proceedings

MDPI

Proceedings

Trajectory Clustering within the Terminal Airspace Utilizing a Weighted Distance Function †

Samantha J. Corrado *, Tejas G. Puranik , Oliva J. Pinon and Dimitri N. Mavris

Aerospace Systems Design Laboratory, Georgia Institute of Technology, Atlanta, GA 30332-0150, USA; tpuranik3@gatech.edu (T.G.P.); olivia.pinon@asdl.gatech.edu (O.J.P.); dimitri.mavris@aerospace.gatech.edu (D.N.M.)

* Correspondence: sjcorrado@gatech.edu

† Presented at the 8th OpenSky Symposium 2020, Online, 12–13 November 2020.

Published: 1 December 2020

Abstract: To support efforts to modernize aviation systems to be safer and more efficient, high-precision trajectory prediction and robust anomaly detection methods are required. The terminal airspace is identified as the most critical airspace for individual flight-level and system-level safety and efficiency. To support successful trajectory prediction and anomaly detection methods within the terminal airspace, accurate identification of air traffic flows is paramount. Typically, air traffic flows are identified utilizing clustering algorithms, where performance relies on the definition of an appropriate distance function. The convergent/divergent nature of flows within the terminal airspace makes the definition of an appropriate distance function challenging. Utilization of the Euclidean distance is standard in aviation literature due to little computational expense and ability to cluster entire trajectories or trajectory segments at once. However, a primary limitation in the utilization of the Euclidean distance is the uneven distribution of distances as aircraft arrive at or depart from the airport, which may result in skewed classification and inadequate identification of air traffic flows. Therefore, a weighted Euclidean distance function is proposed to improve trajectory clustering within the terminal airspace. In this work, various weighting schemes are evaluated, applying the HDBSCAN algorithm to cluster the trajectories. This work demonstrates the promise of utilizing a weighted Euclidean distance function to improve the identification of terminal airspace air traffic flows. In particular, for the selected terminal airspace, if trajectory points closer to the border of the terminal airspace, but not necessarily at the border, are weighted highest, then a more accurate clustering is computed.

Keywords: trajectory clustering; HDBSCAN; machine learning; ADS-B; air traffic management

1. Introduction and Background

Global aviation system modernization efforts have been in progress in the last decades and include the FAA's Next Generation Air Transportation System (NextGen) [1] portfolio in the U.S. and the Single European Sky Air traffic management (ATM) Research (SESAR) [2] program in Europe. The modernization efforts are long-term plans with a large focus on improving the safety and efficiency of aircraft operations. High-precision trajectory prediction and robust trajectory anomaly detection methods are paramount to support efforts to improve the safety and efficiency of airspace systems. The abundance of sensor data collected from aircraft in flight has led to the establishment of structured routine data collection and analysis programs to improve safety. Due to the advances in modern machine learning techniques, operational insights utilizing data-driven techniques are gaining significant momentum to improve aviation safety and efficiency. Operations within the terminal airspace, in particular, are known to greatly impact both individual flight-level and entire airspace system-level safety and efficiency [3]. Consequently, developing data-driven trajectory prediction and anomaly detection methods to operate within the terminal airspace is often a focus. The high traffic density, interdependent utilization of airport and airspace resources, and more complex traffic dynamics in the constrained terminal airspace make the tasks of trajectory prediction and anomaly detection within the terminal airspace challenging [3]. Therefore, in-depth analyses of terminal airspace trajectory operations are required.

Trajectories following a similar spatial path or belonging to the same *air traffic flow* tend to correspond to a specified structured operation within the terminal airspace, as defined by air traffic control (ATC). For each structured operation, the approach and climb paths mandated by ATC typically have very distinct energy profiles. Thus, unless a prediction or an anomaly detection algorithm uses the additional spatial context related to the air traffic flow to which the trajectory belongs, the algorithm may yield more conservative results. For instance, applying anomaly detection methods to an entire terminal airspace dataset without contextual spatial information may result in anomalous flights remaining undetected as their states are embedded within the entire dataset [4]. Additionally, applying prediction methods to an entire dataset without contextual spatial information may result in less precise predictions. Further, in the context of air traffic flow management, it is essential to identify air traffic flows to characterize temporal variations to support system-level air traffic management (ATM) decision-making [3]. Overall, it is evident that the accurate identification of air traffic flows in terms of their spatial context within the terminal airspace is necessary.

The *trajectory clustering* task is generally formulated as an unsupervised machine learning (ML) problem. The task involves partitioning a collection of trajectories into similar groups to define flows [5]. Several different trajectory clustering methods have been employed to identify air traffic flows in previous works. Eckstein [6] presented an automated flight track taxonomy, which identifies air traffic flows by combining Principal Components Analysis (PCA) and k-means clustering. Rehm [7] utilized a single linkage hierarchical clustering technique to identify air traffic flows. Enriquez and Kurcz [8] developed an air traffic flow detection algorithm based on spectral clustering. Gariel et al. [9] proposed two methods for identifying air traffic flows: (1) way-point-based trajectory clustering; and (2) trajectory-based clustering via PCA. Both of Gariel et al.'s proposed methods leverage the DBSCAN (Density-Based Spatial Clustering of Applications with Noise) [10] algorithm. Conde Rocha Murca et al. [3] presented a data mining framework to characterize air traffic flows in metroplex systems, where DBSCAN is utilized to learn the typical network of nominal arrival and departure flows. Olive and Morio [11] proposed a method to identify air traffic flows specifically in the terminal airspace, which recursively applies DBSCAN to cluster significant trajectory points [11]. Further, as a data pre-processing step, Puranik [12], Deshmukh [4], and Olive and Basora [13] utilized DBSCAN to identify air traffic flows prior to performing trajectory anomaly detection. Additionally, Olive and Basora [14] demonstrated the use of a GMM (Gaussian Mixture Model) for trajectory clustering. In another work, Olive and Barsora [15] utilized the HDBSCAN (Hierarchical DBSCAN) algorithm to identify air traffic flows for en-route trajectories. Basora et al. [16] presented a trajectory clustering framework utilizing HDBSCAN to analyze air traffic flows. Additionally, Tanner and Strohmeier [17] applied HDBSCAN to cluster surface trajectories for runway utilization assessments. Most recently, Olive et al. [18] demonstrated the utilization of deep clustering techniques to identify and characterize air traffic flows.

Similar data instances are grouped into clusters according to a defined *distance function*. It has been noted that one of the most critical components of trajectory clustering is the definition of an appropriate distance function [11]. Trajectories are functional in nature, not independent data points, so selection of an appropriate distance function is often less straightforward [18]. Additionally, the computation time of a distance function must scale well with increases in data size to ensure that clustering algorithms are efficient.

The majority of aviation literature focuses on the spatial dimension of air traffic flows (trajectories' latitude, longitude, and, sometimes, altitude features) [4,6–9,11–15,18]. Most distance functions require re-sampling all trajectories to standardize the length of the trajectories such that each trajectory is represented by an n-dimensional vector of points [13]. The Euclidean distance (ED) is most often utilized to compute the distance between two n-dimensional trajectories [13]. Provided two n-dimensional trajectory vectors, T^i and T^j containing x and y features, where $T^i = [(x_1^i, y_1^i), (x_2^i, y_2^i), ..., (x_n^i, y_n^i)]$ and $T^j = [(x_1^j, y_1^j), (x_2^j, y_2^j), ..., (x_n^j, y_n^j)]$, the ED computation proceeds as:

$$D_{ED}^{i,j} = \sqrt{\sum_{k=1}^{n} [(x_k^i - x_k^j)^2 + (y_k^i - y_k^j)^2]}$$

where x_k and y_k indicate an aircraft's horizontal position related to longitude and latitude measurements, respectively, at point k in the sequence of points making up the aircraft's trajectory.

Due to the convergence/divergence of trajectories within the terminal airspace, the distances computed between trajectory points closest to the airport will be relatively small for all trajectories, regardless of the air traffic flow to which the trajectories belong. This is due primarily to runway configuration and assigned arrival and departure runways. On the other hand, distances between trajectories within the same flow may be relatively large between trajectory points considered at the terminal airspace's defined border. The distance between these trajectory points depends on the density of the air traffic flows and the radius around the airport. The uneven distribution of distances as aircraft arrive at or depart from the airport may skew the classification, resulting in misclassification of trajectories or inadequate air traffic flow identification. Therefore, it is noted that the utilization of the ED to cluster trajectory records may not be adequate [11,19].

Other distance functions have been proposed to address the issues related to the limitations of the ED, such as the Symmetrized Segment Path Distance (SSPD) [19] distance. Basora et al. [16] compared the utilization of the ED versus the SSPD to cluster en-route trajectories and found that the utilization of the SSPD results in more precise identification of air traffic flows, yet there exists a significant trade-off with computation time [16]. A large computation time may prevent subsequent applications of the algorithm in real-time. Gariel et al. [9] and Olive and Morio [11] presented iterative point-based clustering methods to identify air traffic flows. However, clustering of entire trajectory records or multi-point segments at once is often a requirement to enable the extension of clustering methods to real-time applications. While real-time application is not the focus of this work, aviation organizations have set goals to ultimately deploy novel trajectory prediction and anomaly detection methods in real-time or near-real-time [20]. Considering the previous observations, the following research objective is proposed for this work:

Research Objective: Demonstrate the implementation of a computation-time-efficient distance that results in a more accurate clustering of the converging and diverging trajectories within the terminal airspace when compared with the Euclidean distance.

2. Method

A primary limitation in the utilization of the ED is that the uneven distribution of distances as aircraft arrive at or depart from the airport may result in inadequate identification of air traffic flows. Therefore, a distance that places more importance on specified trajectory points in the sequence of re-sampled trajectory points could mitigate the limitations of the ED. An alternative distance providing this ability is the weighted Euclidean distance (WED). The WED is a slight modification of the ED such that an extra multiplicative factor is added, which does not significantly increase computation time, nor does it prevent the extension to real-time applications. Considering the same two n-dimensional trajectory vectors, T^i and T^j, and a weight vector, W, where $W = [w_1, w_2, ..., w_n]$, the WED computation proceeds as:

$$D_{WED}^{i,j} = \sqrt{\sum_{k=1}^{n} w_k[(x_k^i - x_k^j)^2 + (y_k^i - y_k^j)^2]}$$

The WED introduces an additional trajectory point weighting term, w_k, which enables an "importance" to be assigned to each trajectory point. The importance may be determined according to the position in the sequence of points relative to the airport's location or bounds of the airspace. Thus, the WED is proposed as a computation-time-efficient distance that results in a more accurate clustering of the converging and diverging trajectories within the terminal airspace compared with the ED through the use of a weighting factor for trajectory points.

Due to the increased prevalence of the HDBSCAN [21] algorithm within the aviation literature [16,22,23], it is selected as the clustering algorithm for which to test the effectiveness of the WED. The HDBSCAN algorithm extends the widely-utilized DBSCAN by converting it to a hierarchical clustering algorithm, where it ultimately extracts a flat clustering (set of clusters without any explicit structure that relates the clusters to each other) based on the stability of the identified hierarchical clusters [24]. The algorithm's primary advantage over DBSCAN is identifying clusters of varying density, which is relevant for applications in the terminal airspace where flows often have varying densities, i.e., some flows appear very tight, while others are more spread out. HDBSCAN requires a single input parameter that sets the minimum cluster size to perform clustering [21]. Additionally, an optional "smoothing" parameter may be set that provides a measure of how "conservative" the clustering should be [24]. The larger is the smoothing parameter value, the more points are declared as outliers, and the clusters are restricted to more dense areas [24]. To demonstrate the proposed method's efficacy, only horizontal spatial features (latitude and longitude) are considered for clustering.

3. Implementation

From the OpenSky Network's [25] historical database, flight records are retrieved for flights arriving at San Francisco International Airport (KSFO) in 2019 via the traffic Python library [26]. Additionally, ASOS (Automated Surface Observing System) data are extracted. ASOS stations take minute-by-minute observations and generate basic weather reports, with the access to the processed weather reports being provided by the Iowa Environmental Mesonet [27].

3.1. Data Processing

First, the data must be extracted from the two data sources. The extraction of ASOS data is straightforward, requiring simple processing to fill and interpolate missing values. Then, OpenSky Network state vectors within a 20 nautical mile radius of KSFO and below 25,000 ft altitude are requested via the traffic Python library for all days in 2019. State vectors that are redundant, contain empty values for latitude, longitude, altitude, heading, or groundspeed, or cannot be associated with a commercial flight utilizing their callsign, are discarded. The ASOS data are then fused with the state vectors based on the state vector timestamp.

Next, state vectors are split by callsign. Then, the state vectors split by callsign are further split into individual flight segments if the time difference between two successive state vectors is greater than 5 min. A unique identifier is assigned to each flight segment. If the median of the first five state vector altitude measurements is greater than the median of the last five state vector altitude measurements, then the flight segment is characterized as an arrival, and departures are discarded. The touchdown point is identified as the first point where the groundspeed reaches below 100 knots or the last point if no state vector groundspeed measurement is recorded to reach below 100 knots.

Utilizing the mean sea level pressure from the fused ASOS data, a corrected airport altitude at the touchdown point time stamp for each flight segment is computed. If the difference between the touchdown state vector altitude measurement and the corrected airport altitude is greater than 100 ft (the flight segment does not reach within 100 ft above the ground level), then the flight segment is discarded. Then, cumulative ground track distance from the touchdown point is computed for each state vector in each flight segment.

The flight segments are then re-sampled on 50 uniform intervals between 0 and the maximum cumulative ground track distance value for the flight segment within a 20 nautical mile radius of KSFO.

Each state vector's latitude and longitude measurements are projected onto a UTM coordinate system. The Cartesian coordinates are then stacked to generate feature vectors for each flight, and the feature vectors for all flights are aggregated to produce the clustering input matrix.

The final dataset contains 178,890 total arrival flight segments. The dataset contains both domestic and international flights on 187 airlines, where 110 unique aircraft types are identified. From the final dataset, 237 days are extracted and separated into daily datasets for clustering, where each day contains around 500 trajectories. Additionally, 30 weeks are extracted and separated into weekly datasets for clustering, where each week contains around 3500 trajectories.

3.2. HDBSCAN Clustering

For each dataset in the data groups (daily and weekly), an HDBSCAN clustering is performed to evaluate the impact of utilizing the WED as opposed to the ED. The hdbscan Python library [24] is leveraged to perform the clustering. To select values of W to assess the performance of the WED compared with the ED, the weights w_k are varied according to functions inspired by the shapes of normal and beta distribution probability density and cumulative distribution functions. Four different weighting schemes are selected, and are displayed in Figure 1. The point 0 is the point in the trajectory closest to the airport location (the touchdown point), whereas point 50 is the first point in which the trajectory enters the terminal airspace.

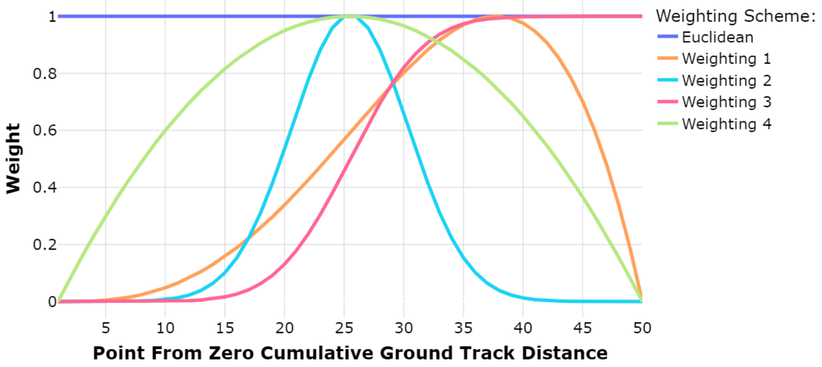

Figure 1. Weighting schemes utilized.

Weighting 2 is included to test the method's robustness and emphasize the importance of selecting an appropriate weighting scheme. Weighting 2 weights the trajectory points in the center of the trajectory higher, yet most other points are weighted near zero on both ends of the trajectory. This should result in a less accurate clustering due to the focus on a limited sub-sequence of the trajectory points.

For the clustering of each dataset, the parameter controlling the minimum cluster size is set to 2% of the total number of flights within the dataset. For the daily datasets, this is an average of 10 trajectories, and for the weekly datasets, an average of 70 trajectories, which is consistent with other studies leveraging HDBSCAN or DBSCAN clustering [14,16]. As the WED weighting schemes and ED are compared against each other directly for each dataset, the bias of the portion of outlying trajectories is removed by utilizing the HDBSCAN smoothing parameter to control the percentage of outlying trajectories identified. Gariel et al. found from 2% to 16% outlying trajectories per day [9]. In this study, for each dataset, an iterative loop is implemented to adjust the smoothing parameter in each loop, such that the percentage of outlying trajectories is restricted to between 10% and 15% of the total number of trajectories within the dataset to remove outlier bias.

To compare the WED weighting schemes' performance with that of the ED, the mean distance between trajectories assigned to a flow and their respective flow centroid is considered. If air traffic flows are more accurately identified, then it follows that the distances between trajectories assigned to a flow and their respective flow centroid will be smaller. Therefore, if the mean of these distances for the WED weighting schemes is lower than that for the ED, and utilization of the WED does not significantly increase computation time, then the proposed method improves terminal airspace air traffic flow identification. Graphic representation of specific daily or weekly datasets enables visual inspection to confirm conclusions.

4. Results and Discussion

When clustering both the daily and weekly datasets, utilizing the hdbscan Python library [24], it was found that the computation time difference between utilizing the ED or WED to cluster the datasets is negligible. For each dataset group (daily and weekly), the comparison of the value of the mean distance between trajectories assigned to a flow and their respective flow centroid for WED weighting schemes and ED is computed. The comparison is represented as a bar chart of the percent of times each of the WED weighting schemes performs better (has a lower value of mean distance between trajectories assigned to a flow and their respective flow centroid) than the ED. Additionally, a box plot is generated to visualize the ratio between the mean distance between trajectories assigned to a flow and their respective flow centroid computed utilizing the ED and WED weighting schemes such that the improvement may be quantified. In this case, if the ratio is above 1, which indicates that the ED resulted in a higher mean distance between trajectories assigned to a flow and their respective flow centroid than the WED weighting scheme presented. Thus, ratios greater than 1 indicate that the WED performed better than the ED and produced tighter clusters. Results for the daily and weekly datasets are presented.

4.1. Daily Datasets

Figure 2 displays the overall results for all 237 days of daily datasets. Three of the weighting schemes, Weightings 1, 3, and 4, appear to perform better than the ED on most daily datasets. Specifically, Weightings 1 and 3 notably perform the best, where both of these weighting schemes have a similar skewed shape, with the primary difference being in the weighting values of the 10 trajectory points closest to the terminal airspace border. As anticipated, Weighting 2 does not perform well, likely due to its emphasis on only a few points in the trajectory sequence's center points. Weighting 4 does not perform as well as Weightings 1 and 3. Similar to Weighting 2, Weighting 4 is symmetric with the highest-weighted points being those in the center of the trajectory point sequence.

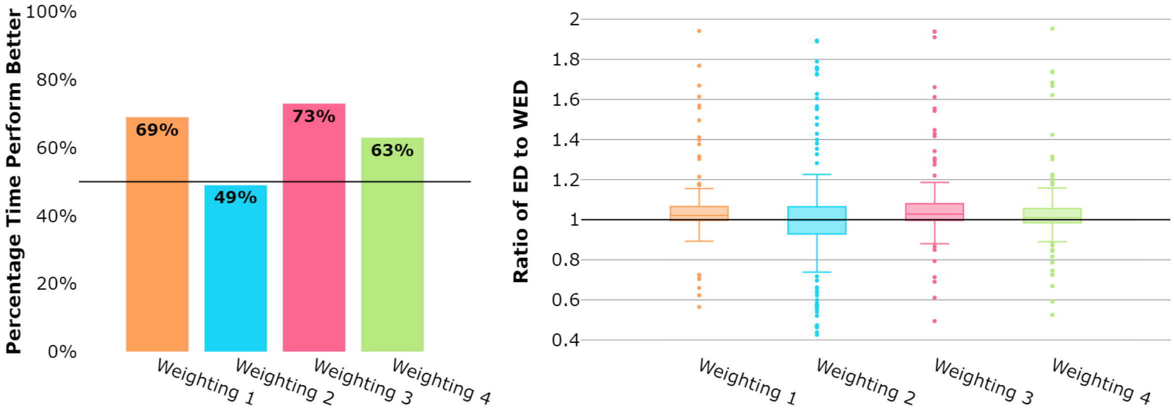

Figure 2. Daily datasets: Mean distance between trajectories assigned to a flow and their respective flow centroid.

To further assess results for individual datasets, a single daily dataset is graphically presented in Figure 3 for the ED and Weighting 1. Both the ED and Weighting 1 clusterings identify five distinct air traffic flows.

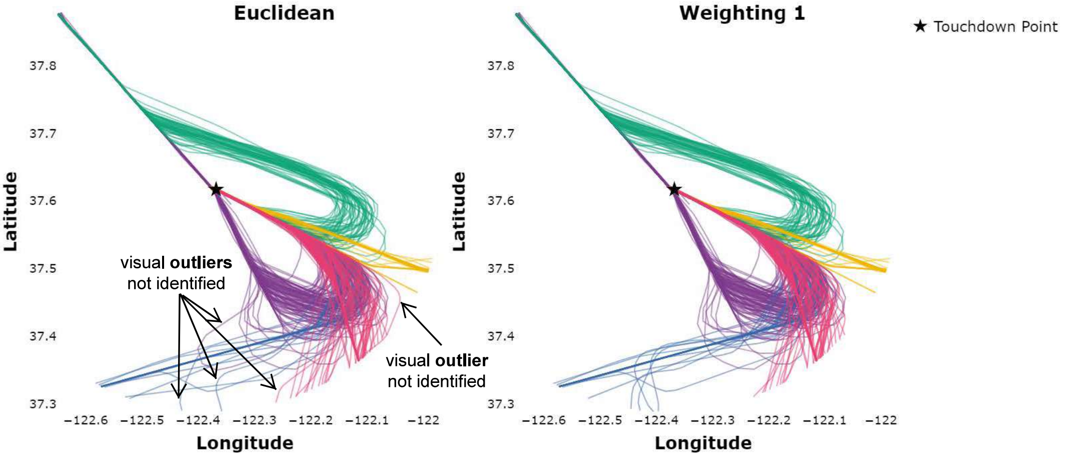

Figure 3. Flows identified utilizing the euclidean distance versus Weighting 1 for a single day.

It is observed that some trajectories are assigned to clusters when utilizing the ED that, visually, appear as likely outliers. On the other hand, the clustering utilizing Weighting 1 does identify these trajectories as outliers. Visually, some clusters (air traffic flows) identified utilizing Weighting 1 appear to be "tighter" or more accurate, which was one of the objectives in utilizing the WED. Figure 4 displays the pink and purple flows in more detail such that the differences may be observed.

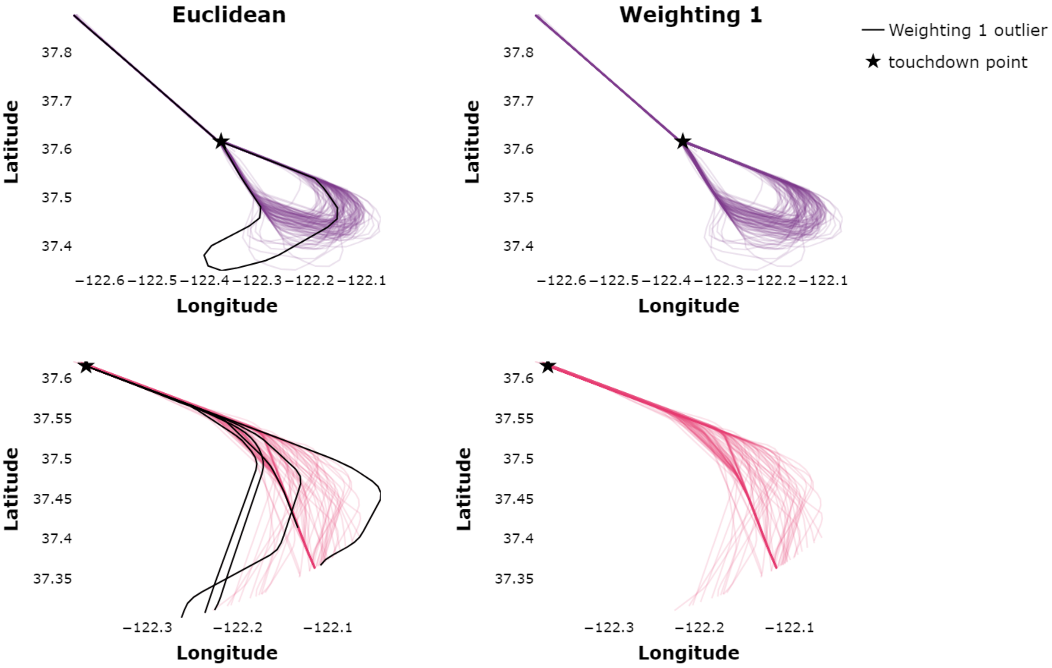

Figure 4. Purple flow identified utilizing the euclidean distance versus Weighting 1 on a single day.

The solid black lines in Figure 4 represent trajectories that are not identified as outliers utilizing the ED but are identified as outliers when utilizing Weighting 1. In Figure 4, it is evident that the solid black trajectories are not as nominally conforming with the dominant flow paths or other trajectories in the flow, and thus should be flagged as off-nominal or outliers. Overall, the quantitative results obtained through Figure 2 are supported through visual inspection of the ED versus Weighting 1 clusterings.

4.2. Weekly Datasets

Figure 5 displays the overall results for all 30 weeks of weekly datasets. Two of the weighting schemes, Weightings 1 and 3, appear to perform better than the ED. In this case, Weighting 4 does not perform better for a majority of the weekly datasets, which is likely due to the inferior performance of symmetric weighting schemes being more pronounced in clustering the weekly datasets that include many more trajectories and, potentially, flows. Again, Weighting 2 performs the worst, which is expected.

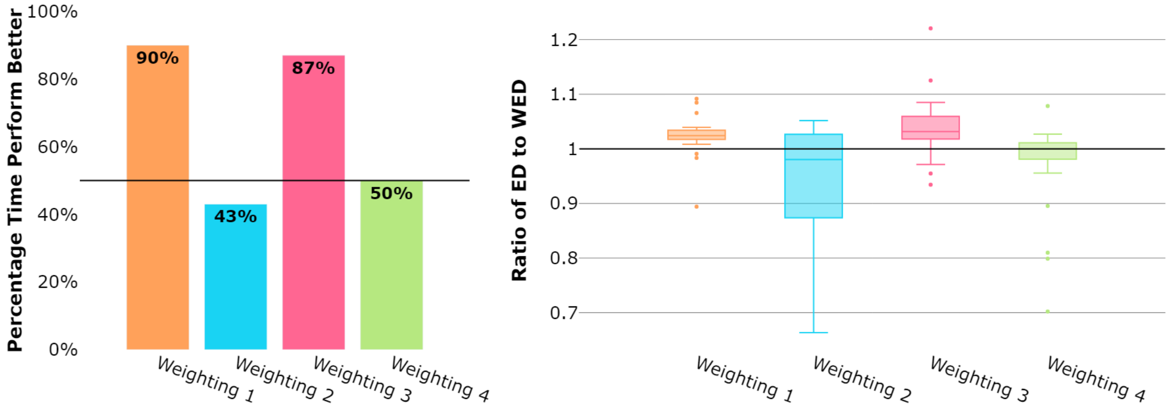

Figure 5. Weekly datasets: Mean distance between trajectories assigned to a flow and their respective flow centroid.

Similarly, the visual inspections of individual weekly datasets for Weightings 1 and 3 are performed, where results similar to those for the daily datasets emerge. Therefore, the WED can be utilized to more accurately identify air traffic flows within the terminal airspace. Additionally, weighting schemes placing more weight on trajectory points between the center of the sequence of trajectory points and those approaching the border of the airspace, but not necessarily at the border, perform better for the KSFO terminal airspace.

5. Conclusions

In summary, the WED shows promise in improving air traffic flow identification compared to the commonly-utilized ED on both daily and weekly datasets. The computation time difference between the ED and the WED is not significant. Further, more accurate flows are identified, which also indicates a more robust identification of outlier trajectories. It is demonstrated that the selection of an appropriate weighting scheme is important. In particular, for the KSFO terminal airspace, if trajectory points closer to the border of the terminal airspace, but not necessarily at the border, are weighted highest, then a more accurate clustering is computed. However, the most appropriate weighting scheme may vary for different airports depending on runway configuration.

This work will subsequently be extended to other terminal airspaces with dissimilar runway configurations. Further, the proposed method will be applied to departing datasets to assess performance compared with utilizing the ED. Additionally, future work includes investigating the impact of utilizing the WED to identify air traffic flows on a subsequent trajectory anomaly detection algorithm compared with utilizing the ED.

Funding: This research received no external funding.

Acknowledgments: The authors would like to acknowledge OpenSky Network for providing access to the data utilized in this work.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Federal Aviation Administration. Modernization of the U.S. Airspace. Available online: https://www.faa.gov/nextgen/ (accessed on 27 November 2020).
2. Single European Sky ATM Research. About SESAR. Available online: https://www.sesarju.ed/discover-sesar/history (accessed on 27 November 2020).
3. Condé Rocha Murça, M.; Hansman, R.J. Data-Driven Modeling of Air Traffic Flows for Advanced Air Traffic Management. Ph.D. Thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, 2018.
4. Deshmukh, R. Data-Driven Anomaly and Precursor Detection in Metroplex Airspace Operations. Ph.D. Thesis, Purdue University, West Lafayette, IN, USA, 2020.
5. Enriquez, M. *Identifying Temporally Persistent Flows in the Terminal Airspace via Spectral Clustering*; Air Traffic Management R&D Seminar: Chicago, IL, USA, 2013.
6. Eckstein, A. Automated flight track taxonomy for measuring benefits from performance based navigation. In Proceedings of the 2009 Integrated Communications, Navigation and Surveillance Conference, Arlington, VA, USA, 13–15 May 2009.
7. Rehm, F. *Clustering of Flight Tracks*; AIAA Infotech@Aerospace: Atlanta, GA, USA, 20 April 2010.
8. Enriquez, M.; Kurcz, C. A Simple and Robust Flow Detection Algorithm Based on Spectral Clustering. In Proceedings of the International Conference on Research in Air Transportation, Berkley, CA, USA, 22–25 May 2012.
9. Gariel, M.; Srivastava, A.N.; Feron, E. Trajectory Clustering and an Application to Airspace Monitoring. *IEEE Trans. Intell. Transp. Syst.* **2011**, *12*, 1511–1524.
10. Ester, M.; Kriegel, H.P.; Sander, J.; Xu, X. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, Portland, OR, USA, 2–4 August 1996; AAAI Press: Palo Alto, CA, USA, 1996, pp. 226–231.
11. Olive, X.; Morio, J. Trajectory Clustering of Air Traffic Flows around Airports. *Aerosp. Sci. Technol.* **2019**, *84*, 776–781.
12. Puranik, T.G. A Methodology for Quantitative Data-driven Safety Assessment for General Aviation. Ph.D. Thesis, Georgia Institute of Technology, Atlanta, GA, USA, 2018.
13. Olive, X.; Basora, L. Detection and Identification of Significant Events in Historical Aircraft Trajectory Data. *Transp. Res. Part C Emerg. Technol.* **2020**, *119*.
14. Olive, X.; Basora, L. A Python Toolbox for Processing Air Traffic Data: A Use Case with Trajectory Clustering. In Proceedings of the 7th OpenSky Workshop 2019, Zurich, Switzerland, 21–22 November 2019.
15. Olive, X.; Basora, L. Identifying Anomalies in past en-route Trajectories with Clustering and Anomaly Detection Methods. In Proceedings of the Thirteenth USA/Europe Air Traffic Management Research and Development Seminar, Vienna, Austria, 17–21 June 2019.
16. Basora, L.; Morio, J.; Mailhot, C. A Trajectory Clustering Framework to Analyse Air Traffic Flows. In Proceedings of the 7th SESAR Innovation Days, Belgrade, Serbia, 28–30 November 2017.
17. Tanner, A.; Strohmeier, M. Anomalies in the Sky: Experiments with traffic densities and airport runway use. In Proceedings of the 7th OpenSky Workshop 2019, Zurich, Switzerland, 21–22 November 2019.
18. Olive, X.; Basora, L.; Viry, B.; Alliger, R. Deep Trajectory Clustering with Autoencoders. In Proceedings of the International Conference on Research in Air Transportation 2020, ICRAT, Online, 15 September 2020.
19. Besse, P.; Guillouet, B.; Loubes, J.M.; Royer, F. Review & Perspective for Distance Based Clustering of Vehicle Trajectories. *IEEE Trans. Intell. Transp. Syst.* **2016**, *17*, 3306–3317.
20. National Academies of Sciences, Engineering, and Medicine. In *In-Time Aviation Safety Management: Challenges and Research for an Evolving Aviation System (2018)*; The National Academies Press: Washington, DC, USA, 2018.
21. Campello, R.J.G.B.; Moulavi, D.; Sander, J. Density-Based Clustering Based on Hierarchical Estimates. In Proceedings of the Advances in Knowledge Discovery and Data Mining, Gold Coast, Australia, 14–17 April 2013; AAAI Press: Palo Alto, CA, USA, 2013, pp. 160–172.
22. Fernandez, A.; Martinez, D.; Hernandez, P.; Cristobal, S.; Schwaiger, F.; Nunez, J.M. Flight Data Monitoring (FDM) Unknown Hazards detection during Approach Phase using Clustering Techniques and AutoEncoders. In Proceedings of the SESAR Innovation Days 2019, Athens, Greece, 2–5 December 2019.

23. Jarry, G.; Delahaye, D.; Nicol, F.; Feron, E. Aircraft atypical approach detection using functional principal component analysis. *J. Air Transp. Manag.* **2020**, *84*, 101787.
24. McInnes, L.; Healy, J.; Astels, S. hdbscan: Hierarchical density based clustering. *J. Open Source Softw.* **2017**, *2*, 205. doi:10.21105/joss.00205.
25. Schäfer, M.; Strohmeier, M.; Lenders, V.; Martinovic, I.; Wilhelm, M. Brining Up OpenSky: A Large-scale ADS-B Sensor Network for Research. In Proceedings of the 13th IEEE/ACM International Symposium on Information Processing in Sensor Networks (IPSN), ICRAT, Berlin, Germany, 15–17 April 2014.
26. Olive, X. traffic, a toolbox for processing and analysing air traffic data. *J. Open Source Softw.* **2019**, *4*, 1518. doi:10.21105/joss.01518.
27. Iowa Environmental Mesonet. Available online: https://mesonet.agron.iastate.edu/request/download.phtml (accessed on 27 November 2020).

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.