AIAA Aviation Forum and ASCEND co-located Conference Proceedings
29 July - 2 August 2024, Las Vegas, Nevada
AIAA AVIATION FORUM AND ASCEND 2024
10.2514/6.2024-4552

A Machine Learning-Based Extraction of Cruise Phase from Trajectory Data

Seokhwan Lee*, Jaeyoung Ryu†, Bae-Seon Park‡, and Hak-Tae Lee§
Inha University, Incheon, Republic of Korea, 21999

When analyzing historical trajectory data, identifying the cruise phase is crucial. However, due to various vertical maneuvers along the course of the flight, it is not straightforward to extract the cruise phase, and rule-based algorithms tend to be inaccurate when the vertical trajectory becomes complicated. This study presents a machine learning-based technique to extract the cruise phase of a flight from recorded trajectory data. The trajectory data are normalized by maximum time and altitude, and then grouped into clusters using an agglomerative hierarchical clustering technique and a Gaussian Mixture Model algorithm. Finally, a rule-based selection criterion is applied to each cluster centroid to identify search regions for top-of-climb (TOC) and top-of-descent (TOD). The TOC and TOD are extracted for the individual trajectory that belongs to the cluster. The study classified a total of 38,051 trajectories into 41 clusters. The cruise phase was extracted for 36,712 flights, accounting for 96.5% of the total trajectories. The proposed method is particularly useful when the cruise phase needs to be extracted for a large data only with the trajectory data without the flight management system information.

I. Introduction

In addition to the constant growth of the air traffic volume, awareness of global warming issues has increased. The impacts of increasing traffic on the environment have also become a major concern for aviation stakeholders. Recently, research aimed at reducing carbon dioxide (CO_2) emissions from burning aviation fuel has been actively conducted. According to [1], a project has been carried out to reduce environmental impact by 12% through efficient route planning, which has saved flight time and fuel, and reduced CO_2 emissions by approximately $1,300\ kg$. Additionally, International Civil Aviation Organization aims to achieve net-zero CO_2 emissions by 2050 [2].

The increase in air traffic and its associated environmental impact necessitate new approaches to enhance the efficiency of Air Traffic Management (ATM) and minimize environmental effects. With the availability of flight trajectory data facilitated by the Automatic Dependent Surveillance-Broadcast (ADS-B) system, historical data are actively being analyzed [3].

For the trajectory analysis, Top of Climb (TOC) and Top of Descent (TOD), where the aircraft begins its cruise and begins its descent, respectively, are important for evaluating an aircraft's climb and descent profiles. For example, TOC marks the end of continuous ascent, while TOD marks the beginning of continuous descent for landing. These points can be benchmarks to compare the performance of continuous climb or descent operations and conventional step operations using real-world data.

Furthermore, the Estimated Time of Arrival (ETA) of an aircraft is a crucial factor in ATM and passenger service. Accurate prediction of an aircraft's arrival time allows for efficient management of airport take-off and landing schedules and provides passengers with precise arrival information, thereby enhancing service quality. Identifying the exact TOD point enables more accurate prediction of the entire flight path conformance and arrival time.

Related studies include [4], which proposes a method using a machine learning approach to predict TOD, aiding in the design of Continuous Descent Approach and Standard Terminal Arrival Route procedures. Another study proposes a method to reduce environmental impact by optimizing the descent path to control the ETA [5]. Additionally, a method to optimize aircraft paths by optimizing vertical navigation during climb, cruise, and descent phases and horizontal navigation considering wind effects to reduce fuel consumption has been proposed [6].

To analyze the various indicators related to the cruise phase, accurately identifying the TOC and TOD is essential. As the flight moves up or down by several thousand feet during the cruise phase, finding the TOC and TOD using only

*M.S. Student, Department of Aerospace Engineering, 36 Gaetbeol-Ro, Yeonsu-Gu, Incheon.
†Ph.D. Student, Department of Aerospace Engineering, 36 Gaetbeol-Ro, Yeonsu-Gu, Incheon.
‡Post-Doctoral Researcher, Industrial Science Technology Institute, 36 Gaetbeol-Ro, Yeonsu-Gu, Incheon. Currently at Hanwha Aerospace.
§Professor, Department of Aerospace Engineering, 36 Gaetbeol-Ro, Yeonsu-Gu, Incheon, AIAA Senior Member.

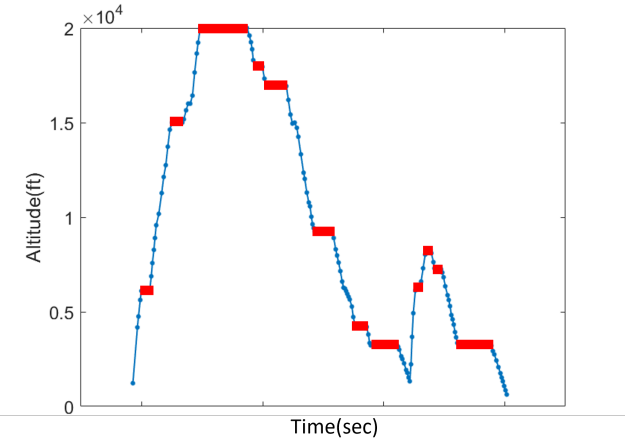

Fig. 1 Extracting constant altitude segments.

the recorded the trajectory data is not straightforward. Consequently, an approach based only on a set of rules is likely to fail to capture many unusual cases.

This paper combines a machine learning approach and a rule-based approach to increase the accuracy of the TOC and TOD extraction from the recorded trajectory data, which is especially useful when no other data such as airline flight data are available.

The trajectory data are normalized by each trajectory's maximum time and altitude, and then grouped into clusters. The number of clusters is determined using an agglomerative hierarchical clustering [7]. Using this information, a Gaussian Mixture Model (GMM) algorithms [8] is applied to group the trajectories. The TOC and TOD of the centroid of each cluster are identified using a rule-based algorithm as the reference points for searching the TOC and TOD of all trajectories that belong to the cluster. For each trajectory within a cluster, points in the vicinity of the TOC and TOD of the cluster centroid are searched to find its own TOC and TOD. This clustering-based approach is more accurate in the sense that unusual patterns are identified during the clustering that cannot be captured by rule-based algorithms.

Following this introduction, Section II describes an overview of the trajectory data. Section III explains the data pre-processing, trajectory clustering, and TOC and TOD extraction methods. Section IV presents the extraction results, and finally, Section V concludes the paper.

II. Trajectory Data

ADS-B data is a common type of trajectory data used in research on air traffic and safety analysis. ADS-B data provides real-time information about an aircraft's position, altitude, ground speed, heading, and others. This study uses the ADS-B data of 38,051 flights between the Gimpo International Airport and Jeju International Airport within the Incheon flight information region in 2019. As of April 2023, this route is the busiest air route in the world [9, 10].

III. Approaches

Figure 1 shows an example of extracting segments that maintain a constant altitude. A subject matter expert is likely to identify that the flat segment at 20,000 ft is the only cruise segment. It seems that this flight had five level outs during the descent and a go-around before landing. As can be seen in this example, a rule-based algorithm that relies only on the constant altitude segments has limitations.

On the other hand, constructing a model using supervised learning is challenging due to the scarcity of accurately identified cruising phases in the flight trajectories unless the airline flight data that contains the flight management system inputs are available. Moreover, it is not practical for subject matter experts to manually evaluate the large number of trajectories.

A. Data Pre-processing

Efficient data pre-processing is essential for implementing effective clustering. The original flight data shows significant variability in flight time and maximum altitude for each flight, and the number of collected data points is inconsistent. For configuring training data for a machine learning model, it is crucial that all data points have the same number of features. In this study, the trajectories are resampled to 500 data points using linear interpolation. Both the time and the altitude values are uniformly adjusted to 500 features. The resampled data is then normalized, transforming both the time and the altitude to a range between zero and one. This data pre-processing allows for the construction of a more consistent dataset and enhances the performance of the clustering algorithm.

B. Clustering

1. Agglomerative Hierarchical Clustering Algorithm

This study clusters trajectory data using the GMM algorithm. However, a significant challenge is determining the number of clusters, K, in advance. To solve this issue, the agglomerative hierarchical clustering is first used to analyze the cluster structure of the data, which helps to determine K to be utilized in the GMM algorithm.

The agglomerative hierarchical clustering treats each data point as a separate cluster and iteratively identifies and merges the most similar clusters according to the given distance metric. In this study, 38,051 trajectories with 500 features are clustered. Each individual trajectory i is shown in Eq. (1).

$$\vec{h}^i = \left(\bar{h}^i_1, \bar{h}^i_2, \cdots, \bar{h}^i_{500}\right), \quad i = 1, 2, \cdots, 38051 \tag{1}$$

where

$$\bar{h}^i_j = \bar{h}^i\left(\bar{t}_j\right) \tag{2}$$

The Ward linkage method is used to measure the similarity between clusters and is calculated using Eq. (3). In particular, this method computes the Euclidean distance between clusters A and B weighted by the number of trajectories in the clusters denoted by $n(A)$ and $n(B)$, respectively, This makes the method less sensitive to outliers as larger clusters reflect less variance increase upon merging. The centroids of clusters A and B are denoted as $\bar{h}^A$ and $\bar{h}^B$, which can be obtained by Eq. (4) [11].

$$d(A,B) = \frac{\|\vec{h}^A - \vec{h}^B\|^2}{\frac{1}{n(A)} + \frac{1}{n(B)}} \tag{3}$$

$$\vec{h}^A = \frac{1}{n(A)} \sum_{i \in A} \vec{h}^i \ , \quad \vec{h}^B = \frac{1}{n(B)} \sum_{i \in B} \vec{h}^i \tag{4}$$

The distances between all the clusters are calculated, and the two clusters with the smallest distance are merged into a single cluster. The centroid of the new cluster is a weighted average of the two centroids as shown in Eq. (5). Once a new cluster is formed, distances to other clusters from this new cluster are calculated. This process is repeated until only one cluster remains.

$$\vec{h}^{A+B} = \frac{n(A)\vec{h}^A + n(B)\vec{h}^B}{n(A) + n(B)} \tag{5}$$

A dendrogram is generated, which visually represents the hierarchical structure and can be analyzed to determine the optimal number of clusters. The dendrogram result of using Eq. (3) is shown in Fig. 2. As can be seen, the number of clusters becomes a function of the distance threshold. In this study, the distance threshold is set to 25, which leads to 41 clusters. the number of clusters is a trade-off between the distinctiveness of each cluster and the computational efficiency. In general, it should be adjusted according to the problem characteristics.

2. Gaussian Mixture Model Algorithm

The GMM algorithm is used to cluster the normalized trajectories. The GMM algorithm is a suitable tool for modeling the latent cluster structure of the data, assuming that the data is generated from multiple Gaussian distributions.

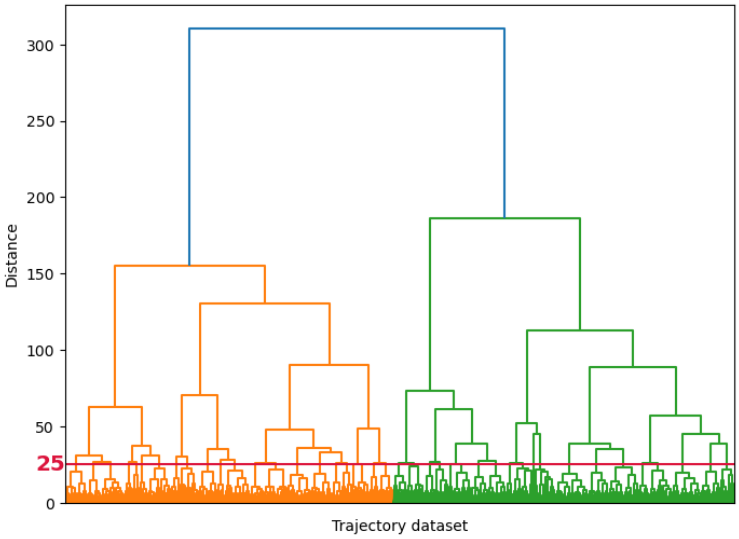

Fig. 2 Dendrogram result.

Each Gaussian distribution is constructed by a mean value, $\vec{\mu}_k$, a covariance matrix, $\mathbf{\Sigma}_k$, and a mixing coefficient, π_k. The mixing coefficient represents the influence of each Gaussian distribution within the overall dataset. Each Gaussian distribution represents a cluster, and the GMM explains the overall distribution of the data through a mixture of these distributions.

The entire model is expressed as a weighted sum of each Gaussian distribution, with the Gaussian probability density function given in Eq. (6), where the input is the trajectory vector, $\vec{h}$.

$$p(\vec{h}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\vec{h} \mid \vec{\mu}_k, \mathbf{\Sigma}_k) \tag{6}$$

The parameters of the GMM are estimated using the Expectation-Maximization (EM) algorithm. The expectation step of the EM algorithm calculates the probability that each data point is generated by the k-th Gaussian distribution. This computes responsibility, as shown in Eq. (7). The function $\gamma_k(\vec{h})$ denotes the probability that the trajectory $\vec{h}$ belongs to the cluster k.

$$\gamma_k\left(\vec{h}\right) = \frac{\pi_k \mathcal{N}(\vec{h} \mid \vec{\mu}_k, \mathbf{\Sigma}_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(\vec{h} \mid \vec{\mu}_j, \mathbf{\Sigma}_j)} \tag{7}$$

In the maximization step, the parameters are updated based on the previously calculated γ_k, maximizing the log-likelihood function. This process is iterated until the parameters π, μ, and $\mathbf{\Sigma}$ parameters converge. The log-likelihood function is provided in Eq. (8).

$$\log p(H \mid \pi, \vec{\mu}, \mathbf{\Sigma}) = \sum_{n=1}^{N} \log \left\{ \sum_{k=1}^{K} \pi_k \mathcal{N}(\vec{h}^n \mid \vec{\mu}_k, \mathbf{\Sigma}_k) \right\} \qquad H = \left\{ \vec{h}^1, \vec{h}^2, \cdots, \vec{h}^N \right\}, N = 30,851 \tag{8}$$

In this study, full matrices are used for the variance, $\mathbf{\Sigma}_k$. Figure 3 shows the number of data points classified into each cluster. Among these, cluster 39 is identified as noise and is distinguished from other clusters. The remaining 40 clusters are identified as cluster groups, each containing at least 100 trajectories. The classified trajectory cluster and their centroids, $\vec{\mu}_k$, are visualized in Fig. 4.

C. Centroid TOD & TOD Extraction

Because the centroids of the identified clusters show simpler and smoother trajectories, the Ramer-Douglas-Peuker (RDP) algorithm [12] is applied to further simplify the 500 points and extract the TOC and TOD. During the cruise

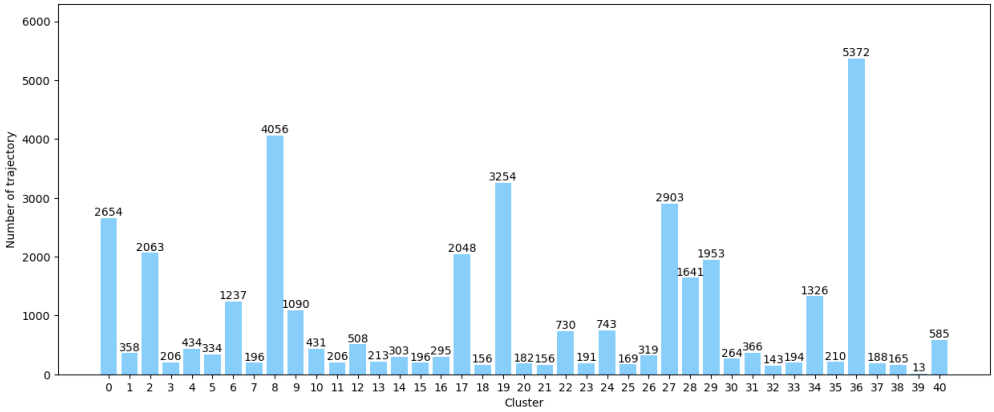

Fig. 3 Distribution of trajectory data by clustering.

phase, an aircraft slightly moves up or down, which makes it challenging to identify major segments. The RDP algorithm generates a simplified version of the altitude data, while maintaining the principal features and removing unnecessary points.

In this process, it is necessary to set a threshold, ϵ, which has a significant impact on the number of points. The larger the epsilon value, the more points are removed, and conversely, the smaller the epsilon value, the fewer points are removed. Figure 5 shows the number of points with respect to ϵ. It can be observed that once the ϵ becomes larger than 0.002 the number of points decreases very slowly. In this study, ϵ was set to 0.01 in the normalized space. Figrue 6 depicts the results of applying the RDP algorithm to the cluster centroids.

The TOC and TOD of the simplified centroid are extracted for each cluster by considering both the maximum altitude and the slope of the vertical trajectory in the normalized space. In Eqs. (9)-(11), $\bar{t}_l$ represents the (l)-th normalized time from the RDP reduction and $\bar{h}(\bar{t}_l)$ means the normalized altitude at time $\bar{t}_l$. The conditions for finding the TOD and TOD are as listed in the following conditions. Points that satisfy all three conditions are considered to be in cruise phase. The minimum and maximum times among this cruise phase are identified as the TOC and TOD of the cluster.

1) Normalized altitude must be greater than or equal to 0.9 as shown in Eq. (9).
2) The slope between the (l)-th point and the $(l+1)$-th point must be less than or equal to 0.5 as indicated by Eq. (10).
3) The slope between $(l+1)$-th point and (l)-th point must be smaller than the slope between (l)-th point and $(l-1)$-th point as indicated by Eq. (11). This condition is added to exclude step climb or descent cases.

$$\bar{h}(\bar{t}_l) \geq 0.9 \tag{9}$$

$$\frac{\bar{h}(\bar{t}_{l+1}) - \bar{h}(\bar{t}_l)}{\bar{t}_{l+1} - \bar{t}_l} \leq 0.5 \tag{10}$$

$$\frac{\bar{h}(\bar{t}_{l+1}) - \bar{h}(\bar{t}_l)}{\bar{t}_{l+1} - \bar{t}_l} - \frac{\bar{h}(\bar{t}_l) - \bar{h}(\bar{t}_{l-1})}{\bar{t}_l - \bar{t}_{l-1}} \leq 0 \tag{11}$$

The results of finding the TOC and TOD of each cluster are presented in Fig. 7. In Fig. 7, cluster 34's centroid shows the flight immediately descending following its maximum altitude. Thus, it can't be considered a cruise phase, and the TOD and TOD could not be extracted.

D. Indivisual TOD & TOD extraction

Individual trajectories may also be affected by data reception problem or noise. Therefore, the RDP algorithm is applied in order to simplify the normalized data. The threshold is set to 0.01, the same as in the analysis of the cluster centroid's TOC and TOD. Figure 8 shows examples of the application of the normalization and the RDP algorithm to individual trajectories.

Using the TOC and TOD of the centroid, two segments can be determined that represent the range where the TOC and TOD exist for all trajectories within the cluster. First, the possible candidate segments are filtered using Eq. (10),

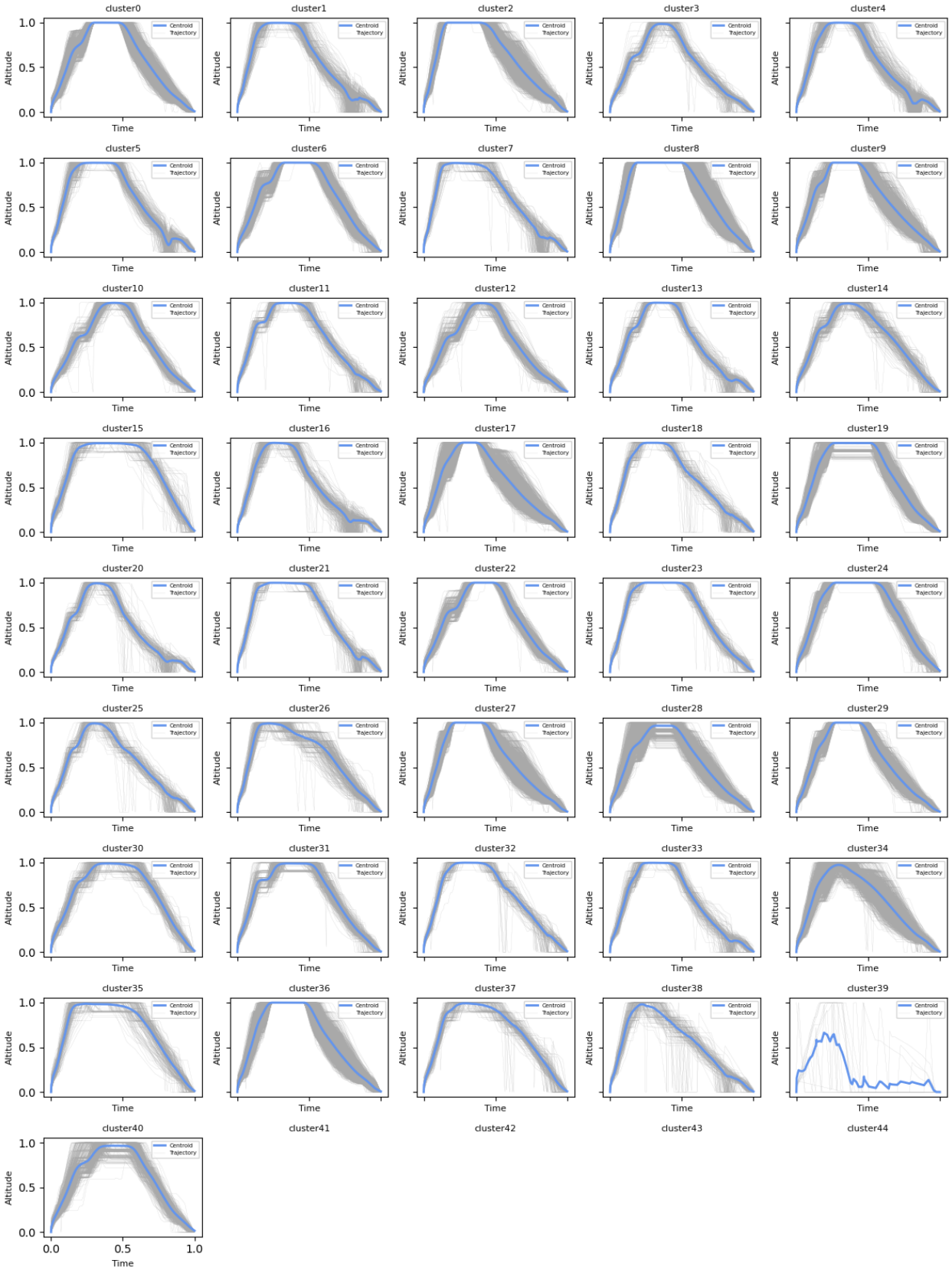

Fig. 4 Clustering results.

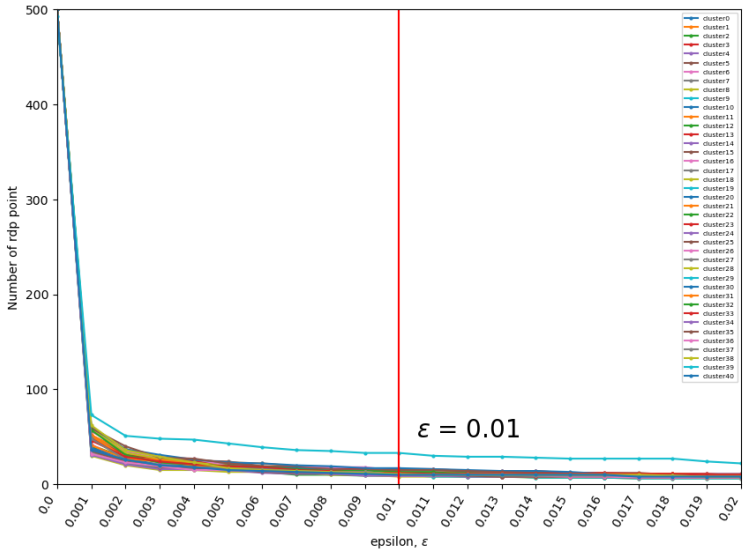

Fig. 5 Number of points with respect to ϵ.

and then two segments are selected from these candidates. The first segment, $Segment1$, which includes the centroid's TOC, represents the TOC range of individual trajectory, while the second segment, $Segment2$, which includes the centroid's TOD, is the TOD range for individual trajectory. The method of defining each segment is visually presented in Fig. 9. Also, Fig. 10 shows examples of individual trajectories with this method applied.

If the cluster centroid's TOC and TOD are not included within the aforementioned segments, the first segment following the cluster centroid's TOC is $Segment1$, and the last segment preceding the cluster centroid's TOD is $Segment2$. This rule-based approach ensures that even if the cluster centroid's TOC and TOD are not within individual trajectory's TOC and TOD range, the cruise phase can still be identified through similar segments.

Finally, once $Segment1$ and $Segment2$ are determined, the earliest altitude point within $Segment1$ is identified as the individual's TOC, and the latest altitude point within $Segment2$ is identified as the individual's TOD. The outcomes are shown in Fig. 11. In particular, the five trajectories in column 3 show that the TOC and TOD are well extracted by applying proposed methods in this study to complex trajectory data.

IV. Results and Discussions

Applying the proposed method in this study resulted in the clustering of 38,051 trajectories into 41 clusters. Except for one noise cluster, the remaining 40 clusters contain more than 100 trajectories. As a result, the cruise phases were extracted from 36,712 individual trajectories. The number of extracted trajectories for each cluster is presented in Fig. 12.

Even though the proposed methods displayed the robustness and some of the extraction results could be visually confirmed, it was not easy to check the accuracy of the results without the flight management system information. During this research, an attempt was made to visually confirm the results using crowd sourcing techniques. 99 undergraduate students majoring in Aerospace Engineering were recruited to confirm the accuracy of the partial dataset with 2665 trajectories. First, the students were briefed about the general climb, cruise, descent phase of flights as well as the typical altitude change scenarios. Each student was then randomly assigned with around 75 trajectories and asked to select True, False, or Unclear for a total of 2665 trajectories. The experiment was designed such that for each trajectory would have inputs from three students. Due to some missing responses, approximately 65% of the trajectories were cross-validated by three or more individuals. Figure 13 visualizes the results of the students' cross-validation.

As shown in Fig. 14, the preliminary validation results with the subset of the data indicate that 82% of the TOC and TOD points are marked as True or Unclear, with 18% marked as False.

Future research will focus on further developing this method and formally validating the extracted data through airline flight data as well as expanding the crowd sourcing approach to more experienced subject matter experts. This approach will enhance the reliability of data validation and improve the accuracy of the extracted cruise phases.

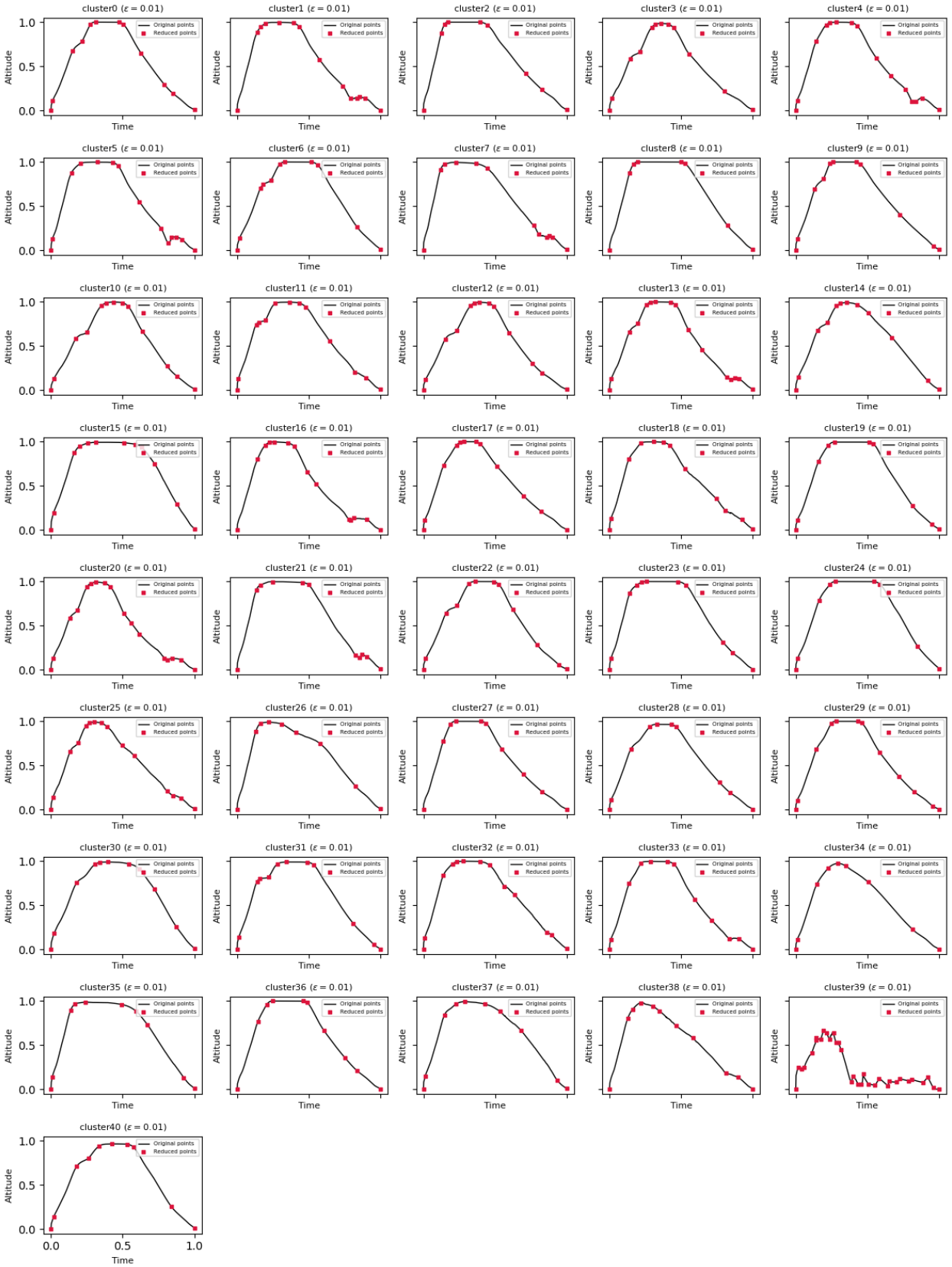

Fig. 6 Centroid's RDP points.

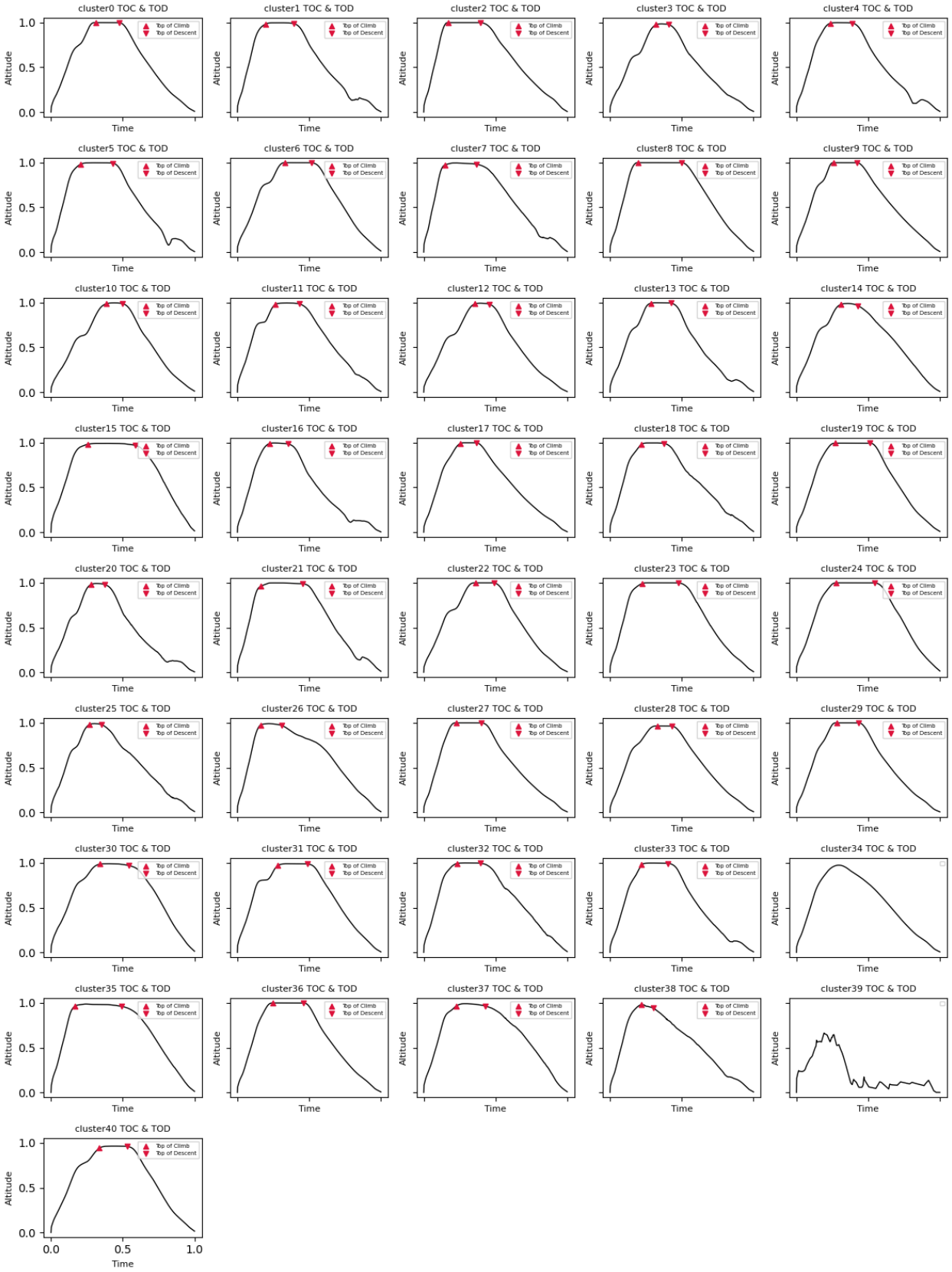

Fig. 7 Centroid's TOC and TOD.

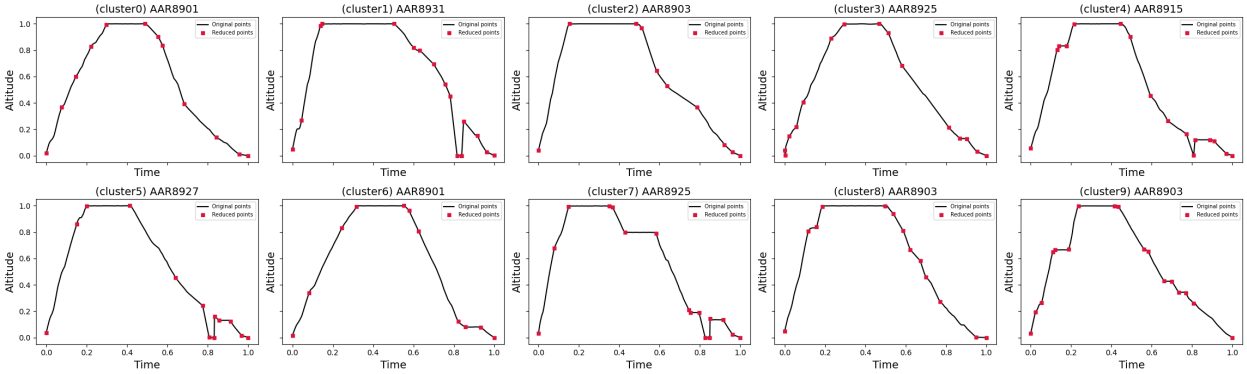

Fig. 8 Individual RDP results.

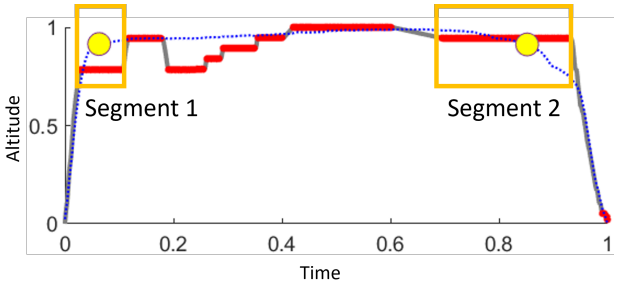

Fig. 9 Finding TOC and TOD in a single trajectory.

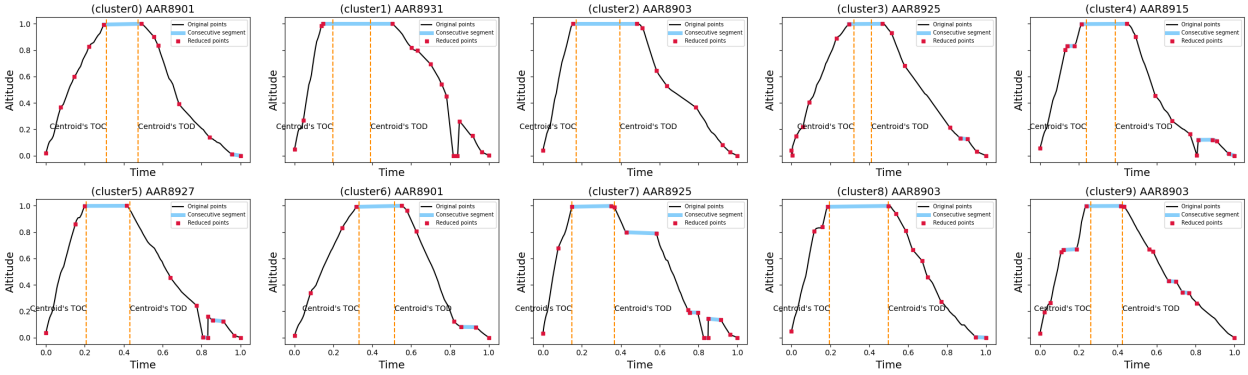

Fig. 10 Individual consecutive segments.

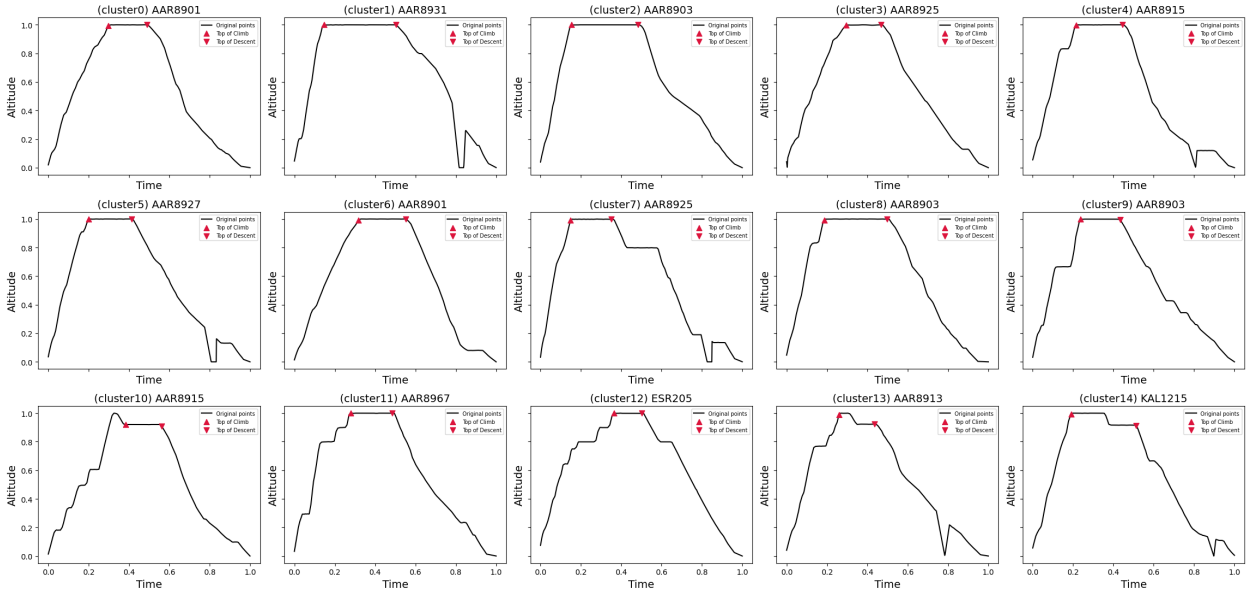

Fig. 11 Individual's TOC and TOD results.

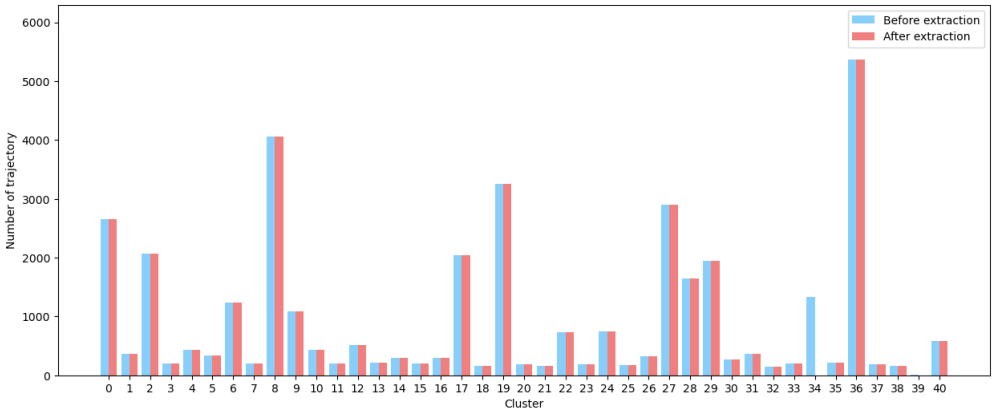

Fig. 12 Distribution of extraction results.

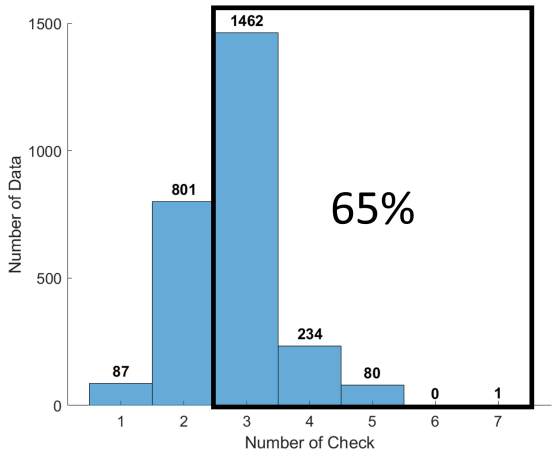

Fig. 13 Distribution of number of validation counts.

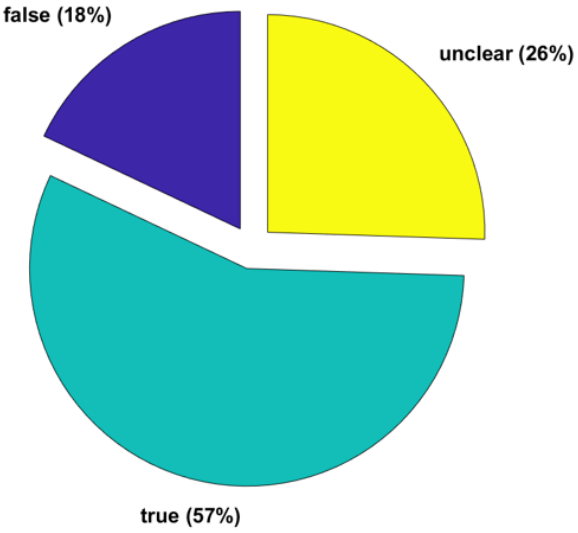

Fig. 14 Validation results.

V. Conclusions

In this study, cruise phase extraction methods using machine learning-based techniques, agglomerative hierarchical clustering and the GMM algorithms, combined with rule-based selection criteria are presented. Vertical trajectory data are clustered using GMM, and the TOC and TOD of the cluster centroids are extracted using a rule-based approach. Additionally, the TOC and TOD of individual trajectories are extracted based on the TOC and TOD of the cluster centroids.

With 40 well populated clusters with one noise cluster, proposed cruise phase extraction methods showed a robust performance processing of 96.5% of the 38,051 trajectories. Without the available flight data, validating the results remains to be a challenge. A crowd sourcing-based validation was attempted, which displayed promising outcomes.

The extraction of cruise phases from this study will be useful for a large scale data-driven studies including the investigation of environmental impacts and estimating the time of arrival among many others.

Acknowledgments

This work is supported by Korea Agency for Infrastructure Technology Advancement (KAIA) grant funded by the Ministry of Land, Infrastructure and Transport (Grant RS-2020-KA158275).

References

[1] ATAG, "Aviation: Benefits Beyond Borders (ABBB)," Tech. rep., ATAG, 2019.

[2] Mithal, S., and Rutherford, D., "ICAO's 2050 net-zero CO2 goal for international aviation," Tech. rep., ICCT, 2023.

[3] Lee, H.-T., Park, B.-S., Ryu, J., Nam, H.-S., Han, S.-M., Hwang, H.-S., Lee, S.-H., Kang, J., and Lee, H., "Data-Driven Aviation Safety Research," *The Korean Society for Aeronautical and Space Sciences (KSAS) 2023 Fall Conference*, 2023, pp. 380–381.

[4] Benjamin, T. Z. Y., Alam, S., and Ma, C. Y., "A machine learning approach for the prediction of top of descent," *2021 IEEE/AIAA 40th Digital Avionics Systems Conference (DASC)*, IEEE, 2021, pp. 1–10. https://doi.org/10.1109/DASC52595.2021.9594470.

[5] Dalmau, R., and Prats, X., "Controlled time of arrival windows for already initiated energy-neutral continuous descent operations," *Transportation Research Part C: Emerging Technologies*, Vol. 85, 2017, pp. 334–347. https://doi.org/https://doi.org/10.1016/j.trc.2017.09.024.

[6] Félix Patrón, R. S., Berrou, Y., and Botez, R., "Climb, cruise and descent 3D trajectory optimization algorithm for a flight management system," *AIAA/3AF Aircraft Noise and Emissions Reduction Symposium*, 2014, p. 3018. https://doi.org/10.2514/6.2014-3018.

[7] Johnson, S. C., "Hierarchical clustering schemes," *Psychomertrika*, Vol. 32, No. 3, 1967, pp. 241, 254. https://doi.org/10.1007/BF02289588.

[8] McLachlan, G. J., and Rathnayake, S., "On the number of components in a Gaussian mixture model," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, Vol. 4, No. 5, 2014, pp. 341–355.

[9] Kang, J.-H., Ryu, J., and Lee, H.-T., "Analysis and Prediction of Aircraft Counts in Korean National Airspace Using Gaussian Mixture Model," *Journal of the Korean Society for Aeronautical & Space Sciences*, Vol. 52, No. 1, 2024, pp. 77–86.

[10] Buchholz, K., "The Busiest Air Routes in the World and Stateside," Tech. rep., statista, 2023.

[11] Vijaya, Sharma, S., and Batra, N., "Comparative Study of Single Linkage, Complete Linkage, and Ward Method of Agglomerative Clustering," *2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon)*, 2019, pp. 568–573. https://doi.org/10.1109/COMITCon.2019.8862232.

[12] Douglas, D., and Peucker, T., "Algorithms for the reduction of the number of points required to represent a digitized line or its caricature," *Cartographica*, Vol. 10, No. 2, 1973, pp. 112, 122. https://doi.org/10.3138/FM57-6770-U75U-7727.