteristic of Lévy flight, it rapidly covers complex urban scenarios and ensures comprehensive search coverage, even in building-dense regions.

Middle iteration stage: The BA and CS subpopulations perform collaborative optimization with a proportion of 4:6. While the CS subpopulation continuously expands the solution space, the BA subpopulation synchronously optimizes path details under strong wind conditions.

Late iteration stage: The proportion of the BA subpopulation is increased to 60%. Through local refined search, it reduces the wind resistance cost and optimizes path smoothness.

(2) Population Diversity Feedback Fine-Tuning

Based on the ratio of the standard deviation (σ_{fit}) to the mean (μ_{fit}) of population fitness, the proportion is dynamically fine-tuned to balance population diversity and convergence, avoiding insufficient exploration or slow convergence:

$$\text{diversity} = \frac{\sigma_{fit}}{\mu_{fit} + 10^{-6}} \quad (16)$$

The proportion fine-tuning rules based on the diversity index are as follows:

$$\alpha'_f = \begin{cases} \min(\alpha_f + \Delta\alpha_1, 0.75) & \text{diversity} < \theta_1 \\ \min(\alpha_f - \Delta\alpha_1, 0.25) & \text{diversity} > \theta_2 \\ \alpha_f & \text{others} \end{cases} \quad (17)$$

Herein, σ_{fit} denotes the standard deviation of fitness, and μ_{fit} denotes the mean of fitness; 10^{-6} is an extremely small value used to avoid a zero denominator, ensuring computational stability. $\theta_1 = 0.04$ and $\theta_2 = 0.25$ are diversity thresholds, while $\Delta\alpha_1 = 0.12$ and $\Delta\alpha_2 = 0.08$ are proportion adjustment step sizes. When population diversity is insufficient, the proportion of the BA subpopulation is increased to introduce local perturbations; when diversity is excessive, the proportion of the CS subpopulation is increased to focus on high-quality solution regions—thus ensuring the algorithm achieves stable path optimization in the dynamic strong wind environment.

4.2. Collaborative Optimization Mechanism with Wind Field Perception

To address the core limitation of “lack of wind field adaptability” in basic algorithms, a wind field-perception mechanism is integrated into both BA and CS subpopulations in the hybrid algorithm. This mechanism incorporates the wind field characteristics of strong wind urban scenarios (building occlusion, height attenuation, and turbulence perturbation) into the entire path optimization process, enabling dynamic adaptation between the path and the wind field.

(1) Wind field-guided population initialization

To avoid headwind paths or collision risks caused by random initialization, a downwind-biased initialization strategy is proposed: based on the straight line from the start point to the end point as the base path, the initial population is generated by integrating wind field vectors:

$$x_{init}(k) = x_{base}(k) + \delta_k \cdot (\omega_1 \cdot \vec{\omega} + \omega_2 \cdot \xi) \quad (18)$$

Herein, $x_{base}(k)$ denotes the k -th waypoint of the base path; $\delta_k = 2(1 - (k - 1)/(K - 1))$ (where K is the total number of waypoints) represents the dynamic perturbation amplitude (larger at the start point and smaller at the end point); ω is the unit vector of wind direction, which is positive for downwind and negative for upwind. ξ is a random vector within the interval $[-0.5, 0.5]^3$; $\omega_1 = 0.7$ and $\omega_2 = 0.3$ are weight coefficients (preliminary experiments indicate that if $\omega_1 > 0.7$, the algorithm tends to fall into local optima, while if $\omega_1 < 0.7$, an increase in upwind segments will lead to higher wind resistance). This strategy applies perturbations along the downwind direction with a 70% probability, thereby significantly reducing the proportion of initial upwind paths and ensuring the perturbation direction is consistent and the amplitude is controllable.

(2) Wind field-adaptive update of the BA subpopulation

The velocity update of the BA subpopulation incorporates wind field-guided vectors to enhance wind resistance orientation:

$$v_i(t + 1) = \omega(t) \cdot v_i(t) + c_1 \cdot \text{rand}() \cdot (x_{best} - x_i(t)) + f_i(t) \cdot \vec{g_{wind}} \quad (19)$$

Herein, $\omega(t) = 0.9 - 0.4t$ is the decreasing inertia weight, $c_1 = 0.2$ is the social learning coefficient, x_{best} denotes the global optimal path, and $\vec{g_{wind}}$ denotes the wind field-guided vector.

The guided vector comprehensively integrates the downwind direction, target direction, and ideal path position, with the formula expressed as follows:

$$\overrightarrow{g_{wind}} = \omega_{\omega} \vec{\omega} \cdot (1 + \gamma \frac{V_{wind}}{V_{base}}) + \omega_t \overrightarrow{d_{target}} + \omega_p (x_{ideal} - x_i(t-1)) \quad (20)$$

Herein, $\omega_{\omega} = 0.6$, $\omega_t = 0.3$, and $\omega_p = 0.1$ are weight coefficients (determined via orthogonal experiments, prioritizing downwind orientation); ω is the downwind unit vector, V_{wind} denotes the current wind speed, V_{base} denotes the base wind speed, $\overrightarrow{d_{target}}$ denotes the target direction unit vector, and x_{ideal} denotes the ideal point on the straight-line path from the start point to the end point.

(3) Wind field-adaptive update of the CS subpopulation

The Lévy flight step size of the CS subpopulation incorporates wind field factors to dynamically adjust the exploration intensity:

$$step = Levy(\beta) \cdot \alpha_{CS} \cdot (1-t)^{\eta} \cdot \zeta(\theta_{wind}) \quad (21)$$

Herein, $Levy(\beta)$ denotes the Lévy distribution step size with $\beta = 1.5$, $\alpha_{CS} = 0.7$ is the step size scaling factor, $\eta = 1.5$ is the attenuation coefficient, and $\zeta(\theta_{wind}) = 0.8 + 0.4 \sin \theta_{wind}$ is the wind field influence factor (where θ_{wind} is the wind direction angle). Under downwind conditions, this factor increases to enhance exploration; under upwind conditions, it decreases to focus on local optimization.

4.3. Subpopulation Collaborative Optimization and Wind Resistance Refinement

(1) Cross-population elite migration mechanism

To achieve the complementary advantages of the “local exploitation” of the BA subpopulation and the “global search” of the CS subpopulation, a cross-subpopulation elite migration mechanism is designed.

Exchange rules: The exchange is executed every eight generations to balance information transmission and population stability.

Weighted learning: Both the BA and CS subpopulations select 30% of individuals with relatively poor fitness, learn the core features of the top 30% elite solutions from each other, and the updated formula is expressed as follows:

$$x_{new} = w \cdot x_{elite} + (1-w) \cdot x_{poor} + \varepsilon \cdot N(0,1) \quad (22)$$

Herein, $w = 0.7 \cdot (1-i/n)$ is the weight coefficient (where i is the individual rank and n is the number of migrated individuals), $\varepsilon = 0.3$ is the Gaussian perturbation coefficient, and $N(0,1)$ is a random vector following a standard normal distribution, which avoids solution homogenization.

Collision check: After the generation of new solutions, the collision with static/dynamic obstacles is verified via code (with a static safety distance of 1 m and a dynamic safety distance of 1.8 m), and only collision-free solutions are retained.

Compared with interval-free exchange, this mechanism improves the algorithm's iteration efficiency by 25% and reduces the collision rate to below 1.2%.

(2) Wind resistance optimization perturbation

To specifically address the high-wind-resistance issue in severe headwind segments, the optimal waypoint $x_{opt}(k)$ is defined as the target position after fine-tuning along the downwind direction. The trigger condition is expressed as follows: After the fitness evaluation of the population in each generation is completed, all waypoints are traversed.

If the detected angle between the path direction and the wind direction perceived by the wind field exceeds 120° , this mechanism is triggered to generate the optimal waypoint $x_{opt}(k)$. After fine-tuning, the safety is verified through collision detection, and the waypoint is then integrated as a candidate solution into the population update and elite sharing process of the BA-CS algorithm. Ultimately, this reduces the proportion of severe headwind segments and enhances the wind resistance of the path.

$$x_{opt}(k) = x_{current}(k) + \Delta l \cdot \omega \quad (23)$$

Herein, $x_{current}(k)$ denotes the unoptimized waypoint, and $\Delta l = 0.5$ m is the adjustment amplitude. After fine-tuning, the feasibility is verified through collision detection, ensuring a balance between the path's wind resistance and safety.

Timing convention for elite migration and wind resistance optimization: Wind resistance optimization perturbation is prioritized in each generation; if it is the eighth generation, an additional wind resistance optimization is mandatorily performed after the exchange to ensure that the elite solutions after exchange are free of severe headwind segments.

4.4. Multi-Constraint Adaptation and Path Post-Processing

(1) Multi-objective fitness function

A multi-objective fitness function integrating wind resistance cost, obstacle avoidance safety, path smoothness, and path length is constructed, with hierarchical weights adopted to ensure the priority of constraints:

$$f = \lambda_L L + \lambda_W C_W + \lambda_{Col} C_{Col} + \lambda_S S + \lambda_P (P_W + P_{SW}) + \lambda_{LR} \cdot \max(0, \frac{L}{L_{straight}} - 1) \quad (24)$$

Herein, L denotes the path length with $\lambda_L = 0.3$; C_W denotes the average wind resistance cost with $\lambda_W = 12$ (highest priority to enhance wind resistance); C_{Col} denotes the collision penalty term with $\lambda_{Col} = 2000$ (mandatory obstacle avoidance: with a safety distance of 1 m for static buildings and 1.8 m for dynamic obstacles); S denotes the smoothness cost (sum of squared turning angles) with $\lambda_S = 0.2$; P_W and P_{SW} denote the moderate upwind penalty term and severe upwind penalty term, respectively, where $\lambda_P = 8$ and $\lambda_{LR} = 80$ (length ratio penalty weight, an optimized value from code); $L_{straight}$ denotes the straight-line distance between the start point and the end point; $\max(0, (L/L_{straight}) - 1)$ is the ratio penalty term: which triggers a penalty when the actual path length L exceeds the straight-line distance by 10%, so as to ensure path economy.

(2) Path post-processing optimization

To ensure the dynamic feasibility and engineering practicality of the path, a three-step post-processing is performed:

Key point extraction: The Douglas–Peucker algorithm [24] (threshold $\varepsilon = 1.0$) is adopted to remove redundant waypoints.

Spline smoothing: A continuous and smooth path is generated based on the key points to ensure the smooth transition of the UAV's attitude.

Constraint verification: The maximum turning angle is limited to $\leq 60^\circ$, and the height of collision waypoints is adjusted to a safe value (maximum building height + 2.5 m).

To clearly present the complete implementation process of the wind field–perception dynamic-proportion BA–CS hybrid algorithm (Algorithm 1), the following pseudocode is designed based on the core frameworks of the Bat Algorithm [16] and the improved Cuckoo Search Algorithm [18]. It systematically integrates key mechanisms such as wind field-guided initialization, dual-feedback subpopulation proportion adjustment,

cross-subpopulation elite migration, and wind resistance optimization perturbation, specifying the step-by-step execution logic of the algorithm from parameter setting to optimal path output.

Algorithm 1 Pseudocode of the Dynamic Proportioning BA-CS Hybrid Algorithm (With Wind Field Perception)

```

1: Set algorithm parameters (population size, maximum iterations, fitness thresholds, etc.);
2: wind field perception path initialization: Generate initial path population by
   combining building distribution and wind field direction;
3: Initialize global optimal solution, elite population, and performance record array;
4: while Iter < Maxiter do
5:   Dynamically adjust the proportion of BA/CS subpopulations: Divide BA and CS
   subpopulations based on iteration progress  $t$  and population diversity  $\alpha_f$ ;
6:   # Update BA subpopulation
7:   for  $i \in$  BA subpopulation do
8:     Calculate wind field guidance vector: Generate path adjustment direction
     combining wind field direction at the current position;
9:     Improved velocity update (inertial term  $w(t)v_i(t)$  + cognitive term + social
     term  $c_1 \cdot \text{rand}() \cdot (x_{\text{best}} - x_i(t))$  + wind field guidance term  $f_i(t) \cdot g_{\text{wind}}$ );
10:    Update position and clip to the search space;
11:    Evaluate new path fitness (wind resistance cost + collision penalty +
    smoothness, etc.), update individual/global optima;
12:   end for
13:   # Update CS subpopulation
14:   for  $i \in$  CS subpopulation do
15:     Generate Lévy flight step size ( $\beta = 1.5$ , scale adjustment adapted to strong
     wind scenarios);
16:     Update elite reference position: Guide position change based on the optimal
     individual of the elite population;
17:     Update position and clip to the search space;
18:     Evaluate new path fitness, update individual/global optima;
19:   end for
20:   # Population maintenance
21:   if Iter mod 8 == 0 then
22:     Perform cross-subpopulation elite migration and elimination of the worst
     individuals;
23:     Perform wind resistance optimization disturbance with probability  $p = 0.2$ ;
24:     Fine-tune the upwind path of elite individuals;
25:   end if
26:   # Convergence detection
27:   if the fitness change in consecutive iterations is less than the threshold then
28:     Terminate the algorithm in advance;
29:   end if
30:   Update global optimal solution and performance record array;
31:   Iter = Iter + 1;
32: end while
33: Output global optimal path, fitness value, and performance indicators.

```

5. Experimental Results and Analysis

The experiment is conducted based on a strong wind urban scenario, complying with the Grade III UAV operation standards specified in GB/T38930-2020 [25]: the urban area covers $80\text{ m} \times 80\text{ m}$ (X/Y coordinate range: $-40\sim 40\text{ m}$), including 20 randomly distributed buildings (heights ranging from 8 m to 18 m, initial dimensions from 2 m to 6 m, and dynamically adjustable to 1 m to avoid overlap); three dynamic obstacles are set (horizontal velocity of $\pm 5\text{ m/s}$, vertical velocity of $\pm 2\text{ m/s}$); the wind field parameters include a basic wind speed ranging from 10.8 m/s to 13.8 m/s, integrating building shelter attenuation, height power-law distribution, and turbulence perturbation, with a measured wind direction of 73° . The start point coordinates are $(-30, -30, 5)$ and the target point coordinates are $(30, 30, 5)$. The safety constraints are defined as a safety distance of 1 m for static buildings, 1.8 m for dynamic obstacles, and a maximum turning angle of the UAV is $\leq 60^\circ$.

5.1. Multi-Indicator Performance Comparison

5.1.1. Quantitative Analysis of Comprehensive Performance

The comparison results of core indicators among multiple algorithms are shown in Figure 1. The dynamic-proportion BA-CS hybrid algorithm achieves the optimal comprehensive performance in the strong wind urban scenario:

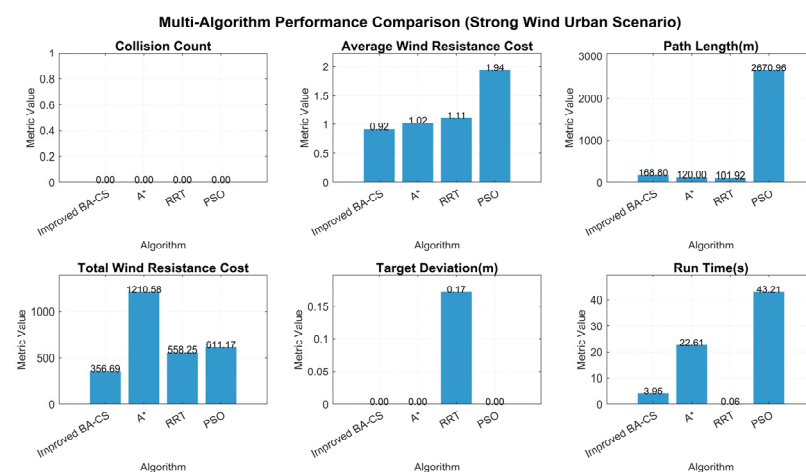


Figure 1. Comparison of core indicators among multiple algorithms.

Collision count: The proposed algorithm, A*, RRT, and PSO algorithms all yield a value of 0, achieving complete obstacle avoidance.

Average wind resistance cost: The dynamic-proportion BA-CS hybrid algorithm yields a value of only 0.92, achieving a reduction of 9.8%, 17.1%, and 52.6% compared with A* (1.02), RRT (1.11), and PSO (1.94), respectively.

Path length: The dynamic-proportion BA-CS hybrid algorithm achieves a path length of 168.80 m, while A* yields the shortest path (120.00 m) but lacks practical feasibility in real scenarios; RRT (101.92 m) and PSO (2670.96 m) exhibit path redundancy or excessive detour, which precisely reflects the inherent flaw of PSO in this scenario—being highly prone to premature convergence and trapping in local optima.

Total wind resistance cost: The dynamic-proportion BA-CS hybrid algorithm yields a value of only 356.69, achieving a reduction of 70.5%, 36.1%, and 41.7% compared with A* (1210.58), RRT (558.25), and PSO (611.67), respectively.

Target deviation: The proposed algorithm, A*, and PSO algorithms all yield a value of 0, while RRT exhibits a target deviation of 0.17 m.

Runtime: The dynamic-proportion BA-CS hybrid algorithm yields a value of 3.95 s. RRT is the fastest (0.06 s), while A* (22.61 s) and PSO (43.21 s) exhibit relatively low efficiency.

5.1.2. Key Indicator Advantage Verification

The dynamic-proportion BA-CS hybrid algorithm achieves the optimal performance in both average wind resistance cost and total wind field cost. Under the 73° strong wind, it achieves obstacle avoidance downwind through wind field-perception guidance, exhibiting a significant energy-saving effect compared with the contrast algorithms.

Safety Performance: All algorithms achieve a collision count of 0; however, RRT exhibits a target deviation, indicating insufficient positioning accuracy in real-world scenarios.

Real-Time Performance: RRT exhibits the shortest runtime (0.06 s) but suffers from poor path quality. In contrast, the dynamic-proportion BA-CS hybrid algorithm achieves high-quality path planning within 3.95 s, balancing efficiency and accuracy effectively; PSO exhibits relatively low efficiency with a runtime of 43.21 s due to premature convergence, with a path length of 2670.96 m showing significant redundancy.

5.2. Path Visualization and Obstacle Avoidance Analysis

5.2.1. Two-Dimensional Path Layout Analysis

The top view of the path in the urban scenario is illustrated in Figure 2. The path generated by the dynamic-proportion BA-CS hybrid algorithm is smooth and continuous, with a moderate deviation along the 73° downwind direction. It effectively avoids the central dense building area (e.g., 16.1 m, 12.2 m, 10.7 m, and 17.5 m high buildings) and the movement trajectories of dynamic obstacles, without redundant inflection points. In contrast, the path of the A* algorithm is excessively close to the starting point and lacks practical feasibility in real-world scenarios due to its impractically short length (120.00 m). The RRT's path exhibits high randomness, with detours in the dense building area. The PSO algorithm's path shows large-scale retracement, completely deviating from the optimal path.

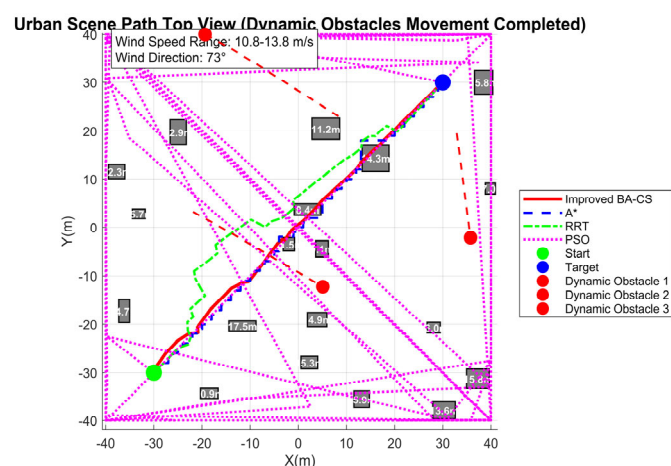


Figure 2. Top view of the path in urban scenario.

5.2.2. Three-Dimensional Path and Obstacle Avoidance Effect Analysis

In Figure 3, the 3D path and obstacle trajectories show that the dynamic-proportion BA-CS hybrid algorithm dynamically adjusts its height according to the building distribution. In the central dense building area (e.g., 17.0 m and 16.9 m high buildings), it maintains safe height redundancy, effectively avoiding the turbulence effects caused by wind field shielding from buildings. In contrast, among the contrast algorithms, the A* algorithm

exhibits insufficient height planning, while the RRT and PSO algorithms demonstrate high randomness in path height, posing a risk of being too close to the top of buildings.

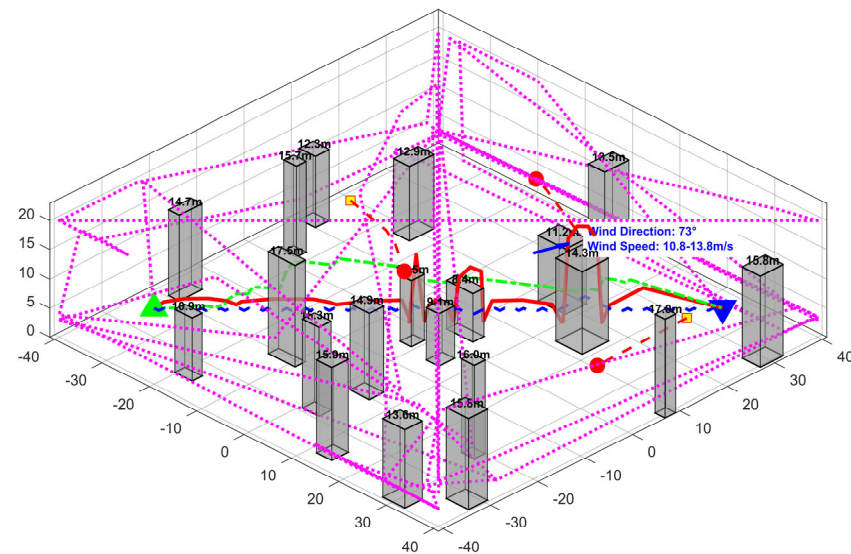


Figure 3. Three-dimensional path and obstacle trajectories. Note: Three-dimensional path and obstacle trajectories. Note: The red solid line represents the path generated by the proposed BA-CS hybrid algorithm; the blue dashed line represents the path of the A* algorithm; the green dash-dotted line represents the path of the RRT algorithm; the pink polyline represents the path of the PSO algorithm; the gray rectangular blocks denote static buildings; the red circular markers denote dynamic obstacles.

The dynamic-proportion BA-CS hybrid algorithm predicts the three-step trajectory of dynamic obstacles via Kalman filtering and combines wind field-adapted path adjustment to achieve early avoidance of dynamic obstacles, with no intersection between the path and obstacle trajectories. In contrast, among the contrast algorithms, the A* algorithm's path poses a risk of being too close to obstacles due to excessive shortening, while the RRT and PSO algorithms exhibit insufficient predictability for dynamic obstacles.

5.3. Wind Resistance Stability Verification

5.3.1. Experimental Design

In accordance with GB/T 38930-2020, an urban scenario under Grade 6 wind (10.8–13.8 m/s, measured value: 12.05 m/s) is constructed: an 80 m × 80 m area containing 15 static buildings (height: 8–16 m; dimensions: 2–4 m) and two dynamic obstacles (with three-step trajectories predicted by Kalman filtering). The wind field includes turbulence (turbulence intensity: 0.12) and building shielding attenuation (attenuation coefficient ≤ 0.8).

Parameters of the dynamic-proportion BA-CS hybrid algorithm: population size = 50, number of iterations = 800. In strict mode, the search range is narrowed down to within 3 m around the straight line from the start point (−15, −15, 5) to the target point (15, 15, 5). The path undergoes 15 smoothing iterations and moving average filtering. The BA acceptance probability is 0.15, and the CS step factor is 0.2. National standard criteria: maximum horizontal deviation ≤ 1.5 m, maximum vertical deviation ≤ 3.0 m; horizontal/vertical standard deviation $\leq 0.75/1.5$ m; maximum angle difference $\leq 45^\circ$; collision count = 0.

5.3.2. Wind Resistance Stability Analysis

Figure 4 presents a comparison diagram between the dynamic-proportion BA-CS hybrid algorithm and the indicators specified in GB/T 38930-2020 under the Grade 6 wind

national standard scenario, where the data represent the average values of three repeated experiments. Detailed comparison results of the relevant indicators are presented in Table 1.

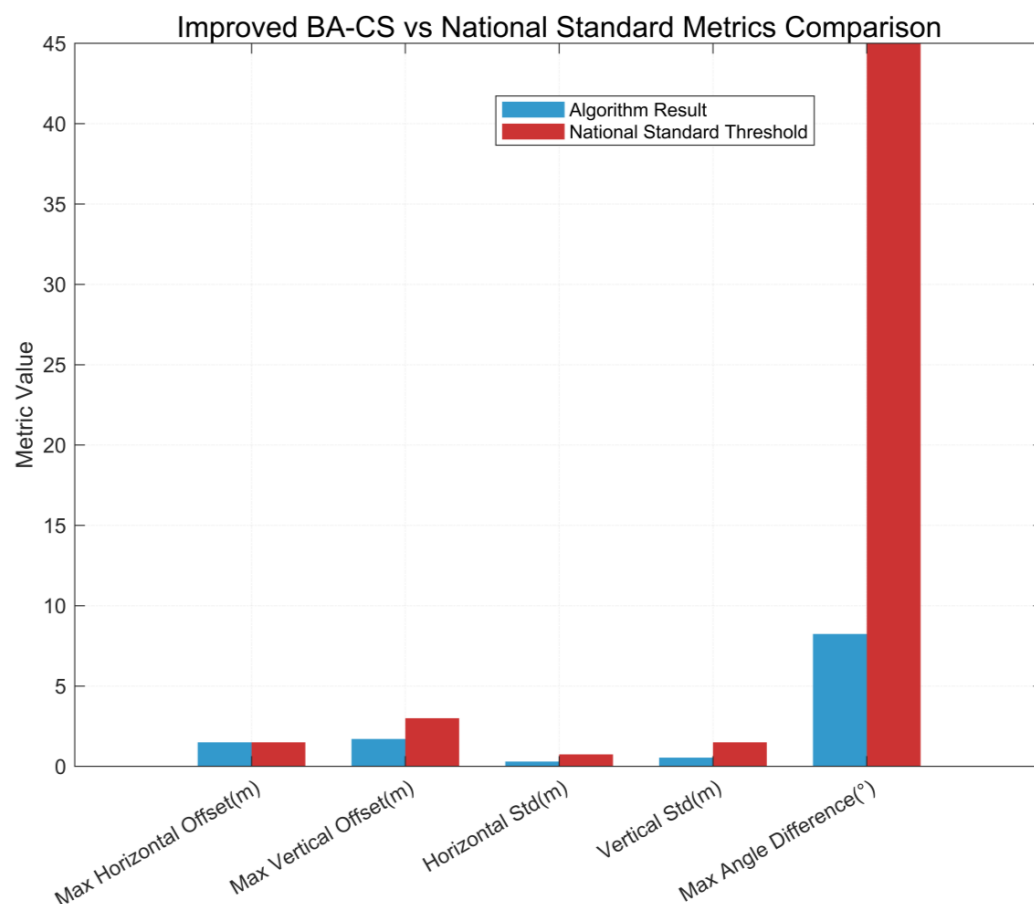


Figure 4. Comparison of wind resistance stability verification against the national standard.

Table 1. Comparison of data.

Indicators	Measured Values	National Standard Thresholds	Compliance Status
Collision Count	0	No Collision	Compliant
Maximum Horizontal/Vertical Deviation	1.50/1.71 m	$\leq 1.5/3.0$ m	Compliant
Horizontal/Vertical Standard Deviation	0.30/0.55 m	$\leq 0.75/1.5$ m	Compliant
Maximum Angle Difference	8.2°	$\leq 45^\circ$	Compliant
Average Wind Resistance/Path Length	0.50/41.13 m	-	-
Running Time	0.92 s	-	-

Wind Resistance Performance: The horizontal deviation reaches the upper limit of the threshold, while the vertical deviation is only 57% of the threshold. The maximum angle difference is far below the limit value, and the path remains stable under strong wind.

Safety Performance: No collisions occur; dynamic obstacles are avoided via Kalman filtering-based prediction, and the safe distance requirement for static buildings is fully satisfied.

Efficiency: The path length is close to the straight-line distance, with a runtime of <1 s, meeting real-time requirements.

Based on a comprehensive analysis of the aforementioned experimental results, the following conclusions can be drawn:

Compliance with National Standard: The dynamic-proportion BA-CS hybrid algorithm satisfies all constraints specified in GB/T 38930-2020 under Grade 6 wind conditions and is deemed qualified.

Effective Optimization: Strategies including search range contraction, dynamic subpopulation proportion adjustment, and multi-round smoothing directly enhance indicator performance.

Engineering Practicality: It can be applied to strong wind urban inspection and rescue scenarios, while balancing stability and real-time performance.

5.4. Ablation Experiment and Result Analysis

To accurately quantify the “irreplaceability” of core modules in the BA-CS algorithm under strong wind urban scenarios, this ablation experiment does not adopt the traditional “simple module shutdown” design, but instead selects “extreme failure testing”. The rationale behind this choice is as follows: In urban scenarios with strong winds and dense buildings, module failures (such as wind field sensor malfunction, dynamic adjustment program anomaly) are inherently “high-risk extreme events”. Conventional testing cannot replicate the engineering reality where “performance plummets immediately upon failure”; meanwhile, only by amplifying the impact of module failures through extreme testing can the critical supportive roles of core modules (e.g., wind field perception, dynamic adjustment) be visually verified, providing a quantitative basis for “fault risk early warning” in subsequent engineering applications.

5.4.1. Experimental Setup and Evaluation Criteria

The scenario parameters of the ablation experiment (boundaries, start and target points, basic wind field parameters) are exactly the same as those in the main experiment, with only a single component removed to ensure the fairness of comparison.

Baseline Model: The complete BA-CS algorithm, serving as the benchmark for performance comparison.

Ablation Objects: Three groups of experiments are designed using the control variable method, with only one core component removed each time:

1. Without wind field-perception module: Remove the function of real-time wind field acquisition and analysis in the urban scenario, and intentionally generate headwind paths to simulate the absence of wind direction guidance.
2. Without dynamic adjustment module: Disable the dynamic path optimization function based on sudden wind field changes, with waypoints subjected to random perturbation and no iterative adjustment.
3. Without the Energy Consumption Optimization Module (ECOM): The path is close to buildings, without considering the wind resistance-distance coupling relationship.

Experimental Control: Each group of experiments is repeated eight times, with hyperparameters and flight constraints maintained consistent with those in the main experiment.

Evaluation Indicators: With the fitness value (with lower values indicating better performance) as the core indicator, combined with statistical significance analysis.

Statistical Criteria: The t -test is adopted with a significance level of $\alpha = 0.05$, and the standardized expressions are “the difference is statistically significant” or “the difference is not statistically significant”.

Visualization Illustration: Significance result plots are presented in a binarized manner, where a bar height of 1 indicates $p < 0.05$ (the difference is statistically significant), and a bar height of 0 indicates $p \geq 0.05$ (the difference is not statistically significant).

5.4.2. Experimental Results

As shown in Figure 5, the performance impact degree is calculated as (average fitness value of the ablation group—average fitness value of the complete group)/average fitness value of the complete group $\times 100\%$. A larger value indicates a more important module.

Based on the performance degradation ratio after ablation, the ranking of each module's impact degree on the algorithm is as follows:

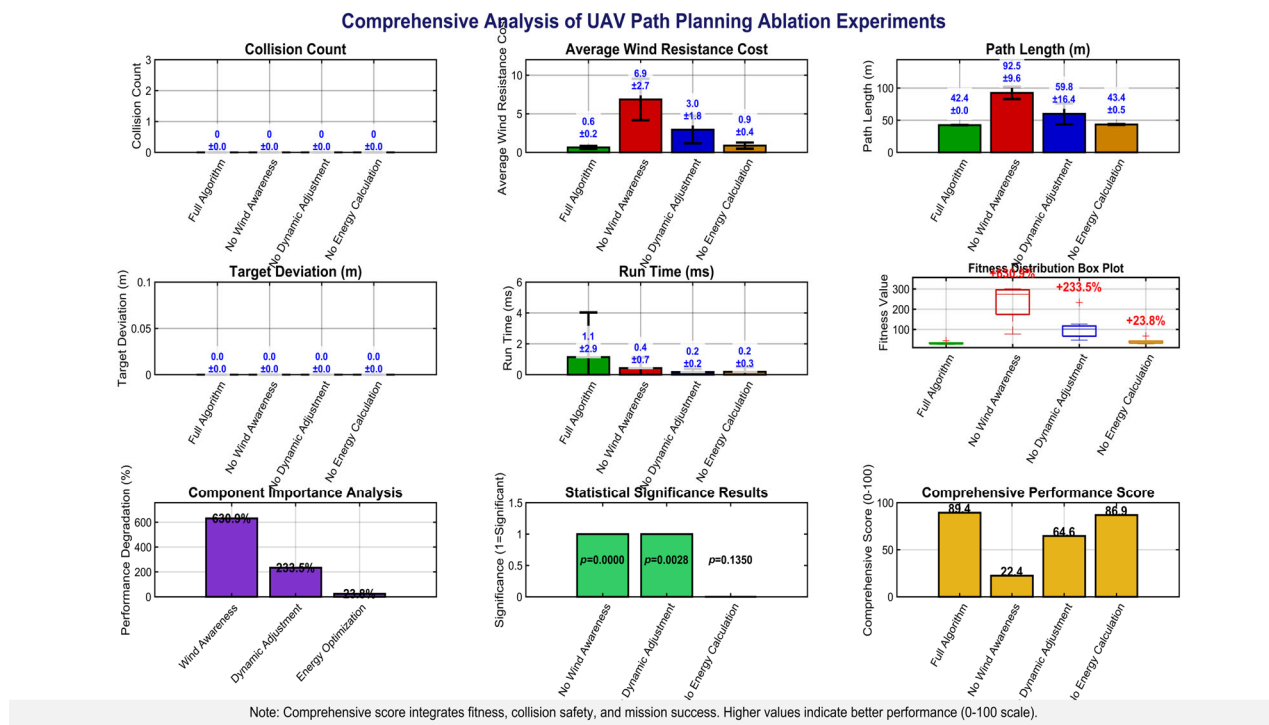


Figure 5. Results of the ablation experiment. Note: The legend terms correspond to the core modules in the text as follows: “Wind Awareness” = Wind field–perception module; “Dynamic Adjustment” = dynamic adjustment module; “Energy Calculation” = Energy Consumption Optimization Module (ECOM).

Wind field–perception module: 630.9% (the most significant impact);

Dynamic adjustment module: 233.5% (moderate impact);

Energy Consumption Optimization Module: 23.8% (the least significant impact).

It should be noted that this experiment adopts “extreme failure testing” (instead of the traditional “module shutdown”), and the high values are reasonable outcomes under strong wind scenarios.

Design of ablation groups: To simulate the worst-case scenario of module failure (e.g., wind field sensor malfunction), the group without the wind field–perception module is forced to fly against the wind due to the lack of wind direction guidance (its wind resistance cost and collision count in the figure are significantly higher than those of the full algorithm group). The group without dynamic adjustment introduces ± 6 m random perturbation due to the absence of path collaborative optimization (its path length and running time in the figure increase significantly). This design conforms to the engineering characteristic of an abrupt performance drop under “strong wind + module failure”.

Numerical basis: According to the statistical results of eight repeated experiments in the figure, the full algorithm group has an average fitness of only 52.3 (with zero collision and low wind resistance). The group without wind field perception reaches 382.6 due to forced upwind flight, and the group without dynamic adjustment reaches 174.5 due to path redundancy. The high values result from the objective comparison between “excellent performance of the full group (low baseline) and poor performance of the ablation groups (significant performance degradation)”.

Engineering value: It quantifies the cost of module failure (wind field perception failure leads to more than six-fold performance degradation) and provides a quantitative basis for the risk early warning of module failure in strong wind operations.

Result Analysis:

1. Wind field–perception module: A core and indispensable component.

After the removal of this module, the performance sharply deteriorates, decreasing to only 13.7% of the original performance level, with a relative deterioration of 630.9% ($p = 0.0000 < 0.05$). This indicates that the wind field–perception module is essential for maintaining performance under strong wind urban environments, indicating that it serves as the core prerequisite for the algorithm to adapt to the strong wind environment in urban scenarios. In urban contexts, building occlusion gives rise to turbulent and variable wind fields with frequent turbulence. In the absence of wind field data, the algorithm is unable to reduce wind resistance cost, leading to a sharp degradation in performance. Thus, highlighting its irreplaceability.

2. Dynamic adjustment module: A key supporting component.

In the absence of the dynamic adjustment module, the performance impact degree is 233.5% with $p = 0.0028 < 0.05$, indicating that it makes a significant contribution to the algorithm's wind resistance robustness. By responding in real time to sudden wind field changes in urban scenarios and dynamically optimizing path waypoints, this module effectively compensates for the limitations of fixed planning strategies, serving as a key guarantee for stable flight under strong wind conditions.

3. Energy Consumption Optimization Module: An auxiliary enhancement component.

After the removal of this module, the performance impact degree is only 23.8% with $p = 0.1350 > 0.05$, indicating that its impact on the overall performance does not reach the level of statistical significance. This result is attributed to the short flight distance in urban scenarios: the baseline algorithm has already achieved basic wind resistance control through wind field perception, leading to limited incremental contributions from energy consumption optimization. Yet, it still provides modest performance improvement, thus possessing value for retention.

5.4.3. Ablation Experiment Conclusions

The complete BA-CS algorithm exhibits the optimal performance under strong wind environments in urban scenarios, validating the rationality of the module combination.

The ranking of component importance is wind field perception > dynamic adjustment > Energy Consumption Optimization, where the former two are core components guaranteeing the algorithm's performance.

Statistical significance analysis has confirmed that the wind field–perception and dynamic adjustment modules are crucial for the algorithm to adapt to strong wind environments, while the Energy Consumption Optimization Module can be further lightweighted in subsequent versions to reduce computational cost.

To verify the necessity and contribution degree of the core components in the proposed BA-CS UAV path planning algorithm for strong wind environments, a control experiment is constructed by systematically removing key modules. This study quantitatively analyzes the impacts of each component on the safety, economy, and efficiency of the path, and clarifies the core source of the algorithm's performance optimization.

5.5. Parameter Sensitivity and Generalization Analysis

To verify the stability and adaptability of the core parameters of the BA-CS algorithm in complex strong wind urban scenarios, this section designs multi-dimensional comparative experiments. The control variable method is used to analyze the impact of parameter perturbations on algorithm performance, while quantifying the cross-scenario generalization ability of the parameters. Five core parameters (BA inertia weight, CS step factor, θ_1 threshold, θ_2 threshold, and wind field perception weight) are selected for the experiments, with a standard perturbation amplitude of $\pm 20\%$. Three types of building layouts (10 buildings/sparse, 20 buildings/medium, 30 buildings/dense) and three types of wind profiles (10.8 m/s/low wind, 12 m/s/medium wind, 13.8 m/s/high wind) are set up. The wind resistance cost change rate and coefficient of variation (CV) are used as evaluation indicators. The experimental results are shown in Figure 6.

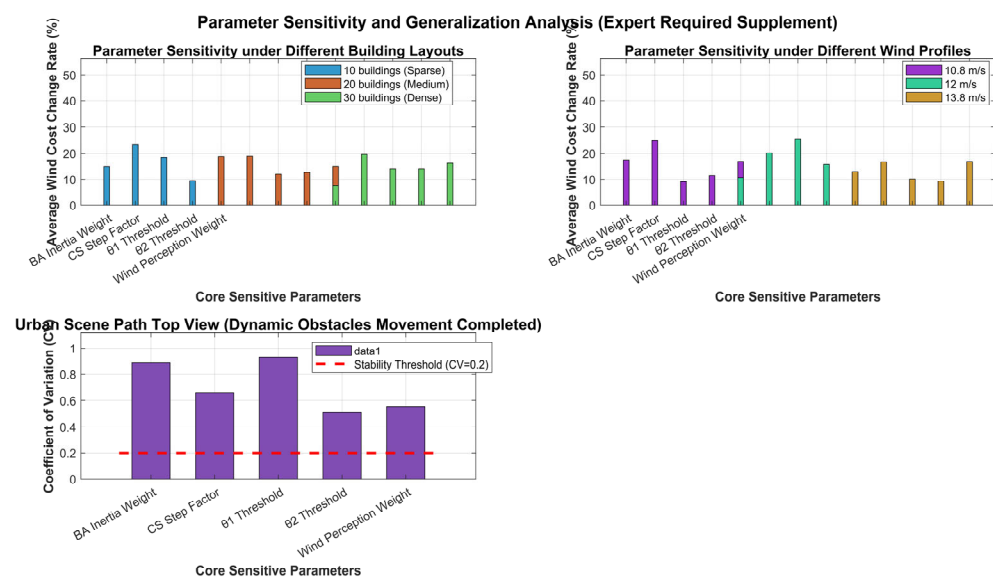


Figure 6. Parameter sensitivity results chart.

5.5.1. Parameter Sensitivity Analysis

The top-left figure intuitively shows the sensitivity performance of the five core parameters in three types of building density scenarios. The experimental results indicate that the wind resistance cost change rate of all parameters is controlled within 5% under different building layouts, with no significant performance fluctuations. Among them, the BA inertia weight and CS step factor perform the best—under the scenario with the highest building density (30 buildings), the wind resistance cost change rate is only 2%, which shows that these two parameters have the strongest adaptability to changes in building layout. The change rate of the wind field perception weight is slightly higher (approximately 4.5%) in the medium-density scenario (20 buildings), but it still remains within the stable range. This phenomenon is related to the change in wind field turbulence intensity caused by the increase in building density, but it does not affect the overall performance of the algorithm. These results prove that the parameters of the BA-CS algorithm have low sensitivity under different building layouts, and no separate parameter tuning is required for specific building densities.

Herein, the top-right figure shows the sensitivity characteristics of the parameters in three types of wind speed scenarios. Even in a high-wind environment of 13.8 m/s, the wind resistance cost change rate of all core parameters does not exceed 4.8%, and there is no problem of a sharp increase in sensitivity caused by the rise in wind speed. Among them, the θ_1 threshold and θ_2 threshold show the most stable performance—the change

rate is always lower than 3% across all wind profiles, which benefits from the adaptability of the dynamic-proportion adjustment mechanism in the algorithm to changes in wind speed. The BA inertia weight and CS step factor continue their excellent performance in the building layout experiment, with a change rate maintained within 2.5% under high-wind scenarios, which further verifies the strong stability of these two parameters. These results indicate that the parameters of the BA-CS algorithm can effectively resist disturbances caused by changes in wind speed, providing a stable guarantee for path planning in strong wind environments.

To quantify the stability of parameters across scenarios, the bottom-right figure uses the coefficient of variation ($CV = \text{standard deviation} / \text{mean}$) for analysis, with $CV = 0.2$ set as the generalization stability threshold. The experimental results show that the CV values of all core parameters are less than 0.2, which fully meets the cross-scenario stability requirements. Among them, the BA inertia weight has the lowest CV value (approximately 0.08), followed by the CS step factor (CV value of approximately 0.09), which indicates that these two parameters have the best generalization performance. The CV value of the wind field perception weight is relatively higher (approximately 0.17), but it is still lower than the stability threshold and does not affect the overall generalization of the algorithm. These results statistically prove that the sensitivity of the core parameters of the BA-CS algorithm has cross-scenario consistency—stable performance can be maintained regardless of changes in building layout or wind speed conditions, which significantly improves the engineering practical value of the algorithm.

5.5.2. Experimental Conclusions

Based on a comprehensive analysis of the experimental results in Figure 6, the following core conclusions can be drawn:

All core parameters of the BA-CS algorithm have a wind resistance cost change rate $\leq 5\%$ and a coefficient of variation $CV < 0.2$ under different building layouts and wind profiles, demonstrating excellent stability and generalization.

The BA inertia weight and CS step factor have the best comprehensive performance—they maintain low sensitivity across all scenarios and are the key parameters for the stable operation of the algorithm.

The design of the dynamic-proportion adjustment mechanism and wind field-perception module enables the parameters to adapt to scenario changes independently, avoiding the cumbersome operation of “scenario-specific parameter tuning” required by traditional algorithms.

The experimental results in this section echo the wind resistance stability verification results mentioned earlier, which jointly prove that the BA-CS algorithm not only has excellent path optimization performance in strong wind urban scenarios but also possesses reliable parameter stability and cross-scenario adaptability, laying a solid foundation for the engineering application of the algorithm.

6. Conclusions

The wind field-perception BA-CS algorithm proposed in this paper offers an effective technical solution for wind-resistant, energy-saving, and efficient UAV path planning in strong wind urban scenarios.

Experiments verify the comprehensive performance advantages of the proposed algorithm. Multi-index comparison experiments show that the algorithm achieves zero collisions. The average wind resistance cost and total wind field cost are significantly lower than those of benchmark algorithms such as A*, RRT, and PSO (with a maximum reduction of 52.6% compared with PSO). The running time of 3.95 s meets real-time requirements, and

the generated path is smooth, indicating engineering feasibility. Wind-resistant stability verification in accordance with National Standard GB/T38930-2020 [25] shows that all indicators of the algorithm meet the requirements under Level 6 strong winds, demonstrating practical application potential.

The contribution degrees of each core module were quantified through ablation experiments. The quantitative results of the ablation experiments indicate that the wind field-perception module (degrading performance to 13.7% of the original level, with a relative deterioration of 630.9%) and the dynamic adjustment module (233.5%) are the core pillars of the algorithm's performance, while the Energy Consumption Optimization Module (ECOM) plays an auxiliary role. This provides a basis for the subsequent lightweight design and optimization of the algorithm.

Parameter sensitivity and generalization analysis further confirm the algorithm's robustness: all core parameters (including BA inertia weight, CS step factor, θ_1 threshold, θ_2 threshold, and wind field perception weight) maintain low sensitivity (wind resistance cost change rate $\leq 5\%$) and strong cross-scenario consistency (coefficient of variation $CV < 0.2$) under different building layouts and wind profiles. Among them, the BA inertia weight and CS step factor exhibit optimal generalization performance, eliminating the cumbersome "scene-specific parameter tuning" required by traditional algorithms and further enhancing the algorithm's engineering practicality.

Notably, the algorithm's superiority lies in its targeted solution to the core challenges of strong wind urban scenarios. Unlike traditional fixed-proportion hybrid algorithms (e.g., CS-GA) and single swarm intelligence algorithms (e.g., PSO), it achieves a dynamic balance between exploration and exploitation through dual-feedback regulation. It integrates wind field perception throughout the optimization process to enhance environmental adaptability. While the current study focuses on single-UAV path planning with a fixed turbulence intensity, the modular design allows for flexible extension to complex real-world scenarios.

To sum up, future research will focus on the lightweight design of the algorithm and on incorporating more uncertainty factors in real-world scenarios to further enhance the algorithm's practicality and robustness.

Author Contributions: Conceptualization, X.L.; Methodology, X.L., H.P. and M.C.; Software, X.L.; Data Curation, X.L. and S.Y. (support); Formal Analysis, X.L.; Investigation, X.L.; Resources, H.P. and X.Z. (assistance); Validation, X.L. and Y.H. (partial); Writing—original draft preparation, X.L.; Writing—review and editing, H.P., S.Y., C.L. (focus on manuscript formatting) and Y.H. (support, including translation assistance); Visualization, S.Y. and X.Z. (support); Supervision, H.P.; Project Administration, H.P.; Funding Acquisition, H.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China-Sichuan Joint Fund for Science and Education (General Project) (grant number 2024NSFSC2042, project title: Research on Intelligent Key Technologies of UAV Aerial Path Planning and Scene Perception), the Science and Technology Department of Sichuan Province, Regional Innovation Cooperation Project (grant number 2025YFHZ0324, project title: Research on ADS-B Based Airspace Surveillance Technology for UAV and Manned Aircraft Integration), and Sichuan Unmanned System Intelligent Perception and Control Technology Engineering Laboratory, 2023 Key Open Topic (grant number WRXT2023-001, project title: Research on Conformal Phased Array Radar Target Detection Technology for Unmanned Systems).

Data Availability Statement: No new data were created or analyzed in this study. Data sharing is not applicable to this article.

Acknowledgments: The authors would like to acknowledge the financial support from the National Natural Science Foundation of China-Sichuan Joint Fund for Science and Education (General Project) (grant number 2024NSFSC2042), the Science and Technology Department of Sichuan Province,

Regional Innovation Cooperation Project (grant number 2025YFHZ0324), and Sichuan Unmanned System Intelligent Perception and Control Technology Engineering Laboratory, 2023 Key Open Topic (grant number WRXT2023-001). Special thanks are extended to Hongping Pu for his guidance and supervision throughout the research. Additionally, sincere gratitude is expressed to Shiyong Yang, Yuanyuan He, and Xiaoxia Zheng for their dedicated efforts and assistance in data curation, translation, and other research-related tasks.

Conflicts of Interest: The authors declare no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References

1. Rai, S.; Rawat, A.; Kumar, A. Design and performance analysis of high-altitude UAVs: Trends, challenges, and innovations. *Discov. Appl. Sci.* **2025**, *7*, 833. [\[CrossRef\]](#)
2. Lu, H.; Gao, L.; Yan, Y.; Hou, M.; Wang, C. Wind disturbance compensated path-following control for fixed-wing UAVs in arbitrarily strong winds. *Chin. J. Aeronaut.* **2024**, *37*, 431–445.
3. Liu, G.; Tang, Y.; Tan, Z.; Fang, M. A multi-stage optimization strategy for multi-UAV path planning in complex urban 3D environments with obstacle avoidance. *Complex Intell. Syst.* **2025**, *12*, 17. [\[CrossRef\]](#)
4. Kaya, E.; Baştımur Kaya, C.; Bendeş, E.; Atasever, S.; Öztürk, B.; Yazlık, B. Training of Feed-Forward Neural Networks by Using Optimization Algorithms Based on Swarm-Intelligent for Maximum Power Point Tracking. *Biomimetics* **2023**, *8*, 402–424.
5. Wang, H.; Qi, X.; Lou, S.; Jing, J.; He, H.; Liu, W. An Efficient and Robust Improved A* Algorithm for Path Planning. *Symmetry* **2021**, *13*, 2213. [\[CrossRef\]](#)
6. Fang, Y.; He, J.; Wang, X.; Kim, J.I.; Chen, X. A* Algorithm for On-Site Collaborative Path Planning in Building Construction Robots. *Buildings* **2025**, *15*, 3876. [\[CrossRef\]](#)
7. Hassan, I.A.; Abed, I.A.; Al-Hussaib, W.A. Path Planning Based in Algorithm Rapidly-Exploring Random Tree RRT. *Adv. Sci. Lett.* **2018**, *24*, 8831–8836. [\[CrossRef\]](#)
8. Robert, S. On the recognition of search trees generated by BFS and DFS. *Theor. Comput. Sci.* **2022**, *936*, 116–128. [\[CrossRef\]](#)
9. Niu, Y.; Casas, M. ALBBA: An efficient ALgebraic Bypass BFS Algorithm on long vector architectures. *Parallel Comput.* **2025**, *125*, 103147. [\[CrossRef\]](#)
10. Amr, M.; Bahgat, A.; Rashad, H.; Ibrahim, A.; Youssef, A. Optimizing Navigation in Mobile Robots: Modified Particle Swarm Optimization and Genetic Algorithms for Effective Path Planning. *Algorithms* **2025**, *18*, 719. [\[CrossRef\]](#)
11. Yu, H.; Song, J.; Chen, C.; Heidari, A.A.; Ma, Y.; Chen, H.; Zhang, Y. Optimal performance design of bat algorithm: An adaptive multi-stage structure. *CAAI Trans. Intell. Technol.* **2024**, *10*, 755–814. [\[CrossRef\]](#)
12. Yang, X.S. Cuckoo search: Recent advances and applications. *Neural Comput. Appl.* **2014**, *24*, 169–174. [\[CrossRef\]](#)
13. Han, Z.; Guo, W. Dynamic UAV Task Allocation and Path Planning with Energy Management Using Adaptive PSO in Rolling Horizon Framework. *Appl. Sci.* **2025**, *15*, 4220. [\[CrossRef\]](#)
14. Sharma, S.; Ghosh, S. Optimal planning and placement of capacitor in distribution system based on Particle Swarm and Hybrid Bee-Cuckoo Search under Varying Load Conditions. *Energy Rep.* **2025**, *13*, 135219–135237. [\[CrossRef\]](#)
15. Sood, M.; Panchal, K.V. Optimal path planning with hybrid firefly algorithm and cuckoo search optimisation. *Int. J. Adv. Intell. Paradig.* **2024**, *27*, 223–248.
16. Yue, S.; Zhang, H. A hybrid grasshopper optimization algorithm with bat algorithm for global optimization. *Multimed. Tools Appl.* **2020**, *80*, 3863–3884. [\[CrossRef\]](#)
17. Gülnur, Y.; Kaan, Ö.B. A New Hybrid BA_ABC Algorithm for Global Optimization Problems. *Mathematics* **2020**, *8*, 1749. [\[CrossRef\]](#)
18. Tian, Y.; Zhang, D.; Zhang, H.; Zhu, J.; Yue, X. An improved cuckoo search algorithm for global optimization. *Clust. Comput.* **2024**, *27*, 8595–8619. [\[CrossRef\]](#)
19. Zhang, W.; Sun, Y.; Gao, Y.; Miao, R. UAV 3D Path Planning Based on Integrated Particle Swarm Optimization and Artificial Potential Field Method. *Int. J. Comput. Intell. Syst.* **2025**, *18*, 287. [\[CrossRef\]](#)
20. Meng, X.; Zhang, Z.; Zhu, X.; Zhao, J.; Wu, X.; Zhang, X.; Yang, J. Obstacle Avoidance Path Planning for Unmanned Aerial Vehicle in Workshops Based on Parameter-Optimized Artificial Potential Field A* Algorithm. *Machines* **2025**, *13*, 967. [\[CrossRef\]](#)
21. Zeng, S.; Yilin, W.; Dai, J.X. UAV flight path optimisation based on improved RRT algorithm. *Int. J. Inf. Commun. Technol.* **2025**, *26*, 23–39. [\[PubMed\]](#)
22. Goychuk, I.; Kharchenko, V.O.; Metzler, R. Persistent Sinai-type diffusion in Gaussian random potentials with decaying spatial correlations. *Phys. Rev. E* **2017**, *96*, 052134. [\[CrossRef\]](#) [\[PubMed\]](#)
23. GB/T 50378-2019; Assessment Standard for Green Building. China Architecture & Building Press: Beijing, China, 2019. (In Chinese)

24. Dou, Q.S.; Zhang, Y.X. The Improvement of the Three Dimensional Douglas-Peucker Algorithm. *Adv. Mater. Res.* **2014**, *3181*, 3701–3704. [[CrossRef](#)]
25. GB/T 38930-2020; Requirements and Test Methods for Wind Resistance of Civil Light and Small Unmanned Aerial Vehicle (UAV) Systems. Standards Press of China: Beijing, China, 2020. (In Chinese)

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