An Optimized UAV Flight Path Planning Method Based on Urban Low-Altitude Navigation Knowledge Graph

Ying Kong[1], Xian Guo[1]

[1] School of Geomatics and Urban Spatial Informatics, Beijing University of Civil Engineering and Architecture
Beijing 102616, China - kongying2025@163.com, guoxian@bucea.edu.cn

Keywords: Urban Low-Altitude Scenario; Knowledge graph; Spatial-Temporal Data Model ; Path Planning ;A* algorithm.

Abstract

With the rapid growth of the low-altitude economy, path planning for Unmanned Aerial Vehicles (UAVs) in complex urban low-altitude environments has become increasingly critical. However, urban low-altitude scenarios are influenced by buildings, meteorological conditions, regulatory restrictions and numerous factors. Traditional path planning methods struggle to effectively consider the impact of multiple constraints, making it challenging to provide effective and interpretable decision support for flight operations. This study proposes an optimized UAV flight path planning method based on an Urban Low-Altitude Navigation Knowledge Graph (ULAN-KG). Utilizing the knowledge graph, it structures the association between low-altitude flight route elements and low-altitude flight constraint factors in the urban space. The experiment selects a densely built area of Beijing for validation, where the proposed method is compared with traditional algorithms. The experimental results show that the A* algorithm improved by ULAN-KG can effectively avoid flight segments affected by strong wind conditions. When conflicting with controlled airspace events, the path planning results prioritize avoiding no-fly zones. This approach offers efficient and reliable technical support for UAV applications in complex urban low-altitude scenarios, such as logistics and emergency response.

1. Introduction

The rapid development of unmanned aerial vehicle (UAV) technology has significantly expanded its applications in urban scenarios, including logistics delivery, emergency response, and infrastructure inspection. Compared to traditional ground transport, UAV operations at low altitudes offer distinct advantages in resource efficiency and transportation flexibility, enabling UAVs to bypass ground-level constraints and reduce travel time and costs.

Aiming to identify optimal paths between origin and destination that fulfil mission objectives such as minimal distance, maximum safety, or shortest flight time, path planning is essential for UAV autonomous flight(Meister et al., 2008; Sun et al., 2016). However, precise navigation in low-altitude urban environments requires efficient computational methods. The complexity of urban spaces (e.g., terrain, buildings, irregularly distributed obstacles, varying meteorological conditions, and stringent airspace regulations) creates substantial challenges(Du et al., 2024). Additionally, low-altitude flights demand rapid responsiveness and high safety standards, necessitating real-time environmental awareness and precise computational methods for path planning(Liao et al., 2023; Zhang, 2021).

Traditional path planning algorithms face limitations in addressing these urban complexities. Conventional heuristic methods, such as A*(Kong et al., 2020), primarily rely on geometric distances, limiting their ability to integrate dynamic environmental factors like building structures, weather conditions, and airspace rules effectively (Khoufi et al., 2019; Yu et al., 2021). In addition, these algorithms experience scalability issues within dense urban areas, where an increasing number of obstacles and varying constraints significantly impact computational efficiency and real-time responsiveness(Du et al., 2024).

Knowledge Graphs(Chen et al., 2024) have emerged as a viable solution by integrating heterogeneous data through semantic networks. This unified representation supports effective querying and fusion of diverse information. Hence, this study introduces the Urban Low-Altitude Navigation Knowledge Graph (ULAN-KG), designed to comprehensively represent the spatiotemporal and semantic relationships among various urban elements influencing low-altitude UAV flight. By providing flexible, adaptive, and rapid data processing, the proposed ULAN-KG can potentially enable precise modeling of UAV navigation scenarios under multiple constraints.

By integrating environmental risk indicators from ULAN-KG into the classic A* algorithm’s cost function, path planning now extends beyond geometric distances, explicitly considering environmental constraints such as urban structures, weather conditions (e.g., wind strength, visibility), and dynamic events (e.g., temporary no-fly zones). The proposed method quantifies environmental impacts and adjusts weight coefficients within the cost function, enabling optimization toward specific goals such as the shortest or safest path. This method’s effectiveness is demonstrated through a case study conducted in a densely built area of Beijing, where obstacle avoidance, weather adaptation functions, and regulatory restrictions are integrated. A comparison with traditional methods verifies its practical significance and effectiveness under real-world constraints. Consequently, this optimization enhances computational efficiency, reduces algorithm complexity, and improves interpretability, providing robust technical support for UAV operations in complex urban low-altitude environments.

https://doi.org/10.5194/isprs-archives-XLVIII-4-W14-2025-97-2025

2. Methodology

2.1 Ontology Modeling for Urban Low-Altitude Navigation Scenarios

Urban low-altitude scenarios involve various spatial locations and dynamic and static influencing factors. The complexity of the environment presents challenges for UAV low-altitude navigation. Therefore, a spatiotemporal model is needed to integrate these elements, providing support for accurate and efficient path planning. This study constructs a data model for low-altitude navigation, unifying the modeling and storage of spatial topological relationships, as well as the physical constraints and rules in low-altitude scenarios. Referring to relevant literature(Garrow et al., 2021), The urban low-altitude public airways can effectively reduce flight conflicts under high traffic systems, improving the safety and efficiency of UAV flights. Therefore, the model is built around flight route elements, using a top-down approach to define low-altitude flight elements and influencing factors. These elements, their relationships, and rules are abstracted, and an ontology model for urban low-altitude navigation scenarios is developed. To enable unified data management and efficient utilization, Neo4j graph database is used for data storage and management, providing support for flexible and efficient queries in subsequent stages.

2.1.1 Semantic Element Classification and Definition: Based on relevant standards and specifications, the concepts, classification, attributes, and relationships of elements are defined. As shown in Figure 1, the model consists of three types of node elements (environmental elements, functional elements, and flight path elements) and the relationships between these elements (spatiotemporal relationships and semantic relationships).

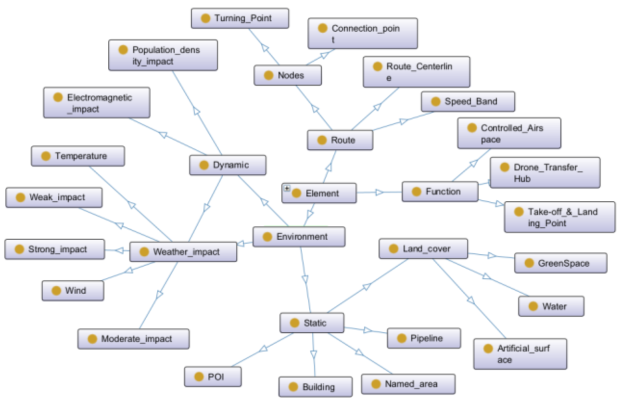


Figure 1. Urban Low-Altitude Navigation Scenarios Conceptual Model.

-Low-altitude environmental elements are the facilities and environments in the low-altitude airspace and on the ground that influence flight, where facility environment mainly includes buildings, roads, and associated infrastructure. The scene environment primarily includes points of interest, land cover and land use, named regions, terrain and landforms, meteorological conditions, electromagnetic environment, population density, etc.

-Low-altitude functional elements are elements related to low-altitude flight management and services, including take-off & landing points, flight control airspace, and connecting facilities. Functional elements are physically attached to environmental elements.

-Low-altitude flight path elements are the elements that define the linear airspace in which low-altitude aircraft can fly in three-dimensional space, including route centerlines, turning points, connection points, and velocity layers. The flight path connects with facility elements, scene elements, functional elements, and ground traffic elements through connection points.

2.1.2 Property Constraints and Semantic Relationships:In the proposed model, relationships serve as critical connectors that bind discrete entities and their properties into an integrated framework, enabling coherent representation and efficient querying of complex urban low-altitude scenarios. Unlike conventional models that prioritize static geometric relationships, this geographic model emphasizes dynamic spatiotemporal interactions while accommodating semantic associations. Given the inherent volatility of low-altitude environments — where entities undergo frequent spatial transformations, temporal state changes, and contextual meaning shifts—the relationship taxonomy is systematically divided into three categories: spatial, temporal, and semantic relationships.

Spatial relationships form the geometric foundation of the model, capturing three fundamental dimensions of entity interaction: topological connections (intersecting/adjacent/disjoint relationships), distance metrics (Euclidean/network-based proximity), and directional orientations (azimuthal bearings/altitude differentials). These relationships enable precise modeling of physical interactions, such as the topological adjacency between connection points and points of interest (POIs), or the directional alignment of turning points along flight route centerlines. The model employs RCC-8 calculus for continuous spatial reasoning, enhanced with UAV-specific constraints like minimum vertical clearance thresholds.

Temporal relationships introduce the dynamic dimension, distinguishing between absolute and relative time references. Absolute time relationships anchor events to specific calendar instances (e.g., "NO_FLY_ZONE_ACTIVE from 2025-06-18T09:00 to 2025-06-18T18:00"), providing unambiguous temporal boundaries for regulatory constraints. Relative time relationships express duration-based associations (e.g., "WIND_SHEAR_EVENT persists for 15 minutes after detection") or periodic patterns (e.g., "WEEKLY_AIRSPACE_RESTRICTION every Monday 08:00-10:00"). These temporal constructs are formalized using Allen's interval algebra extended with metric temporal logic operators.

Semantic relationships bridge the gap between raw data and contextual meaning, encompassing attribute associations (e.g., "Building_47 hasHeight 52m"), causal dependencies (e.g., "HEAVY_RAIN causes REDUCED_VISIBILITY"), and cognitive relationships (e.g., "HOSPITAL_HElipad isPreferredLandingZone"). The model leverages OWL 2 DL ontologies to formalize these semantic associations, enabling rule-based reasoning such as inferring flight path validity based on building height thresholds or prioritizing medical delivery routes during emergencies.

Under the three core relationship frameworks of space, time, and semantics mentioned above, the model further defines the secondary relationship system of low altitude scene elements (as shown in Figure 2), and introduces constraint attribute parameters to achieve dynamic environment perception. These secondary relationships include but are not limited to: facility

https://doi.org/10.5194/isprs-archives-XLVIII-4-W14-2025-97-2025 | © Author(s) 2025. CC BY 4.0 License.

attachment relationships (such as physical binding between functional elements and buildings), path connection relationships (topological associations between flight routes and turning points), event triggering relationships (causal linkage between meteorological changes and control areas), etc., which achieve fine-grained modeling of complex scenes through a multi-level relationship network. At the same time, the model integrates real-time constraint attribute data collected by sensors, including key parameters such as three-dimensional coordinates (X/Y/Z), path length (m), wind speed level, electromagnetic strength, etc. These data are continuously updated through a dynamic threshold detection mechanism to provide real-time environmental status feedback for path planning algorithms. For example, when the wind sensor detects that the instantaneous wind speed of a certain segment exceeds the preset threshold, the system will automatically trigger the path replanning mechanism and quickly query alternative routes through the knowledge graph. This collaborative design of relationship modeling and attribute constraints enables the model to both perform long-term planning through semantic relationship inference and achieve short-term obstacle avoidance based on real-time data, thus constructing a full cycle navigation support system from the strategic layer to the tactical layer (see Figure 2 for specific classification).

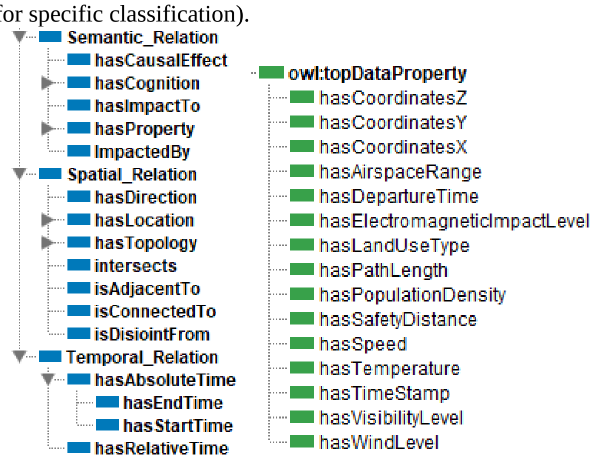


Figure 2. Object Properties and Dara Properties.

2.1.3 Knowledge Graph Construction：Based on the ontology model, the knowledge graph is constructed in a top-down manner. Terrain data, building data, and meteorological data obtained from sensors are processed into triplet data containing environmental information, which serves as the basic unit for knowledge graph modeling. These data are imported into the Neo4j database. The constructed "Urban Low-Altitude Navigation Knowledge Graph" (ULAN-KG) provides support for visualization and parameter queries during path computation.

The relationships between nodes and edges in the knowledge graph are shown in Figure.3:

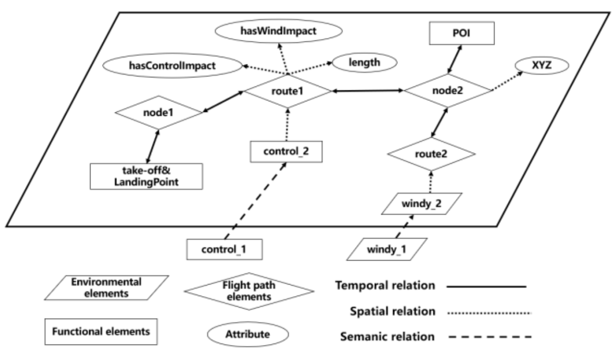


Figure3. Node and Relationship Diagram.

The knowledge contained in each slice includes:

Nodes: The ULAN-KG node contains five different elements: Environmental elements, Functional elements, and Flight path elements. Each type of node carries attributes.Environmental elements include wind, visibility, and electromagnetic environments.Functional elements include take-off & landing points, Points of Interest (POI), etc.Flight path elements are the core elements for path planning, including nodes (Turning Point and Connection Point) and the Route Centerline connecting two nodes. The Flight path elements can be represented by Eq.1:

$$Route = \{ Route_{name}, Route_{class}, Route_{attributes} \} \quad (1)$$

$Route_{name}$ is the unique identifier for the route, $Route_{class}$ is the category of the route, and $Route_{attributes}$ is the set of attributes related to the route, which can include length, wind impact, control impact, etc. Specific attributes can be expanded as Eq.2:

$$Route_{attributes} = \{ length,\ hasWeatherImpact, hasControlImpact \} \quad (2)$$

Length represents the length of the route,hasWindImpact indicates whether the route is affected by wind,hasControlImpact indicates whether the route is subject to airspace control.These attribute details describe the state of the route, serving as important parameters in the cost function for path planning.

Edges: Edges in the knowledge graph are designed to reflect temporal relations, spatial relations, and semantic relations.Temporal relations are represented by long dashed lines, such as windy_1 to windy_2, indicating the change in wind between two time points.Spatial relations are represented by solid arrows between routes and nodes, showing the spatial connection between routes and nodes, which can form the road network structure for path planning.Semantic relations are represented by short dashed lines, such as control_1 connected to route1, indicating that the route is subject to control impact; node2 connected to attribute XYZ represents the coordinates of the point, used in path planning for heuristic function calculations.

2.2 Optimized A* Path Planning Algorithm

UAV path planning is essentially the process of finding the optimal or relatively optimal solution for the objective function while meeting the mission flight requirements, and solving the objective function is the path planning process. The A* algorithm is a heuristic graph search path planning algorithm that combines the advantages of breadth-first search and best-first search. (Liu et al., 2020), It adds a heuristic function on the basis of Dijkstra's algorithm to estimate the cost to the target point.

The A* algorithm improves search efficiency by avoiding exhaustive traversal of all nodes and is well-suited for complex or dynamic environments.Its core mechanism lies in the cost function, defined as Eq. 3:

$$f(n) = \alpha_g g(n) + \alpha_h h(n) \quad (3)$$

Where n represents a navigational node within the airspace visited during the search process; the coefficients α_g and α_h are weight balancing the contributions of actual and estimated costs; $f(n)$ represents the total cost estimate for reaching the goal through node n, used to determine which nearby node is more suitable for the next step of expansion, combining: $g(n)$ is the actual accumulated cost from the start node to n; $h(n)$ is the heuristic estimate of the remaining cost from n to the goal node, typically computed using Euclidean or Manhattan distance based on the graph structure.

This study proposes incorporating the node attributes from ULAN-KG into the low-altitude navigation constraint factors to improve the traditional A* algorithm.*

2.2.1 Heuristic Function h(n): $h(n)$ is the heuristic function, representing the estimated cost from the current point to the target point. Calculate using the coordinates from the current node to the target point obtained from the knowledge graph. It is calculated using Eq. 4:

$$h(n) = |x_1 - x_2| + |y_1 - y_2| + |z_1 - z_2| . \tag{4}$$

In this study, the calculation method for h(n) uses Manhattan distance, which is the shortest straight-line distance between two points; (x_1, y_1, z_1) and (x_2, y_2, z_2) epresent the coordinates of the current point and the target point, respectively.

2.2.2 Cost Function g(n): $g(n)$ is obtained by accumulating the replacement values of the initial path segments, and its calculation method is shown in Eq. 5:

$$g(n) = \sum_{i=1}^{n} g(r_i) . \tag{5}$$

$g(r_i)$ represents the cost value from the neighboring node of node i to the current node, while r_i represents the segment of the UAV's path in the i-th step of the path planning. $g(r_i)$ consists of three parts: directional cost g_{dir} environmental cost g_{env} and control cost g_{con} .As shown in Eq. 6:

$$\begin{cases} g(r_i) = \alpha_1 g_{dir} + \alpha_2 g_{env} + g_{con} \\ \alpha_1 + \alpha_2 = 1 \end{cases} \tag{4}$$

g_{dir} is the cost represents the consideration of the UAV's ascent/descent and travel distance in this cost function, while the turning cost is calculated based on the turning points (node type) that connect the centerlines of adjacent flight paths. g_{env} is the environmental impact cost, which represents the cost calculated based on surrounding information of flight path elements, including meteorological conditions, electromagnetic interference, and the type and danger level of nearby obstacles. The weight is adjusted according to the actual situation. g_{con} represents the control event cost. When no control event occurs on this segment, the value of g_{con} is 0; when a control event occurs on this segment, the value of g_{con} is ∞.

g_{dir} considers two influencing factors: ascent/descent cost and turning cost, and is calculated as shown in Eq.7:

$$g_{dir} = \beta L_{AB} + \Delta d . \tag{7}$$

β is the directional coefficient, representing the different consumption per unit distance during UAV's horizontal flight and ascent/descent. L_{AB} represents the actual distance of the flight path between points A and B.

The calculation of g_{env} is shown in Eq. 8:

$$\begin{cases} g_{env} = \gamma_1 \rho_{wea} + \gamma_2 \rho_{obstacle} + \gamma_3 \rho_{disturb} + \ldots \sum_{i=1}^{n} \gamma_i \rho \\ \sum_{i=1}^{n} \gamma_i = 1 \end{cases} \tag{8}$$

Using the above method, the UAV flight cost function model $h(n)$ is constructed, and by establishing a path planning algorithm, the path with the minimum cost function can be computed. γ the weight coefficient, which can be obtained by referring to relevant civil aviation standards and through the level coefficients of different obstacles stored in the knowledge graph. For example, when the obstacle is a high-voltage power tower, its weight is set higher at 0.7, and when the obstacle is a residential building, the weight coefficient is 0.4.

ρ_{wea} represents the impact of meteorological factors on the flight path. Low-altitude meteorological factors include temperature, pressure, humidity, wind, precipitation, cloud cover, visibility, etc., as well as the distribution patterns of these factors over time and space. Taking strong wind as an example, the maximum wind strength W that the UAV can withstand is first determined. When the wind strength exceeds W, the ρ_win value for that segment is set to ∞. When the wind strength is between level 0 and W, ρ_{win} represents the value of its impact indicator. The calculation is shown in Eq. 9:

$$\begin{cases} \rho_{win} = \dfrac{w_i}{W} & 0 < w_i \le W \\ \rho_{win} = \infty & w_i > W \end{cases} . \tag{9}$$

w_i represents the actual wind strength at node i. When $w_i > W$, ρ_{win} is infinite, indicating that the segment of the flight path is prohibited.

$\rho_{obstacle}$ represents the impact of obstacles near the flight path on the path's danger level, including buildings, roads, and

https://doi.org/10.5194/isprs-archives-XLVIII-4-W14-2025-97-2025 | © Author(s) 2025. CC BY 4.0 License.

associated infrastructure. The calculation of $\rho_{obstacle}$ is shown in Eq. 10:

$$\rho_{obstacle} = \frac{D_{safe} - D_i}{D_{safe}} \; (D_i < D_{safe}) \tag{10}$$

When an obstacle exists within the absolute safe distance D_{safe} from the flight path centerline, the actual distance Di from the obstacle boundary to the flight path centerline is used.

$\rho_{disturb}$ represents the impact of the electromagnetic environment on the flight path. The electromagnetic environment refers to the overall distribution of electric fields, magnetic fields, and electromagnetic wave power spectral density generated by various natural and man-made electromagnetic sources in different frequency bands within a specific spatiotemporal range, and their variations over time.

2.3 A* Path Planning Process with Knowledge Graph-Based Spatial Modeling

To enhance spatial awareness and adaptability in complex low-altitude environments, we integrate knowledge graphs into the A* algorithm for flight path planning. The core principle of A* lies in iteratively exploring neighboring nodes and selecting the one with the lowest cost function to construct an optimal path.

Prior to planning, a structured route network is built by querying the knowledge graph for the coordinates of nodes, the lengths of connecting segments, and their topological relationships. These serve as the foundational nodes and edges for the search graph.

During the pathfinding process, the algorithm evaluates neighboring nodes of the current position and calculates their composite cost functions to determine the most promising direction. The improved cost function, denoted as g_{env}, incorporates various environmental and regulatory constraints, including obstacle risk, meteorological influence, and airspace control restrictions. By continuously querying the knowledge graph, the algorithm dynamically retrieves node-level attributes in real-time, enabling context-aware path computation.

The overall planning procedure is illustrated in Figure 4.

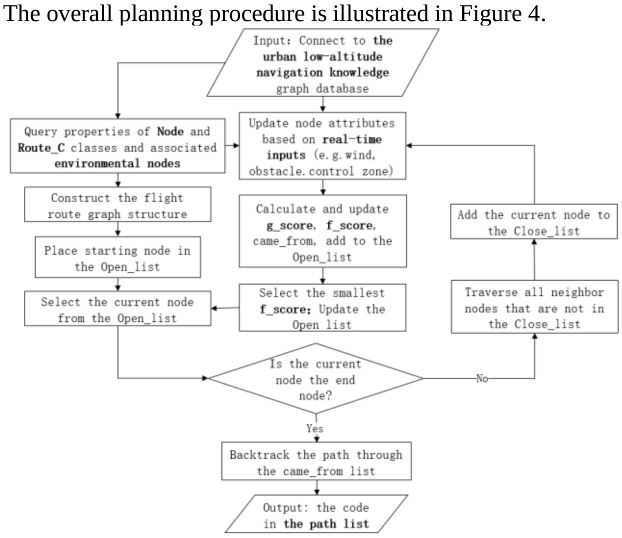


Figure 4. Flight path planning algorithm process considering low-altitude environmental influencing factors.

The revised flowchart enhances A-based path planning through five key improvements. System initialization establishes dual data streams for static geographic data (building heights/road networks) and real-time sensor inputs (weather/ADS-B signals). Knowledge graph queries are refined into spatial data retrieval, attribute extraction, and dynamic API integration. The cost function Eq. 11

$$f(n) = h(n) + \alpha \cdot g_{safety}(n) + \beta \cdot g_{regulation}(n) + \gamma \cdot g_{weather}(n) \tag{11}$$

is visually encoded with color-coded parameters. Path selection incorporates neighbor exhaustion checks and iteration limits to prevent infinite loops, with reliability indicated through color gradients. Exception handling triggers automatic replanning, obstacle recording, and risk model updates when no feasible path exists. These modifications explicitly demonstrate knowledge graph integration with A algorithms, showing how real-time environmental data and regulatory constraints dynamically adjust path calculations. The diagram now provides traceable connections between visual elements and code implementations, supporting reproducible research.

3. Experimental Validation and Results

3.1 Study Area Overview

This section demonstrates the proposed method through a case study conducted in an urban area. The study area is a hospital complex located within Beijing’s Second Ring Road, covering approximately 33,664 square meters, as shown in Figure 5. Following Chen et al.'s (2020) theoretical framework for iterative path network construction, we implemented a three-tier altitude stratification:

1. Lower Layer (12m): Avoids streetlights (12m), utility poles (10m), and trees (<12m)
2. Middle Layer (30m): Clears 80% of buildings (field survey data)
3. Upper Layer (70m): Ensures full building clearance while maintaining communication signal coverage

Figure 5. Schematic Diagram of Local Urban Buildings and Road Network.

Based on the theoretical system of iterative construction of UAV low-altitude flight path networks in urbanized areas(Chenchen et al., 2020), this study divides the flight path network into three main altitude layers. According to field survey statistics, streetlights in the area are 12 meters high, utility poles are 10 meters, trees are below 12 meters, and 80% of the buildings are under 30 meters in height. Therefore, the first altitude layer is set at 12 meters, allowing UAVs to avoid most streetlights, utility poles, and trees. The third layer is set at

70 meters, which allows UAVs to avoid all buildings while remaining within the communication signal coverage range.

Obtain the building footprint data for the Beijing area in 2025 from OpenStreetMap, with the coordinate system set to WGS84. Obtain the coordinates of the building vertices by processing the shp file in ArcGIS. Meanwhile, the coordinates of the defined take-off & landing points and turning points in the flight network are also obtained.

3.2 Construction of the Scene Knowledge Graph

Based on the ontology model, the ULAN-KG for the area is constructed, and the local query visualization results are shown in Figure 6.

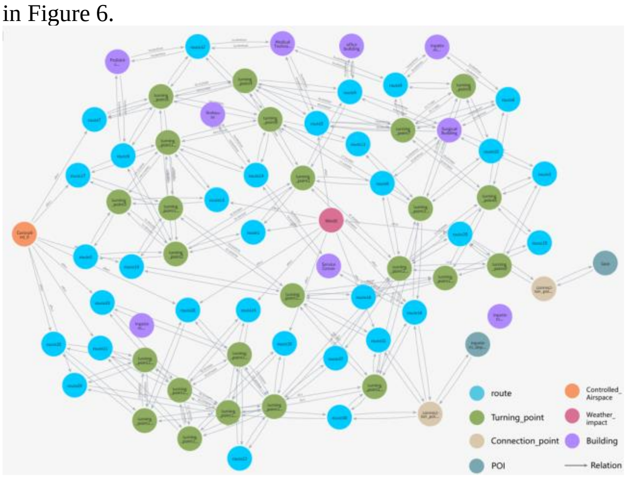


Figure 6. The knowledge graph of the Low-Altitude Navigation Scenario.

In the ULAN-KG architecture, the term "node" carries dual meanings that require precise contextual differentiation. Within the knowledge graph framework, a "node" represents a semantic entity with defined attributes and relationships, while in the path planning algorithm context, "node" refers to a vertex in a graph-based route network. The knowledge graph contains 71 semantic nodes categorized into three types:

Route Nodes (32): Represent flight path centerlines as graph edges, characterized by attributes including length (geometric distance), hasWindImpact (aerodynamic interference coefficient), and hasControlImpact (airspace restriction severity). These nodes form the connectivity backbone of the urban airspace model.

Waypoint Nodes (28): Comprise 2 Connection Points, 2 POIs (Points of Interest), and 24 Turning Points. POIs specifically denote UAV take-off & landing zones at the hospital's main gate and inpatient department entrance, linked to Connection Points through connected relationships.

Environmental Nodes: Include buildings with isAdjacentTo relationships to flight paths, capturing spatial proximity constraints.

At the algorithm execution level, A-path search maps Waypoint Nodes of the knowledge graph to vertices of the graph structure, each vertex carrying spatial attributes such as coordinates and obstacle distances; The Route Nodes of the knowledge graph are abstracted as edges of the graph structure, whose dynamic properties directly determine the weights of the edges. This design achieves decoupling of semantic constraints and spatial optimization: the semantic layer ensures path legitimacy through knowledge inference, while the algorithm layer performs quantitative calculations based on real-time updated edge weights. For example, when the meteorological API detects that the wind speed of a certain Route Node exceeds the threshold, the system will automatically set the weight of that edge in Figure A to infinity, forcing the path to detour.

Figure 7 serves as a simplified view of the knowledge graph, highlighting the algorithm level graph structure by removing environmental nodes (such as buildings): green vertices correspond to Waypoint Nodes of the knowledge graph, and blue edges correspond to Route Nodes of the knowledge graph. This hierarchical architecture ensures the rigor of knowledge reasoning while maintaining the efficiency of algorithms.

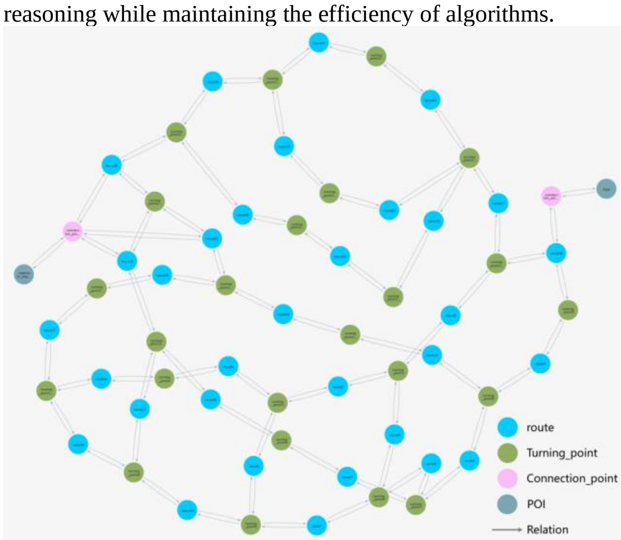


Figure 7. Relationships and Attributes of the Turning_point Node Class.

The coordinate values of flight path elements are static and do not change over time. However, as time progresses, various environmental adaptive factors in the airspace may change, leading to updates in the attribute values of the associated relationships. The attribute "hasControlImpact" is a Boolean value: 0 indicates no airspace control and the segment is navigable, while 1 indicates that flight is restricted in that area. Querying these attribute values can provide essential support for the A* algorithm in flight path planning.

3.3 Path Planning Results Using the Optimized A * Algorithm

A one-way path planning is performed between the logistics delivery lockers on the rooftop of the hospital building "inpatient_department" near the POI "Gate" and the hub airport.

3.3.1 Traditional Path Planning Algorithms: For the same time period, with the same starting and ending points, three traditional methods were selected for path calculation. The concept of the Genetic Algorithm (GA) is to simulate the process of natural selection and biological genetic evolution for search optimization. Ant Colony Optimization (ACO) utilizes the principle of pheromone usage by ants when searching for food(Ma and Xiong, 2019), which gives ACO good parallelism and collaboration.

3.3.2 Optimized A* Algorithm: A* uses a traditional relational database to store data and applies the classic A* algorithm for path planning. KG-A* stores data in a knowledge graph and uses the classic A* algorithm for path planning. ULAN-A* uses

https://doi.org/10.5194/isprs-archives-XLVIII-4-W14-2025-97-2025 | © Author(s) 2025. CC BY 4.0 License.

a traditional relational database to store data and applies an A* algorithm that considers low-altitude environmental information during path planning. According to the principle of the classic A* algorithm, its path planning result is the shortest path between two points.

When there is no explicit destination $\alpha_1 = \alpha_2 = \gamma_1 = \gamma_2 = 0.5$ .At the same time, 20% of the route nodes are randomly set as no-fly zones, and 25% of the nodes are randomly set as windy segments, with wind levels ranging from 1 to 6. The maximum acceptable wind level for the route is set to level 7. The path planning results are Path (a) and (b).

3.3.3 Safety-Oriented Path Planning: When the safest path needs to be selected, the setting for α is changed to $\alpha_1 = 0, \alpha_2 = 1$. Other conditions remain unchanged. The path planning result is Path (c).

To evaluate the effectiveness of various algorithms under different constraints, the path planning results are evaluated using five indicators: path length, meteorological risk zones, number of path turns, computation time, and crossing restricted airspace. The results are shown in the table below.

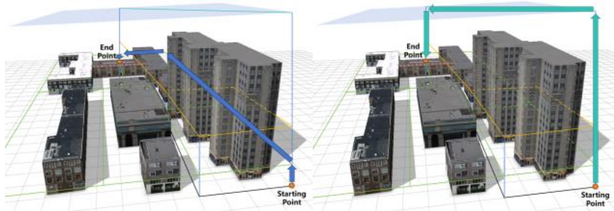
Path planning algorithm	Path length (m)	Weather Risk Zone (m)	Turing_point Number	Computation time (s)	Traverse Controlled Zone
A*	271.82	65.5138	5	0.009010	1
GA	271.82	65.5138	5	0.007004	1
ACO	326.04	42.3849	2	0.018998	0
KG-A*	271.82	65.5138	5	0.006721	1
ULAN -A*	289.44	102.388	4	0.010047	0
ULAN-KG-A*	289.44	102.388	4	0.007516	0
ULAN-KG-A*	400.42	0	3	0.008194	0

Table 1. Comparative Statistics of UAV Path Planning Algorithms in Urban Low-Altitude Environments

From the path planning results in Table 1, it can be seen that the traditional A* and GA methods are unable to effectively integrate environmental factors, resulting in paths that pass through restricted areas. Although ACO can avoid restricted zones through heuristic influence factors, its pre-search time and path length are longer. The path planning result is Path (C). None of these three traditional methods are able to effectively integrate dynamic environmental factors.

The nodes of the road network are encoded as 0-22, and the nodes passed through can represent the planned path.When environmental factors are not considered, the optimal path is (0, 23, 6, 7, 8, 22, 21). It is the shortest path. When environmental factors are included, the resulting path is (0, 23, 14, 16, 17, 18, 21). It can be seen that after incorporating environmental constraints, the route avoids controlled zones, but the total flight distance increases from 271.82, and the length of routes affected by wind also increases. Since the flight path mainly lies in the mid-altitude corridor, avoiding most buildings, the number of path turning points is reduced. This indicates that the algorithm prioritizes the calculation of no-fly zones over meteorological impacts. When flight safety is prioritized, the result is (0, 23, 6, 11, 12, 21). It does not pass through the Weather Risk Zone or Traverse Controlled Zone, meeting the safety requirements. However, the total length of this path is significantly increased, reaching 1.47 times that of the shortest path.

(a) ULAN -A* and ULAN-KG-A* (b) A*&GA and KG-A*



(c) ULAN-KG-A*(Safest Path) (d) ACO

Figure 8. Schematic of the optimal path in path planning algorithm. The orange points represent the starting point and endpoint, the line segments represent the flight routes, and the arrows indicate the planned path. (a) ULAN -A* and ULAN-KG-A* (b) A*&GA and KG-A* (c) ULAN-KG-A*(Safest Path) (d) ACO

Regarding computation time, the computation time of ULAN-KG-A*and ULAN-A* is slightly higher than that of KG-A* and A*, but the query time using the knowledge graph is significantly lower than that of traditional methods.

In conclusion, the ULAN-KG-A* algorithm demonstrates good adaptability, scalability, and high computational efficiency in path calculation within complex low-altitude environments.

4. Conclusions and Future Work

This study constructs an Urban Low-Altitude Navigation Knowledge Graph based on low-altitude flight routes and environmental characteristics and proposes a knowledge graph-optimized path planning algorithm. The algorithm reconstructs the cost function to quantify low-altitude risk factors, effectively eliminating impassable paths, thereby addressing the dynamic response and multi-objective optimization issues of UAVs flying in urban environments. The experiments show that the ULAN-KG-A* algorithm better adapts to multi-objective path planning tasks in complex urban scenarios.

In the future, integrating historical data and probabilistic information (such as congestion probability) from the knowledge graph into the cost function is expected to further improve the accuracy of path selection. When expanding nodes in the A* algorithm, the knowledge graph can be used to predict the reachability of neighboring nodes, dynamically excluding failed nodes (such as no-fly zones) and avoiding high-risk areas (such as strong wind zones), which is expected to reduce the number of search nodes and improve computational efficiency.

Acknowledgements

This study was supported by The R&D Program of Beijing Municipal Education Commission (KM202210016004), The Pyramid Talent Training Project of Beijing University of Civil Engineering and Architecture (JDYC20200322), The Fundamental Research Funds for Beijing University of Civil Engineering and Architecture (X20044).

References

Chen, Z., Wan, Y., Liu, Y., and Valera-Medina, A.,2024: A knowledge graph-supported information fusion approach for multi-faceted conceptual modelling, *Information Fusion*, 101, 101985.

Chenchen, X. U., Huping, Y. E., Huanyin, Y. U. E., Xiang, T. A. N., and Xiaohan, L.,2020: Iterative construction of UAV low-altitude air route network in an urbanized region: Theoretical system and technical roadmap, *Acta Geographica Sinica*, 75, 917-930, 10.11821/dlxb202005003.

Du, S., Zhong, G., Wang, F., Pang, B., Zhang, H., and Jiao, Q.,2024: Safety Risk Modelling and Assessment of Civil Unmanned Aircraft System Operations: A Comprehensive Review, 10.3390/drones8080354.

Garrow, L. A., German, B. J., and Leonard, C. E.,2021: Urban air mobility: A comprehensive review and comparative analysis with autonomous and electric ground transportation for informing future research, *Transportation Research Part C: Emerging Technologies*, 132, 103377.

Khoufi, I., Laouiti, A., and Adjih, C.,2019: A Survey of Recent Extended Variants of the Traveling Salesman and Vehicle Routing Problems for Unmanned Aerial Vehicles, 10.3390/*drones*3030066.

Kong, X., Pan, B., Cherkashin, E., Zhang, X., Liu, L., and Hou, J.,2020: Multi-constraint UAV Fast Path Planning Based on Improved A* Algorithm, *Journal of Physics: Conference Series*, 1624, 042009, 10.1088/1742-6596/1624/4/042009.

Liao, X., Qu, W., Xu, C., and He, H.,2023: A review of urban air mobility and its new infrastructure low-altitude public routes, 10.7527/S1000-6893.2023.28521.

Liu, Q., Xu, H., Wang, L., Chen, J., Li, Y., and Xu, L.,2020: Application of Dijkstra Algorithm in Path Planning for Geomagnetic Navigation, 2020 IEEE 11th Sensor Array and Multichannel Signal Processing Workshop (SAM), 1-4, 10.1109/SAM48682.2020.9104382.

Ma, F. and Xiong, F.,2020: Research on Path Planning of Plant Protection UAV Based on Grid Method and Improved Ant Colony Algorithm, 6th International Conference on Advanced Composite Materials and Manufacturing Engineering (ACMME), Yunnan, China.

Meister, O., Frietsch, N., and Ascher, C.,2008: Adaptive path planning for a VTOL-UAV, Position, Location and Navigation Symposium, IEEE/ION, 10.1109/PLANS.4570046.

Sun, Z., Wu, J., and Yang, J.,2016: Path Planning for GEO-UAV Bistatic SAR Using Constrained Adaptive Multiobjective Differential Evolution, IEEE Transactions on Geoscience and Remote Sensing, 10.1109/TGRS.2016.2585184.

Yu, Z., Sun, F., Lu, X., and Song, Y.,2021: Overview of Research on 3D Path Planning Methods for Rotor UAV, 2021 International Conference on Electronics, Circuits and Information Engineering (ECIE), 368-371, 10.1109/ECIE52353.2021.00081.

Zhang, H.,2021: Research on UAV Route Optimization Method Based on Double Target of Confidence and Ambiguity, Frontiers in Neurorobotics, 10.3389/fnbot.2021.694899.