

A Helicopter Path Planning Method Based on AIXM Dataset

Lai Xin, Civil Aviation Flight University of China, China

Liang Chang Sheng, China Aviation Navigation Data Co., Ltd., China

 <https://orcid.org/0009-0004-2027-5025>

Jiayu Feng, Civil Aviation Flight University of China, China

Hengyan Zhang, Civil Aviation Flight University of China, China

ABSTRACT

ICAO has emphasized that aeronautical information agencies should provide digitized aeronautical data and information, and realize that aeronautical data exchanging internationally in AIM. The AIXM structured aeronautical information dataset will be the main source of aeronautical basic data in the aeronautical information exchange network. In this article, the authors first analyze the spatio-temporal attributes of AIXM dataset and design the query method of AIXM structured obstacle data based on the research of AIXM coding specification. Secondly, the helicopter path planning is taken as the research scenario. Using the AIXM obstacle dataset and route dataset, combining the helicopter performance constraints to construct the envelope frame for collision judgment, and a new path planning method with improving the classical A* algorithm based on the AIXM dataset is proposed. The proposed method is validated and visualized. The validation results show that the proposed method reduces the frequency of helicopter turning, and ensures the safe distance between the flight path and the obstacles.

KEYWORDS

A* Algorithm, Air Transportation, AIXM Dataset, Helicopter Path Planning, Spatio-temporal Data

INTRODUCTION

In 2008, ICAO put forward the requirement to transform from Aeronautical Information Services (AIS) to Aeronautical Information Management (AIM). To regulate data management and release data under the AIM system, the Aeronautical Information Exchange Model (AIXM) specification was adopted as the underlying data structure for all kinds of aeronautical information (ICAO 2014, 2018). The aeronautical data in the AIXM structure will become the primary data to support general aviation operations in the future. As AIXM is essentially a spatiotemporal dataset, research on AIXM should

DOI: 10.4018/JCIT.333469

*Corresponding Author

This article published as an Open Access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0/>) which permits unrestricted use, distribution, and production in any medium, provided the author of the original work and original publication source are properly credited.

also study the related spatiotemporal data model research and spatiotemporal data retrieval research in addition to the research on the AIXM specification, which can provide a deeper understanding of AIXM from basic geographic information knowledge. Scholars have proposed solutions for the field based on research on spatiotemporal data models and data retrieval.

For big data information in the logistics and transportation industry, Wen and Yan (2018) established a data mining method based on a spatiotemporal data model, using real-time transmission of vehicle location information and image data to generate the optimal route of the corresponding scene information, which at the same time can be used for path planning. Zhang et al. (2016) proposed a spatiotemporal data prediction model based on deep learning and built a real-time people flow prediction system called UrbanFlow. Huynh et al. (2017) proposed a parallel R-tree construction scheme based on the Hadoop framework to improve the retrieval of big data. Azqueta-Alzúaz et al. (2017) proposed a scheme to load big data in parallel to improve the data loading efficiency of HBase. Laksmiwati et al. (2015) proposed a general architecture for spatiotemporal unpredictable data processing systems in disaster management information systems. Zhang et al. (2020) proposed a 3DPS-based spatiotemporal data model service sharing scheme with a research objective of spatiotemporal data model sharing. They applied it to two scenarios of ground settlement monitoring and railroad emergency rescue simulation in integrated disaster mitigation. Liu et al. (2021) researched and designed a set of spatiotemporal data model construction methods for natural resources based on hybrid modeling. This solved the need for integrated expression of natural resources in time, space, semantics, management, and services. To meet the needs of spatiotemporal analysis of the battlefield environment, Zhu et al. (2018) proposed establishing an object-oriented spatiotemporal data organization model through a task process-driven approach.

On the other hand, helicopter operations are characterized by flexible path planning and are deeply affected by terrain obstacles and other factors. Using the AIXM dataset for obstacle and path planning pre-flight hints can provide safety information for navigation operations. Helicopter path planning, which requires a combination of start and end locations and the operational spatial environment for path planning before operation, has become a hot research topic, with scholars proposing various algorithms in 3D path planning. Cicibaş et al. (2016) summarized the main obstacles, meteorological constraints, and model objectives, such as distance and time fuel for path planning. They improved the A* algorithm and applied it to 4D path planning. Jaishankar and Pralhad (2011) used greyscale images of 3D environments as a data environment for path planning. They combined the spatial multi-criteria decision analysis (MCDA) technique with distance transformation to generate optimal paths. Hara and Tomono (2020) proposed a method to remove moving objects in dynamic 3D environments and reconstruct maps using graph search and surface grid maps for path planning. Cao et al. (2022) proposed an improved artificial potential field algorithm and extended it to 3D space to solve the problem of unmanned helicopter trajectory planning in a 3D environment.

To apply AIXM data more broadly in aeronautical information management and to enable AIXM data sets to serve additional areas of aeronautical operations, we investigated the AIXM dataset coding specification and applied the AIXM dataset in a helicopter operation scenario. First, based on the above analysis of the path specification approach, we considered that the underlying data for the path planning of helicopter operations comes from the AIXM dataset, including the obstacle and route datasets. Second, based on analyzing the spatial relationship between the AIXM obstacle dataset and the route dataset, we constructed a query algorithm for the obstacle dataset according to AIXM specifications and constructed enclosing boxes for trajectory and collision judgments. Finally, we proposed a helicopter path planning algorithm based on the AIXM dataset using an improved classical A* algorithm and conducted simulation verification.

SPACE-TIME ATTRIBUTE ANALYSIS OF AIXM DATASETS AND QUERY METHODS

Space-Time Attribute of AIXM Datasets

Starting from AIXM version 5.1 (EUROCONTROL and FAA, 2022b), all geometry-related attributes in AIXM are described by GML objects. According to ICAO Annex 15, aeronautical geographic coordinates should be expressed in latitude and longitude, using the World Geodetic System WGS-84 as the datum. Based on the GML specifications, AIXM uses points, lines, and surfaces to describe aeronautical entities. For example, airport reference points and waypoints are defined using points, routes and linear obstacles using lines, and airspace using surfaces. To take the aerodrome reference point (ARP) of an airport as an example, the geographic attribute of the AIXM data corresponding to the ARP is a point representation; for a simple location point like the ARP, it uses the `aixm:Point` or `aixm:ElevatedPoint` elements, which are inherited from `gml:Point` and extend on top of it, as exemplified in Figure 1.

In the AIXM data structure, temporal attributes are proposed as a temporal characteristic model, which consists of the interpretation, sequence number, and correction number (EUROCONTROL and FAA, 2022b, 2023), where the interpretation specifies the type of time slice, and four different types of time slice are defined in the interpretation attribute, which are `BASELINE`, `TEMPDELTA`, `SNAPSHOT`, `PEMDELTA`. The `TEMPDELTA` and `BASELINE` attributes are prepared according to the state information of the aeronautical entity in different periods due to the event changes, the temporary time slice is prepared when the event causes the aeronautical entity to have a temporary change, and the baseline time slice is prepared when the event causes the aeronautical entity to have a permanent change. The temporary time slice has a higher priority than the baseline time slice; when both time slices are valid simultaneously, the information described by the temporary time slice is considered valid. The sequence number is used to update the previous time slice, and the correction number is used to correct the previous time slice. Both numbers are assigned starting with “1” and increase incremented by 1 (“2,” “3,” “4,” etc.). Figure 2 below shows an example of a temporal representation model in AIXM.

Dataset Query Methods

As can be seen from the data structure analysis, AIXM data is essentially a spatiotemporal dataset that can be used to retrieve specific data through spatiotemporal data retrieval methods. To use the AIXM dataset for path planning, it is necessary to efficiently retrieve the AIXM obstacle dataset in the specified range. In this paper, we describe a query method designed to extract obstacle data from the AIXM dataset based on the start and end points of the route and based on file retrieval and temporal retrieval. The specific steps of the proposed method are as follows.

Figure 1. Point representation in AIXM spatial attribute

```
<aixm:APR>
  <aixm:ElevatedPoint gml:id="ID0001">
    <gml:pos>52.38 -31.94</gml:pos>
    <aixm:elevation uom="M">30.0</aixm:elevation>
  </aixm:ElevatedPoint>
</aixm:APR>
```

Figure 2. AIXM temporal feature model

```

<aixm:timeslice>
  <aixm:VerticalStructureTimeSlice gml:id="VST0001">
    <!--valid Time-->
    <gml:validTime>
      <gml:TimePeriod gml:id="vt0001">
        <gml:beginPosition>2021-08-05T15:00:08Z</gml:beginPosition>
        <gml:endPosition indeterminatePosition="unknown"/>
      </gml:TimePeriod>
    </gml:validTime>
    <!--The Time Slice model:BASELINE,TEMPDELTA,SNAPSHOT,PERMDELTA-->
    <aixm:interpretation>BASELINE</aixm:interpretation>
    <!--Sequence Number-->
    <aixm:sequenceNumber></aixm:sequenceNumber>
    <!--Correction Number-->
    <aixm:correctionNumber>0</aixm:correctionNumber>
    <aixm:featureLifetime>
      <gml:TimePeriod gml:id="ft0001">
        <gml:beginPosition>2021-08-05T15:00:00Z</gml:beginPosition>
        <gml:endPosition indeterminatePosition="unknown"/>
      </gml:TimePeriod>
    </aixm:featureLifetime>
    ...
  </aixm:timeSlice>

```

Step One: Document Search

Determine the search radius, determine the search range (x_3, y_3, x_4, y_4) based on the helicopter start and end point coordinates and the search radius, and construct the minimum outsourcing rectangle of the search range. Based on the geographic range (x_1, y_1, x_2, y_2) of the obstacle data provided by the `<gml:boundedby>` tag in the obstacle dataset, we constructed geometrically minimal outsourced rectangles of the search range. Compare the minimum outsourcing rectangle of the search range with the minimum outsourcing rectangle of the geographic range of the obstacle dataset and obtain the comparison result. If the comparison result is that the obstacle data overlaps with the search range, or the search range is in the obstacle data, then the obstacle dataset intersects with the search range. Then, the obstacle that may impact the route is screened, and this data is used as the underlying data for the algorithm.

If there is an overlap between the obstacle data and the search range, equation (1) is satisfied:

$$\begin{aligned}
 a_1 &= 1(x_4 < x_1 < x_3) \\
 a_2 &= 1(x_4 < x_2 < x_3) \\
 a_3 &= 1(y_4 < y_1 < y_3) \\
 a_4 &= 1(y_4 < y_2 < y_3) \\
 a_1 + a_2 + a_3 + a_4 &\geq 2
 \end{aligned} \tag{1}$$

If the obstacle data is in the search range, equation (2) is satisfied:

$$\begin{aligned}
 a_1 &= 1(x_2 \leq x_3 \leq x_1) \\
 a_2 &= 1(x_2 \leq x_4 \leq x_1) \\
 a_3 &= 1(y_2 \leq y_3 \leq y_1) \\
 a_4 &= 1(y_2 \leq y_4 \leq y_1) \\
 a_1 + a_2 + a_3 + a_4 &= 4
 \end{aligned} \tag{2}$$

Step Two: Time Retrieval

According to the universal unique identifier (UUID) of the obstacle data associated with the temporal attributes of the obstacle entity and the input query time point, screen out the obstacle data that is valid at this time point and replace the attributes described in the temporary time slice in the valid obstacle data with the baseline time slice. If the valid obstacle data only exists in the baseline or temporary time slice, select its maximum sequence and correction numbers corresponding to the time-slice attribute information. The algorithm is shown in Algorithm 1.

Analysis of Helicopter Operational Performance

Helicopter path planning in the context of the AIXM dataset requires consideration of helicopter operational performance. The movement direction of the helicopter can be simplified into nine directions, as shown in Figure 3 below, with the X-axis as the forward positive direction of movement and the remaining directions as shown in Equation 3:

$$\begin{aligned}
 f_1 &= (1, 0, 0) \\
 f_2 &= (1, \tan \theta, 0) \\
 f_3 &= (1, -\tan \theta, 0) \\
 f_4 &= (1, 0, \tan \alpha_1) \\
 f_5 &= (1, 0, -\tan \alpha_2) \\
 f_6 &= (1, \tan \theta, \tan \alpha_1 \sqrt{(\tan \theta)^2 + 1}) \\
 f_7 &= (1, -\tan \theta, \tan \alpha_1 \sqrt{(\tan \theta)^2 + 1}) \\
 f_8 &= (1, \tan \theta, -\tan \alpha_2 \sqrt{(\tan \theta)^2 + 1}) \\
 f_9 &= (1, -\tan \theta, -\tan \alpha_2 \sqrt{(\tan \theta)^2 + 1})
 \end{aligned} \tag{3}$$

where α_1 and α_2 are the climb and descent angles of the helicopter, and θ is the turning angle. To ensure that the planned path does not turn too often and that this constraint is consistent with the motion performance of the helicopter, the length of the helicopter's shortest route is set to d . The constraint formula is given in Equation 4:

$$d = \frac{TAS^2}{127094 \tan \theta} + TAS \times T \tag{4}$$

Here, d consists of two parts: the radius of the out-of-turn change and the technical error, which is the distance the pilot flies in reaction time plus the radius of the turn. TAS is the vacuum speed of the helicopter flight, θ is the slope of the turn, and T is the pilot's reaction time (0-6s).

Algorithm 1.

<p>Input: <i>UUID</i> of the obstacle in the segment, obstacle dataset <i>OBS</i>, query time <i>T</i>, obstacle attributes <i>A</i> Output: Resulting datasets <i>result</i></p> <pre> <i>result</i> ← {} for <i>ID</i> ∈ <i>UUID</i> do <i>OBSY</i> ← <i>timein</i>(<i>OBS</i>(<i>ID</i>), <i>T</i>) /*Filtering out valid data*/ end for <i>repeat</i> ← <i>duplicate</i>(<i>OBSY</i>) /*Filter out obstacle data that has a duplicate UUID*/ if <i>len</i>(<i>repeat</i>) > 0 then <i>flag</i> ← 0 for <i>feature</i> ∈ <i>repeat</i> do for <i>a</i> → <i>A</i> do if <i>feature</i>(<i>a</i>, <i>I</i>) == 'TEMPEDLTA' then <i>flag</i> ← 1 endif if <i>flag</i> == 1 then <i>feature1</i> ← <i>drop</i>(<i>feature</i>(<i>a</i>, <i>I</i>) == 'BASELINE') <i>OBS2</i> ← <i>merge</i>(<i>feature1</i>) /*Merge datasets*/ end if end for end for end if <i>OBS3</i> ← <i>merge</i>(<i>OBS2</i>, <i>OBS</i>) /*Merge datasets by UUID*/ <i>result</i> ← <i>getMax</i>(<i>OBS3</i>) return <i>result</i> </pre>

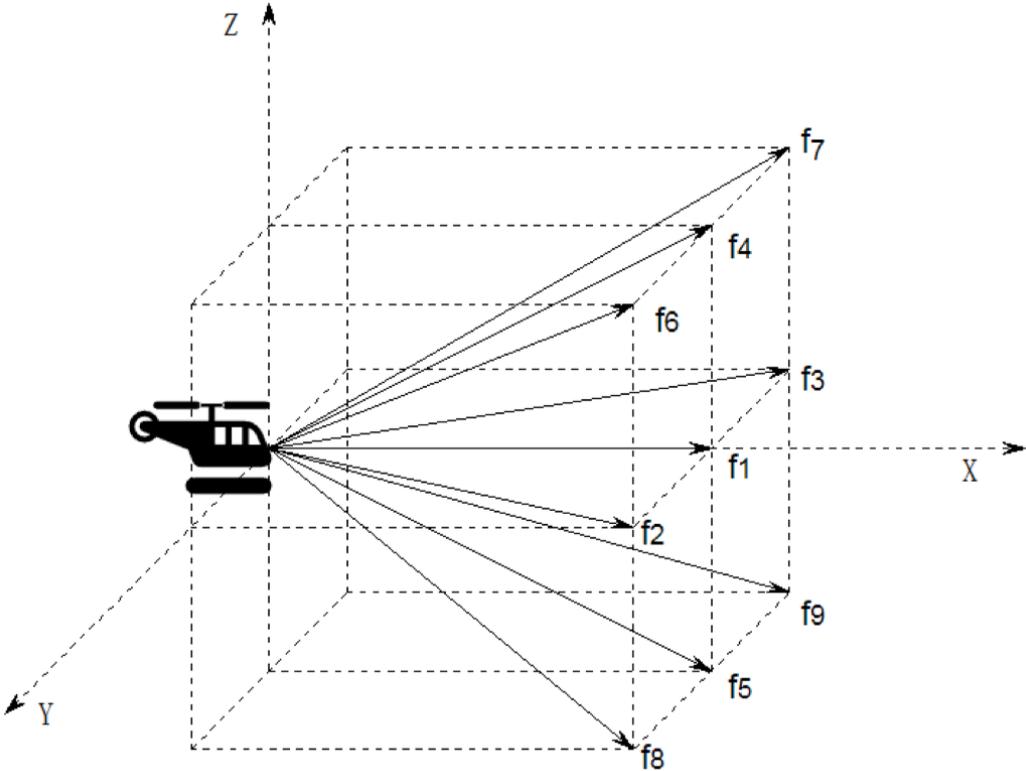
A* Improvement Algorithms Based on AIXM Dataset and Helicopter Performance Constraints

Meanwhile, in practice, the helicopter's forward-looking terrain warning system can calculate the position distance between the helicopter and the obstacle and provide a timely warning to ensure flight safety. The forward-looking terrain warning of the helicopter generates a virtual three-dimensional safety envelope in the forward direction space and, at the same time, obtains the terrain and obstacle data information provided by the database, compares the spatial position relationship between the warning envelope and the surrounding terrain in real time, and triggers the warning when the warning envelope is in contact with the surrounding terrain. In this paper, concerning the helicopter forward-looking terrain alarm, we create a simple enclosing box and provide judgment in the A* algorithm to determine whether the node can pass. If the obstacle is inside the enclosing box, it cannot pass, which makes the algorithm align more accurately with reality and provides the helicopter with a safer path.

Figure 4 shows the created enclosing box, which has the following main parameters:

- Minimum safe altitude *h*, a safe altitude below the helicopter that ensures the protection of the helicopter and any obstacles.

Figure 3. Direction of Helicopter Movement



- Forward-looking distance d , the straight-line distance traveled by a helicopter in a turn to avoid an obstacle. It consists of two parts: the distance traveled during the pilot's reaction time and the radius of the turn.
- Side boundary. In the helicopter alarm system, the alarm envelope is determined by the starting width, the center line of the oblique angle, and the side boundary deviation angle. In this paper, it is simplified as a trapezoidal shape; the two ends of the side lengths are L_1 and L_2 , and the relationship between the two is shown in Equation 5. The reference selected is 45m, where α is the side boundary deviation angle:

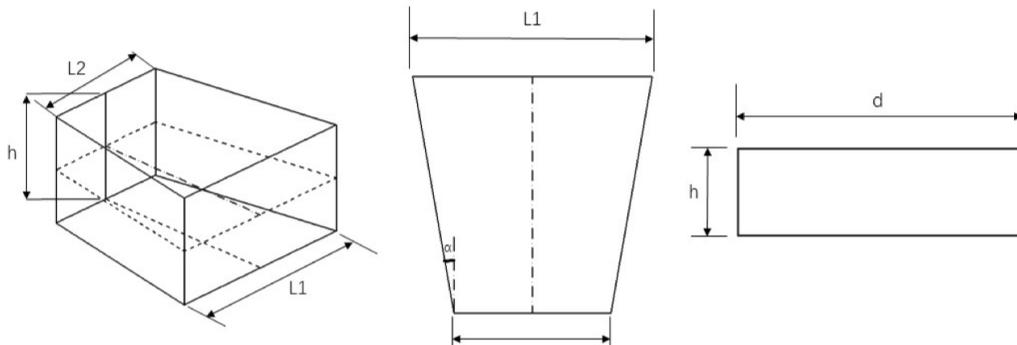
$$L_1 = L_2 + 2d \tan \alpha \tag{5}$$

Improvement of A* Algorithm

Hart et al. (1972) showed that the A* algorithm is a heuristic search algorithm which has a cost function:

$$f(n) = g(n) + h(n) \tag{6}$$

Figure 4. Obstacle enclosing box



where $f(n)$ is the total cost of the node; $g(n)$ is the actual path cost from the start point to the current node; and $h(n)$ is the predicted path cost from the current node to the end point, which is the key heuristic function in the algorithm. When calculating the distance from the current node to the end point to use as a heuristic function, the function typically has three choices: its Manhattan distance, diagonal distance, and Euclidean distance. The Manhattan distance applies to objects that can only move in the up, down, left, right, and four directions; the Euclidean distance is the line distance between two points, and the distance formula for three dimensions is as follows:

$$h(n) = \sqrt{(x1 - x2)^2 + (y1 - y2)^2 + (z1 - z2)^2} \tag{7}$$

Based on the AIXM dataset’s spatiotemporal attributes and the helicopter’s operational performance limitations, we propose an improved A* algorithm based on the spatial relationship between obstacle data retrieval and the enclosing box. In the improved algorithm, the heuristic function in the cost function is chosen to calculate the Euclidean distance between the point and the end point, and the corresponding formula is:

$$f(n) = d \times (n - 1) + \sqrt{(x_n - x_{n-1})^2 + (y_n - y_{n-1})^2 + (z_n - z_{n-1})^2} \tag{8}$$

where n is the current node, $n-1$ is the previous node, and d is the distance of each move.

In the classical A* algorithm, the judgment of whether the current node can be passed is mainly based on the points and obstacle surfaces. However, the actual operation of the helicopter’s forward-looking terrain warning system is based on the helicopter’s flight status. This requires generating a virtual three-dimensional safety envelope in space in the direction of its advancement and, at the same time, obtaining the information on terrain and obstacle data provided by the database and comparing in real-time the spatial location of the warning envelope in relation to the surrounding terrain. That is, the safety of the helicopter is ensured by the judgment of the 3D envelope box and obstacle positions, so the node judgment of points and surfaces in the classical A* algorithm cannot accurately simulate the state of the helicopter when it is in operation. Therefore, to ensure the path obtained through the algorithm aligns more closely with the actual operation requirements and safety, this paper refers to the helicopter’s forward-looking terrain alarm, constructs the enclosing box for collision judgment, and changes the position judgment of points and surfaces to the position judgment of polygons and polygons, which is divided into the following two steps:

Step One

Generate the bounding box using the coordinates of the current node $(x_{n-1}, y_{n-1}, z_{n-1})$ and the next point. The coordinates of the next point are set to $(x_{n\beta}, y_{n\beta}, z_{n\beta})$ by the next moving direction and the shortest route length, and the coordinates of the vertex of the enclosing box are generated by using the direction vector perpendicular to the moving direction. The formula is shown in Equation 9:

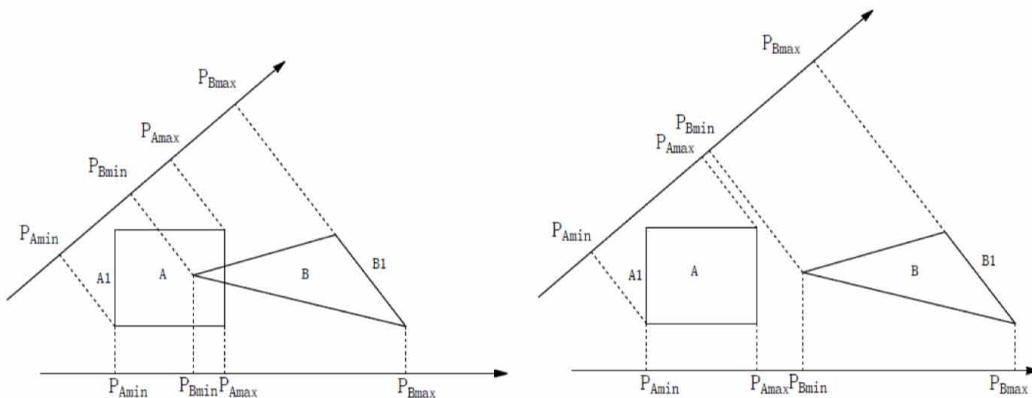
$$\begin{aligned}
 (x_{n\beta}, y_{n\beta}, z_{n\beta}) &= \frac{f_{\beta}}{|f_{\beta}|} \times d + (x_{n-1}, y_{n-1}, z_{n-1}) \\
 f_{\beta\perp r} &= f_{\beta} \times \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \\
 f_{\beta\perp l} &= f_{\beta} \times \begin{pmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \\
 (x_{r\varphi}, y_{r\varphi}, z_{r\varphi}) &= f_{\beta\perp r} \times \frac{L_{\varphi}}{2} + (x_n, y_n, z_n) \\
 (x_{l\varphi}, y_{l\varphi}, z_{l\varphi}) &= f_{\beta\perp l} \times \frac{L_{\varphi}}{2} + (x_n, y_n, z_n) \\
 (x_{r\alpha}, y_{r\alpha}, z_{r\alpha}) &= f_{\beta\perp r} \times \frac{L_{\varphi}}{2} + (x_n, y_n, z_n) - (0, 0, h) \\
 (x_{l\alpha}, y_{l\alpha}, z_{l\alpha}) &= f_{\beta\perp l} \times \frac{L_{\varphi}}{2} + (x_n, y_n, z_n) - (0, 0, h) \\
 \varphi &= 1, 2 \\
 \alpha &= 3, 4 \\
 \beta &= 1, 2, 3 \dots 9
 \end{aligned} \tag{9}$$

where $f_{\beta\perp r}$ and $f_{\beta\perp l}$ are the vertical vectors of the moving direction vectors, $(x_{r\varphi}, y_{r\varphi}, z_{r\varphi})$ and $(x_{l\varphi}, y_{l\varphi}, z_{l\varphi})$ are the coordinates of the right vertex and the left vertex of the enclosing box, and $(x_{r\alpha}, y_{r\alpha}, z_{r\alpha})$ and $(x_{l\alpha}, y_{l\alpha}, z_{l\alpha})$ are the coordinates of the right bottom endpoint and the left bottom endpoint of the enclosing box, respectively.

Step Two

Generate the encircling box and obstacle judgment algorithm. The generated enclosing box and obstacle judgment algorithm are divided into vertical and horizontal projection directions—the vertical direction to compare the height of the obstacle with the height of the bottom of the enclosing box and the horizontal direction to judge the positional relationship between the obstacle and the enclosing box—using the separation axis algorithm, which is a collision detection algorithm applicable to convex polygons. Figure 5 shows an example of the steps of the separation axis algorithm: Two polygons, A and B, are first used to determine the A1 side of the unit normal vector and the separation axis, and then the two polygons are projected to the separation axis to determine whether the projection of the two polygons intersect. When $P_{A_{\max}} < P_{B_{\min}}$, the two polygons in this direction do not intersect, as shown on the right side of Figure 5. As shown on the left side of Figure 5, the two polygons intersect

Figure 5. Example of separation axis algorithm



the projection. After the cycle of the two polygons with the edge generates the separation axis, project the two polygons onto the separation axis and determine whether they intersect. If the two projections do not intersect in one direction, the two polygons do not intersect.

THE PROPOSED METHOD AND SIMULATION VERIFICATION

The Proposed Method

This paper proposes a helicopter path planning method based on the given AIXM data retrieval method and the improved A* algorithm. The workflow diagram of the planning method is shown in Figure 6.

The method can be divided into three steps.

Step One

According to the helicopter start and end point coordinates, effective and accurate obstacle data are extracted from the AIXM dataset of the helicopter route through file and time searches to provide obstacle information for the algorithm.

Step Two

The helicopter start point and end point coordinates and the obstacle coordinates in the valid obstacle data are projected from the WGS84 coordinate system to the Universal Mercator Grid System coordinate system. Then, the helicopter enveloping box for obstacle collision judgment is established, and the improved A* algorithm is used to calculate the helicopter path planning node set in combination with the helicopter performance constraints.

Step Three

All node coordinates are reprojected in the helicopter path planning node set into the WGS84 coordinate system and then converted to the AIXM file.

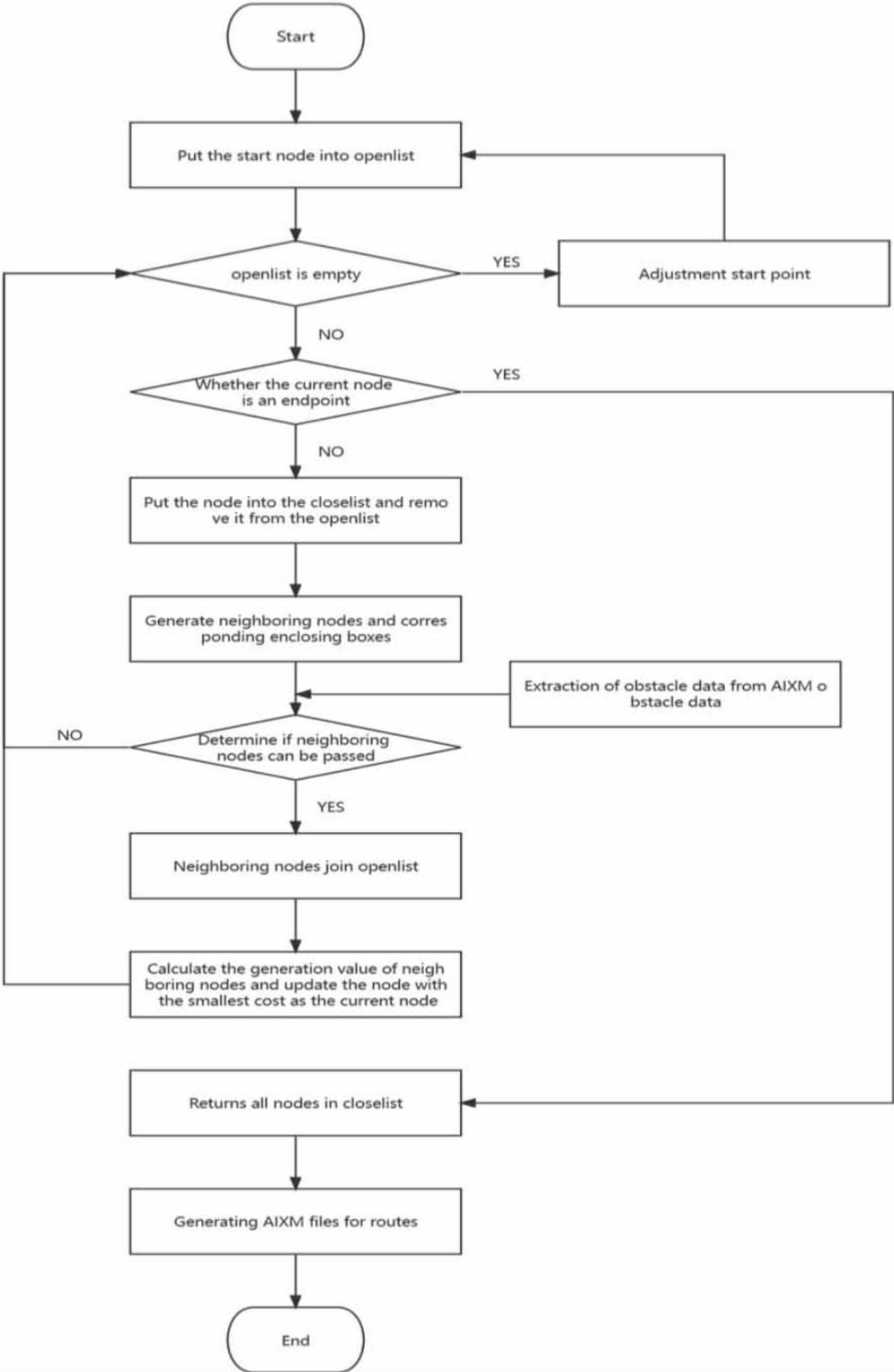
Simulation Validation of the Proposed Method

To verify the usability and execution performance of the proposed algorithm, we designed the following experiments.

Experimental environment: Using the Visual Studio Code platform and Python programming language, we set up a simulation environment of the helicopter operation.

The experimental parameter settings are as follows:

Figure 6. Work flow of the proposed method



- According to the helicopter-related regulations in DOC8168, we selected the following parameters: helicopter climb gradient of 13.3%, optimal descent gradient of 6.5%, turn angle of 30°, and speed of 130km/h.
- The coordinate system transformation was completed using the Python open-source package to convert the start and end point coordinates to the Universal Transverse Mercator Grid System (UTM) coordinate system. For example, the coordinates (119.630, 44.400) are in the projected belt of 50N, and the converted UTM coordinates are (709437.7807428013, 4919665.880332163).

We set the obstacle and its midpoint coordinate data as shown in Table 1. Meanwhile, the obstacle data were converted to an AIXM file to provide the obstacle data to improve the A* algorithm, as shown in Figure 6. Meanwhile, the experimental results generated the route KML file, which can be used directly for visually presenting the data.

The purpose of the experiment is to run the improved A* algorithm and the unimproved A* algorithm in the same environment and to project the resulting experimental data from each algorithm onto the x-y, x-z, and y-z projection surfaces according to the coordinate axes to compare the two algorithms.

Analysis of the Experimental Results

The visualization of the AIXM route data generated by the improved A* algorithm can be shown in the Luaid AIXM Viewer, as shown in Figure 8. This demonstrates that the algorithm creates route AIXM files in compliance with the data structure regulations of the AIXM dataset and that the improved algorithm can be used to perform path planning under the AIXM dataset.

The two algorithms were compared regarding the number of nodes and the path length in terms of two metrics; Table 2 lists specific data. The improved A* algorithm generates fewer path nodes than the Classical A* algorithm. The length of the path, or the gap between the two algorithms, is not significant; the improved A* algorithm-generated path difference is 40.093.

A more intuitive comparison can be obtained from Figure 9, which compares the improved A* algorithm and the classical A* algorithm in the x-y projection plane. On the left side, in the improved A* algorithm, the dotted line is constructed by the encircling box of the collision judgment. In the path obtained by the improved A* algorithm, a safe distance has been maintained between the path and the obstacle. On the right side, the Classical A* algorithm's path is very close to the obstacle. At the same time, the improved A* algorithm obtains a path with fewer turning points, and the path

Table 1. Experimental data settings

Parameter name	Coordinate data
Start point	(44.400, 119.630, 500)
End point	(44.408, 119.650, 600)
Obstacle 1	(44.404, 119.631, 700), (44.405, 119.631, 700), (44.405, 119.632, 700), (44.404, 119.632, 700), (44.404, 119.631, 700)
Obstacle 2	(44.400, 119.633, 650), (44.401, 119.633, 650), (44.401, 119.634, 650), (44.400, 119.634, 650), (44.400, 119.633, 650)
Obstacle 3	(44.405, 119.637, 700), (44.407, 119.637, 700), (44.407, 119.638, 700), (44.405, 119.638, 700), (44.405, 119.637, 700)
Obstacle 4	(44.406, 119.640, 700), (44.407, 119.640, 700), (44.407, 119.641, 700), (44.406, 119.641, 700), (44.406, 119.640, 700)
Obstacle 5	(44.403, 119.639, 455), (44.404, 119.639, 455), (44.404, 119.640, 455), (44.403, 119.640, 455), (44.403, 119.639, 455)

Figure 7. Partial obstacle data for the AIXM files used in the experiments

```
<message:hasMember>
  <aixm:VerticalStructure gml:id="uuid.cb648833-a786-43da-9e83-b171096e9e69">
    <gml:identifier codeSpace="urn:uuid:">cb648833-a786-43da-9e83-b171096e9e69</gml:identifi
    <aixm:timeSlice>
      <aixm:VerticalStructureTimeSlice gml:id="OBS-0002">
        <gml:validTime>...
      </gml:validTime>
      <aixm:interpretation>BASELINE</aixm:interpretation>
      <aixm:sequenceNumber>1</aixm:sequenceNumber>
      <aixm:correctionNumber>0</aixm:correctionNumber>
      <aixm:featureLifetime>...
    </aixm:featureLifetime>
    <aixm:name>RODSAND I</aixm:name>
    <aixm:type>WINDMILL_FARMS</aixm:type>
    <aixm:part>
      <aixm:VerticalStructurePart gml:id="vsp0002-1">
        <aixm:type>WINDMILL_FARMS</aixm:type>
        <aixm:markingFirstColour>GREEN</aixm:markingFirstColour>
        <aixm:horizontalProjection_surfaceExtent>
          <aixm:ElevatedSurface srsName="urn:ogc:def:crs:EPSG::4326" gml:id="e
            <gml:patches>
              <gml:PolygonPatch>
                <gml:exterior>
                  <gml:Ring>
                    <gml:curveMember>
                      <gml:Curve gml:id="c1234">
                        <gml:segments>
                          <gml:GeodesicString>
                            <gml:posList>44.404 119.631
                              44.405 119.631
```

Figure 8. Visualization of the experimental results of the proposed method in Luaid AIXM viewer

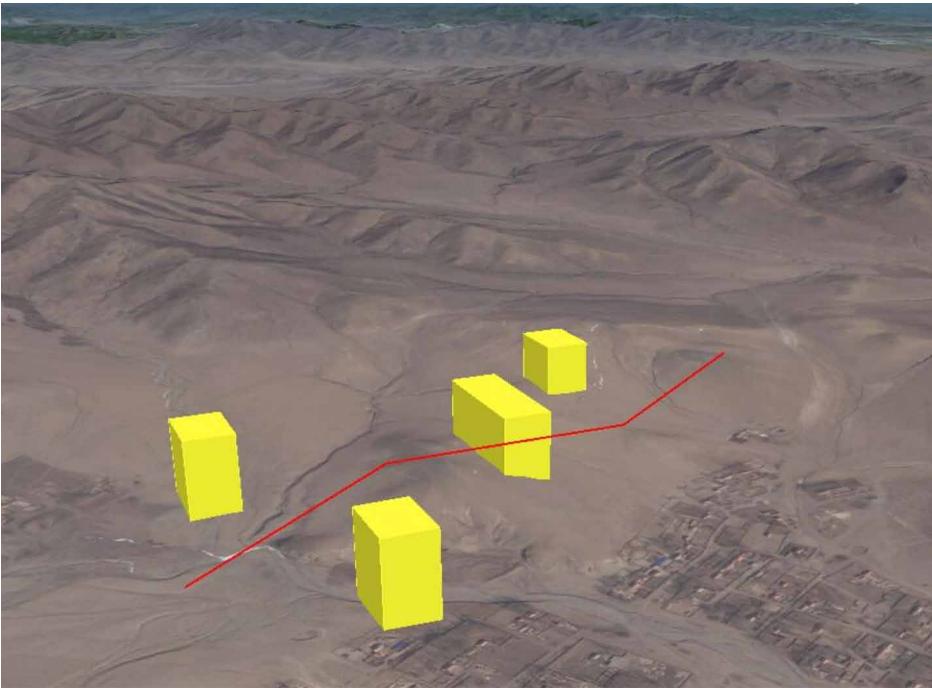


Table 2. Comparison of the experimental results of the two algorithms

Algorithm	Number of nodes	Path length
Improved A*	4	2160.003
Classical A*	23	2200.096

is smoother. This means that the path involves fewer heading adjustments from a flight operation perspective, which is more conducive to the pilot’s operation.

A further comparison can be obtained from Figures 10 and 11, in which the paths generated by the two algorithms are projected onto the x-y, x-z, and y-z projection planes. A comparison shows that the two algorithms have the same effect on the x-z and y-z projection planes. In these planes, the path of the improved A* algorithm maintains a safe distance from obstacles in the vertical profile, and there are only three flight segments and two inflection points. Therefore, compared with the Classical A* algorithm, the routes generated by the improved A* algorithm, with fewer up and down path fluctuations, are more in line with the helicopter motion characteristics. In addition, the routes have fewer nodes, which reduces the frequency of helicopter turns and climbs. At the same time, the path guarantees a safe distance between the route and the obstacle.

CONCLUSION AND DISCUSSION

This paper proposes a helicopter path planning method based on the AIXM dataset. The study first analyzes the AIXM dataset from the data structure and designs a query method to extract obstacle data from the AIXM dataset using the dataset’s spatiotemporal attributes, laying the foundation for the subsequent path planning data retrieval. Regarding the principle of a helicopter’s forward-looking terrain alarm, we propose an envelope box construction method for the AIXM dataset, which generates a virtual 3D safety envelope box in the space of the helicopter’s forward direction. It also utilizes the previously constructed obstacle data query method to obtain the information on terrain and obstacle data provided by the database. It compares the spatial positional relationship between the warning envelope box and the surrounding terrain in real time. Thus, by improving the A* algorithm based on the envelope box construction of the AIXM dataset, we proposed and validated a method for helicopter path planning in the environment of the AIXM dataset.

Figure 9. Comparison of the 2D projections of the results of the two algorithms

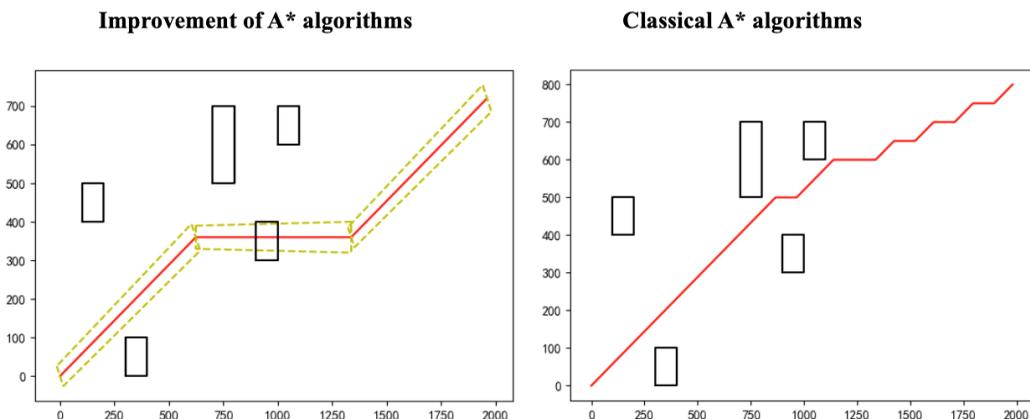


Figure 10. 3D projection results of improved A*

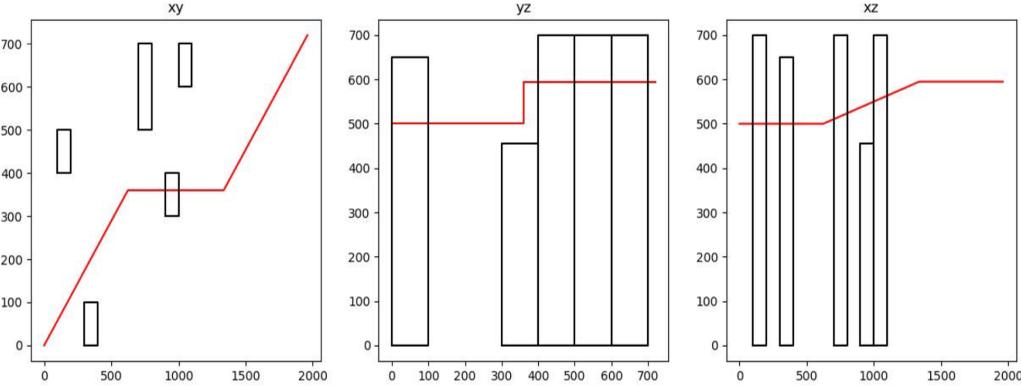
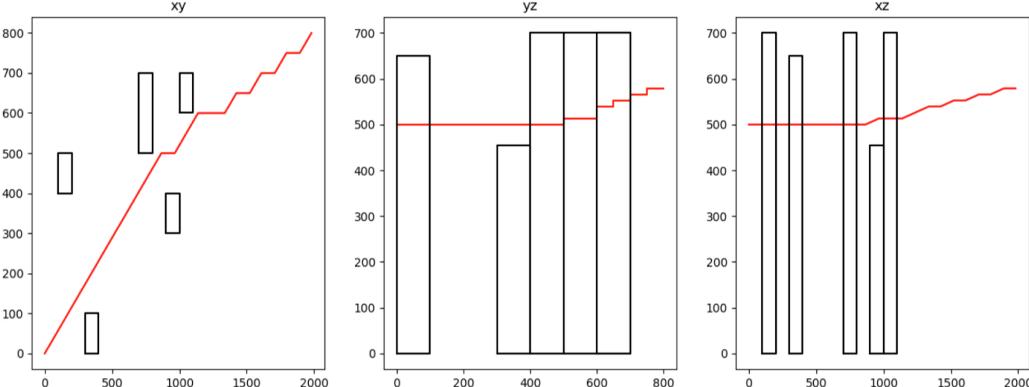


Figure 11. 3D projection results of classical A*



The experimental results show that the proposed method can perform path planning based on the AIXM obstacle and route datasets. The proposed method can efficiently perform retrieval and spatial analysis of the AIXM dataset, and its completed path-planning pairs can achieve obstacle avoidance and smoother routes. Using this method, the helicopter’s situational awareness of obstacles around the planned route can be improved, and the helicopter’s operational risk can be reduced by lowering the number of turning points. This method is a positive attempt to apply the AIXM dataset to multi-aeronautical operations. In our future research, we aim to explore the informatization applications of the AIXM dataset in more areas of aviation operations and build an application model for aviation information interaction and visualization on multiple information platforms using the AIXM dataset.

FUNDING

This research was supported by the Natural Science Foundation of Sichuan Province (2023NSFSC0903), the Special Fund for Civil Aviation Education Talents (MHJY2023001), and the Special Fund for Scientific Research of Central Universities (ZJ2023-003).

REFERENCES

- Azqueta-Alzúaz, A., Patiño-Martínez, M., Brondino, I., & Jimenez-Peris, R. (2017). Massive data load on distributed database systems over HBase. In *Proceedings of the 17th IEEE/ACM International Symposium on Cluster, Cloud, and Grid Computing (CCGRID)* (pp. 776–779). IEEE. doi:10.1109/CCGRID.2017.124
- Cao, L., Wang, L., Liu, Y., & Yan, S. (2022). 3D trajectory planning based on the rapidly-exploring random tree-connect and artificial potential fields method for unmanned aerial vehicles. *International Journal of Advanced Robotic Systems*, 19(5), 172988062211188. doi:10.1177/17298806221118867
- Cicibaş, H., Demir, K., & Arica, N. (2016). Comparison of 3D versus 4D path planning for unmanned aerial vehicles. *Defence Science Journal*, 66(6), 651–664. doi:10.14429/dsj.66.9575
- EUROCONTROL and FAA. (2022a). *AIXM5 data coding guidelines*. [EB/OL]. aiXm. <https://www.aixm.aero/page/data-coding-guidelines>
- EUROCONTROL and FAA. (2022b). *AIXM5 guide*. [EB/OL]. iZm. <https://www.aixm.aero/>
- EUROCONTROL and FAA. (2023). *AIXM 5 temporality model*. [EB/OL]. aiXm. <https://www.aixm.aero/>
- Hara, Y., & Tomono, M. (2020). Moving object removal and surface mesh mapping for path planning on 3D terrain. *Advanced Robotics*, 34(6), 1–13. doi:10.1080/01691864.2020.1717375
- Hart, P., Nilsson, N., & Raphael, B. (1972). A formal basis for the heuristic determination of minimum cost paths. *Intelligence/Sigart Bulletin*, 37, 28–29. doi:10.1145/1056777.1056779
- Huỳnh, N., Lee, K.-W., Joo, I.-H., Kwon, O.-H., & Song, H.-J. (2017). Improving the quality of an R-tree using the map-reduce framework. In *Proceedings of the International Conference on Multimedia and Ubiquitous Engineering International Conference of Future Information Technology* (pp. 164–170). doi:10.1007/978-981-10-5041-1_29
- ICAO. (2014). *Manual on the digital exchange of aeronautical meteorological information (ICAO Doc 10003)*. ICAO.
- ICAO. (2018). *ICAO Aeronautical information management (ICAO Doc 10066)*. ICAO.
- Jaishankar, S., & Pralhad, R. N. (2011). 3D off-line path planning for aerial vehicle using distance transform technique. *Procedia Computer Science*, 4, 1306–1315. doi:10.1016/j.procs.2011.04.141
- Laksmiwati, H., Widyani, Y., Hafidhoh, N., & Yusuf, A. (2014). Modeling unpredictable data and moving object in disaster management information system based on spatio-temporal data model. In *Proceedings of the International Conference on Data and Software Engineering (ICODSE)* (pp. 1–6). IEEE. doi:10.1109/ICODSE.2014.7062662
- Liu, J., Gao, Y., Zhao, W., Zhai, Z., & Wu, C. (2021). Research on construction method of natural resources spatial-temporal data model. *Geomatics World*, 5, 42–46.
- Wen, R., & Yan, W. (2018). Spatio-temporal mining with scene data integration for urban transportation navigation. In *Proceedings of the IEEE International Conference on Big Data* (pp. 3175–3179). IEEE. doi:10.1109/BigData.2018.8622244
- Zhang, J., Zheng, Y., Qi, D., Li, R., & Yi, X. (2016). DNN-based prediction model for spatio-temporal data. In *Proceedings of the 24th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems* (pp. 1–4). ACM. doi:10.1145/2996913.2997016
- Zhang, X., Huo, L., Shen, T., Zhu, Y., & Duan, Y. (2020). Research on service sharing method of dynamic spatiotemporal data model. *Science of Surveying and Mapping*, 12, 161–167+190. doi:10.16251/j.cnki.1009-2307.2020.12.024
- Zhu, J., You, X., & Xia, Q. (2018). Battlefield environment object spatio-temporal data organizing model based on task-process. *Geomatics and Information Science of Wuhan University*, 11, 1739–1745. doi:10.13203/j.whugis.20170074.10.13203/j

Lai Xin is an associate professor and research scholar at the Civil Aviation Flight University of China with a PhD. She has interests in aeronautical information management and big data mining. She has published more than 20 technical research papers in journals. Email: 7261240@qq.com Mailing Address: Civil Aviation Flight University of China, Air Traffic Management College, Guanghan City, Sichuan Province, China.

Liang Changsheng obtained his postgraduate degree from the Civil Aviation Flight University of China. Now, he is an assistant engineer at China Aviation Navigation Data Co., Ltd. His primary research interests are aviation operation management and aeronautical information and data.

Feng Jiayu studied at the Civil Aviation Flight University of China. His primary research interests are aviation management, aeronautical information, and big data.

Zhang Hengyan studied at the Civil Aviation Flight University of China. Her primary research interests are aviation management, aeronautical information, and big data.