Available Online: 1 April 2026 DOI: 10.54254/2977-3903/2026.32481

Multi-UAV fire rescue path planning based on Monte Carlo city modeling

Rui Qin**, *Hanjin Zhou

School of Transportation Science and Engineering, Civil Aviation University of China, Tianjin, China

*Corresponding Author. Email: rtan@cauc.edu.cn

Abstract. To address the issues of insufficient universality in algorithm validation and slow convergence in current path planning for multiple Unmanned Aerial Vehicles (UAVs) involved in firefighting operations in urban fire scenarios, this study proposes a path planning and simulation method that integrates a Monte Carlo simulation–based urban model with an improved A* algorithm. A generalized urban model is constructed to reduce the path length and operational time of multi-UAV firefighting missions. The spatial distribution patterns of urban clusters are analyzed, and the A* algorithm is adopted as the path search method, with the shortest time and path length defined as the objective functions. In addition, a time-window optimization mechanism is incorporated to prevent collisions. Simulations and comparative analyses are conducted under three urban scenarios: large metropolitan cities, dual-core cities, and small-scale cities. The simulation results demonstrate the efficiency of the improved A* algorithm and enhance the operational performance of multi-UAV systems in urban fire scenarios. According to the numerical results, the improved A* algorithm achieves the fastest convergence across the three urban scenarios, reducing convergence time by up to 6.89%, 5.78%, and 1.27%, respectively, compared with other algorithms.

Keywords: multi-UAV cooperation, path planning, Monte Carlo simulation, urban fire rescue, improved A* algorithm

1. Introduction

Urban fires occurring in cities characterized by dense buildings and numerous super-high-rise structures are typically complex and highly variable. Such incidents often feature rapid fire spread, large numbers of affected individuals, and significant risks during rescue operations. At present, urban firefighting mainly relies on a strategy combining personnel and fire engines, with coordinated interior and exterior suppression. However, urban areas are prone to traffic congestion, and the height limitations of ladder trucks restrict their operational capability. Consequently, this approach cannot efficiently address fires in super-high-rise commercial districts or densely populated residential areas. Unmanned Aerial Vehicles (UAVs) have already been successfully deployed in various firefighting tasks, including fire monitoring, search and rescue in fire scenes, and auxiliary fire suppression. UAVs possess numerous advantages, such as high mobility, ease of remote operation, strong adaptability to complex fire environments, and high operational precision [1]. Compared with traditional firefighting methods, UAV-based fire suppression in urban fire scenarios—particularly in super-high-rise commercial areas and densely populated residential zones—can significantly improve rescue efficiency. Moreover, compared with the use of a single UAV as an auxiliary firefighting tool, coordinated multi-UAV formations can accomplish rescue missions more efficiently. Under the complex constraints of urban fire scenarios [2, 3], how to deploy multiple UAVs for rescue operations and how to design effective path planning strategies have become major research topics.

Both domestic and international studies have conducted extensive research on UAV path planning and the application of multi-UAV formations in fire scenarios, including topics such as application scenarios, path planning, and task allocation. Guo et al. [4] proposed a task allocation scheme for multi-UAV formations. Zhang et al. [5] applied Nash optimality to UAV obstacle-avoidance research. Yang et al. [6] integrated the concept of regional partitioning with an improved search strategy for path planning. Li et al. [7-10] investigated obstacle-avoidance strategies and path-planning methods for multiple UAVs based on different theoretical frameworks. Ali et al. [11] proposed an algorithm for obtaining the global optimal solution in UAV-based firefighting scenarios. Song et al. [12-14] also discussed various approaches for applying UAVs to fire scenarios and corresponding path-planning methods. However, relatively few studies have examined UAV applications in urban environments, particularly in urban fire scenarios. Li et al. [15], in the context of urban logistics, applied the A* algorithm to UAV path planning in urban areas. Yue et al. [16] constructed a model of a small city and applied a dynamic search algorithm for UAV path planning. Zhang [17] designed a variable step-size A* algorithm to study path planning for urban logistics UAVs. These studies demonstrate that UAVs can be effectively applied to both fire-rescue and urban scenarios and have explored various approaches to solving multi-UAV path-planning problems. However, most validation experiments have been conducted in specific predefined scenarios. Since urban fire incidents may occur within cities of different structural configurations, whether these approaches remain applicable across diverse urban layouts remains an open question.

In this study, a grid-based method is used to construct an urban fire probability model, and typical urban scenarios are generated by incorporating building cluster characteristics. An improved A* algorithm, integrated with dynamic obstacle avoidance and trajectory-smoothing optimization, is employed for cooperative path planning of multiple UAVs. Random fire scenarios are generated using the Monte Carlo method for simulation validation. The results demonstrate that the proposed algorithm exhibits stronger adaptability in complex urban environments and can provide useful references for related research.

2. Modeling of the urban basic scenario

Considering the operational environment of multiple UAVs in urban fire scenarios, this study analyzes the spatial distribution patterns and mathematical characteristics of urban building clusters in accordance with fundamental urban morphology principles. On this basis, mathematical methods are employed to construct a basic urban scenario model.

2.1. Description of the basic urban scenario

In urban fire rescue operations involving multiple UAVs, once a fire is detected and reported, rotary-wing rescue UAVs are typically dispatched from a base and transported to the fire area to participate in firefighting tasks. The rescue UAVs are divided into two groups responsible for different functions: fire monitoring and fire suppression. UAVs assigned to fire monitoring enter the vicinity of the fire scene and conduct continuous surveillance while transmitting situational information. In contrast, UAVs responsible for fire suppression carry a certain number of fire-extinguishing water bombs into the fire area and release them to extinguish the flames. The urban fire rescue scenario is illustrated in Figure 1. Two UAVs perform different tasks: one is responsible for extinguishing residual flames, while the other conducts fire monitoring. Sensors enable position awareness and communication both between the UAVs and the fire scene and among the UAVs themselves.

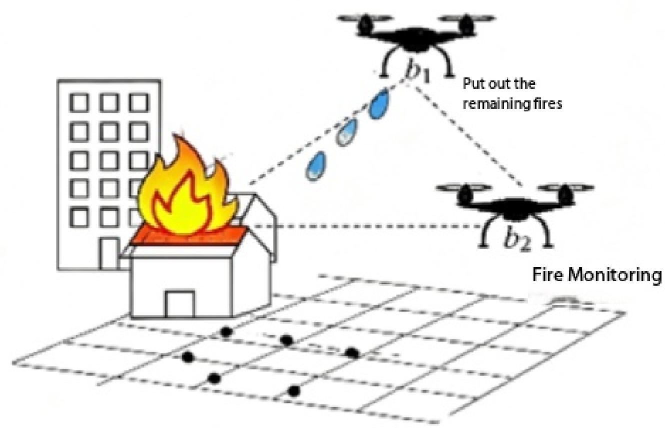


Figure 1. Multi-UAV collaborative fire rescue

2.2. Modeling of the urban low-altitude airspace

Since the aforementioned fire scenarios occur with a certain probability in complex urban environments, it is first necessary to construct a corresponding urban environmental model. In this study, a three-dimensional Digital Terrain Model (DTM) is adopted to discretize the real urban environment into grids. To ensure the continuity of UAV path planning, the center point of each grid cell is treated as a waypoint in the UAV flight path, and UAVs are assumed to travel between adjacent points at a constant and stable velocity.

To represent the static conditions of each grid cell, a three-dimensional indicator variable is introduced. This variable denotes the presence of static obstacles in the i -th airspace grid whose center is located at a given three-dimensional coordinate (x_i, y_i, z_i) . If any obstacle exists within the grid cell, or if any type of no-fly zone prevents UAV flight within that region, the indicator variable for that grid is assigned a value of 1; otherwise, it is assigned 0.

During urban fire rescue operations, multiple UAVs must operate cooperatively. At every time step, each UAV must ensure not only that its position avoids no-fly zones and obstacle regions but also that its relative distance from other UAVs remains within a safe range. Once a particular grid cell at a given time has been occupied by a previously planned flight path, that grid cell will be marked as occupied at that moment and cannot be traversed in subsequent path planning.

As illustrated in Figure 2, when planning a path from point A to point B, static obstacle grids must always be avoided. If, at a given time, the airspace between points C and D is occupied by another UAV, that region is treated as a dynamic obstacle grid during planning, and the planned path will circumvent that area.

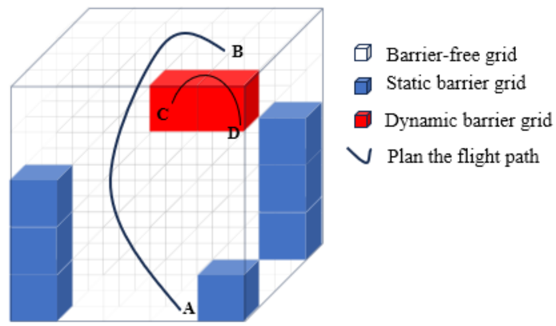


Figure 2. Low-altitude airspace modeling grid map

2.3. Modeling of urban clusters

Urban buildings typically exhibit a clustered spatial distribution. Although the overall building distribution varies considerably among cities, clusters with similar functions tend to follow certain regular patterns. In this study, ten major cities are selected as research samples. By comparing cluster characteristics both within the same city and across different cities, and by conducting parametric analysis of the urban spatial structure, a mathematical model is derived. Through analysis and simulation, urban clusters are categorized into commercial center clusters and residential center clusters. These two categories follow different discrete distribution models while also exhibiting overlapping spatial patterns. When such overlaps occur, buildings in these areas tend to exhibit relatively greater heights. Building heights in commercial districts are assumed to follow a lognormal distribution with parameters (4.5,0.6). The distribution is expressed as:

$$p\left(h\right)=\frac{1}{4.5\sqrt{2\pi}h}exp\left(-\frac{\left(lnh-0.6\right)^{2}}{2\times4.5^{2}}\right),h\in\left(0,h_{max}\right) \tag{1}$$

where h_{max} denotes the maximum building height specified for the region in the model. In residential areas, building heights typically exhibit smaller variation and often appear in settlement-like clusters. A Weibull distribution is therefore adopted for clustering analysis, expressed as:

$$F\left(h;\lambda,k,h_{min}\right)=1-exp\left[-\left(\frac{h-h_{min}}{\lambda}\right)^{k}\right] \tag{2}$$

The parameter values λ,k,h_{min} corresponding to different residential building types are shown in Table 1.

Table 1. Residential value table

Residential Type	$\lambda\,(m)$	k	h_{min}
High-rise residential buildings	50 ± 5	2.3 ± 0.2	30
Single-story houses	25 ± 3	1.7 ± 0.2	10
Villa districts	10 ± 2	3.2 ± 0.3	8

3. Modeling of UAV urban firefighting scenarios

The characteristics of urban environments determine certain patterns in the occurrence and spread of fires. Based on this, UAV-based urban firefighting scenarios are constructed by integrating fire probability with the urban model. This yields a mathematical model of urban fire scenarios along with the associated objective functions and constraints.

3.1. Fire probability in urban clusters

Urban fires typically exhibit distinct spatial distribution patterns. Fire scenarios that require coordinated UAV intervention are mainly concentrated in two high-risk areas: Fires in high-rise buildings within commercial districts, where building heights exceed 80 meters, surpassing the reach of ladder trucks. Fires in residential areas with dense spatial layouts and complex environments. Accordingly, a Probability Density Function (PDF) for fire distribution is established. By comprehensively considering factors influencing fire likelihood and existing distribution formulas, an urban fire risk model is formulated. The probability that a fire occurs at a given point (x, y, z) during a specific time period can be expressed as:

$$P_{ign}\,(x, y, z, t) = P_{base}\,(x, y, t) f_h\,(z) \tag{3}$$

where $P_{\text{base}}\,(x, y, t)$ is the baseline probability density function representing the intrinsic risk of a fire occurring at point x at time t, calculated as:

$$P_{base}\left(x, y, t\right) = \frac{\alpha B(x,y) + \beta N(x,y)}{\int_{\Omega} \alpha B(x,y) + \beta N\left(x,y\right) d_x d_y} \tag{4}$$

Here, $N\,(x, y)$ is the cluster fire occurrence density, and represents the local population density. $f_h\,(z)$ is a height adjustment function reflecting the additional risk posed to rescue operations by building height and structural complexity. Its calculation is given by:

$$f_h\,(z) = \begin{cases} exp\left[-\frac{(z-100)^2}{200}\right]; z \geq 80 \\ \frac{1}{1+0.2(z-15)^2}\,; z < 80 \\ 1; others \end{cases} \tag{5}$$

3.2. Objective functions

The overall objective is to minimize both the total flight distance of all UAVs and the time taken for the last UAV to reach the fire scene, thereby optimizing the rescue efficiency. Based on this goal, the objective functions are constructed as follows.

3.2.1. Path length function $L(x_i, y_i, z_i)$

Considering UAV deployment in a fire scenario, the origin of the grid (x_0, y_0, z_0) is set as the takeoff point for all firefighting UAVs, serving as the starting point for every UAV path. The fire location is denoted as (x_n, y_n, z_n) . The waypoints along the UAV path, defined as the centers of traversed grid cells, are denoted as (x_i, y_i, z_i) . By summing the lengths of each segment along the UAV's path, the total path length for a single UAV can be expressed as:

$$L\left(x_i, y_i, z_i\right) = \sum_{\text{i=1}}^{j} \sqrt{\left(x_i - x_{\text{i-1}}\right)^2 + \left(y_i - y_{\text{i-1}}\right)^2 + \left(z_i - z_{\text{i-1}}\right)^2} \tag{6}$$

By recording the path length function for each drone and summing these values, the total path length can be obtained. This objective function is denoted as:

$$C_1\left(x\right) = min \sum_{\text{i=1}}^{n} L\left(x_i, y_i, z_i\right) \tag{7}$$

3.2.2. Time function $T(x_i, y_i, z_i)$

Arrival times of different UAVs at the fire scene may vary. Setting fixed time intervals between UAVs can prevent collisions and streamline flight paths, though this may not always yield the minimum time. To improve rescue efficiency, the time taken by the last UAV to reach the fire scene is denoted as $T\left(x_i, y_i, z_i\right)$, expressed as:

$$T\left(x_i, y_i, z_i\right) = \sum_{\text{i=1}}^{j} \frac{L(x_i, y_i, z_i) - L(x_{i-1}, y_{i-1}, z_{i-1})}{v_i} + t \tag{8}$$

$$v_i = \begin{cases} v_p; z_i = z_{\text{i-1}} \\ v_s; z_i < z_{\text{i-1}} \\ v_j; z_i > z_{\text{i-1}} \end{cases} \tag{9}$$

where t accounts for total time losses due to speed adjustments, obstacle avoidance, and other operational delays $v_p; v_s; v_j$ represents cruising, ascent, and descent speeds along the UAV flight path. This function is denoted as:

$$C_2\left(x\right) = minT(x_i, y_i, z_i)_{\max} \tag{10}$$

3.3. Constraints

UAVs are subject to upper and lower flight altitude restrictions:

$$H_{min} \leq z_i \leq H_{max}, \forall_i \tag{11}$$

During flight, UAVs must maintain a safe distance from obstacles and other UAVs:

$$\sqrt{\left(x_i - x_1\right)^2 + \left(y_i - y_1\right)^2 + \left(z_i - z_1\right)^2} \geq D_{\min}, \forall_i \tag{12}$$

where (x_1, y_1, z_1) is the coordinate of the nearest obstacle and $D_{\min}$ is the minimum safe separation distance.

The UAV flight program imposes strict constraints on pitch angles, expressed as: $|\theta_{i,n,k}| \leq \theta_{max}$,The maximum pitch angle θ_{max} is set here. The calculation formula can be written as:

$$\theta_{i,n,k} = arctan\left(\frac{z_{i,n,k+1} - z_{i,n,k}}{\sqrt{\left(x_{i,n,k+1} - x_{i,n,k}\right)^2 + \left(y_{i,n,k+1} - y_{i,n,k}\right)^2}}\right) \tag{13}$$

For a path segment ($G_{i,n,k}$, $G_{i,n,k+1}$) of UAV i along the n -th sub-path, the pitch angle is denoted as $\theta_{i,n,k}$.

Similarly, horizontal turning angles (yaw) are constrained: $|\omega_{i,n,k}| \leq \omega_{max}$,The maximum pitch angle ω_{max} is set here. The calculation formula can be written as:

$$\omega_{i,n,k} = arccos\left(\frac{\boldsymbol{G_{i,n,k+1}} - \boldsymbol{G_{i,n,k}}}{|\boldsymbol{G_{i,n,k}}||\boldsymbol{G_{i,n,k+1}}|}\right) \tag{14}$$

In the formula: $\omega_{i,n,k}$ denote the horizontal turning angles of UAV i on the nth sub-path segment ($G_{i,n,k}$, $G_{i,n,k+1}$); ω_{max} represents the maximum turning angle constraint.

4. Algorithm design

To address the multi-UAV rescue path planning problem, an improved A* algorithm is employed. This algorithm integrates both path-smoothing and obstacle-avoidance techniques into the standard A* framework, enabling faster computation of optimal paths for multiple UAVs operating in coordination.

4.1. Path optimization algorithm design

4.1.1. Principle of the A algorithm*

The A* algorithm is a widely used method for path planning. It is based on Dijkstra's algorithm, enhanced by the introduction of heuristic and evaluation functions to estimate the total cost. In the present model, the heuristic function evaluates each adjacent grid cell center relative to the current UAV location, selecting the most favorable next step. This process is repeated iteratively until the UAV reaches the destination. The evaluation function can be expressed as:

$$f(x) = g(x) + h(x) \tag{15}$$

where $g\left(x\right)$ is the actual cost, representing the accumulated distance and time from the start point to the current waypoint along the computed path. $h\left(x\right)$ is the heuristic function, representing the estimated cost from the current waypoint to the fire location.

4.1.2. Improved A algorithm design*

In this study, the problem includes a three-dimensional spatial variable and a time variable. Using the standard grid-based computation, each grid has many adjacent neighbors, resulting in slow computation. To improve efficiency, the algorithm exploits UAV directionality and height stability, simplifying the search process. When selecting adjacent grids, both shortest path and minimum time objectives are considered. The improved evaluation function is formulated as:

$$f(x) = a\left[C_1^m\left(x\right) + C_1^n\left(x\right)\right] + b\left[C_2^m\left(x\right) + C_2^n\left(x\right)\right] \tag{16}$$

Among these, $C_1^m\left(x\right)$ represents the actual shortest path cost, $C_1^n\left(x\right)$ estimated as the path cost to reach the endpoint. $C_2^m\left(x\right)$ denotes the time required for the last drone to arrive at the current point. $C_2^n\left(x\right)$ signifies the time needed for the last drone to reach the endpoint along the currently planned path. Parameters a and b are proportional coefficients that decrease and increase in direct proportion to time, respectively. The program performs $\left(x_i, y_i, z_i\right)$ forward calculation, ultimately traversing every intermediate point to reach the endpoint, outputs the values, and completes path planning. This yields the minimum value for the path, representing the globally optimal solution.

4.2. Path smoothing optimization

Paths obtained through the grid method are typically polyline-shaped, which does not reflect actual UAV flight behavior and can affect the objective function. Therefore, path smoothing is required. The spline interpolation method inserts curves between consecutive waypoints to replace straight segments. Among these, B-spline interpolation is commonly used for UAV path planning. By selecting an appropriate number of control points, the path can be smoothed; the longer the path, the better the smoothing effect. In this study, several key points along the previously computed optimal path are selected as control points for cubic B-spline interpolation.

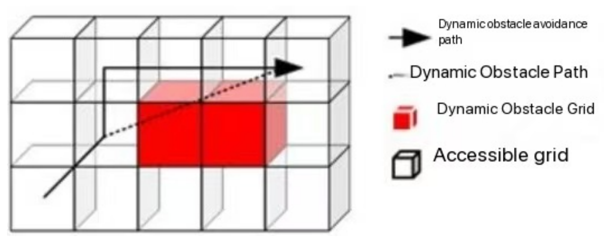
The matrix formulation for the cubic B-spline is:

$$\boldsymbol{C}(x) = \frac{1}{6}\begin{bmatrix}1 & x & x^2 & x^3\end{bmatrix}\begin{bmatrix}1 & 4 & 1 & 0\\ -3 & 0 & 3 & 0\\ 3 & -6 & 3 & 0\\ -1 & 3 & -3 & 1\end{bmatrix}\begin{bmatrix}L_0\\ L_1\\ L_2\\ L_3\end{bmatrix} \tag{17}$$

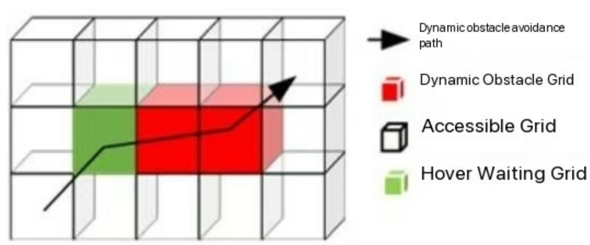
where L_0~L_3 represent four consecutive segments of the original path, divided by three control points (x_0, y_0, z_0) ; (x_1, y_1, z_1) ; (x_2, y_2, z_2) . Connecting all interpolated segments yields the smoothed, improved path.

4.3. Time-window obstacle avoidance optimization

To prevent collisions, this study adopts two strategies: hover-waiting and altitude adjustment. For each detected conflict point, the appropriate avoidance method is selected, generating a new path. After path adjustment, all grid points along the path are marked as temporary static obstacles, and the environment status is updated. Through this dynamic obstacle handling mechanism, the algorithm iteratively eliminates all path conflicts, completing the embedded algorithm solving process. The principles of dynamic obstacle avoidance are illustrated in Figure 3:



(1) Altitude adjustment avoidance



(2) Hovering avoidance

Figure 3. Dynamic obstacle avoidance method diagram

5. Simulation experiments

This section constructs a simulation environment based on Monte Carlo random simulation, incorporating parameters such as urban building density and height distribution, and validates the proposed improved algorithm through testing.

5.1. Introduction to interactive simulation design

The simulation process uses an interactive algorithm. Grid data of urban scenarios generated via Monte Carlo simulation are input into the path planning algorithm, which integrates both path smoothing and obstacle avoidance modules.

5.2. Urban environment modeling and background settings

5.2.1. Simulation method

Based on the previously established urban cluster and fire probability models, three typical urban forms are selected as simulation environments: small cities, dual-center cities, and large cities. Monte Carlo random sampling is used to generate corresponding urban models. The points within urban clusters generated by the Monte Carlo method are imported in tabular form. Grids with a length of 5 meters are used to read grid states, and fire rescue operations are simulated over a 7-day period. Each simulation deploys one UAV carrying water bombs for firefighting and one UAV for reconnaissance, with experiments repeated 20 times.

5.2.2. Simulation process

The improved A* algorithm is implemented in Python 3.10. Its results are compared with three commonly used path planning algorithms: traditional A*, RRT* (Rapidly-Exploring Random Tree), and APF (Artificial Potential Field).

To address the needs of fire detection and rescue operations, this study employs the DJI Matrice 300 RTK model as the simulation drone. Its key specifications are as follows: maximum range of 15 km; endurance of 55 minutes; maximum cruise speed of 23 m/s; maximum ascent rate of 6 m/s; maximum descent rate of 5 m/s; maximum pitch angle of 30 degrees; and dimensions of (810 × 670 × 430) mm.

5.2.3. Monte Carlo generation of urban fire scenarios

Based on the urban fire model constructed previously, the urban area is defined according to Table 1 data and distribution characteristics for the three city types. Fire points are randomly generated within areas suitable for multi-UAV rescue operations. Each scenario simulates a 7-day period, during which all fire events are addressed by multiple UAVs. The number of fire events studied is shown in Figure 4.

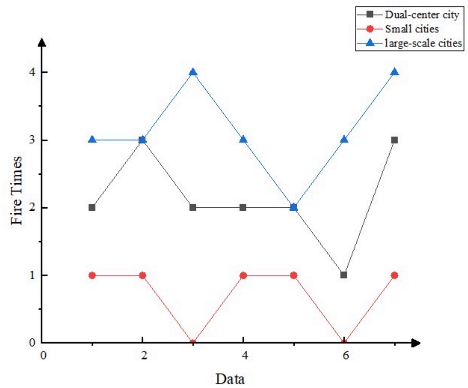


Figure 4. Number of fires studied

5.3. Simulation results analysis

5.3.1. Iteration results analysis

To verify the robustness of different algorithms, solution parameters are fixed, and the number of iterations required to reach the objective function is recorded (Figure 5). A* and improved A* reach optimal solutions through node expansion. RRT* and APF converge to stable solutions as iterations increase. The improved A* algorithm requires fewer iterations to reach the optimal solution, demonstrating superior convergence efficiency.

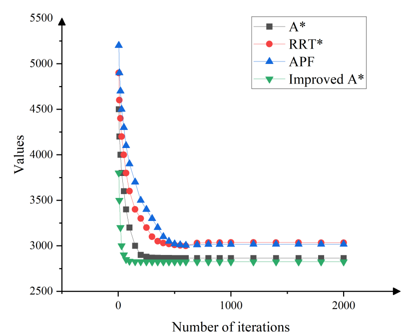


Figure 5. Graph of iteration times for algorithm solution

5.3.2. Analysis of total cost

Using the Monte Carlo-generated fire scenarios, each fire in the three urban types is simulated for UAV deployment and firefighting. Path planning and UAV assignment follow the interactive algorithm design from Figure 5. The total cost is calculated based on time cost and distance cost, normalized across experiments. Box plots of the total cost for 20 repetitions are shown in Figure 6. The improved A* algorithm achieves the lowest total cost. Combined with the iteration results, this indicates that the improved A* algorithm provides superior path planning performance (see Table 2).

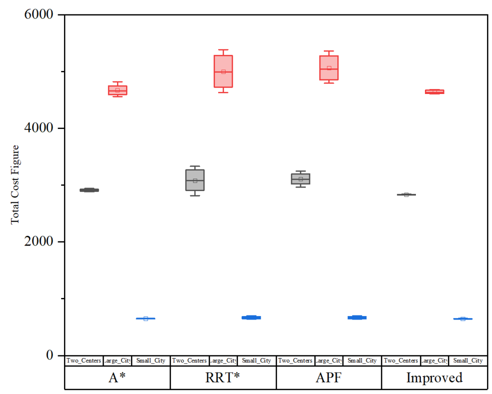


Figure 6. Total cost box chart

Table 2. Cost solution table

Algorithm	Dual-Center City	Large City	Small City
Traditional A*	2865.11	4624.3	645.13
RRT*	3033.56	4871.69	651.05
APF	3016.62	4892.15	653.08
Improved A*	2824.67	4609.42	644.8

6. Conclusion

This study addresses the multi-UAV fire rescue path planning problem in urban scenarios. While adhering to standard operational constraints for multiple UAVs, it takes into account urban layout and fire characteristics, and introduces an innovative approach to generate fire scenarios using the Monte Carlo method within urban environments. The traditional A* algorithm is improved and embedded with path smoothing and dynamic obstacle avoidance. Simulation results across large cities, dual-center cities, and small-scale cities demonstrate that the proposed method outperforms conventional algorithms in terms of shorter flight time, reduced path length, and faster convergence. Overall, the improved A* algorithm provides superior path planning performance compared with other methods. For future research, integrating intelligent algorithms into the A heuristic function* could further enhance path planning efficiency and yield even better results.

References

[1] Ding, M., Xia, X., & Zou, Y. (2023). A UAV 3-D trajectory planning method based on an improved butterfly optimization algorithm. *Journal of Nanjing University of Aeronautics and Astronautics, 55*(5), 851–858.

[2] Wang, Z., Wang, L., & Xu, H. (2025). Three-dimensional UAV path planning based on a multi-strategy improved dung beetle optimization algorithm. *Journal of Nanjing University of Aeronautics and Astronautics (Natural Sciences Edition), 57*(3), 475–486.

[3] Jayapandian, N. (2019). Cloud-enabled smart firefighting drone using Internet of Things. In *2019 International Conference on Smart Systems and Inventive Technology* (ICSSIT) (pp. 1079–1083). IEEE.

[4] Guo, Y., Wang, Y., & Zhou, Y. (2023). Joint optimization strategy for multi-UAV cooperative computing and resource allocation in edge networks. *Journal of Nanjing University of Aeronautics and Astronautics, 55*(5),

757–767.

[5] Zhang, H., Fu, M., & Gong, R. (2017). Cooperative collision avoidance control technology for multiple UAVs based on distributed model predictive control. *Journal of Nanjing University of Aeronautics and Astronautics, 49*(S1), 96–102.

[6] Yang, H., Li, A., & Guo, Y. (2024). UAV planar search research based on dynamic sensors. *Journal of Nanjing University of Aeronautics and Astronautics, 56*(6), 1074–1081.

[7] Li, Y. (2024). *Design of a UAV swarm trajectory simulation system based on the Dubins algorithm* [Master's thesis, Xi'an Technological University].

[8] Bailon-Ruiz, R., Bit-Monnot, A., & Lacroix, S. (2022). Real-time wildfire monitoring with a fleet of UAVs. *Robotics and Autonomous Systems, 152*, 104071.

[9] Yang, X., Wang, C., & Zhang, Y. (2023). UAV path planning based on incremental developmental deep reinforcement learning. *Flight Dynamics, 41*(3), 40–46.

[10] Feng, G., Zhao, X., & Gao, G. (2018). UAV route planning based on A* ant colony algorithm. *Flight Dynamics, 36*(5), 49–52.

[11] Ali, H., Xiong, G., Haider, M. H., Tamir, T. S., Dong, X., & Shen, Z. (2023). Feature selection-based decision model for UAV path planning on rough terrains. *Expert Systems with Applications, 232*, 120713.

[12] Song, R., Zhu, P., & Zhang, J. (2024). UAV path planning for forest fire suppression in plateau regions. *Science Technology and Engineering, 34*, 14863–14870.

[13] Ji, L., Chen, C., & Chen, H. (2025). UAV three-dimensional trajectory planning based on an improved dung beetle optimization algorithm. *Acta Armamentarii, 46*(9), 15–28.

[14] Aggarwal, S., Budhiraja, I., & Garg, S. (2025). A blockchain-based secure path planning in UAV communication networks. *Alexandria Engineering Journal, 113*, 451–460.

[15] Li, Y., & Zhao, R. (2024). Multi-objective UAV path planning in complex urban environments. *Journal of Nanjing University of Aeronautics and Astronautics, 56*(6), 1002–1012.

[16] Yue, R., & Hou, B. (2025). Urban logistics UAV route planning based on an improved artificial fish swarm algorithm. *Journal of Civil Aviation University of China, 43*(1), 89–96.

[17] Zhang, L. (2022). *Research on trajectory planning technology for multiple logistics UAVs in urban low-altitude environments* [Doctoral dissertation, Nanjing University of Aeronautics and Astronautics].