Visualizing Corridors in Terminal Airspace using Trajectory Clustering

Carlos Paradis
KBR Wyle Services, LCC
Moffett Field, USA
carlos.paradis@us.kbr.com

Misty D. Davies
Intelligent Systems Division
NASA Ames Research Center
Moffett Field, USA
misty.d.davies@nasa.gov

***Abstract*—Context: Advances in battery and automation technology have made routine air taxi and cargo transport in urban areas a business model that can be attained by emerging aviation innovators. The community vision and work to enable these novel operations is discussed using the term 'Urban Air Mobility' or UAM. Small, piloted, airspace vehicles that fly with a few passengers do operate in urban areas today, and these vehicles can be studied as an early proxy for this future UAM traffic.**

Aim: We seek to identify corridors already in daily operation and their properties.

Method: We applied DBSCAN and HDBSCAN to Dallas Forth-Worth TRACON flight data to identify corridors in use, their density, and devised a method to annotate landing sites used in these corridors with site metadata.

Results: While DBSCAN was unable to group similar trajectories, we we were able to successfully identify corridors using HDBSCAN, measure their density and annotate them.

Conclusion: The applied method can successfully identify corridors in daily operation with additional metadata to help domain expert understand the intent of UAM corridors.

***Index Terms*—UAM, Trajectory, TRACON, Clustering, DBSCAN, HDBSCAN**

I. Introduction

Urban centers have begun encouraging novel aviation missions to transport goods and people for relatively short distances and at low altitudes over dense populations. These novel aviation missions, known collectively as Urban Air Mobility or UAM, require community consensus and potentially airspace rules in order to enable the scale of operations that are envisioned. Industry, academic, and government committees are currently working through definitions and specifications for this airspace [1]. A common assumption is that, at least in the first-approved regular operations, this traffic will be primarily confined to corridors. Early concepts of operation imagine these corridors between smaller suburban airports, larger airports, and downtown centers.

UAM at scale is being driven by innovations in electric propulsion, battery density, and automation, and there are very few of these novel, highly-automated, electric vehicles in operation today, with even fewer being given waivers to fly over dense populations. However, small piloted air vehicles that fly with a few passengers do operate in urban areas today, and these vehicles can be studied as an early proxy for this future UAM traffic.

In this work, we use recent historical data from the Sherlock Data Warehouse [2] in the Dallas metropolitan area (within 30 nautical miles of the Dallas-Fort Worth Terminal Radar Approach Control (DFW-TRACON)) to identify these low-altitude, primarily rotorcraft flights that can be used as proxies for the future UAM traffic. We examine this traffic to identify de facto corridors that already exist in flight operations today. We created this UAM proxy dataset by identifying and filtering flights that both originate and terminate within the Dallas-Fort Worth metropolitan area and never exceed an altitude of 5000 feet. When the trajectories of these flights are visualized, clear corridors emerge.

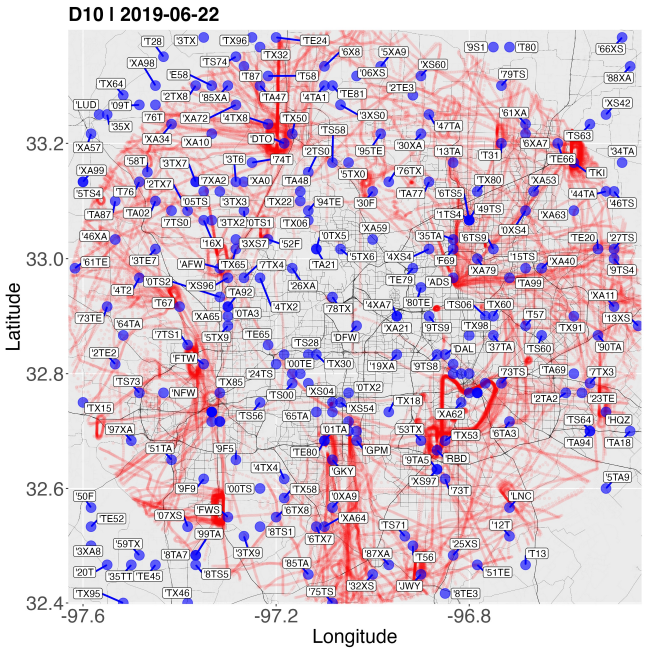

Fig. 1. Dallas Forth Worth Terminal Airspace (D10 TRACON) — 2019-06-22.

To motivate our work, Figure 1 displays trajectories flown by aircraft whose origin and destination is not specified in the flight plan for the Dallas Forth Worth Terminal Airspace (D10 TRACON) on June 22, 2019. The data is augmented with FAA metadata of vertiports and airports represented as blue circles, while the flown trajectories are shown in red. When

trajectory points overlap, the overlapping region is darkened. As can be seen in the figure, common trajectory patterns emerge, which hereafter we define as corridors. While many of the corridors suggest landing maneuvers, others such as the "heart shaped" corridor (-96.8,32.8) suggests the presence of pre-existing corridors in current use. Our interest in this work is to automatically identify such corridors, while providing sufficient information for domain experts for sense making. While the emergence of corridors may be manually visualized by the overlay of their track points' density, the study of the set of track points as trajectories is required for automating the identification of the corridors, and is non-trivial.

In this work, we utilize trajectory clustering to identify UAM corridors, and subsequently study their properties given these trajectories within the Dallas-Forth Worth metropolitan area. We have prototyped an automated way to identify these corridors, and have begun using statistical and automated means for their analysis and specifications. This automated toolchain allows us to replicate our analysis more quickly for other metropolitan areas and enables rapid historical and future change analysis.

The novelty of our work, we argue, lies in a method to study the emerging properties of these corridors, such as the flights' degree of separation, their likely locations, and the current-day capacity. We also augment the identified corridors with landing sites of each corridor, which facilitates understanding the corridor's purpose. The impact of the work lies in understanding and specifying this traffic in the airspace today, prior to operations at the scale envisioned by UAM. This will allow us to compare today's operations against the concepts of operation of future air traffic, as it is proposed. Additionally, it will give us a baseline of current operations to use as we evolve traffic towards these novel large-scale UAM operations.

Towards this vision, we ask the following research questions:

> RQ1: Can we automatically identify existing corridors using trajectory clustering?

While we may visually observe corridors in Figure 1 by overlaying track points, it is not possible to immediately identify which flights participated in different corridors as they are represented mathematically as individual overlaid track points. By identifying corridors, which cluster track points as overlapping trajectories, we can leverage additional metadata pertaining to each individual flight.

> RQ2: Can corridors be automatically annotated for domain expert sense making?

In Figure 1 we display FAA sites annotated in blue. If the mapping of flights to corridors is possible, it would be useful to know what landing sites are also part of the corridors, as the purpose of the sites can provide further insight on the corridor purpose. For instance, if a corridor uses landing sites such as airports and heliports near tourist centers, its usage may be associated to tourism. Other information such as time and day of the week, combined with traffic, may also further enable analysts to label the primary purpose of the corridor.

> RQ3: Can the density of traffic within the corridors be measured?

The density of airspace is inversely proportional to safety, with increased density directly adding complexity to the airspace. We derive a simple measure of corridor density, by counting the number of flights in a corridor normalized by the number of flights in a given a day.

> RQ4: Do flights that participate in the same corridor share the full trajectory, or only parts of it?

While visually adjacent track points of multiple flights may suggest all start and end at the same airports or vertiports, it may be possible smaller corridors may be used independently by the flights. It would be therefore useful to dynamically observe the formation of corridors from the individual flights to assess their trajectory overlap.

II. Related Work

The rising interest in Urban Air Mobility (UAM) has led to a variety of studies concerning the airspace architecture. Broadly, we consider studies whose primarily purpose is concept proposals [1], simulations (e.g. NASA's Terminal AutoResolver [3], [4], MATSim [5]), and pattern finding (e.g. identifying unused spaces [6], [7]). Our work fits under pattern finding, and more specifically trajectory clustering. When comparing trajectories for grouping based on similarity, each combination of airspace and flight phase presents its own challenges in choice of pre-processing techniques [8]. For example, airport surfaces require the handling of stop points which can lead to an over-estimate of taxi distance [9]. The types of airspace trajectory clustering that have been applied can be broadly divided by terminal airspace [10]–[12], transition en-route airspace [13], and airport surfaces [8]. In this work, we are interested in Urban Air Mobility in the terminal airspace using trajectory clustering. Related work also reflects variability in how data is pre-processed before trajectory clustering is performed. As such, we dedicate the remainder of our related work discussion to enumerate the pre-processing steps that have been applied for trajectory clustering in terminal airspace (irrespective of being related to urban air mobility) to contextualize our work contributions and the possibilities for future work.

Dataset sub-setting. At its most basic form, the data available at a TRACON provides us with a set of flights in which the trajectories can be grouped. Different data

sources and areas have been used in related work such as the Chicago Terminal Radar Control Airspace (C90 TRACON) using FlightAware [10], [11]. Furthermore, trajectories have been partitioned in different ways before clustering, such as their arrival and departure, weather conditions [10] and visual flight rules (VFR) [11]. In comparison to related work, we did not perform any differentiation between arrival and departure trajectories, as one of the criterion for filtering UAM flights in our dataset was that, for the low-altitude small flights of interest we are using as proxies, arrival and departure points are often not specified in the given flight plan dataset. We also have not performed any further subsetting of trajectories, as we wish to assess what minimal set of data transformations would suffice to identify UAM corridors.

Feature Engineering. When grouping trajectories, we must decide what information of the trajectories should be used for similarity. Minimally, latitude and longitude is used. Both Li and Ryerson [10] and Gariel et al. [11] used the planar coordinates (x,y) for trajectory clustering. Gariel et al. [11] justifies not using altitude due to their definition of "at or above ft", being less accurate in contrast to the planar localization for one of the two provided algorithms. For the second algorithm, a combination of planar position, distance from center of the TRACON, distance from "the top left corner" of the bounding box containing the TRACON, the angular position in cylindrical coordinates and heading of the aircraft are used. In Li and Ryerson the planar coordinates are used "as-is" for trajectory clustering, while in Gariel et al. the planar coordinates (x,y) are used to derive the turning points of each trajectory as features for clustering. The original trajectories are then expressed as a sequence of clusters for longest common sequence comparisons. In our work, we preferred to only use planar coordinates, as the information is expected to be available in other datasets, and thus allow for easier replication.

Data Collection Irregularities. Beyond the choice of dataset and partitions, the process of data collection itself may require different pre-processing methods. For instance, if collected data is not accurate, the use of a low-pass filter may be desirable [11]. Another issue in the data collection process is the irregularity of sampling interval. For example, if one aircraft has a 30 second interval between two consecutive observations, whereas another aircraft has a 1 minute interval [10]. Such irregularities may result from the rotational speed of the radar [11]; and when extreme, deriving a common time index may be necessary. Li and Ryerson address the irregularity by interpolation, whereas Gariel et al. use re-sampling to reduce each feature vector to a size *n=50* (smaller vectors were discarded). The choice of size is stated to provide the best accuracy for the model.

Trajectories of varying length. Another issue reported in the literature is the variation of data points between trajectories, since it is not guaranteed that each aircraft will have the same number of observations [6, p. 60], even if the sampling interval was invariant [10]. While trajectories may vary in size, a number of clustering methods, including those used in this work, may require trajectories to be of the same observation length. Trajectories can vary wildly in size (e.g. 10 to 550 points [11]) due to shorter flight duration. Re-sampling vectors to constrain their length is a popular approach; however, loss of the original trajectory's smoothness to high length variation may occur [6, p. 60]. This re-sampling can be done by randomly choosing any trajectory's track points, or by choosing trajectory points indexed by time (temporal normalization). For temporal normalization, the re-sampling normalizes the time stamps for each trajectory in the interval $\rho = [0, 1]$, divides ρ into a fixed number of equally sized time intervals, and linearly interpolates the spatial position for the fixed number of normalized time stamps in ρ. The result is a feature vector $F_i = (x_{i1}, y_{i1}, x_{i2}, y_{i2}, ..., x_{in}, y_{in})$ [6][p.62]. In our dataset, we addressed the trajectories' varying length by re-sampling larger trajectories, and discarding smaller trajectories. We chose this approach over temporal normalization to be conservative, and avoid introducing artificial track points.

Clustering Pre-processing. Clustering algorithms may require the use of normalization. To guarantee that features used in trajectory clustering are considered with the same importance, their values are commonly normalized between 0 and 1 [11], [6, p. 63]. An additional reported pre-processing step is the use of dimensionality reduction, specifically principal component analysis [11], to reduce the number of features. For instance, [11] reported using *p=5* components provided best accuracy results. In our work, we opted to evaluate if a simpler pipeline could provide useful results for domain experts, and decided to not perform normalization or dimensionality reduction.

Model and Evaluation. Both the use of K-Means and DBSCAN [14] have been reported in the literature for clustering trajectories. Li and Ryerson [10] used DBSCAN for each combination of STAR procedure and weather condition. The authors assessed the quality of the obtained trajectory clusters and choice of hyperparameters using the Davies-Bouldin index, the percentage of trajectories classified as outliers [13], and comparisons of the resultant clusters against common runaway configurations in the chosen airport. In Gariel et al. [11], the authors use K-Means or DBSCAN depending on the density of points in subsets of the data. While we have tested K-Means, DBSCAN and HDBSCAN [15], we were only able to obtain meaningful results using HDBSCAN, and therefore focus the presentation of the results on the latter.

III. Method

Trajectory clustering pipeline steps varied in related literature; we decided to build our model with the smallest number of steps that would provide meaningful results. We elaborate the steps in our pipeline in this section.

A. Dataset and Sub-setting.

To ensure replication and clarity of the dataset, we detail in this section how the data can be obtained and what transformations were performed.

Sherlock Data Warehousing. We obtained our flight dataset from the Sherlock Data Warehouse [2]. Sherlock's raw data include a variety of flight information from live streams of FAA operational systems, weather observations and forecasts, and NAS advisories and statistics. Sherlock also provides modified data, which comprise parsed and merged data sources and metadata [2]. In the scope of this work, data files are provided in the Integrated Flight Format (IFF) version of November 25, 2020. Each IFF file can be parsed into three tables containing *header records*, *flight plan records*, and *track point records*. Moreover, the three tables can be combined via a flight id ('ftKey') common to them.

We obtained 24 hours of flight data records from June 22, 2019 within 30 nautical miles of the DFW-TRACON and an altitude of 5000 feet or below. Furthermore, to identify candidate UAM flights, we preserved in the final dataset only flights ('ftKey') which do not specify an origin or destination (i.e. the header record table *orig* and *dest* fields are specified as '?'). Of the remaining flight dataset we used their latitude ('coord1'), longitude ('coord2'), altitude ('alt'), and ground speed ('groundSpeed') data obtained from the *track point records* table. The plot in figure Figure 1, for example, showcases the latitude and longitude information in red.

Sites Dataset. The sites dataset, displayed as blue circles in Figure 1, was originally obtained from an FAA public website [1]; however, we noticed post-analysis the interface has since changed. To obtain the current site dataset, the ADIP interface can be used[2]. Specifically, a .csv table can be obtained by clicking the 'Go To Advanced Facility Research', choosing the State field as 'Texas', clicking 'Execute Search', 'Download Results' and 'Facility Data'. From this table, we utilized the 'Loc ID', 'ARP Latitude' and 'ARP Longitude' data for visualization. In addition, we converted the sites' latitude and longitude from Degrees Minutes Second (DMS) format to Decimal Degrees (DD), which is the format adopted by the IFF dataset.

B. Trajectory Representation

In order to identify overlapping flight trajectories from the IFF dataset flight track points, we must first define a representation for the trajectories suitable for using in clustering algorithms. This definition, in turn, directly impacts the choice of distance function used to compare how similar two trajectories are. Because we expect our dataset trajectories to be of different sizes, that is, the number of track points per flight to vary, we chose a fixed-size vector to represent each trajectory [6, p. 60]. Specifically, we took the median size of all flight trajectories in our dataset as our fixed-size vector. Trajectories which contained more track points than the specified threshold (which is discussed in the results section) were reduced by track point re-sampling. Trajectories which contained fewer track points than the median were removed from the dataset. We believe the choice of removing smaller trajectories is a more conservative approach than generating artificial track points, as an incorrect track point may potentially bias the properties of the identified corridors. The impact of this decision on the number of flights is further discussed in the results.

C. Feature Engineering

By defining a fixed n-sized vector of track points, we define a table of size $2 * n$, containing per flight n-longitude and n-latitude values. We defer to future work to examine how other information (e.g. altitude) impacts the identification of corridors.

D. Clustering Algorithm

As noted in the related literature section, several types of clustering algorithms have been proposed for trajectory clustering, including hierarchical, partitioning (e.g. k-means), density-based (e.g. DBSCAN) and grid-based [6, p. 61]. We decided to evaluate both DBSCAN [14] and HDBSCAN [15] which have observed successful results in related literature, and defer to future work more sophisticated methods such as Dynamic Time Warping [16] which can account for varying length trajectories. Both DBSCAN and HDBSCAN implementations are available in R[3] [17].

E. Model Tuning

The DBSCAN model introduces two thresholds that must be specified a priori, the number of neighbors *MinPts* and the radius *Eps* [14]. A intuitive idea of the two parameters is to consider that for a trajectory to be considered a corridor (cluster), it is required to have at least a minimum number of neighbor trajectories *MinPts* in an *Eps*-neighborhood of that trajectory (point) [18]. For HDBSCAN [15], only *MinPts* is required to be specified a priori. The full details of the algorithm can be found in its seminal work [18] and revisions [14].

F. Site to Trajectory Mapping

To facilitate the exploration of corridors and answer RQ2, it would be helpful for experts to know which sites are used for landing in the identified corridors. We hypothesize and subsequently evaluate if a flight landing at a given site can be inferred from the chosen dataset if the altitude or ground speed of the aircraft falls below a threshold. Specifically, for each track point of a given flight, we evaluate if the track point can be considered as an intent to land. For each of these track points, the closest site location is then verified using the Vincenty inverse formula for ellipsoids [19]. Note that because multiple track points of the same flight may be in close proximity, they may collectively map to the same closest site. To the unique set of sites obtained in this manner, we consider them the landing sites of the corridor. In the Results section, we assess the altitude and ground speed separately and combined.

[1] https://web.archive.org/web/20200303055712/http://www.faa.gov/airports/airport_safety/airportdata_5010/

[2] https://adip.faa.gov/agis/public/

[3] https://cran.r-project.org/web/packages/dbscan/index.html

G. Dynamic Reconstruction of Trajectories

To evaluate if identified corridors' trajectories overlap over the full length of the corridor in order to answer RQ4, we used moveVis [20]. The R library moveVis provides tools to visualize movement data (e.g. from GPS tracking) and temporal changes of environmental data (e.g. from remote sensing) by creating video animations[4]. By first identifying the corridors, and their associated trajectories, we can recreate the corridors separately to inspect their formation.

IV. Results

We begin the results discussion by motivating our choice of pre-processing and model parameters in Figure 2 and Figure 3. As we would expect and as shown in Figure 1, trajectories come in different sizes (i.e. number of track points), due to different flight duration. From figure Figure 2, we can see that in our single day dataset (n. trajectories = 593), the vast majority of flights contained 250 track points or less. To construct our feature vector in order to group the trajectories, we chose to use trajectories with at least 50 or more track points. The intent is that we preserve as many flights as possible, while still providing a sufficient number of features (track points) to be grouped for similarity. This choice of vector size is also consistent with [11], and resulted in a final dataset of 414 trajectories. As noted in our method section, and similar to related work, larger trajectories are downsampled to match a size of 50 track points (and hence 50 longitude features and 50 latitude features).

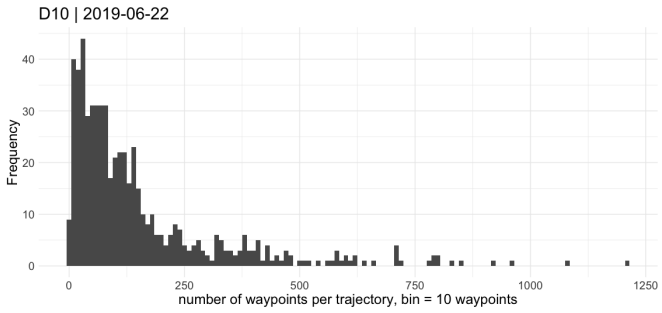

Fig. 2. Number of track points per trajectory.

While trajectories may be of similar sizes, their sampling rate may vary. The average and standard deviation cumulative distribution of the trajectories' sampling rate is shown in Figure 3. We can see nearly 100% of the trajectories in our one day dataset have an average sampling rate of 25 seconds or less; however, 25 seconds only accounts for nearly 75% or the trajectories standard deviation, with an upper bound of 200 seconds. Based on these statistics, we can conclude our dataset suffers from irregularity of sampling interval. As we noted in the related work section, a common approach to address this issue is the use of interpolation; however, this is not without its shortcomings due to generating artificial track points. Therefore, we decided to keep the data "as-is" to assess if both DBSCAN and HDBSCAN could still identify corridors with minimal pre-processing.

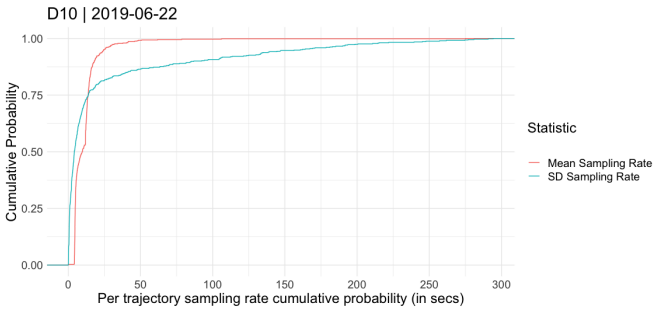

Fig. 3. Sampling Rate Statistics.

> RQ1: Can we automatically identify existing corridors using trajectory clustering?

When testing DBSCAN over varying parameters of *MinPts* and *Eps*, we observed all trajectories were consistently mapped to a single cluster. We therefore dedicate the remainder of the discussion to HDBSCAN.

When using HDBSCAN, we must decide on the choice of the *MinPts* hyperparameter. Intuitively, we are modifying the minimum number of trajectories within close planar proximity which we wish to consider a cluster (and thus a candidate corridor). Figure 4 shows how the number of identified trajectory clusters varies as we increase *MinPts*. In the interest of space, we selected three representative values of the *MinPts* hyperparameter to explore our first research question. Figure 5, Figure 6, and Figure 7 display the results for 2, 5, and 10 clusters respectively. As the minimum number of trajectories requirement increases, the number of clusters declines. Moreover, from the figures, we can see the identified clusters are consistent. Therefore, we chose to examine for the remainder of this work *MinPts* = 5, *N. clusters* = 10. Regardless, the answer to our first research question is yes, we can identify potential corridors already in daily operation, using HDBSCAN on a single day's worth of trajectories.

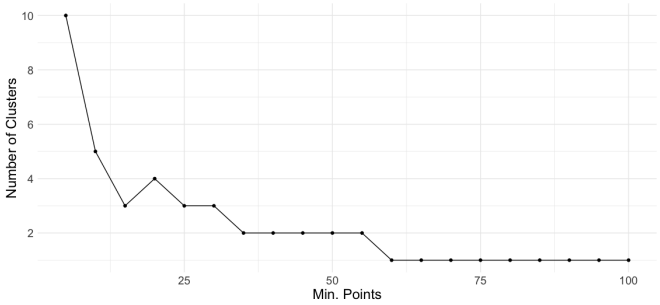

Fig. 4. HDBSCAN Number of clusters vs *MinPts*.

[4]http://movevis.org/

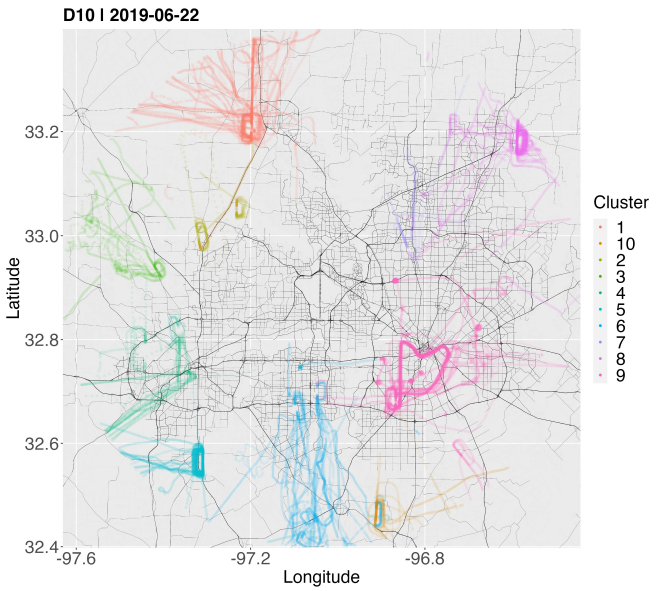

Fig. 5. HDBSCAN - *MinPts* = 5, *N. clusters* = 10.

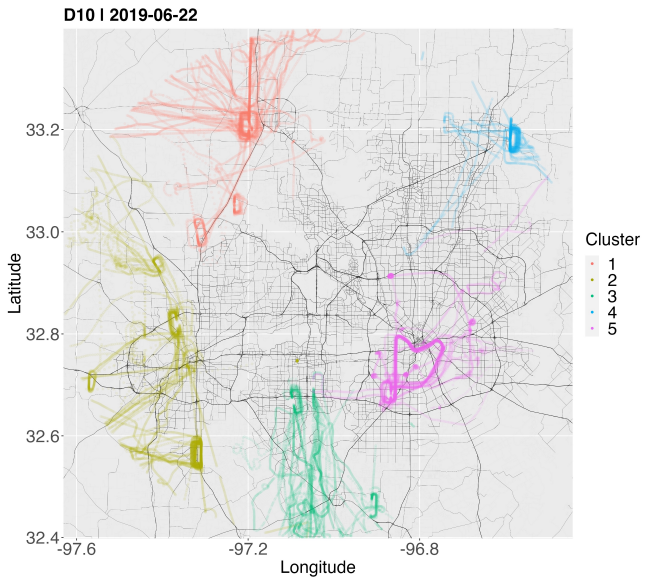

Fig. 6. HDBSCAN - *MinPts* = 10, *N. clusters* = 5.

> RQ2: Can corridors be automatically annotated for domain expert sense making?

We experimented separately with different thresholds of altitude and with ground speed to determine which track points to classify as intention to land. These track points were used in order to map the 'landing' track points to the nearest FAA landing site. The full set of track points available can be seen in Figure 1. Figure 8 shows our choice for ground speed threshold and Figure 9 for altitude threshold, which provide two different subsets of the track points of Figure 1. We can see both thresholds filter out some of the sites in

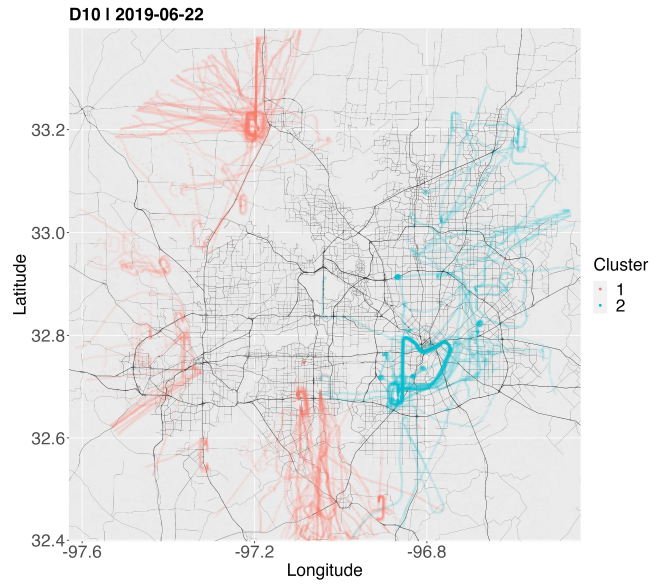

Fig. 7. HDBSCAN - *MinPts* = *35*, *N. clusters* = 2.

Figure 1. Moreover, a small amount of sites do not overlap when we consider low altitude or ground speed track points as landing intent. We speculate this difference may be due to sites in which sightseeing may happen in the air over the landing site, or simply the threshold choice leading to false positives. Regardless, both variables, separately and or in combination to others can be adjusted in this manner for corridor interpretation.

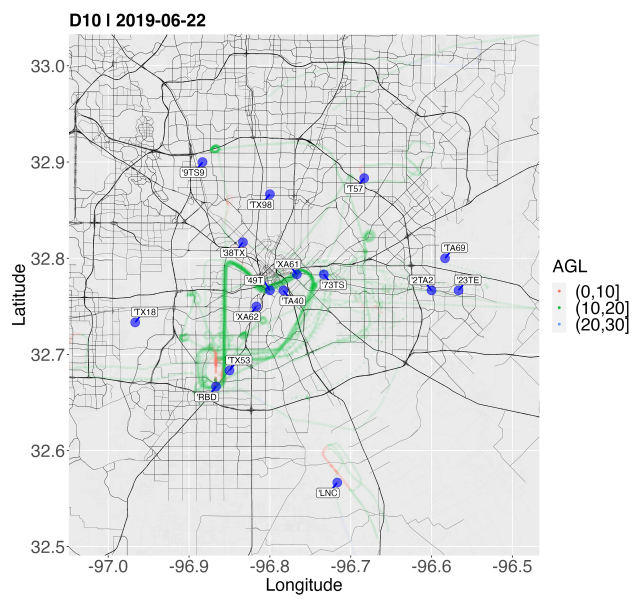

Fig. 8. Automatically annotated sites based on planar proximity. Blue dots represent FAA landing sites and green track points had a speed threshold of lower than 30 knots.

With the identified sites, we can further look into their metadata for corridor sense-making. Table I displays the sites identified in Figure 8. As shown, the corridor display con-

TABLE I
CLOSEST SITES USING GROUND SPEED THRESHOLD

	Location ID	Facility Name	Owner
1	’66R	ROBERT R WELLS JR	COLORADO COUNTY
2	’XA61	BAYLOR UNIVERSITY MEDICAL CENTER DALLAS	BAYLOR HEALTH CARE SYSTEM
3	’TX98	CRESPI HELISTOP	BRAD OELLETMAN
4	’49T	DALLAS CBD VERTIPORT	CITY OF DALLAS
5	’TA40	DALLAS CITY HALL	CITY OF DALLAS
6	’RBD	DALLAS EXEC	CITY OF DALLAS
7	’73TS	FIRE DEPARTMENT TRAINING CENTER	CITY OF DALLAS
8	’XA62	METHODIST DALLAS MEDICAL CENTER	METHODIST DALLAS MEDICAL CENTER
9	’TX53	POLICE H PORT-REDBIRD	CITY OF DALLAS
10	’TX18	REDMOND TAYLOR AHP	TEXAS ARMY NATIONAL GUARD
11	’9TS9	TOYOTA OF DALLAS INC	WAYNE STEPHENS
12	’38TX	UT SOUTHWESTERN MEDICAL CENTER	UNIV OF TEXAS SW MEDICAL CENTER
13	’T57	GARLAND/DFW HELOPLEX	CITY OF GARLAND
14	’TA88	PREMIER AVIATION INC	PREMIER AVIATION, INC.
15	’LNC	LANCASTER RGNL	CITY OF LANCASTER
16	’2TA2	THE MEDICAL CENTER OF MESQUITE	PRIME HEALTH CARE SERVICES-MESQUITE
17	’5TA0	HAMILTON AIRCRAFT, INC	DONALD P. KUBECKA
18	’TA69	LUPTON FARMS	LUPTON FAMILY CAPITAL CORP
19	’23TE	TEXAS RGNL MEDICAL CENTER	ROCKWALL HOSPITAL CORP
20	’TT00	TREE TOP AIR	WAYNE THURMAN

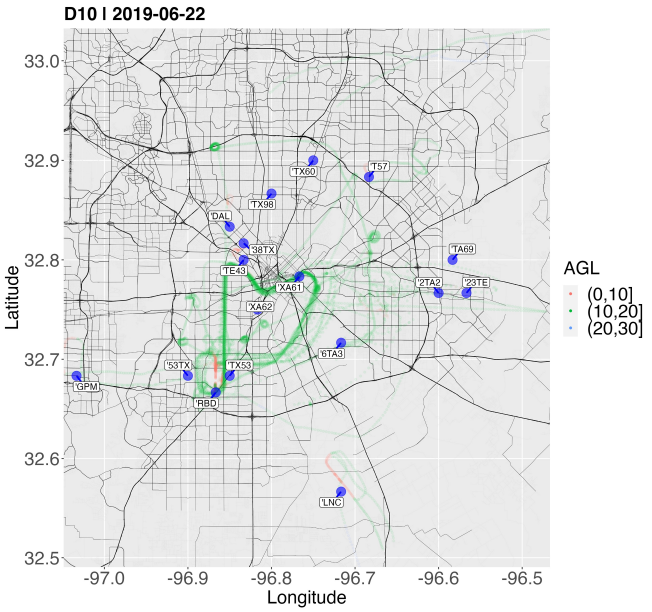

Fig. 9. Automatically annotated sites based on planar proximity. Blue dots represent FAA landing sites and green track points had an a altitude threshold of lower than 1000 feet.

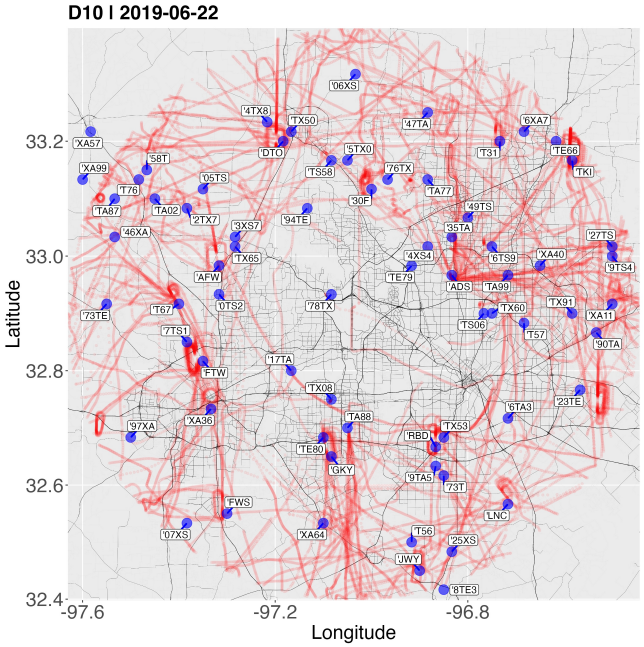

Fig. 10. Annotated Outliers.

tains corriders that are used for both tourism and emergency services.

For completeness, we also present the set of outliers with annotated landing sites in Figure 10. We observed that the general pattern of outliers is similar to Gariel et al. [11, i.e. Fig 14], in the sense that they cover track points across all identified clusters. More generally, the outlier clusters appear to capture trajectories that bridge two or more clusters in DFW.

RQ3: Can the density of corridors be measured?

When comparing the identified clusters by HDBSCAN in Figure 7 to the motivating example in Figure 1 in our introduction, we can see we are now able to mathematically distinguish close and distant trajectories in an automated manner. Because each trajectory is assigned a cluster identifier, we can define a candidate corridor density simply as the cluster size, i.e. the number of trajectories assigned to each cluster.

Table II shows the relative assignment numbers of the

TABLE II
CLUSTER ID TRAJECTORIES

Cluster ID	N. Trajectories	Proportion
0	217	52.4%
1	34	8.2%
2	7	1.6%
3	10	2.4%
4	15	3.6%
5	15	3.6%
6	25	6 %
7	5	1.2 %
8	20	4.8 %
9	60	14.4 %
10	6	1.4 %

trajectories to each of the 10 clusters. Cluster 0, which is not shown in the prior figures, aggregates outlier trajectories. In Figure 7, for example, 60 flights create the heart-shaped trajectory displayed on the bottom right of the figure out of the 404 analyzed flights in that day. Therefore, the answer to our research question 3 is yes, we can measure the density of corridors over a day. Further research is needed to refine these density metrics so that they are of value for evaluating efficiency and safety.

RQ4: Do flights that participate in the same corridor share the full trajectory, or only parts of it?

To answer RQ4, we created a visualization that dynamically reconstructs a given cluster output by HDBSCAN. The reconstruction draws one trajectory assigned to the cluster at a time in a different color, with a thicker line indicating the direction of the flight. A white trace is left behind by each colored trajectory, which over time reconstructs a given cluster identified in Figure 7. Due to medium limitations and space, we present here three frames of this visualization for cluster 9 (the cluster with the distinctive heart shape): The first (Figure 11), second (Figure 12) and final two trajectory frames (Figure 13) are displayed. We see clearly that the automatically-identified corridors consist of both a.) flights that participate in only part of the corridor (Figure 11 and Figure 12), and b.) flights that trace the corridor from beginning to end (Figure 13).

V. CONCLUSION AND FUTURE WORK

In this work, we demonstrated a method to identify corridors already in use, and augmented the data with visualization for sense making. Identifying the corridors allows for the study of their properties, such as quantifying their density on a given day, and also allows an analyst to reconstruct each trajectory dynamically to verify if the corridor is being used differently by different aircraft.

One limitation of this work is the use of only a single day's worth of data. In future work, we intend to extend the framework to perform a more comprehensive analysis of UAM corridors over time and cross-site. We also intend to assess the use of other metadata, such as weather and traffic congestion, to better understand the motivation behind the existing corridors.

As acknowledged in our related work section, different pre-processing, feature engineering, and clustering algorithm schemes (such as dynamic time warping, and utilizing the hierarchical relationship of the identified clusters) may provide further insight, and are will be explored in future work. Finally, extending the 2D visualizations of corridors into 3D and the study of their properties and overlap may lead to further insight.

ACKNOWLEDGMENTS

This research was partially conducted at NASA Ames Research Center. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not constitute or imply its endorsement by the United States Government.

REFERENCES

[1] A. Bauranov and J. Rakas, "Designing airspace for urban air mobility: A review of concepts and approaches," *Progress in Aerospace Sciences*, vol. 125, p. 100726, 2021.
[2] M. M. Eshow, M. Lui, and S. Ranjan, "Architecture and capabilities of a data warehouse for atm research," in *2014 IEEE/AIAA 33rd Digital Avionics Systems Conference (DASC)*, pp. 1E3–1–1E3–14, 2014.
[3] A. Nikoleris and H. Erzberger, *Autonomous System for Air Traffic Control in Terminal Airspace*.
[4] A. Nikoleris, H. Erzberger, R. Paielli, and Y.-C. Chu, *Performance of an Automated System for Control of Traffic in Terminal Airspace*.
[5] C. Bosson and T. A. Lauderdale, *Simulation Evaluations of an Autonomous Urban Air Mobility Network Management and Separation Service*.
[6] M. C. R. Murca, *Data-Driven Modeling Air Traffic Flows for Advanced Air Traffic Management*. PhD thesis, Massachusetts Institute of Technology, 2018.
[7] M. C. R. Mura, "Identification and prediction of urban airspace availability for emerging air mobility operations," *Transportation Research Part C: Emerging Technologies*, vol. 131, p. 103274, 2021.
[8] A. Churchill and M. Bloem, *Hierarchical Clustering of Aircraft Surface Trajectories*. 2020.
[9] M. Bloem and A. Churchill, *Filtering of Aircraft Surface Trajectories*.
[10] M. Li and M. Ryerson, "Novel terminal arrival airspace robustness metrics via topological density clustering," in *2018 International Conference on Research in Air Transportation*, 06 2018.
[11] M. Gariel, A. N. Srivastava, and E. Feron, "Trajectory clustering and an application to airspace monitoring," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 4, pp. 1511–1524, 2011.
[12] K. Leiden and S. Atkins, *Trajectory Clustering for Metroplex Operations*.
[13] M. C. R. Murca, R. DeLaura, R. J. Hansman, R. Jordan, T. Reynolds, and H. Balakrishnan, *Trajectory Clustering and Classification for Characterization of Air Traffic Flows*.
[14] E. Schubert, J. Sander, M. Ester, H. P. Kriegel, and X. Xu, "Dbscan revisited, revisited: Why and how you should (still) use dbscan," vol. 42, jul 2017.
[15] R. J. G. B. Campello, D. Moulavi, A. Zimek, and J. Sander, "Hierarchical density estimates for data clustering, visualization, and outlier detection," *ACM Trans. Knowl. Discov. Data*, vol. 10, jul 2015.
[16] J. Taylor, X. Zhou, N. M. Rouphail, and R. J. Porter, "Method for investigating intradriver heterogeneity using vehicle trajectory data: A Dynamic Time Warping approach," *Transportation Research Part B: Methodological*, vol. 73, no. C, pp. 59–80, 2015.
[17] M. Hahsler, M. Piekenbrock, and D. Doran, "dbscan: Fast density-based clustering with r," *Journal of Statistical Software*, vol. 91, no. 1, p. 130, 2019.
[18] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "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*, KDD'96, p. 226231, AAAI Press, 1996.

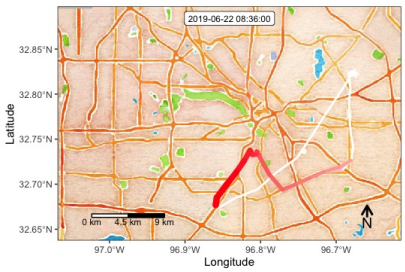

Fig. 11. First trajectory reconstruction.

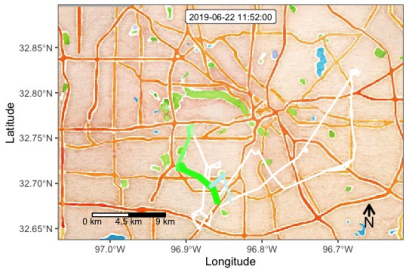

Fig. 12. Second trajectory reconstruction (after first).

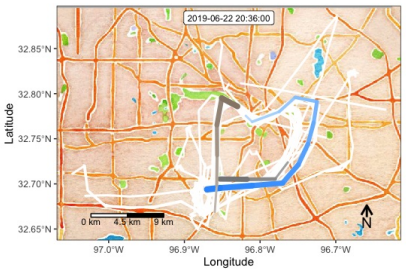

Fig. 13. Last two trajectories reconstruction.

Fig. 14. Three snapshots for the dynamic reconstruction of cluster 9 (heart-shape) trajectories. A white line traces the path of each trajectory after being drawn.

[19] T. Vincenty, "Direct and inverse solutions of geodesics on the ellipsoid with application of nested equations," *Survey Review*, vol. 23, no. 176, pp. 88–93, 1975.

[20] J. Schwalb-Willmann, R. Remelgado, K. Safi, and M. Wegmann, "movevis: Animating movement trajectories in synchronicity with static or temporally dynamic environmental data in r," *Methods in Ecology and Evolution*, vol. 11, no. 5, pp. 664–669, 2020.