Flight Trajectory Option Set Generation Based on Clustering Algorithms

WANG Shijin, SUN Min, LI Yinglin, YANG Baotian*

New Generation Intelligent Air Traffic Control Laboratory, College of Civil Aviation, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, P. R. China

(Received 30 March 2025; revised 11 September 2025; accepted 12 November 2025)

Abstract: Addressing the issue that flight plans between Chinese city pairs typically rely on a single route, lacking alternative paths and posing challenges in responding to emergencies, this study employs the "quantile-inflection point method" to analyze specific deviation trajectories, determine deviation thresholds, and identify commonly used deviation paths. By combining multiple similarity metrics, including Euclidean distance, Hausdorff distance, and sector edit distance, with the density-based spatial clustering of applications with noise (DBSCAN) algorithm, the study clusters deviation trajectories to construct a multi-option trajectory set for city pairs. A case study of 23 578 flight trajectories between the Guangzhou airport cluster and the Shanghai airport cluster demonstrates the effectiveness of the proposed framework. Experimental results show that sector edit distance achieves superior clustering performance compared to Euclidean and Hausdorff distances, with higher silhouette coefficients and lower Davies-Bouldin indices, ensuring better intra-cluster compactness and inter-cluster separation. Based on clustering results, 19 representative trajectory options are identified, covering both nominal and deviation paths, which significantly enhance route diversity and reflect actual flight practices. This provides a practical basis for optimizing flight paths and scheduling, enhancing the flexibility of route selection for flights between city pairs.

Key words: flight trajectory clustering; trajectory option set; sector edit distance; density-based spatial clustering of applications with noise (DBSCAN) algorithm; deviation trajectories

CLC number: V355 **Document code:** A **Article ID:** 1005-1120(2025)06-0767-22

0 Introduction

With the rapid development of China's civil aviation transportation industry, the number of flight operations continues to rise. In 2023, the annual flight movements at national airports reached 11.708 2 million, surpassing pre-pandemic levels (11.660 4 million in 2019)[1]. However, the limitations of finite airspace resources have become increasingly apparent. The constraints of single-route operations in city-pair flights are magnified, particularly when planned trajectories are disrupted by convective weather or flow control. The lack of alternative routes often leads to delays or cancellations, increasing airline operational costs and compromising passenger experience.

The next generation air transportation system (NextGen) proposed by the Federal Aviation Administration (FAA) has advanced this concept through trajectory-based operations (TBO)[2]. Its core component, the Collaborative Trajectory Options Program (CTOP), introduces the trajectory options set (TOS)[3]—A weighted set of preferred routes submitted by airlines to air traffic control. This system has been widely implemented in international busy air routes, significantly reducing planning complexity and operational costs while optimizing flow management[4-8]. Building upon this foundation, the introduction of multi-path trajectory options provides potential solutions for domestic chal-

*Corresponding author, E-mail address: sm45430220@163.com.

How to cite this article: WANG Shijin, SUN Min, LI Yinglin, et al. Flight trajectory option set generation based on clustering algorithms[J]. Transactions of Nanjing University of Aeronautics and Astronautics, 2025,42(6):767-788.

http://dx.doi.org/10.16356/j.1005-1120.2025.06.005

lenges. By generating preferred route sets for specific city-pairs, multi-path trajectory options enhance route planning flexibility, mitigate pressure from sudden operational adjustments, optimize fuel efficiency and punctuality rates, and facilitate collaborative decision-makings between air traffic management and airlines.

The TOS generation traditionally relies on airline experience and predefined routes. In the United States, FAA playbook routes serve as common implementations, while domestic research and standardization in this area remain underdeveloped. Recent advancements in data-driven technologies have positioned trajectory clustering as a novel approach for identifying frequently used routes from massive historical data, establishing a more scientific foundation for strategic-phase trajectory option generation. For example, Pham[9] introduced a network model-based user-preferred path generation method that considers both airline and air traffic control requirements. Zhu et al.[10] developed a Transformer-based trajectory option generator that automatically generates candidate paths conforming to planning principles using historical trajectory data. Furthermore, Evans et al.[11] implemented an automated TOS generation method with high operational acceptance probability by combining hierarchical clustering and machine learning based on historical flight plan revision data. Mateos Villar et al.[12] further proposed a novel approach that leverages machine learning to extract airspace users' route preferences and predict new routes not observed during the model training phase, thereby providing a more adaptive optimization solution for TOS generation.

By analyzing actual flight patterns, the trajectory clustering can reveal commonly adopted deviations from predefined routes, addressing the limitations of single-path planning. Research indicates that the data-driven preference path mining can optimize airspace resource allocation and reduce flight delays[4]. Current research demonstrates the prevalent use of density-based clustering algorithms in trajectory analysis due to their robustness and elimination of pre-defined cluster requirements. For instance, Ye et al.[13] developed a multidimensional aviation trajectory clustering method using Hausdorff distance and the density-based spatial clustering of applications with noise (DBSCAN) algorithm to identify abnormal trajectories, extracting central trajectories through statistical methods combined with flight distance and similarity metrics. Wang et al.[14] employed kernel principal component analysis (KPCA) and the DBSCAN to enhance the trajectory type differentiation and complete clustering after the interference removal spectral clustering also emerges as a prominent approach. Ma et al.[15] classified terminal area trajectories using heading factor-based Euclidean distance similarity measures combined with improved spectral clustering. Wang et al.[16] investigated spectral clustering-based identification of prevalent traffic flows in terminal areas using three-dimensional trajectory data. Li et al.[17] addressed computational efficiency and parameter selection challenges in traditional spectral clustering through resampling and natural neighbor methods for effective terminal area traffic flow identification. Recent years have witnessed the emergence of deep learning applications. Rao et al.[18] implemented deep clustering combining autoencoders with bidirectional long short-term memory (Bi-LSTM) networks for air traffic flow classification and anomaly detection through Q-distribution. Zeng et al.[19] integrated deep autoencoders (DAEs) with Gaussian mixture models (GMMs) for terminal area trajectory clustering. These methodologies provide theoretical and technical references for this study.

Although the existing studies have demonstrated the usefulness of Euclidean distance, Hausdorff distance, and other traditional similarity measures for clustering flight trajectories, these methods face notable limitations when applied to long-sequence city-pair trajectories. Euclidean distance is sensitive to local deviations and requires strict temporal alignment, while Hausdorff distance emphasizes extreme points and may overestimate the dissimilarity. Both approaches will incur high computational costs when dealing with full-route trajectories comprising tens of thousands of points, reducing their efficiency and robustness. To address these challenges, this study introduces the sector edit distance, which represents

each trajectory as a sequence of sectors controlled by air traffic management. By transforming similarity measurement into sequence editing operations, the sector edit distance reduces computational complexity, better reflects controller decision-making preferences, and enhances clustering performance for long-sequence trajectories. The innovation of this research therefore lies in systematically comparing traditional distance measures with the proposed sector edit distance, and in demonstrating its effectiveness for generating trajectory option sets in city-pair operations.

This study proposes a trajectory clustering-based framework for generating city-pair trajectory options, as illustrated in Fig.1. Firstly, the planned path data and actual trajectory data are jointly processed to identify deviation trajectories, where a deviation marking method is applied to distinguish flights that significantly diverge from their planned paths. Secondly, these deviation trajectories are clustered using multiple similarity measurement methods combined with a DBSCAN-based clustering algorithm, in order to extract typical deviation patterns. Finally, the central trajectories of both deviation and non-deviation clusters are integrated to determine a trajectory option set, which incorporates commonly used deviation paths and provides alternative route options. The proposed TOS refers to a collection of typical flight paths preferred by airlines and pilots based on historical data, operational experience, and considerations of safety, economy, and punctuality. It aims to provide diversified references during flight planning while maintaining high acceptability in air traffic management. Trajectory options originate from two sources: Trajectories adhering to planned routes and frequently deviated trajectories from plans.

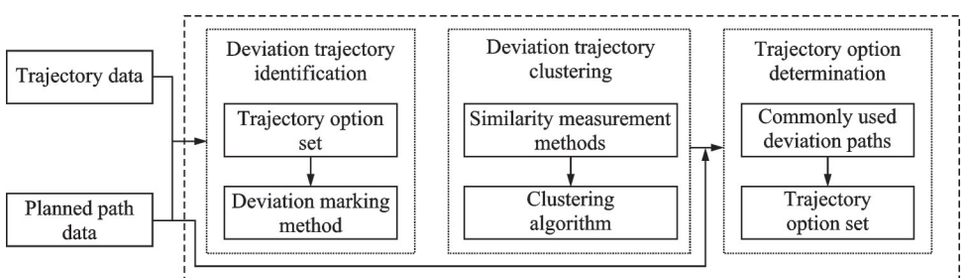


Fig.1 Framework for determining the trajectory option set

1 Flight Trajectory Option Identification Basic

1.1 Fundamentals of flight operations

Chinese flight plans follow the Civil Aviation Pre-flight Plan Management Regulations (CCAR-93)[20], and are prepared by the airline's operations control center based on seasonal schedules and real-time conditions (such as weather and airspace restrictions). The plan includes information such as flight number, aircraft type, route, altitude, and departure/arrival times. After approval by air traffic control (ATC), the plan is executed. The process includes airlines, based on seasonal schedules and real-time conditions (such as weather and airspace restrictions), and preparing flight plans through the operations control center, which include flight number, aircraft type, route, altitude, and expected departure/arrival times. These plans are submitted to ATC for approval at least 5 d before the flight, typically via AFTN/SITA messages. After the review, ATC allocates a calculated takeoff time (CTOT) slot and issues the permit. During the execution phase, the crew follows the planned route and coordinates with tower, terminal, and area control in sequence, reporting positions through data link or voice communication with controllers. In case of dynamic adjustments (such as flow control), ATC issues instructions, which the pilot follows. After landing, actual flight data is archived for

future optimization. The operational process of a flight is shown in Fig.2. The flight plan ensures operational safety and efficiency. The proposed trajectory option set can provide multiple path references for airlines when preparing flight plans, enhancing the adaptability of route selection. As shown in Fig.2, both ATC and airlines play central roles in the flight operation process. Airlines are responsible for preparing and submitting flight plans, while ATC approves, allocates time slots, and issues adjustment instructions during the execution. Deviation trajectories often arise from the collaborative interaction between ATC and airlines, reflecting both regulatory control and operational requirements.

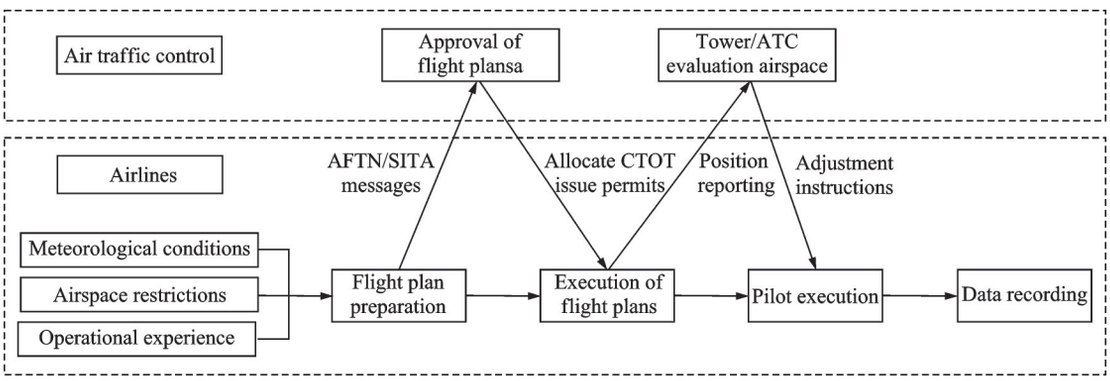


Fig.2 Flight plan operation process

1.2 Deviation trajectory features

To identify trajectory options that can provide more alternatives for reference when a flight faces complex airspace, this study's trajectory option set considers two types of flights: One that follows the planned trajectory as per the flight plan, and the other that deviates from the original planned path (deviation trajectory). During the flight, factors such as weather, airspace restrictions (e.g., temporary military no-fly zones), flow control (ATC adjustments to avoid congestion), navigation errors, airline optimization, and pilot operations may cause the flight to deviate from the planned route. Under strict definition, any deviation (such as navigation errors) is considered as a deviation from the planned trajectory.

However, the proposed trajectory option set must exhibit significant geographical variance to ensure that it does not fail entirely under different weather conditions. Therefore, under good weather and normal traffic conditions, flights typically follow the planned route, but slight deviations may occur due to route width tolerance and pilot operational errors. Such deviations are not considered deviations from the planned trajectory. This study focuses on intentional, long-distance deviation trajectories, as they reflect proactive decisions made by pilots and controllers in response to unexpected situations or operational optimizations, showing significant geographical deviations and strong tendencies. These deviation trajectories differ from minor, arbitrary deviations within a sector: When a flight operates within a sector, it is controlled by the sector's controller, and the flight has the flexibility to adjust its trajectory. Although it may not exactly match the planned route, it is still considered as a non-deviation trajectory. However, if the flight deviates into another sector, cross-sector coordination is required between controllers, and since the airspace covered by sectors is large, changing sectors often results in significant distance deviations, making it easier to recognize as a deviation trajectory.

Fig.3 illustrates the characteristics of deviation and non-deviation flights. In Fig.3(a), the flight does not perfectly align with the planned path, but there is no significant long-distance deviation from the planned route. In Fig.3(b), the flight experiences a long-range deviation due to strong convective weather in the southwestern part of Fujian, which is a clear and intentional deviation from the planned path. Such a trajectory is considered a deviation trajectory.

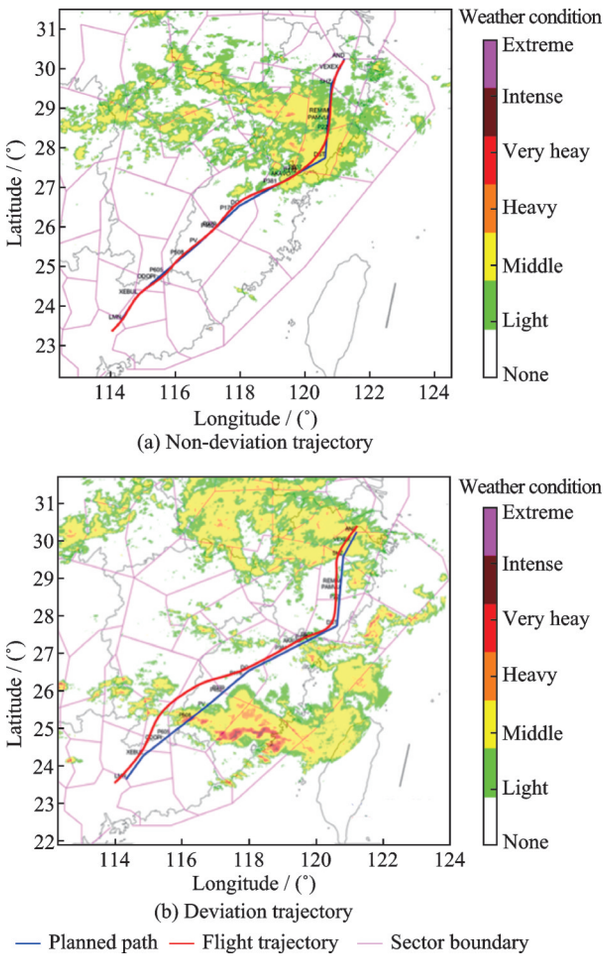


Fig.3 Schematic diagrams of deviation trajectory characteristics

2 Deviation Trajectory Identification

To identify cases where a flight deviates from the planned path under normal conditions, it is necessary to determine a threshold distance for such deviations. To accurately recognize the phenomenon of deviation, that is, under what circumstances a flight diverges from the planned path, this section employs statistical methods to establish the specific threshold for deviation behavior. This allows for the identification of deviation trajectories, laying the foundation for subsequent analysis of deviation paths.

2.1 Trajectory preprocessing

The raw data used in this study consists of radar fused trajectory data, including flight number, latitude and longitude of trajectory points, departure airport, destination airport, trajectory point altitude , time, speed, direction, and other relevant information. Due to the large volume of data per hour and the fact that the trajectory points are not completely continuous, the data processing poses certain challenges.

To improve the data processing efficiency, this study merges the hourly fusion trajectory data into daily data and extracts the trajectory of each flight to construct a complete flight path. For flights spanning multiple days, the data is processed based on the departure date. The merged data files typically contain a large number of trajectory points. To reduce computational resource consumption and improve the processing efficiency, the multi-processing technology is employed to process the daily data. Given that the trajectory data for a single flight is large, further memory optimization is achieved by increasing the time span between trajectory point samples for resampling.

Figs.4 and 5 show the trajectory of Flight CSN3569 before and after compression, respectively. The original number of trajectory points is 642, and after compression, it is reduced to 128 points. From Fig.5, it can be observed that the shape of the compressed trajectory differs slightly from the original trajectory only at the turns, while the overall shape remains highly consistent with the original trajectory.

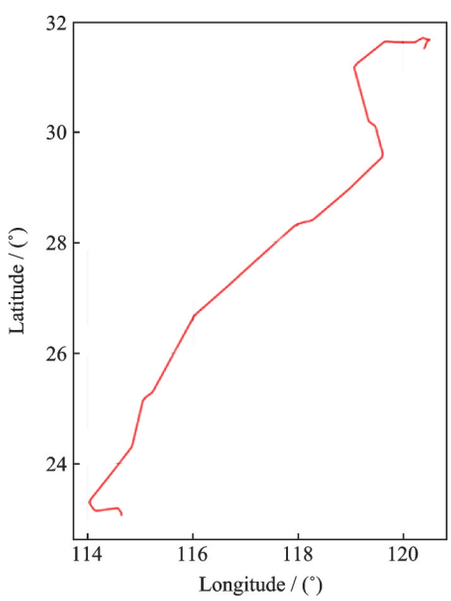


Fig.4 Original trajectory of CSN3569

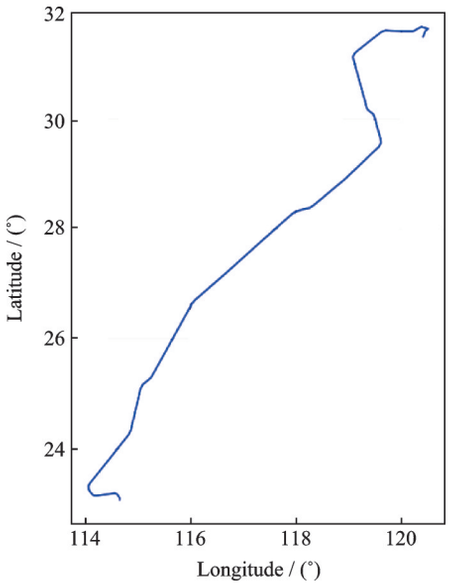


Fig.5 Compressed trajectory of CSN3569

2.2 Deviation threshold determination

To calculate the deviation distance of a flight, this study determines the centerline of each route segment based on the planned flight path. The planned flight path of a flight is typically composed of trajectory data from multiple segments. To match each trajectory point to its corresponding route segment, this study calculates the distance from each trajectory point to all route segments and selects the segment with the minimum distance as the matched segment for that trajectory point.

Based on this, the shortest distance from each trajectory point to the route centerline (the line segment formed by connecting adjacent route points of the planned path) is calculated, in order to quantify the deviation degree of the flight. Fig.6 provides a schematic for calculating trajectory deviation, which involves computing the Euclidean distance d between each flight trajectory point and the route centerline. If d exceeds a certain threshold, the trajectory point is considered to have deviated from the planned trajectory.

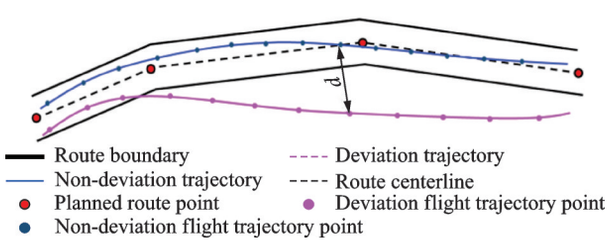


Fig.6 Schematic diagram of deviation trajectory calculation

To determine the deviation threshold, the quantile-inflection point method is applied. The procedure is as follows:

(1) Calculate deviation distances for all trajectory points relative to their planned routes.

(2) Select a high quantile (e.g., 80th percentile) as the initial candidate threshold, which filters out normal variations while retaining potential deviations.

(3) Incrementally increase the threshold, and at each step, compute the proportion of points exceeding the threshold to generate a frequency-threshold sequence.

(4) Construct the cumulative frequency curve and observe its growth trend.

(5) Identify the inflection point of the curve, where the growth rate slows significantly, as the deviation threshold.

This method ensures that the final threshold reflects both the statistical distribution of deviations and the operational distinction between normal variations and substantial trajectory deviations.

2.3 Deviation trajectory marking

This study proposes a method for identifying deviation trajectories to determine whether a flight's trajectory has deviated from its planned path. During the cruising phase, if the number of points deviating from the planned path exceeds the set threshold, the trajectory is considered a deviation. A single deviation point is not considered a deviation and the number of consecutive deviation points must be calculated. If the proportion of consecutive deviation points in the entire segment exceeds the threshold, the behavior is identified as a deviation.

Moreover, this study focuses on the set of trajectory options between cities, primarily determining whether the flight deviates from its path during the cruising phase. Since the trajectory points during the takeoff and landing phases are outside the flight plan path and controlled by airport flight procedures, these phases are not considered in this study. Assuming a trajectory sequence as shown in Fig.7, where 0 represents no deviation and 1 a deviation,

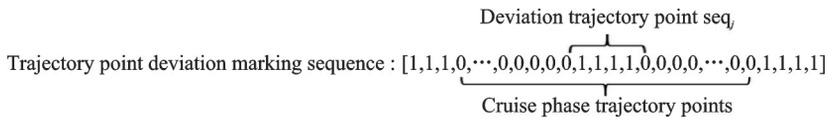


Fig.7 Trajectory deviation calculation

the calculation method for identifying a deviation trajectory follows

$$\text{Ratio}=\frac{\sum_{j=1}^{K}\text{seq}_j}{\sum_{i=1}^{N}P_i} \tag{1}$$

where Ratio represents the proportion of deviation points, P_i the ith trajectory point where the cruising altitude exceeds 6 000 m; seq_j the jth sequence of consecutive deviation points, with $\text{seq}_j>1$; N the total number of trajectory points during the cruise phase (altitude>6 000 m) and K the number of continuous deviation segments, where each segment consists of consecutive points deviating beyond the threshold. When Ratio exceeds the set threshold (Thre), the trajectory is considered a deviation trajectory.

For clarity, the procedure is summarized as follows. (1) Only cruise-phase points (altitude> 6 000 m) are retained. (2) For each point, the shortest distance to the planned route is computed. (3) A distance threshold d_{th} is determined using the quantile-inflection method. (4) Each point is binarized: 1 if its distance exceeds d_{th}, otherwise 0. (5) Runs of consecutive 1 s (seq_j) longer than one point are extracted, and Ratio in Eq.(1) is calculated. If Ratio exceeds ratio of Thre, the trajectory is classified as a deviation trajectory. (6) The deviation trajectory is mapped into a sector sequence and compared with the planned sequence. A change in the sector sequence not only reflects geometric deviation, but also indicates an ATC responsibility handover, thereby linking deviation trajectories to both spatial displacement and operational significance.

3 Trajectory Option Set Generation Method

3.1 Similarity measurement

In the clustering analysis of flight trajectories, accurate similarity measurement is the foundation for determining the similarity between trajectories. This study uses multiple similarity measurement methods to ensure precise evaluation of clustering results. Below are several commonly used similarity measurement methods.

3.1.1 Euclidean distance

When processing flight trajectory data, a trajectory consists of hundreds of trajectory points, and directly calculating the Euclidean distance between trajectories can result in high computational complexity. To address this issue, this study first applies the uniform manifold approximation and projection (UMAP) algorithm for dimensionality reduction of high-dimensional trajectory data. The core assumption of UMAP is that high-dimensional data points locally approximate some low-dimensional manifold, and the manifold is a geometric object locally resembling Euclidean space. Through this mapping, UMAP can reveal the underlying structure of high-dimensional data, while retaining both global and local geometric information, thus reducing the computational complexity when calculating distances.

In the reduced low-dimensional space, the dimensionality of trajectory points is significantly reduced, making the computation of Euclidean distances between trajectories more efficient. The Euclidean distance between two trajectories is calculated by

$$d(x,y)=\sqrt{\sum_{i=1}^{n}(x_i-y_i)^2} \tag{2}$$

where x and y represent two dimensionality-reduced trajectory points, and n is the reduced space dimension. By reducing the dimensionality, the computational load when calculating Euclidean distances is minimized, thereby effectively improving the efficiency while maintaining an effective measure of similarity between trajectories.

3.1.2 Hausdorff distance

The Hausdorff distance is a measure of the similarity between two sets, especially suitable for handling trajectories with a different number of points. In flight trajectory clustering, the Hausdorff distance can measure the maximum distance between two trajectories, specifically the distance between the farthest points on the two trajectories. Let A and B represent two trajectories, with their respective point sets being $A=\{a_1, a_2, \cdots, a_m\}$ and $B=\{b_1, b_2, \cdots, b_n\}$, the Hausdorff distance between the two sets is defined as

$$d_{\mathrm{H}}(A,B)=\max\left(\sup_{a\in A}\inf_{b\in B}d(a,b), \sup_{b\in B}\inf_{a\in A}d(a,b)\right) \tag{3}$$

where $d(a,b)$ represents the distance between points a and b, sup the supremum (the least upper bound), and inf the infimum (the greatest lower bound). The Hausdorff distance calculation takes into account the maximum distance between trajectory points, making it suitable for irregular or differently sized trajectory data. It helps identify whether two trajectories have similarity in shape or path, and it is commonly used as a distance metric in flight trajectory clustering.

3.1.3 Sector edit distance

In flight trajectory similarity measurement, the Euclidean distance, while computationally intuitive and straightforward, requires two trajectories to be of equal length (having the same number of points) and necessitates a strict one-to-one correspondence between points. It is highly sensitive to local deviations. When applied to long-sequence full-flight trajectories, significant deviations at individual points can markedly inflate the overall distance. The Hausdorff distance, capable of measuring the maximum deviation between the shapes of two trajectories, does not require the number of points to be identical and is suitable for comparing irregular trajectories. However, it is sensitive to outliers and incurs high computational overhead when processing large-scale data. Consequently, the above two traditional methods exhibit certain limitations in clustering long-sequence, city-pair flight trajectories.

To address these issues, this paper introduces the sector edit distance. This method represents flight trajectories as sequences of transited sectors and measures the dissimilarity between trajectories through sequence edit operations. This approach not only effectively reduces computational complexity but also captures the logic of air traffic control sector division and managerial decision-making, making it more suitable for the clustering and analysis of long-range trajectories.

The edit distance refers to the minimum number of editing operations required to convert one string into another. The types of operations include inserting a single character, deleting a single character, and replacing a single character. This distance measurement is relatively simple and intuitive, and it effectively reflects the similarity between two strings. The calculation of the number of operations is typically achieved using dynamic programming.

The construction of the distance matrix based on edit distance is as follows. Firstly, for a given set of strings $S=\{s_1, s_2, \cdots, s_n\}$, the edit distance between each pair of strings needs to be calculated. Then, the dynamic programming method is used to compute the edit distance, utilizing a two-dimensional array edit[i][j] to represent the edit distance between the first i characters of string s_j and the first j characters of string s_j.

The steps for calculating the edit distance are as follows.

(1) Initialize edit[0][0] = 0.

(2) For the distance between an empty string and a non-empty string

$$\begin{cases}\text{edit}[i][0]=i(i\text{ characters need to be deleted})\\ \text{edit}[0][j]=j(j\text{ characters need to be inserted})\end{cases} \tag{4}$$

(3) Fill the entire matrix based on the dynamic programming equation. For strings A and B, where the indices of strings start from 1:

If $A[i-1]=B[j-1]$, edit[i][j] = edit[$i-1$][$j-1$];

If $A[i-1]\neq B[j-1]$

$$\text{edit}[i][j]=\min\begin{cases}\text{edit}[i-1][j]+1(\text{delete})\\ \text{edit}[i][j-1]+1(\text{insert})\\ \text{edit}[i-1][j-1]+1(\text{replace})\end{cases} \tag{5}$$

The sector distance described in this paper is based on edit distance, where each trajectory is represented as a sequence of sectors. Fig.8 illustrates how the trajectory of Flight CCA1734 is mapped to a sector string representation based on the sectors it passes through. When calculating the distance, each operation is not applied to a single character but rather to modify the sector string, where the sector string is the basic element in the trajectory sequence.

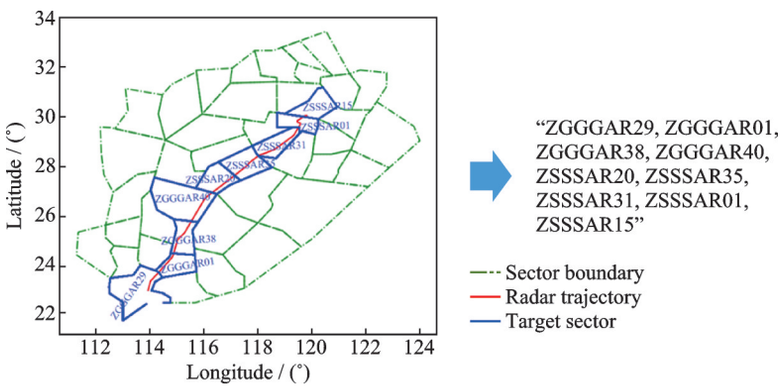


Fig.8 Sector sequence representation of Flight CCA1734

The use of sector edit distance to measure the similarity between flight trajectories transforms the editing operations of individual trajectory points into edit operations based on sector sequences. Since each trajectory typically involves no more than ten sectors, and these sectors are formed through coordination and handover by air traffic controllers, this approach not only avoids high computational complexity encountered in traditional methods for long trajectory similarity calculations, but also reflects the decision-making preferences of air traffic controllers.

Sector edit distance emphasizes local features and control strategies. By focusing on the transitions of trajectories between different control sectors, it effectively captures local differences and reflects the decision-making characteristics of controllers. This method provides a more reasonable and computationally efficient similarity measurement and clustering for the entire flight trajectory, considering the tendencies of air traffic controllers, thereby compensating for the limitations of traditional methods.

3.2 Clustering algorithm

In the field of machine learning, the unsupervised learning does not rely on labeled sample information. Its core goal is to reveal the intrinsic structure and patterns of data by analyzing unlabeled data, laying the foundation for subsequent data analysis. To mine commonly used deviation trajectories from trajectory data, clustering techniques in unsupervised learning are essential, and this method has been widely used in trajectory recognition.

The DBSCAN is a density-based clustering algorithm. Fig.9 illustrates the principle of DBSCAN clustering. This algorithm groups samples into clusters based on regions with high-density sample points in the sample space. It relies on two key parameters: ε (Epsilon, also denoted as Eps), which defines the neighborhood radius, and MinPts, the minimum number of points required to form a cluster. The specific process of the algorithm is as follows.

(1) Type determination: Eqs.(6, 7) provide the criteria for identifying core sample points. For any sample p in the dataset D, if the number of neighbors within its radius ε is greater than or equal to MinPts, p is considered a core point; if the number of neighbors is less than MinPts, but p lies within the neighborhood of a core point, it is considered a border point. Otherwise, it is treated as a noise point.

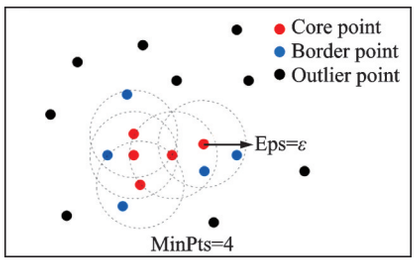


Fig.9 Schematic diagram of DBSCAN clustering principle

(2) Cluster formation: From the dataset D, arbitrarily select an unclassified core point q, and mark q and all core and border points within its ε neighborhood as belonging to class i. For all core points in class i, further classify all points in their ε neighborhoods as belonging to class i. This process repeats until no more core or border points are found to belong to class i.

(3) Dataset update: Remove the already classified samples from D and set $i=i+1$.

(4) Repeat clustering: Continue Steps 2 and 3 until all core and border points have been classified (noise points are not assigned to any class).

$$N_\varepsilon(p)=\{q\in D|\text{dist}(p,q)\leqslant\varepsilon\} \tag{6}$$

$$|N_\varepsilon(p)|\geqslant\text{MinPts} \tag{7}$$

where dist (p, q) represents the distance between points p and q, and $|N_\varepsilon(p)|$ the number of points within the ε neighborhood of p. In this study, the ε neighborhood represents the similarity distance between each trajectory and other trajectory samples. Using different trajectory similarity metrics, the same neighborhood radius can produce different results. This study selects the DBSCAN algorithm over traditional clustering methods such as K-means, primarily based on the following considerations: Unlike algorithms such as K-means, DBSCAN does not require predefining the number of clusters. Given that the number of deviation patterns between Guangzhou and Shanghai is unknown, the DBSCAN can automatically detect clusters of any shape and size. Additionally, the DBSCAN can effectively identify and exclude noise points, which are prevalent in radar data due to anomalies or errors, ensuring that the analysis focuses solely on meaningful deviation patterns.

3.3 Trajectory option set determination method

This study aims to systematically identify common trajectory patterns and construct a set of flight trajectory options based on these patterns through clustering analysis of flight deviation trajectories. First, clustering algorithms are used to classify deviation trajectories and identify different trajectory clusters. Then, for each cluster, the centerline is fitted using polynomial methods to represent its trajectory characteristics. Finally, the resulting center trajectories are used to generate trajectory option sets.

However, considering that the non-deviation trajectories in actual flight operations do not perfectly align with the planned route, in addition to deviation trajectories, it is necessary to calculate the center trajectory for the actual flight trajectories under each planned route. For each planned route, the same polynomial fitting method is applied to calculate the center trajectory. This method helps identify the regular deviations and patterns of flights along the planned route in different flight scenarios. Ultimately, by combining the center trajectories of deviation trajectories and planned routes, a set of flight trajectory options is constructed. The specific steps are as follows.

(1) Deviation trajectory clustering: Use clustering algorithms to classify deviation trajectories and identify different trajectory clusters.

(2) Deviation trajectory centerline calculation: For each deviation trajectory cluster, calculate the center trajectory using polynomial fitting and the least squares method.

Suppose a cluster contains N trajectories, and each trajectory consists of M_i trajectory points. The coordinates of the points are denoted as (x_{ij}, y_{ij}), where $i=1, 2, \cdots, N$ and $j=1, 2, \cdots, M_i$. The fitting function for the central trajectory can be expressed as

$$y=f(x)=a_0+a_1x+a_2x^2+\cdots+a_kx^k \tag{8}$$

where k is the order of the fitted polynomial. The parameter vector $\boldsymbol{a}=(a_0, a_1, \cdots, a_k)$ is determined using the least squares criterion shown as

$$\min_{\boldsymbol{a}}\sum_{i=1}^{N}\sum_{j=1}^{M_i}(y_{ij}-f(x_{ij};\boldsymbol{a}))^2 \tag{9}$$

Solving this optimization problem yields the central trajectory function. Finally, all points in the trajectory cluster are projected onto the fitted curve to obtain the central trajectory of the cluster.

(3) Planned route centerline calculation: For each planned route's actual flight trajectory, apply the same fitting method to calculate the center trajectory.

(4) Trajectory option set generation: Integrate the center trajectories of deviation trajectories and planned routes to generate the trajectory option set.

4 Case Study

4.1 Experimental data and clustering evaluation metrics

The Pearl River Delta and Yangtze River Delta are China's most economically developed regions, home to mega-cities and their respective airport clusters. Table 1 lists the airports included in each cluster along with their ICAO codes. The distance between these two regions is approximately 1 200 km, with complex terrain (such as the Nanling and Huangshan mountains), and long ground transportation times (approximately 8—10 h by high-speed rail), making air travel essential. As a result, the flight route from Guangzhou to the Shanghai airport cluster has become one of the busiest flight routes in China. The airspace in this region is complex, and the route is often affected by typhoons during the summer, with significant deviation phenomena, high flight demand, and diverse patterns, exhibiting rich flight behavior characteristics. Therefore, this study focuses on the flight trajectories between Guangzhou and the Shanghai airport cluster, analyzing their trajectory option set.

Table 1 Airports involved in each airport cluster

Guangzhou airport cluster	ICAO code	Shanghai airport cluster	ICAO code
Guangzhou Baiyun International Airport	ZGGG	Shanghai Hongqiao International Airport	ZSSS
Shenzhen Bao'an International Airport	ZGSZ	Shanghai Pudong International Airport	ZSPD
Zhuhai Jinwan Airport	ZGSD	Nanjing Lukou International Airport	ZSNJ
Huizhou Pingtan Airport	ZGHZ	Hangzhou Xiaoshan International Airport	ZSHC
		Wuxi Shuofang Airport	ZSWX

Fig.10 shows the planned airways from Guangzhou airport cluster to Shanghai airport cluster, along with the involved sectors. Each integrated trajectory file used in this study contains approximately 300 000 trajectory data points per hour, and the merged daily trajectory data typically exceeds seven million points. The study period is set from 6 April to 15 July, 2023, when convective weather was frequent, providing representative conditions for deviation analysis. The dataset consists of flights from the Guangzhou airport cluster to the Shanghai airport cluster. After screening and cleaning, including the removal of incomplete or abnormal trajectories, 23 578 valid flight trajectories are retained. Fig.11 shows the distribution of trajectories during this period, with the densest trajectories located near the

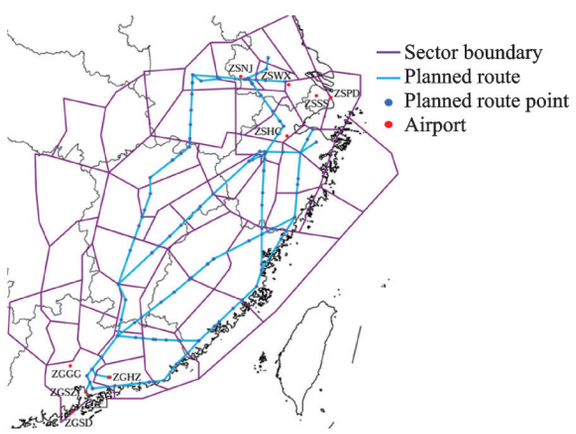


Fig.10 Distribution of Guangzhou/Shanghai airport clusters and sectors

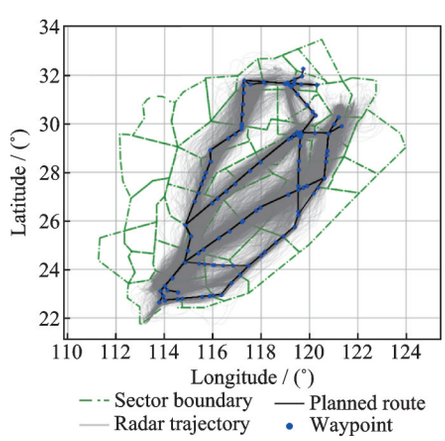


Fig.11 Planned routes and trajectory distribution from Guangzhou airport cluster to Shanghai airport cluster

planned route, although some trajectories are concentrated in non-planned route areas.

To identify the deviation threshold, this study selects five days in 2023 (April 16th, 17th, May 1st, 28th, and July 11th) in the East China and Central South regions, with no weather influence, covering flights between major cities such as Shanghai, Nanjing, Hangzhou, Ningbo, Xiamen, Fuzhou, Nanchang, Changsha, Guangzhou, and Shenzhen. Table 2 shows the filtered radar data, which results in 3 456 trajectory data points, corresponding to 3 456 flights.

Table 2 Flight information table

Date	Number of flights
16 April	711
17 April	690
1 May	633
28 May	703
11 July	719
Total	3 456

In the study of flight trajectory clustering, evaluating the clustering quality is crucial, as high-quality clusters can effectively reveal the underlying structure of the data. An ideal clustering result should have high similarity within clusters and low similarity between clusters. Therefore, this study uses various quality evaluation metrics. These metrics measure clustering performance by analyzing the distance between sample points and the cluster center, without relying on external benchmark datasets or reference models.

To assess the clustering quality, this study adopts the silhouette coefficient as an important evaluation criterion for the clustering method. The silhouette coefficient is a measure of clustering quality that reflects both the cohesion of samples within a cluster and the separation between clusters. The silhouette coefficient ranges from $[-1, 1]$, with values closer to 1 indicating better clustering, where the samples within a cluster are highly similar and the distinction between clusters is large. A value closer to -1 indicates poor clustering with substantial overlap between clusters. The silhouette coefficient S for a given sample can be calculated by

$$S=\frac{b-a}{\max(a,b)} \tag{10}$$

where a is the average distance between the sample and other samples within its own cluster, and b the average distance between the sample and the closest sample from any other cluster. The sum of the silhouette coefficient (SC) is used to evaluate the overall quality of the clustering result. The calculation process is given by

$$\mathrm{SC}=\frac{1}{n}\sum_{i=1}^{n}S_i \tag{11}$$

When the silhouette coefficient is close to 0, it suggests that there may be overlap between clusters, indicating poor clustering performance. When the silhouette coefficient is large, it indicates that the samples within the cluster are more concentrated and the separation between clusters is clearer.

Additionally, the Davies-Bouldin index (DBI) and Calinski-Harabasz index (CHI) are two other commonly used clustering performance evaluation metrics, which are often used to compare the effectiveness of different clustering algorithms. The DBI measures the quality of the clustering result by calculating the dispersion within clusters and the distance between clusters. A smaller DBI value indicates that the samples within a cluster are tighter and the separation between clusters is greater. Its calculation formula is shown as

$$\mathrm{DBI}=\frac{1}{K}\sum_{i=1}^{K}\max\left(\frac{s_i+s_j}{M_{ij}}\right) \tag{12}$$

where s_i represents the dispersion of sample points within the ith cluster, and d the distance between the centers of clusters i and j. A smaller DBI value indicates better clustering performance, meaning higher intra-cluster compactness and greater inter-cluster separation.

The CHI index evaluates the clustering performance by calculating the ratio of the inter-cluster variance to the intra-cluster variance. The higher the CHI index, the better the clustering performance, as it indicates higher intra-cluster compactness and greater inter-cluster separation. Its calculation formula is shown as

$$\mathrm{CHI}=\frac{\mathrm{tr}(\boldsymbol{B}_k)(N-K)}{\mathrm{tr}(\boldsymbol{W}_k)(K-1)} \tag{13}$$

where $\boldsymbol{B}_k$ represents the between-cluster covariance matrix, $\boldsymbol{W}_k$ the within-cluster data covariance matrix, N the total number of samples in the dataset, and K the number of clusters. Using Eq.(13), the

CHI index quantifies the quality of the clustering results.

4.2 Deviation trajectory identification and analysis

The distance calculation is implemented using geometric methods. For each trajectory point, this study calculates the shortest distance from the trajectory point to the flight path centerline using the Euclidean distance. After organizing the calculated deviation distance data, a total of 1 973 573 trajectory sample points' distances to the planned path centerline are obtained. Fig.12 shows the histogram distribution of the deviation distances between flight trajectories and the centerline, where the largest number of trajectories falls within a deviation distance of 5 km, with frequencies exceeding 200 000. The number of trajectory points with a deviation distance greater than 10 km significantly drops, with frequencies within 30 000.

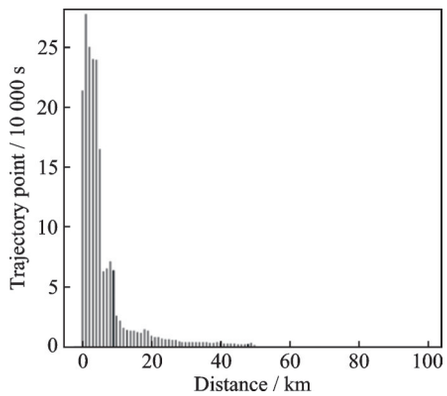


Fig.12 Histogram distribution of trajectory point deviation distances

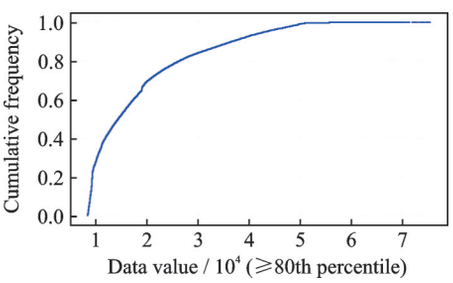


Fig.13 Cumulative frequency relationship at the 80th percentile

According to the results in Fig.13, starting from the 80th percentile, as the deviation distance increases, the frequency initially rises rapidly. Before the deviation distance reaches 20 km, there is a sudden change in the growth rate of the cumulative frequency. After this point, the growth rate of the cumulative frequency gradually slows down, thus the final deviation threshold is determined to be 19.2 km.

This study also analyzes the threshold for the number of yawed trajectory points, denoted as Thre. The value of Thre is set within the range [20%—60%] to discuss the number of yawed trajectories. Fig.14 shows the trajectory distribution corresponding to different thresholds, while Fig.15 shows the trajectory count distribution corresponding to the Thre threshold.

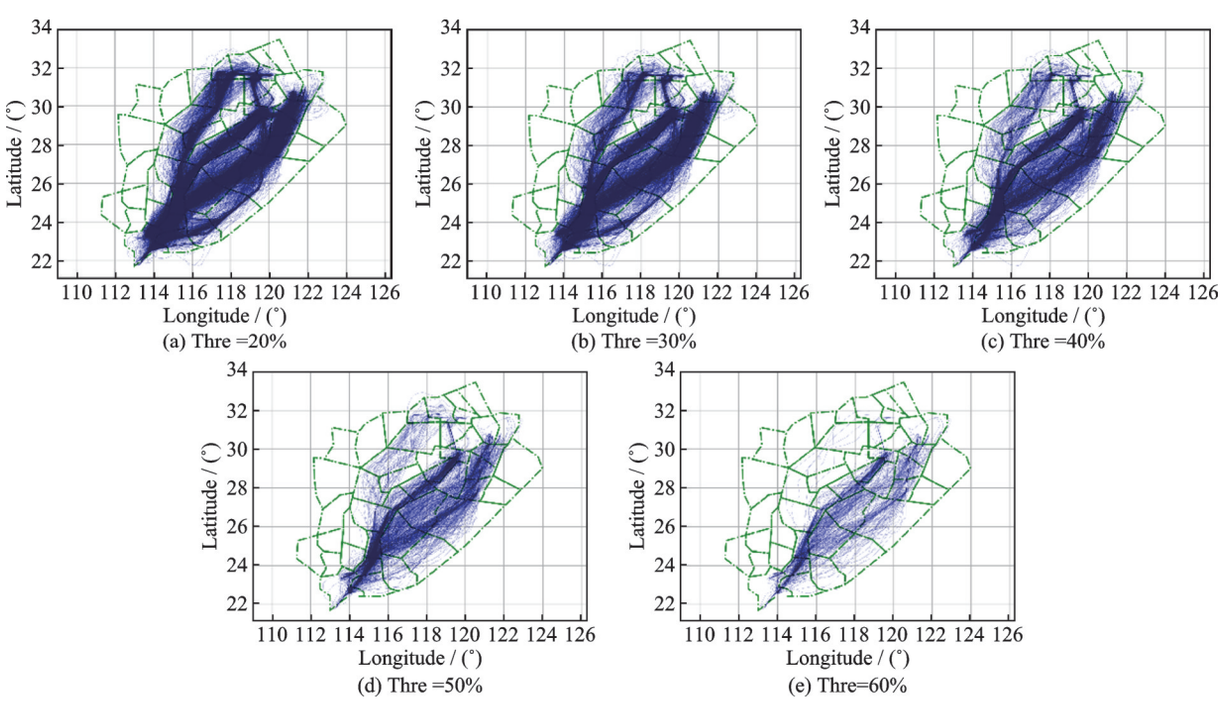


Fig.14 Trajectory distributions corresponding to different Thre thresholds

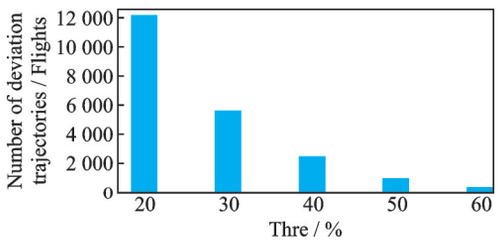


Fig.15 Relationship between Thre threshold and number of deviation trajectories

By combining the results from Figs.13 and 16, it can be seen that most yawed behaviors occur in a specific segment of the flight, with the remainder of the flight still following the planned path. When the Thre threshold is set too low (i.e., when Thre $\leqslant$ 30%), it leads to the recognition of flights that largely follow the planned path as yawed trajectories, which is not conducive to the subsequent use of clustering methods to distinguish yawed trajectories from planned trajectories. On the other hand, when the Thre threshold is set too high (i.e., when Thre$\geqslant$50%), the number of identified yawed trajectories becomes too small, resulting in some cases where trajectories significantly deviating from the original planned path are incorrectly classified as non-deviated. When Thre=40%, the identified yawed trajectory results align more closely with actual conditions. Therefore, the Thre threshold is set to 40%, resulting in 2 436 identified yawed trajectories.

4.3 Clustering experiment result analysis

To determine the optimal hyperparameters for DBSCAN, this study employed a grid search method to evaluate clustering performance under different similarity measures. The parameter range for the grid search is set as: Neighborhood radius $\varepsilon \in [0.1, 1]$, and minimum number of samples MinPts $\in [3, 10]$. The performance metrics from various parameter combinations obtained through grid search (as shown in the evaluation results based on Euclidean distance) are analyzed using DBI, CHI, and silhouette coefficient.

According to the results in Fig.16, when $\varepsilon \in [0.7, 1]$ and MinPts $\in [3, 6]$, the DBI reaches its minimum value of 0.25, indicating high intra-cluster compactness and significant inter-cluster separation.

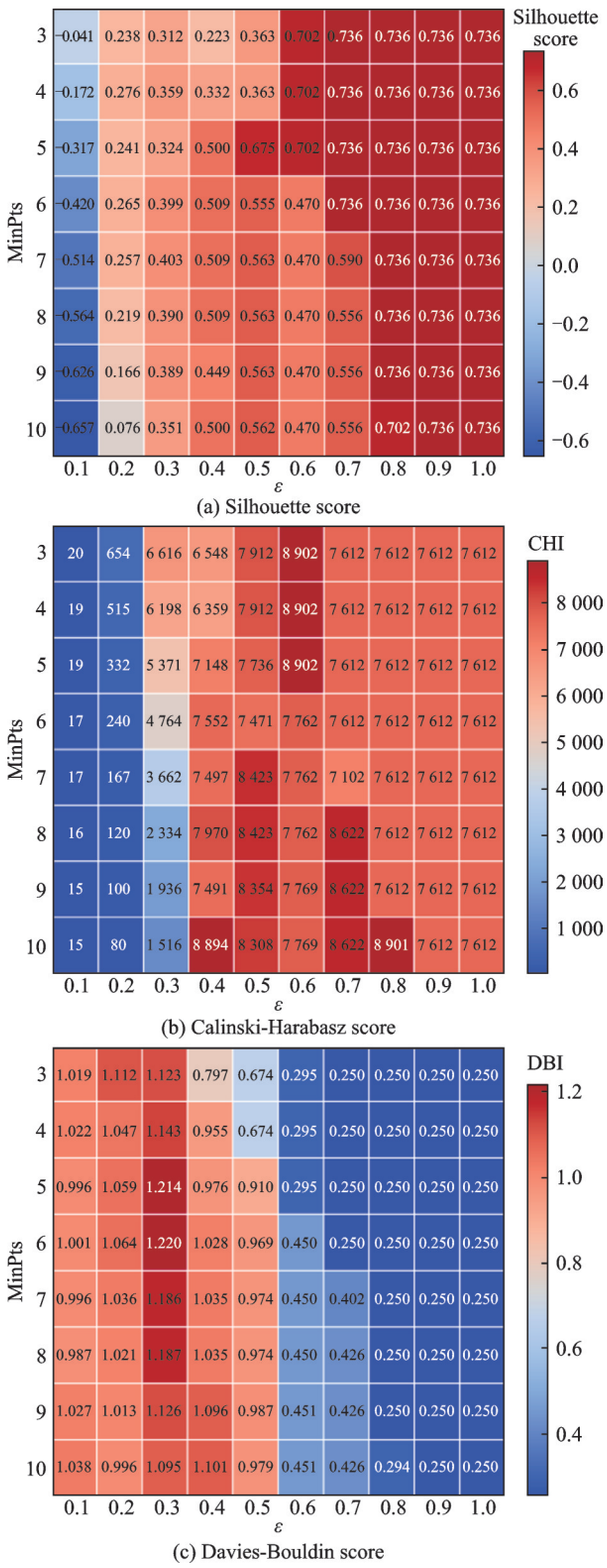


Fig.16 Results of Euclidean distance metrics

Within this range, the silhouette coefficient reaches 0.736, further confirming the superiority of the clustering performance. However, the CHI index shows that when $\varepsilon = 0.4$ and MinPts $\in [3, 6]$, intra-cluster compactness is optimal (CHI=8 901).

Under this parameter combination, the DBI and silhouette coefficient are 0.295 and 0.703, respectively, slightly lower than the optimal values, but the difference is minimal. The optimal parameter range for DBI and silhouette coefficient has a corresponding CHI value of 7 612, which differes significantly from the maximum CHI value.

Considering the consistency of the three metrics and the actual clustering performance, this study ultimately selects $\varepsilon=0.6$ and MinPts=3 as the optimal hyperparameters. This parameter combination performs nearly optimally on DBI and silhouette coefficient while achieving a relatively high CHI value, balancing intra-cluster compactness and inter-cluster separation. The corresponding clustering results are shown in Fig.17, where eight clusters are identified. Overall, the clusters are clearly separated, but Labels 2 and 3 show overly coarse distributions, requiring further separation.

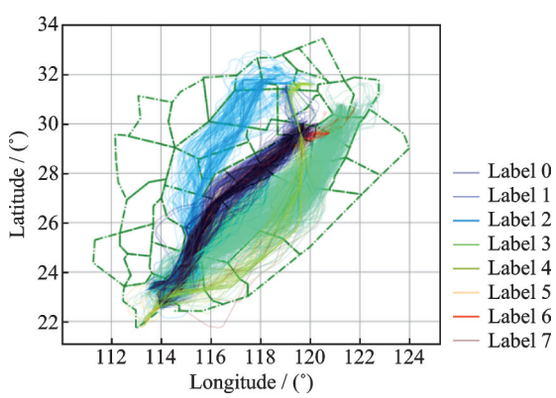


Fig.17 Clustering results based on Euclidean distance

Fig.18 shows the trend of the performance metrics corresponding to the clustering hyperparameters based on Hausdorff distance. Similar to the results with Euclidean distance, the trend follows a comparable pattern, though there are numerical differences. The significant difference lies in the DBI and CHI optimal values for Hausdorff distance (DBI=0.225, CHI=19 111), which are notably better than the results based on Euclidean distance. This suggests that shape-based trajectory similarity measures, such as Hausdorff distance, outperform traditional Euclidean distance in assessing the quality of flight trajectory clustering and are better suited to the spatial characteristics of trajectories. Nevertheless, the optimal parameter range for Hausdorff distance is consistent with that for Euclidean distance ($\varepsilon\in[0.7, 1]$, MinPts$\in[3, 6]$). Considering consistency and computational efficiency, this study still selects $\varepsilon=0.6$ and MinPts=3 as the hyperparameters.

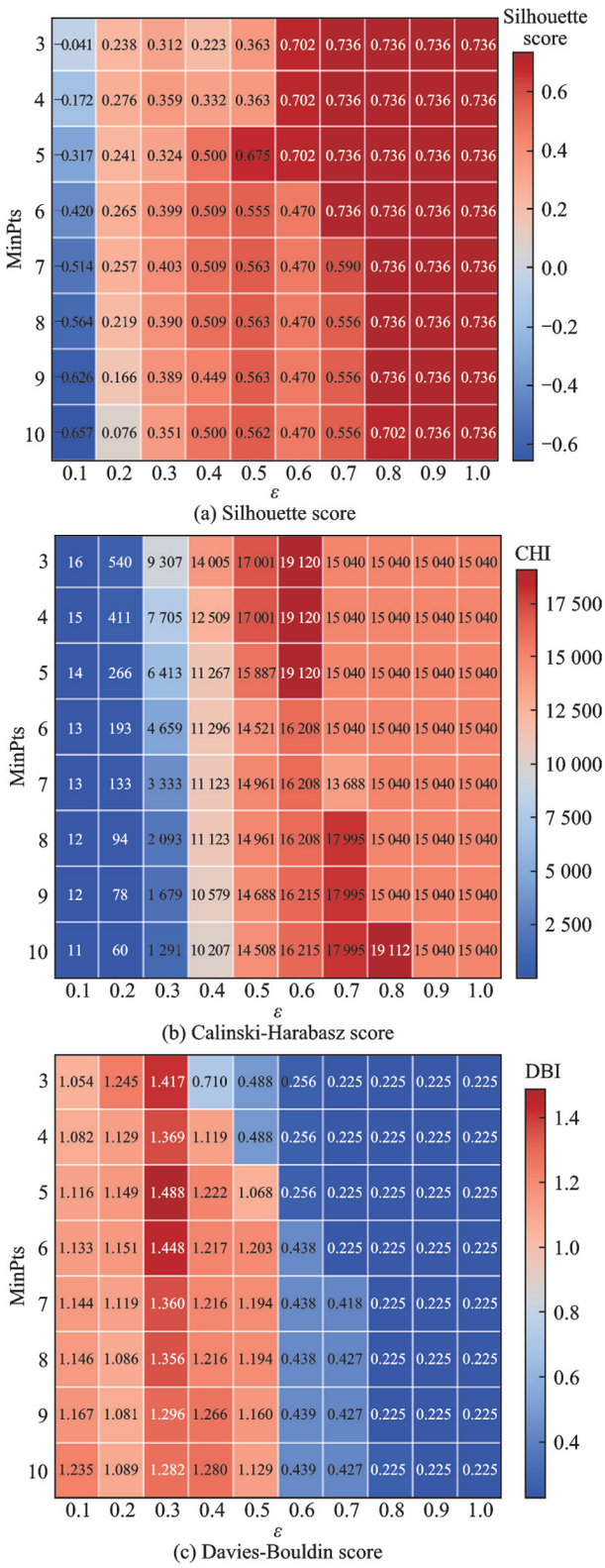


Fig.18 Results of Hausdorff distance metrics

The clustering results using Hausdorff distance are shown in Fig.19, where nine clusters are identified. Compared to the Euclidean distance clustering, this method identifies more clusters in the Label 0 region, demonstrating higher discriminative ability. However, the distribution of Labels 2 and 3 remains too coarse, with insufficient separation, indicating that clustering performance still needs further optimization in certain regions.

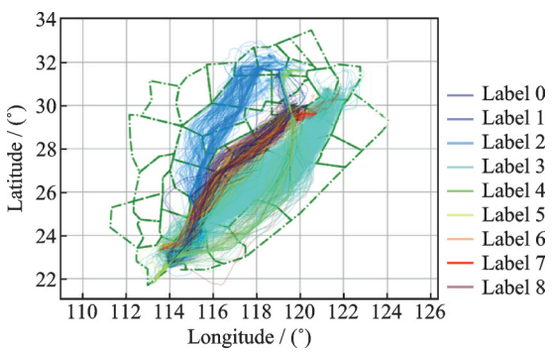


Fig.19 Clustering results based on Hausdorff distance

Based on the results from Fig.20, the parameters for the sector edit distance method are set as $\varepsilon=0.7$ and MinPts=3. Compared to the clustering methods based on Euclidean distance and Hausdorff distance, this method demonstrates better clustering performance. The neighborhood radius is smaller ($\varepsilon=0.4$), and the number of core sample points is higher (MinPts=6). Additionally, in terms of clustering evaluation metrics, both the CHI and DBI outperform the first two methods, indicating that the sector edit distance achieves a higher level of intra-cluster compactness and inter-cluster separation.

The clustering results are shown in Fig.21, where the sector edit distance successfully classifies the trajectories on the northwest and southwest sides of the Guangzhou to Shanghai route. The first two methods fails to further segment these areas, recognizing them only as a single cluster. In contrast, the sector edit distance method divides these areas into five distinct clusters (Labels 3, 6, 9, 13, 19), demonstrating a stronger discriminatory capability. Among them, the trajectories in Label 9 are more dispersed, likely due to the wide range of the individual sector passed, which results in multiple

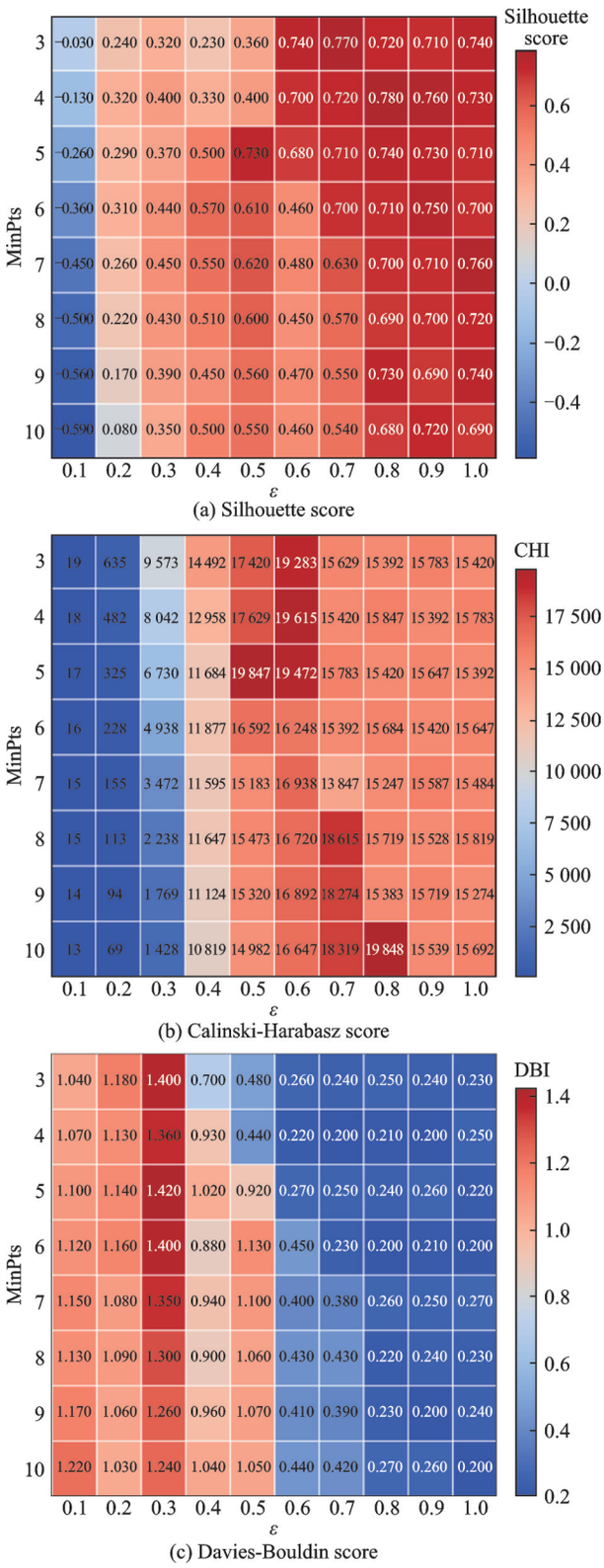


Fig.20 Results of sector edit distance metrics

paths within the same sector. Although the edit distance might consider these paths similar, this method still significantly improves trajectory classification compared to the previous two. In summary, experimental results of the three similarity measures

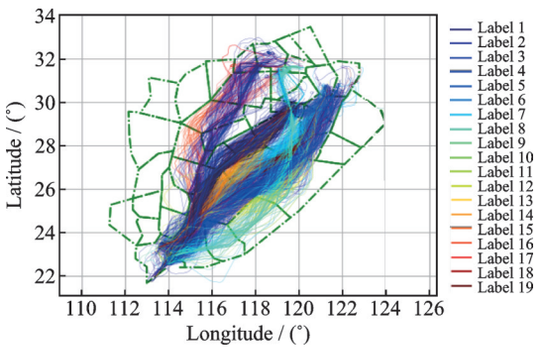


Fig.21 Clustering results based on sector edit distance

are consistent with the theoretical analysis. Euclidean distance is sensitive to local fluctuations and identified fewer clusters, while Hausdorff distance improves cluster discrimination in some regions but still shows coarse separation. The sector edit distance demonstrats superior performance by identifying representative deviation patterns more effectively, as reflected in higher CHI values and lower DBI values. Therefore, this study adopts sector edit distance as the measure of trajectory similarity and uses the clustering results obtained from this method to generate the trajectory options set.

4.4 Trajectory option set generation

Through three trajectory similarity measures (sector similarity, Euclidean distance, and Hausdorff distance), combined with the DBSCAN clustering results under optimal parameters, it is evident that the sector similarity-based method outperforms the traditional Euclidean distance and Hausdorff distance in clustering effectiveness. Using this method, this study identifies several typical deviation paths commonly used by flights. Additionally, the analysis shows that although most flight trajectories still follow the planned path, flights do not strictly adhere to the planned path during actual flight. To address, it this study employs polynomial fitting on non-deviation trajectories to generate the central trajectory for each planned path, replacing the original planned path, thereby forming path options that are closer to actual flight conditions. By combining the central trajectories of non-deviation paths with commonly used deviation trajectories, a set of flight path options from the Guangzhou city cluster to the Shanghai airport cluster is ultimately generated.

There are as many as 49 planned paths from the Guangzhou airport cluster to the Shanghai airport cluster. As analyzed in Fig.11, most planned paths largely overlap, with only slight differences in route points during the takeoff and landing phases. For example, Route A may differ from Route B due to the presence of one additional route point; similarly, the planned path from Guangzhou to Nanjing is almost identical to the planned path from Guangzhou to Shanghai Pudong Airport, except that the latter path includes one additional route point at the end. Moreover, certain paths, while differing due to an additional route point in the segment or due to differences in airport locations during takeoff and landing, often share the same route during the cruising phase. These factors collectively result in a large number of planned paths. However, considering that the cruising phase paths are essentially highly consistent, this study categorizes the 49 planned paths into nine main types based on the characteristics of the cruising phase. The trajectory distribution shown in Fig.22 reduces redundant trajectories and highlights the core patterns of the flight routes.

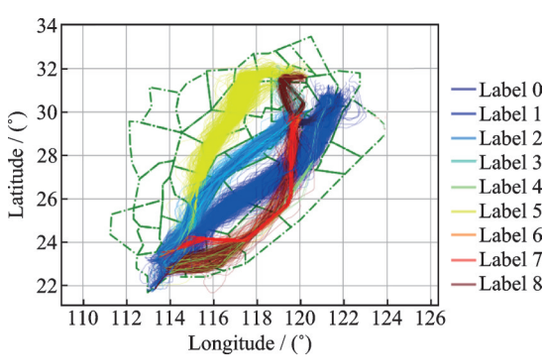


Fig.22 Nine types of non-deviation flight trajectories

Fig.23 shows the central trajectories of nine types of non-deviation flights based on the polynomial fitting method. Fig.24 presents the central trajectories of 19 clusters of identified deviation trajectories obtained using the same method. The flight clusters of non-deviation trajectories are relatively concentrated around the central trajectory, while the deviation trajectories are more dispersed. However, the clusters derived from the sector similarity meth-

od are more concentrated compared to those obtained using Euclidean distance and Hausdorff distance.

Comparing Figs.23 and 24, the central trajectories of the deviation clusters show some similarity to the central trajectories of flights that follow the

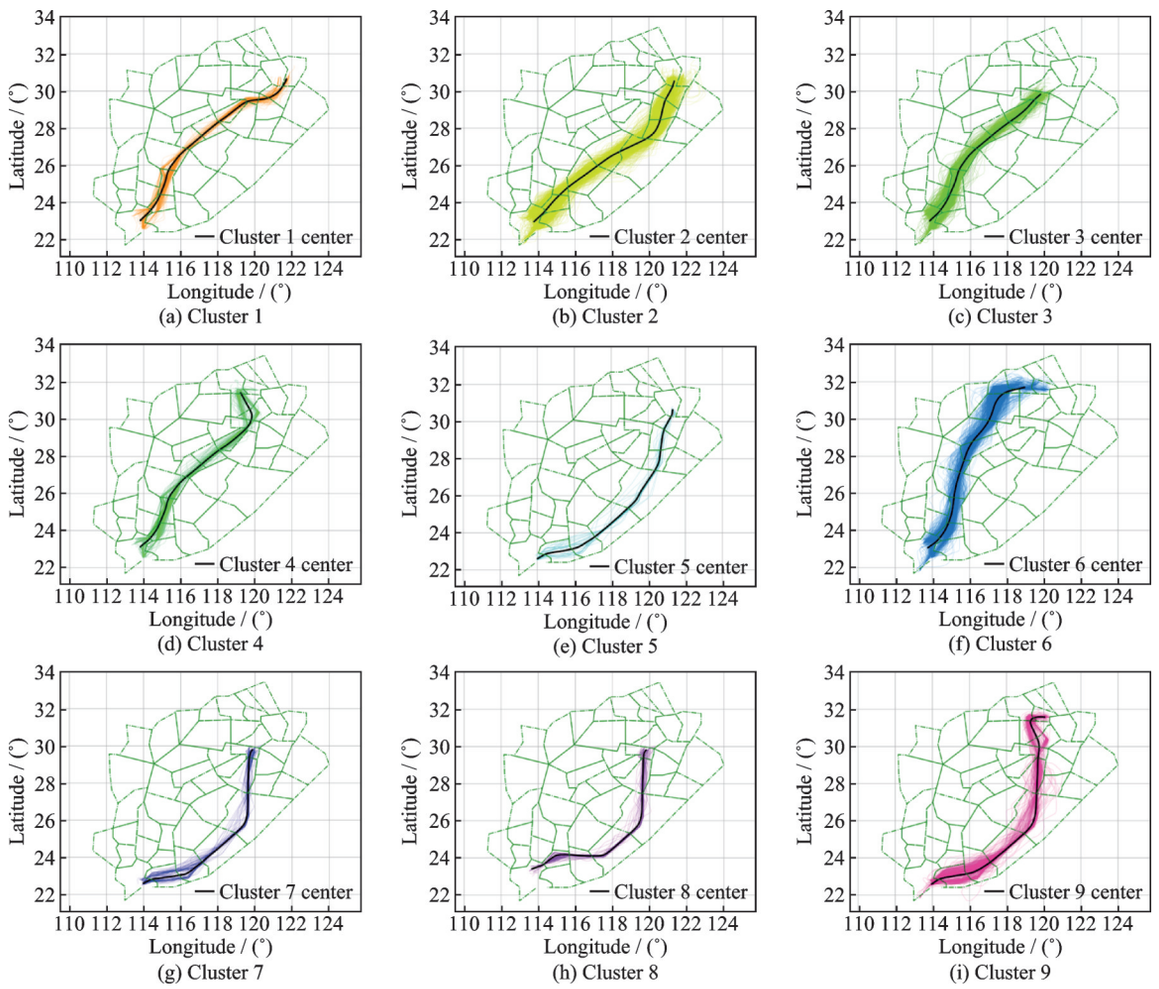
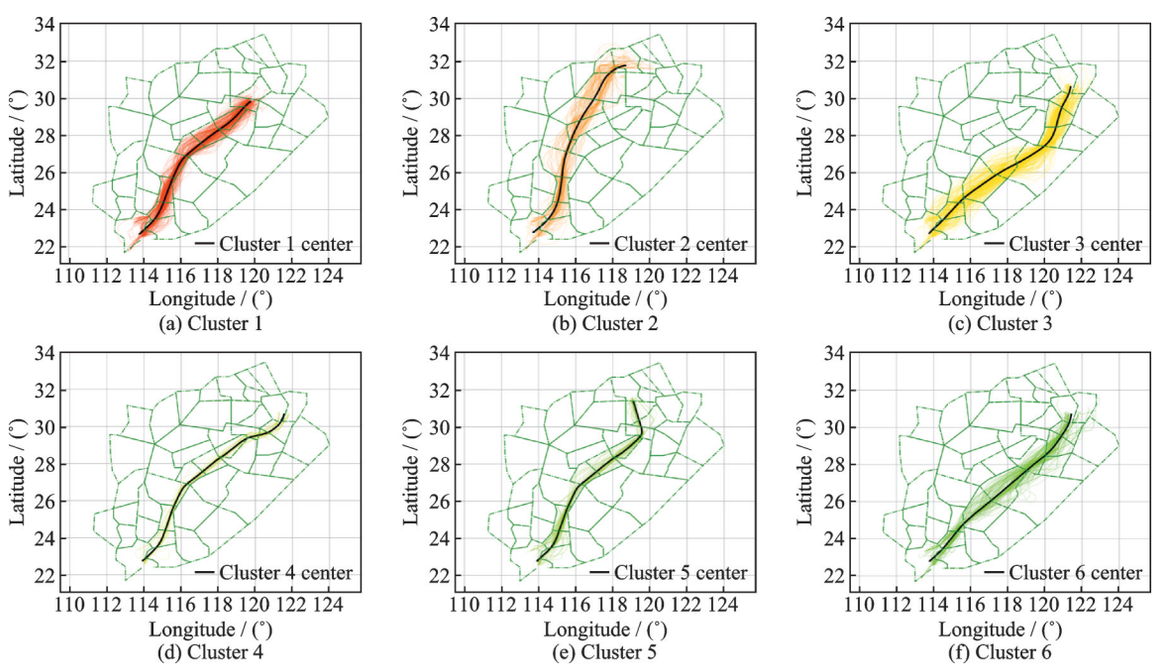


Fig.23 Central trajectories of nine types of non-deviation flights



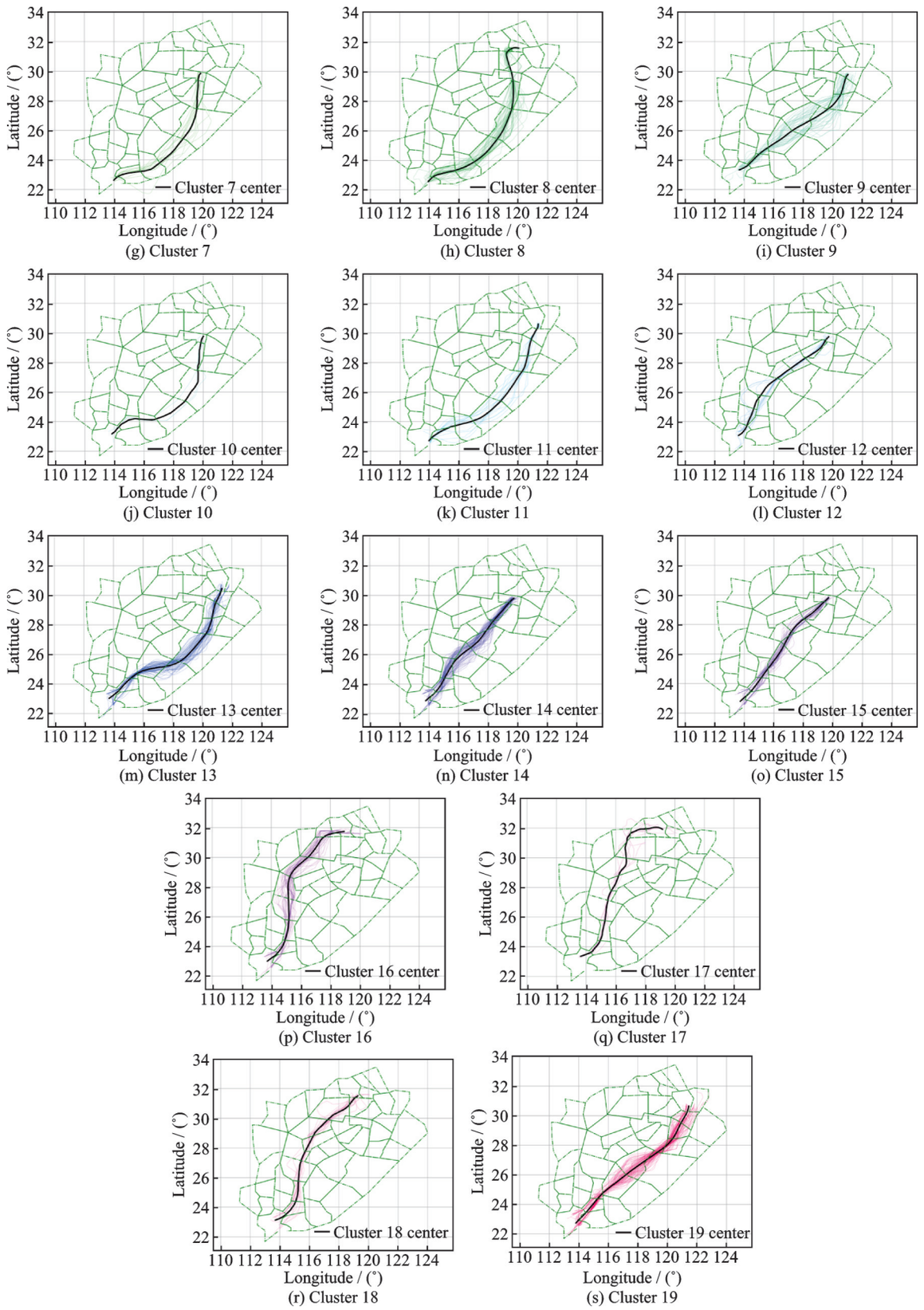


Fig.24 Central trajectories of 19 clusters of deviation flights

planned path. As a result, this study merges the clusters with similar central trajectories into a single category, incorporating the similar patterns of both deviation and non-deviation trajectories. The final trajectory option set from Guangzhou airport cluster to Shanghai airport cluster is thus generated. As shown in Fig.25, a total of 19 trajectory options are generated.

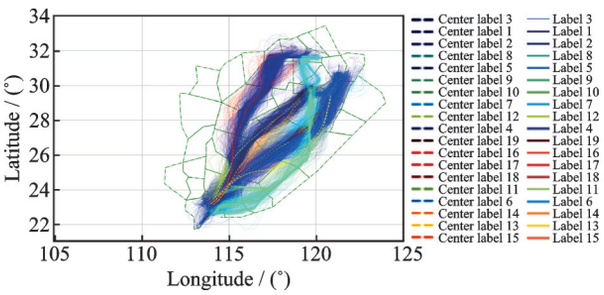


Fig.25 Trajectory option set from Guangzhou airport cluster to Shanghai airport cluster

5 Conclusions

This study develops a trajectory option set generation method based on sector edit distance, using data preprocessing and clustering analysis. The DBSCAN algorithm is employed to identify deviation trajectories and generate a set of trajectory options from the Guangzhou airport cluster to the Shanghai airport cluster. Case studies show that the method effectively extracts preferred paths and aligns with actual flight patterns. The main conclusions are as follows.

(1) Deviation trajectory identification and definition: This study successfully identifies and defines deviation trajectories. By statistically analyzing the deviation of flight trajectories from the planned path in the Central South and East China regions under clear weather conditions, the deviation threshold is established. Using the quantile inflection method based on this threshold, deviation and non-deviation trajectories are distinguished, providing a foundation for subsequent clustering.

(2) Similarity measurement comparison: In the comparison of similarity metrics, sector edit distance combined with the DBSCAN clustering outperforms Euclidean distance and Hausdorff distance. Using silhouette coefficient, CHI, and DBI as evaluation metrics, sector edit distance demonstrates higher intra-cluster compactness and inter-cluster separation when $\varepsilon=0.7$ and MinPts=3, verifying its performance advantages.

(3) Trajectory option generation: Based on the central trajectories of non-deviation and deviation trajectories, this study generates 19 trajectory option sets from the Guangzhou airport cluster to the Shanghai airport cluster. These paths exhibit significant geographical diversity and deviation characteristics, providing practical support for flight path optimization and scheduling.

The findings of this study not only enhance the flexibility of flight planning by providing multiple path options, but also offer data support for collaborative decision-making and dynamic traffic management by air traffic control. In practice, the proposed method has the potential to contribute to the development of more adaptive flight operation strategies in complex airspace environments. However, the method has certain limitations. The sector edit distance has limited clustering separation when there are significant trajectory differences within sectors, and it may be constrained by the granularity of sector division or the complexity of local trajectory patterns. Future research could improve in the following directions: (1) Increasing the volume of historical flight trajectory data to enhance the model's generalization ability; (2) considering the impact of weather forecasts and dynamic airspace factors (such as wind and traffic distribution) on deviation recognition. Additionally, the trajectory option set generated by this study can provide data support for future flight plan path selection, promoting collaborative decision-making between air traffic control and airlines.

References

[1] Civil Aviation Administration of China. 2023 statistical bulletin on the development of China's civil aviation industry[R]. Beijing: Civil Aviation Administration of China, 2023.

[2] Federal Aviation Administration. The Future of the NAS[R]. Washington D C, USA: Department of Transportation, FAA, 2016.

[3] Federal Aviation Administration. Vision for trajectory-based operations (Version 2.0)[R]. Washington D C, USA: Department of Transportation, FAA, 2017.

[4] RODIONOVA O, ARNESON H, SRIDHAR B, et al. Efficient trajectory options allocation for the collaborative trajectory options program[C]//Proceedings of 2017 IEEE/AIAA 36th Digital Avionics Systems Conference (DASC). [S.l.]: IEEE, 2017: 1-10.

[5] ZHANG Baocheng, HU Wei, LIU Wanchun. Optimization method for collaborative route allocation consid-

ering airline preferences[J]. Journal of Transportation Systems Engineering and Information Technology, 2024, 24(4): 243-252. (in Chinese)

[6] ZHU G, WEI P. An interval-based TOS allocation model for collaborative trajectory options program (CTOP)[C]//Proceedings of 2018 Aviation Technology, Integration, and Operations Conference. Atlanta, USA: AIAA, 2018: 3042.

[7] YOO H S, BRASIL C, SMITH N M, et al. Impact of different trajectory option set participation levels within an air traffic management collaborative trajectory option program[C]//Proceedings of 2018 Aviation Technology, Integration, and Operations Conference. Atlanta, USA: AIAA, 2018: 3040.

[8] GALINDO K C G, LEE P U, BRASIL C L. Modeling relative trajectory costs for airborne trajectory reroutes using trajectory option sets[C]//Proceedings of 2021 IEEE/AIAA 40th Digital Avionics Systems Conference (DASC). [S.l.]: IEEE, 2021: 1-10.

[9] PHAM V V. Optimisation algorithms and heuristics for aircraft user-preferred routes and their environmental impact[D]. Sydney, Australia: The University of New South Wales, 2012.

[10] ZHU J, CAI K, LI Y, et al. A novel Transformer-based trajectory options generator for collaborative air traffic flow management[C]//Proceedings of 2023 IEEE/AIAA 42nd Digital Avionics Systems Conference (DASC). Barcelona, Spain: IEEE, 2023: 1-9.

[11] EVANS A, LEE P. Using machine-learning to dynamically generate operationally acceptable strategic reroute options[C]//Proceedings of Air Traffic Management Research and Development (ATM R&D) Seminar. Vienna, Austria: Eurocontrol, 2019.

[12] MATEOS VILLAR M, MARTÍN I, ALCOLEA R, et al. Unveiling airline preferences for pre-tactical route forecast through machine learning. An innovative system for ATFCM pre-tactical planning support[C]//Proceedings of the 11th SESAR Innovation Days. Brussels, Belgium: SESAR Joint Undertaking, 2021.

[13] YE Zijian, LI Nan, LI Jiayi, et al. Aircraft anomaly recognition based on trajectory clustering[J]. Journal of Wuhan University of Technology, 2021,43(7): 42-47.(in Chinese)

[14] WANG Zhisen, ZHANG Zhaoyue, FENG Zhaohui, et al. Terminal area flight trajectory clustering analysis and abnormal trajectory recognition[J]. Science Technology and Engineering, 2022, 22(9): 3807-3814. (in Chinese)

[15] MA Yong, HU Minghua, GU Xin, et al. Terminal area flight trajectory analysis based on spectral clustering[J]. Aeronautical Computing Technique, 2015 (5): 46-50. (in Chinese)

[16] WANG Chao, HAN Bangcun, WANG Fei. Terminal area prevalent traffic flow recognition method based on trajectory spectral clustering[J]. Journal of Southwest Jiaotong University, 2014, 49(3): 546-552. (in Chinese)

[17] LI Shuren, LU Chaoyang, REN Guangjian. Terminal area aircraft flight trajectory analysis based on improved spectral clustering[J]. Journal of Wuhan University of Technology (Transportation Science & Engineering), 2019, 43(6): 1130-1134. (in Chinese)

[18] RAO Dan, SHI Hongwei. Research on air traffic flow recognition and anomaly detection based on deep clustering[J]. Computer Science, 2023, 50(3): 121-128. (in Chinese)

[19] ZENG W, XU Z, CAI Z, et al. Aircraft trajectory clustering in terminal airspace based on deep autoencoder and Gaussian mixture model[J]. Aerospace, 2021, 8(9): 266.

[20] Civil Aviation Administration of China. Civil aviation pre-flight plan management regulations (CCAR-93)[EB/OL]. (2015-11-02)[2025-02-11]. https://www.acaac.gov.cn/XXGK/XXGK/MHGZ/201511/t20151102_8466.html.

Acknowledgements This work was supported in part by Boeing Company and Nanjing University of Aeronautics and Astronautics (NUAA) through the Research on Decision Support Technology of Air Traffic Operation Management in Convective Weather under Project 2022-GT-129, and in part by the Postgraduate Research and Practice Innovation Program of NUAA (No.xcxjh20240709).

Authors

The first author Dr. **WANG Shijin** received her B.S. degree in radio technology from Harbin Engineering University, M.S. degree in signal and information processing from Naval University of Engineering, and Ph.D. degree in transportation planning and management from Nanjing University of Aeronautics and Astronautics (NUAA) in 1996, 2002, and 2011, respectively. From 2002 to present, she has been with College of Civil Aviation, NUAA. Her research interests include airspace planning and management technology, data analytics in aviation, and airspace safety analysis.

The corresponding author Ms. **SUN Min** received the B.S. degree from College of Business Administration, Civil Aviation University of China, in 2024. She is currently pur-

suing the M.S. degree with NUAA. Her research interests include re-planning of flight routes and predicting delays.

Author contributions Dr.**WANG Shijin** designed the study, developed the models, conducted the analysis, interpreted the results, and wrote the manuscript. Ms. **SUN Min** contributed to data and model components for the clustering algorithm. Ms. **LI Yinglin** provided data for the analysis of flight trajectory patterns. Mr. **YANG Baotian** contributed to the discussion and background of the study. All authors commented on the manuscript draft and approved the submission.

Competing interests The authors declare no competing interests.

(Production Editor: WANG Jing)

基于聚类算法的航班轨迹选项集合生成

王世锦，孙 敏，李迎淋，杨宝田

(南京航空航天大学民航学院新一代智能空管实验室，南京 211106，中国)

摘要：针对中国城市对间航班计划通常设置单条航路，缺乏备用路径，难以有效应对突发情况的问题，本文采用“分位数-拐点法”分析特定偏航轨迹，确定偏离阈值，并识别常用偏航路径。通过将欧氏距离、豪斯多夫距离和扇区编辑距离等多种相似性度量与基于密度的噪声应用空间聚类(Density-based spatial clustering of applications with noise, DBSCAN)算法相结合，对偏航轨迹进行聚类，构建面向城市对的轨迹选项集合。以广州机场群与上海机场群之间23 578条航班航迹为案例，验证了所提框架的有效性。实验结果表明，相较于欧氏距离和豪斯多夫距离，基于扇区编辑距离的相似度度量在聚类性能上更优，具有更高的轮廓系数和更低的戴维斯-博尔丁指数，能够实现更好的类内紧凑性和类间分离度。基于聚类结果，共识别出19条代表性轨迹选项，涵盖无偏轨迹与偏航轨迹，显著提升了航路多样性。该研究真实反映了实际飞行运行特征，为飞行路径优化与调度提供了实用依据。

关键词：航班轨迹聚类；轨迹选项集合；扇区编辑距离；基于密度的噪声应用空间聚类算法；偏航轨迹