Central Trajectory Extraction Based on a Simplified Feature Trajectory Model (Postprint)

Authors: Wang Lili, Peng Bo

Date: 2018-05-20T00:00:00+00:00

Abstract

To address the issues in existing center trajectory extraction methods: missing approach turning points, 2. large data volume and time-consuming computation, and some trajectory points lacking representativeness. Based on the concept of "replacing curves with straight lines" to simplify trajectory point data, then performing clustering based on the heading of characteristic trajectories, calculating the average heading to obtain center trajectory positioning points, and finally using B-spline curve fitting to obtain the center trajectory. Simulation results demonstrate that the new method reduces the number of trajectory points by 78.92%, while accurately obtaining the center trajectory, thereby solving the problems of missing turning points and time consumption, and verifying the accuracy and effectiveness of the new method for center trajectory extraction.

Full Text

Research on Central Track Extraction Based on Simplified Track Model

Wang Lili, Peng Bo
(Tianjin Key Laboratory of ATC Operation Planning & Safety Technology, Civil Aviation University of China, Tianjin 300300, China)

Abstract: Existing central track extraction methods suffer from three main problems: (1) missing approach turning points, (2) huge data volume and time-consuming computation, and (3) some track points lack representativeness. This paper proposes a novel method that first simplifies track point data based on the "replace curve with straight line" principle, then clusters feature tracks by their heading to obtain average headings and central track locating points, and finally fits a B-spline curve to obtain the central track. Simulation results demonstrate that the new method reduces the number of track points by 78.92% while accurately extracting the central track, effectively solving the problems of turning point omission and computational inefficiency. The results validate the accuracy and effectiveness of the proposed method for central track extraction.

Key Words: central track extraction; feature track; approach turning point; track clustering; replace curve with straight line

0 Introduction

By analyzing historical radar data of aircraft, airlines can objectively evaluate pilot performance, aircraft capabilities, and flight operations, thereby enabling scientific adjustments to flight plans [3-8]. For air traffic management authorities, trajectory data can be utilized to study flight procedure design, optimization, and noise prediction [9,10]. However, with air traffic volume increasing exponentially, aircraft trajectory data has grown geometrically, making manual processing impractical. Consequently, mining valuable information from massive trajectory datasets has attracted increasing attention from researchers in the civil aviation domain.

Numerous scholars have conducted relevant research on trajectory clustering. Zhao et al. [3] modeled trajectories as discrete point cloud collections in a three-dimensional coordinate system centered on the airport radar, using computational geometry skeleton extraction algorithms to obtain representative points for central track identification. Wang et al. [4] analyzed existing trajectory clustering algorithms and identified their neglect of temporal information, proposing a time-space clustering method that recognizes the close relationship between track point distribution and approach time. Wang et al. [6] studied basic clustering algorithms, noting that trajectory clustering is prone to local optima, and proposed optimizing the fuzzy C-means algorithm using a genetic simulated annealing algorithm. Xu et al. [8] proposed a CURE algorithm for trajectory similarity measurement based on area-weighted inter-trajectory distances, achieving more precise clustering results. Rehm et al. [11] directly utilized existing trajectories to define similarity for clustering approach procedures across different runways at the same airport. Gariel et al. [12] performed radar trajectory clustering for real-time airspace monitoring. Kenneth et al. [13] employed grid-based clustering methods to analyze trajectories from different airports in busy terminal areas, obtaining departure and arrival central tracks.

Currently, extracting and fitting central tracks at approach turning points represents a significant challenge in trajectory clustering research. When clustering approach trajectories to extract central tracks, traditional methods typically succeed in obtaining central tracks for straight segments while failing to capture turning segments, resulting in omissions. Therefore, effective central track extraction and fitting for historical trajectories at approach turning points holds substantial theoretical and practical value. This paper proposes a central track extraction method based on a simplified feature track model. Following the "replace curve with straight line" principle, we simplify track points to obtain feature points, cluster feature tracks by heading, calculate average headings to obtain central track locating points, and finally fit a smooth central track using B-spline curves. Experimental analysis using real historical approach trajectory data for turning segments validates the accuracy and effectiveness of the proposed method.

---

1 Radar Data Processing and Approach Procedure Description

The data source for this study is raw secondary radar data. The processing pipeline involves: (1) decoding raw secondary radar data into plain text format, (2) extracting all useful text entries by keywords and converting required fields into binary integer or floating-point formats while preserving original units, and (3) organizing the data into standardized trajectory format as shown in .

Although heading information can be obtained from radar data, the 4-second sampling interval is insufficient for accurately capturing heading change patterns for Category D aircraft with high turning speeds. Therefore, Section 2 redefines concepts including feature points, feature tracks, and heading variation. Since secondary radar returns latitude/longitude information, conversion to a Cartesian coordinate system is necessary to accurately reflect actual flight trajectories. Using the airport navigation station as the origin, we establish a rectangular coordinate system and convert latitude/longitude coordinates to geodetic coordinates. The converted Cartesian coordinates (in kilometers) are shown in .

Plotting the converted data in MATLAB reveals the actual operational profile of the approach procedure, as illustrated in [Figure 1: see original paper]. We selected three turning segments, designated Segment 1, Segment 2, and Segment 3. The flight procedure proceeds as follows: after exiting the airway, the aircraft flies toward the station, turns over the navigation station (Segment 1), then flies directly to the next navigation station (IAF), turns over the station to the next navigation station, and turns again (Segment 2), before entering the baseline turn procedure until landing (Segment 3). The data shows that an average trajectory contains approximately 20 points per turn, and with approximately 300,000 annual takeoffs and landings at this airport, the accumulated approach turning data reaches nearly four million points annually. Processing such massive datasets is not only time-consuming but also memory-intensive. Therefore, a method is needed that can significantly reduce trajectory data volume while maintaining accuracy.

---

2 Simplified Model Based on Feature Tracks

The "replace curve with straight line" principle from mathematics and physics suggests that a curve can be approximated by a polyline connecting points on the curve. For turning trajectories, this principle can be applied: when several consecutive track segments share similar headings, they can be approximated by a straight line connecting the start and end points. We term these start and end points "feature points," and the connecting line segment a "feature track." To ensure accuracy and maximize fidelity to the original trajectory, feature points should represent locations where the trajectory direction changes significantly.

Consider a turning trajectory composed of points P_1 to P_8 as shown in [Figure 2: see original paper]. Points P_3 and P_6 exhibit distinctly different trajectory directions on either side, making them ideal feature points. The feature track from P_3 to P_6 captures the overall trend of the original trajectory.

Feature point selection follows two principles: the accuracy principle (minimizing difference between feature tracks and original trajectories) and the simplicity principle (minimizing the number of feature points). These principles conflict and must be balanced to achieve optimal feature point selection with adequate computational precision. We introduce a difference metric E, the feature track difference matrix, defined as the difference between the approximate length of the original trajectory and the characteristic length of the feature track. The original trajectory' s approximate length is the sum of straight-line distances between consecutive track points. The feature track' s characteristic length comprises three components: (a) feature track length d, (b) vertical difference d_v (sum of distances from original track points to the feature track), and (c) angular difference d_a as defined in Equation (1):

$$d_a = \sum_{i=1}^{n} l_i \times \sin\theta_i$$

where l_i is the length of each original track segment and θ_i is the angle between each original track segment vector and the feature track vector. When $\theta_i = 90^\circ$, $\sin\theta_i = 1$, making d_a equal to the difference in vertical differences between the segment' s endpoints, as shown in [Figure 3: see original paper]. The original trajectory length is $d' = \sum_{i=1}^{n} l_i$, while the feature track characteristic length is $d + d_v + d_a$, yielding the feature track difference $E = d' - (d + d_v + d_a)$. By controlling E, the selected feature points satisfy both principles. The new method' s advantage in addressing big data computational challenges lies in this simplification model, which streamlines data during initial processing to reduce algorithmic complexity.

3 Heading-Based Clustering

Traditional clustering algorithms perform poorly at extracting turning points for two reasons: (a) they rely on mathematical evaluation metrics (typically distance or density) without considering aircraft operational characteristics, often missing turning portions and only approximating linear trends; (b) some track points fail to represent actual aircraft operational patterns and must be discarded. This paper proposes a heading-based clustering algorithm that captures the turning process by grouping track segments with similar headings as the same turning phase, then reconstructing the complete turning central track from each phase' s central track.

Aircraft turning trajectories are characterized by large turning radii (typically 2-5 km), which can be approximated as several straight segments. Since turning fundamentally involves gradual heading change, we cluster feature tracks by heading, grouping segments with small heading differences as the same turning phase. The clustering process converts all feature track segments to vector form, randomly selects one segment, calculates its angle with other vectors, and uses an angle threshold θ_{sita} to group directionally similar vectors into the same cluster.

Traditional clustering algorithms that process all trajectory data simultaneously without classification often produce inaccurate or missing central locating points. To address this, after clustering, each cluster represents a turning phase. We calculate the average of trajectories within each cluster and combine all clusters to reflect the historical trajectory distribution trend—the central track. The average trajectory can be obtained using a perpendicular line (called an intercept line) to cut through trajectory segments and calculate the average of intersection points. However, the intercept line cannot be arbitrarily selected; if its angle with cluster trajectories is too large, portions on either side cannot be accurately averaged. Therefore, the intercept line should be perpendicular to most trajectories' direction, which we represent using an average vector.

The average vector is defined in Equation (2):

$$\vec{v}_{avg} = \frac{1}{n}\sum_{i=1}^{n}\vec{v}_i$$

After obtaining the average vector, central track extraction involves sweeping a line perpendicular to the average vector across each cluster' s segments and averaging the intersection points. Since trajectories consist of line segments between track points, we consider all track points in each cluster, using lines through each point to intersect with trajectories and averaging these intersection points to obtain central track locating points. However, not all such averages qualify as valid locating points.

As shown in [Figure 4: see original paper], averaging every point can introduce significant deviations. When the number of intersections between the intercept line and trajectories falls below a threshold (e.g., points and), the resulting central track locating points are unreliable and should be discarded. Therefore, an intersection point threshold must be set—only when the intersection count exceeds this value is the locating point considered valid. Additionally, to ensure smoothness, the spacing between central track locating points must be controlled. As shown in [Figure 4: see original paper], points and are too close, so one should be skipped. The final central track should be formed by locating points , , and .

5 Spline Curve Fitting

The previous step yields central track locating points that can be connected to form the central track. To produce a smooth central track conforming to aircraft dynamics constraints, this paper employs cubic B-spline curves. Spline curves are widely used in engineering design as smooth curves passing through given points, with continuous first and second derivatives at each point and uniform curvature variation—characteristics that match flight performance requirements.

The cubic B-spline formulation is as follows: Given a partition on interval (a, b):

$$a = x_0 < x_1 < \cdots < x_n = b$$

A function $S(x)$ on (a, b) is called an interpolating cubic spline function if it satisfies:

1. On each interval $[x_i, x_{i+1}]$, $S(x)$ is a cubic polynomial $S_i(x)$ for $i = 1, 2, \dots, n$;
2. At nodes x_i $(i = 1, 2, \dots, n-1)$, $S(x)$ has continuous second derivatives: $S^{(k)}(x_i^-) = S^{(k)}(x_i^+)$ for $k = 0, 1, 2$;
3. Interpolation conditions: $S(x_i) = y_i$ for $i = 0, 1, \dots, n$.

When $n = 3$, the cubic B-spline basis functions are:

$$\begin{cases} G_0(t) = \frac{1}{6}(1-t)^3 \\ G_1(t) = \frac{1}{6}(3t^3 - 6t^2 + 4) \\ G_2(t) = \frac{1}{6}(-3t^3 + 3t^2 + 3t + 1) \\ G_3(t) = \frac{1}{6}t^3 \end{cases}$$

for $t \in [0, 1]$. The cubic B-spline curve segment is:

$$P(t) = \begin{bmatrix} G_0(t) & G_1(t) & G_2(t) & G_3(t) \end{bmatrix} \begin{bmatrix} p_0 \\ p_1 \\ p_2 \\ p_3 \end{bmatrix}$$

6 Simulation Analysis

Simulations were conducted in MATLAB 8.3 using one week of trajectory data for Segment 3 (the turning portion) of an approach procedure at a certain airport. The dataset contains 15,341 track points. The algorithm requires three parameters: (a) k, which determines the number of feature points and approximation accuracy; (b) θ_{sita}, which determines the number of clusters and clustering effectiveness; and (c) $dist$, which controls the distance between central track locating points and the number of intersection points, affecting the smoothness of the fitted trajectory.

First, different k values were tested for point reduction to satisfy both accuracy and simplicity principles. The results for various k values, remaining points, and reduction rates are shown in . To validate the reduction effect, a single trajectory from Segment 3 was selected. The reduction results for $k = 0.20, 0.21, 0.22$ are shown in [Figure 5: see original paper] through [Figure 7: see original paper], where blue hollow circles represent original track points and red solid points represent retained feature points. The results show that $k = 0.20$ and $k = 0.22$ produce overly concentrated points with adjacent duplicates, while $k = 0.21$ yields well-distributed points that satisfy both principles. Therefore, $k = 0.21$ was selected for feature point reduction.

Clustering analysis was performed on the reduced points using different angle thresholds θ_{sita}. As shown in [Figure 8: see original paper], $\theta_{sita} = 20^{\circ}$ successfully clusters the segment into eight classes with complete coverage and no missing segments. Different distance thresholds $dist = 0.5, 1.0, 1.5$ were then tested, as shown in [Figure 9: see original paper] through [Figure 11: see original paper]. The results indicate that $dist = 1.0$ produces appropriately spaced points with good distribution.

With other parameters fixed, $k = 0.20, 0.21, 0.22$ all yield computation times around 5 seconds. To validate the method' s advantages, we compared it with the K-means algorithm. As shown in [Figure 12: see original paper], K-means required 44 seconds to process the same data, extracting 17 central track locating points (marked as black crosses). While K-means performs well on straight segments, it exhibits missing turning points, particularly points 10 and 11, which deviate significantly from the central trajectory. This demonstrates that traditional clustering algorithms are inadequate for extracting approach turning points, solving computational inefficiency, and obtaining accurate central tracks.

The 17 central track points obtained by our method were then fitted using B-splines with an interpolation interval of 0.5, yielding 33 control points. Partial control point coordinates are listed in . Connecting these control points produces the final central track shown in [Figure 13: see original paper], where blue hollow circles represent the 17 central track points and red crosses represent the 33 B-spline control points.

---

7 Conclusion

This paper proposes a central track extraction method based on a simplified feature track model following the "replace curve with straight line" principle. The method involves: (1) decoding raw secondary radar data and converting coordinates, (2) eliminating non-feature track points to obtain simplified feature points, (3) clustering feature tracks by heading to extract central track locating points, and (4) fitting a smooth, flight-performance-compliant central track using cubic B-spline curves. Simulation using real secondary radar approach data and comparison with other algorithms demonstrates that the proposed method eliminates missing turning points, reduces processing time for equivalent datasets, and solves existing method limitations. The results validate the method' s accuracy and effectiveness, offering significant value for trajectory clustering analysis, particularly for central track extraction in turning segments. Future research will focus on central track extraction for multiple arrival flights in terminal areas.

References

[1] Gudmundsson J, Thom A, Vahrenhold J. Of motifs and goals: mining trajectory data [C]// Proc of the 20th International Conference on Advances in Geographic Information Systems. New York: ACM Press, 2012: 129-138.

[2] 吴笛, 杜云艳, 易嘉伟, 等. 基于密度的轨迹时空聚类分析 [J]. 地球信息科学学报, 2015, 17 (10): 1162-1171.

[3] 赵元棣, 孙禾, 王洁宁, 等. 终端区航迹簇的中心航迹提取方法研究 [J]. 图学学报, 2014, 35 (3): 379-386.

[4] 王洁宁, 孙禾, 赵元棣. 基于时间-空间的进场航迹聚类分析 [J]. 科学技术与工程, 2013, 13 (33): 10078-10081, 10088.

[5] 王超, 徐肖豪, 王飞. 基于航迹聚类的终端区进场程序管制适用性分析 [J]. 南京航空航天大学学报, 2013, 45 (1): 130-139.

[6] 王涛波, 黄宝军. 改进模糊聚类算法在航迹分析中的应用 [J]. 中国安全科学学报, 2014, 24 (2): 23-27.

[7] 王超, 王明明, 王飞. 基于改进的模糊 C-means 航迹聚类方法研究 [J]. 中国民航大学学报, 2013, 31 (3): 14-18.

[8] 徐涛, 李永祥, 吕宗平. 基于航迹点法向距离的航迹聚类研究 [J]. 系统工程与电子技术, 2015, 39 (9): 2198-2204.

[9] 李永祥. 面向机场噪声预测的航迹聚类研究及应用 [D]. 天津: 中国民航大学, 2015.

[10] 徐涛, 陈雪蕊, 等. 基于航迹聚类的终端区飞行程序轨迹表示 [J]. 四川大学学报: 工程科学版, 2016, (06): 188-196.

[11] Rehm F. Clustering of flight tracks [C]// AIAA infotech @ Aerospace, Atlanta: AIAA 2010-3412.

[12] Gariel M, Srivastava A N, Feron E. Trajectory clustering and an application to airspace monitoring [J]. IEEE. Trans on Intelligent Transportation Syatems, 2011, 12 (4): 1 511-1 524.

[13] Kenneth Leiden, Stephen Atkins. Trajectory clustering for metroplex operations [C]// Proc of the 11th AIAA Aviation Technology, Integration, and Operation Conference. 2011.

[14] Lee J G, Han J. , et al. Trajectory clustering: a partition-and-group framework [C]// Proc of ACM SIGMOD International Conference on Management of Data. 2007: 593-604

Note: Figure translations are in progress. See original paper for figures.