A Map-Matching Algorithm for Ground Movement Trajectory Representation using A-SMGCS Data

Thanh-Nam Tran*, Duc-Thinh Pham†, Sameer Alam‡
Air Traffic Management Research Institute, School of Mechanical and Aerospace Engineering
Nanyang Technological University, Singapore
Email: {*thanhnam.tran | †dtpham | ‡sameeralam}@ntu.edu.sg

***Abstract*—Increasing availability of air traffic data has opened new opportunities for better understanding of Air Traffic Management (ATM) system. At Airport-Air side, A-SMGCS (Advanced Surface Movement Guidance & Control System) data may provide useful insights to improve efficiency and safety of airport operations by understanding traffic patterns, taxi-way usage, ground speed profiles and any anomaly behaviour. However, A-SMGCS data comes from the fusion of several sensors such as MLAT, ADS-B and SMR. This leads to high and variable noise, missing data values, and temporal and spatial misalignment. In this study, we proposed a new and simplified representation of ground movement trajectories using a map-matching algorithm applied on A-SMGCS data. The proposed approach not only overcomes above mentioned issues of data, but also takes into consideration airport specific operational constraints. The algorithm shows a good matching results with mean percentage error of approximate 8.13% . The matching trajectories and sequences of nodes in resulting graph, supports a variety of analysis about airport operations. To show the effectiveness of proposed approach, we performed some analysis such as traffic patterns, taxi-way usages, speed profiling and anomaly detection, using one month of A-SMGCS data at Singapore Changi Airport.**

***Index Terms*—airport surface movement, map-matching, airport graph, data analytics, A-SMGCS Data**

I. Introduction

According to forecasts of the International Air Transport Association (IATA), the number of passengers transported by airlines will reach 8.2 billion in 2037, doubling compared to 4.1 billion in 2017 [1]. With the current infrastructure, the airports around the world are facing the capacity challenge, with the busiest airports observing prolonged congestion and delay problems [2]. To address this concern, apart from operational improvements, many airports are rapidly expanding their airside infrastructure with additional runways and taxiways being constructed. However, this leads to a complex airport-airside environment for aircraft to navigate and for Ground Movement Controllers to manage.

To address this concern, many airports have now installed Advanced-Surface Movement Guidance and Control System (A-SMGCS) which comprises of a combination of systems that assist Air Traffic Tower Controllers to maintain safe movement of aircraft on airport airside. This system comprise of several sub-systems, such as Multi-Lateration System (MLAT), Automatic Dependant Surveillance Broadcast (ADS-B) and Surface Movement Radar (SMR), which provides surveillance data. Therefore the surface movement data coming from A-SMGCS can be of high variance, noisy and even misaligned in space and time. [3] points out the three main challenges for working directly with A-SMGCS data: a huge volume of data, noise in observation, and uneven sampling. The noise in surface movement data occurs because of the inherent errors in the environment (large metal bodies of aircraft in close proximity), tracking system (MLAT uses triangulation) and mathematical models that requires approximations. It affects the accuracy of the tracking position and causes the missing data problem. Thus, standardizing data to find a good representation for ground movement trajectory is necessary in order to make best use of A-SMGCS data, specially for research purposes.

Although, surface movement data shares the similar standardizing problem with flight trajectory data. Most of the work in the literature focused on flight trajectories, due to the increasing and public availability of track data (ADS-B). Whereas A-SMGCS data is still prerogative of Air Navigation Service Providers leading to limited research on analyzing surface movement data. Some recent research proposed clustering approach [4] [5] [6], [3] to learn simplified representation of flight trajectories using two-level clustering algorithm. Although clustering approaches are quite successful in flight trajectory representation, the surface movement trajectories have three main characteristics comparing to flight trajectories, that makes it challenging:

- Big variance in taxing speed, between aircraft, since the aircraft can stop/accelerate/slowdown during taxiing.
- Significant noise in the data (data jump) due to close proximity of aircraft on the taxiway system.
- Aircraft need to follow the physical taxi-ways structure with complex crossings (leads to ghosting) and Taxiway-Design Group (TDG) restrictions (interpolation do not work) for a given airport.

Those characteristics make surface movement data similar to GPS data that track the position of the vehicle on a 2D surface with a fixed physical network road. Although there are differences in system, based on those nature of the operation and data, analyzing surface trajectories can be inspired by similar problems from urban transportation where vehicles'

GPS trajectories are collected and analyzed. From the urban transportation view, map-matching algorithms often are the first step to project trajectories onto a road network before employing any methods like clustering algorithm [7]. The advantage of standardizing data by matching coordinate data to the traffic graph is that it can be used to measuring road speeds and building statistical models like predicting traffic delays [8]. Map matching is long-standing research problem with different algorithms [9] [10] [11], each algorithm has different strengths and weaknesses [12] depending on context and characteristic of data. From our limited view, there is few studies that extends the map matching problem for surface movement data at an airport-airside. [13] apply the rule-based method to identify two kinds of taxi route segments (straight and turn). However, it is only a simple pre-processing to support analyzing surface movement trajectory. [14] recently proposed the point-to-point map-matching algorithm as a standardize module in the surface movement pipeline to predict the taxi-time. Nevertheless, their algorithm has limited accuracy in large data set and cannot account for noise in the trajectory data.

In this study, we propose a map-matching algorithm[1] specific for airport surface movement data to construct a simplified representation of aircraft ground movement trajectories. The proposed map-matching algorithm can deal with noises and missing of surface movement data on dense segment network. Moreover, the framework is highly flexible and the proposed algorithm can easily be integrated with operational constraints to account for operational complexity of a given airport. We also conduct data analysis on graphs to demonstrate the benefits of graphical representation of trajectory. We show the application of the clustering method on the graph to analyze usage patterns for taxiways, taxi-routes, and detecting anomalous trajectories in different aspects. The average moving speed on each taxiway segment is an important information identified from the graph which may help identify potential risks, such as taxiway incursions.

II. Data for study

A. Dataset

9 features
Gate, Longitude, Latitude, Velocity in Longitude, Velocity in Latitude, Time of Track, Measured flight level, Type of Aircraft, Wake Turbulence Categorisation

TABLE I: The list of data features.

In this study, we have used the Advanced Surface Movement Guidance and Control System (A-SMGCS) [15] data. The dataset comprise of one month for the period of October 2017, collected at Singapore Changi Airport. The dataset contains about 24000 trajectories (aircraft movements). A-SMGCS data contains information about movements of aircraft, their trajectories and flight plan. For the movement part, every data point records the calculated position with the corresponding velocity of aircraft in the airport in WGS-84 coordinates and Cartesian coordinates. The original data are compressed in EDR format that needs extracting and preprocessing before performing analysis. The features of the final dataset are listed in TABLE I

[1]Source code available at: https://github.com/strnam/airport_mapmatching

B. Graph of the Airport Network

The graph of the airport network in this study is constructed from public OpenStreeet map data [16] using OSMNX package [17] that can apply the same method to get the graph network of any airport in the world. We use all physical infrastructure with tag aeroway but filter out the tags: aerodrome, apron and terminal. The node and edge of the graph correspond to the line marking in the taxiway. The graph is simplified by eliminating all nodes that are not intersections or dead-ends [17]. Because the graph is constructed automatically, there is some error at the intersection that are fixed manually. The resulting Changi Airport graph contains 518 nodes and 1686 edges (segments), the mean length of edges is 91.34 meters, minimum length is 3.678 meter, and the maximum length 990.14 meters.

C. Data Challenges

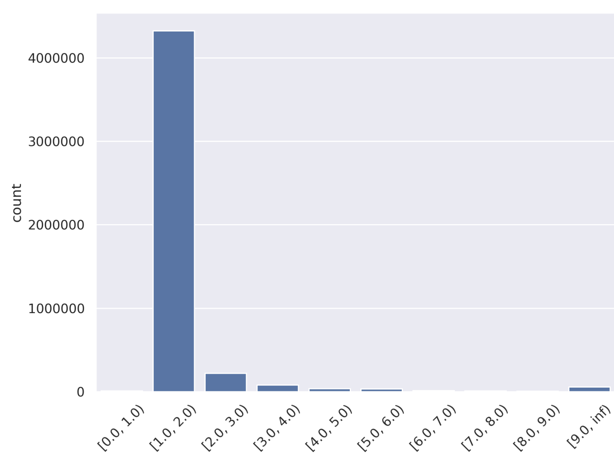


Fig. 1: Sampling rate distribution

1) High sampling rate position data in the dense segments network: Fig. 1 shows the sampling rate distribution in our dataset. The default sampling rate of the system is 1 second per tracked data point, but in the case of noise, the sampling rate can be bigger than 9 seconds. From the urban traffic view, the sampling rate of surface movement data is extremely high compared with the high-sampling rate standard in urban traffic that one point often records in every 10-30 second [18]. When the aircraft is moving with low speed, the high sampling rate causes the high density of tracked points in a specific area. Those areas often are parking slots and intersection that also have dense edges, and it makes the map-matching algorithm, especially for the segment-based algorithm, very hard to detect the right segment.

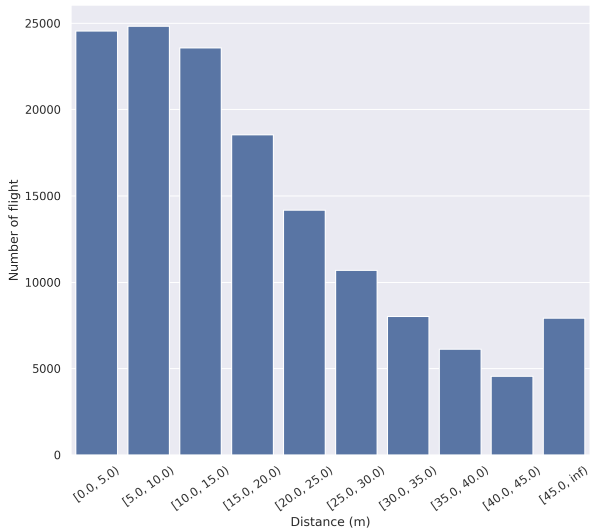


Fig. 2: Analyze the number of flight contains the periods of distance difference

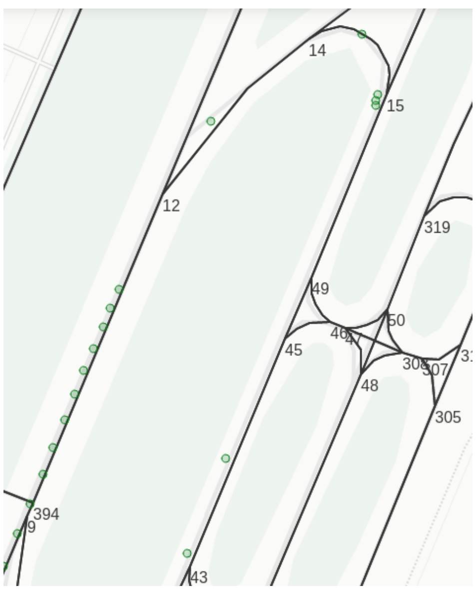


Fig. 3: The case of high distance difference

2) High distance difference of position data in the dense segments network: Although the A-SMGCS system is designed for tracking the position of the aircraft every second, we found that it is not stable. There is the missing data and the big variation of speed of aircraft which can lead to a significantly large difference in distance between two consecutive points. The number of flight which contains the periods of distance difference is given in Fig. 2 (except the points tracked on runways), it shows that more than 10000 flights contain the gap between two consecutive points are from 25 to 30 meters, and about 7000 flights contain the distance difference bigger than 45 meters. With the dense segments of the airport graph network, the big distance difference can cause the missing connectivity of the segments in the trajectory. The bigger the distance difference, the harder in filling the missing segment information. As in Fig. 3, it displays the case the distance gap between two consecutive points suddenly changes from short distance to long distance, and it misses several segments between (segment (45, 49)).

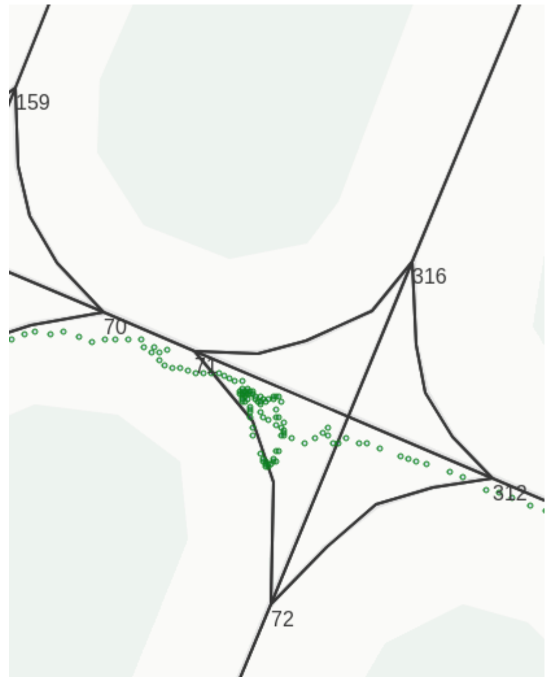


Fig. 4: Dense position data in intersection

Fig. 5: Noise in position data

3) The noise in data: We found that the noise of data often occurs when the movement speed of aircraft is low, especially in the intersections (Fig. 4) and the parking slots. As in Fig. 5, we can see the aircraft move abnormally from edge (390, 153) to the parking position, then edge (303, 313). The noise in data is the most challenge of the map matching algorithm.

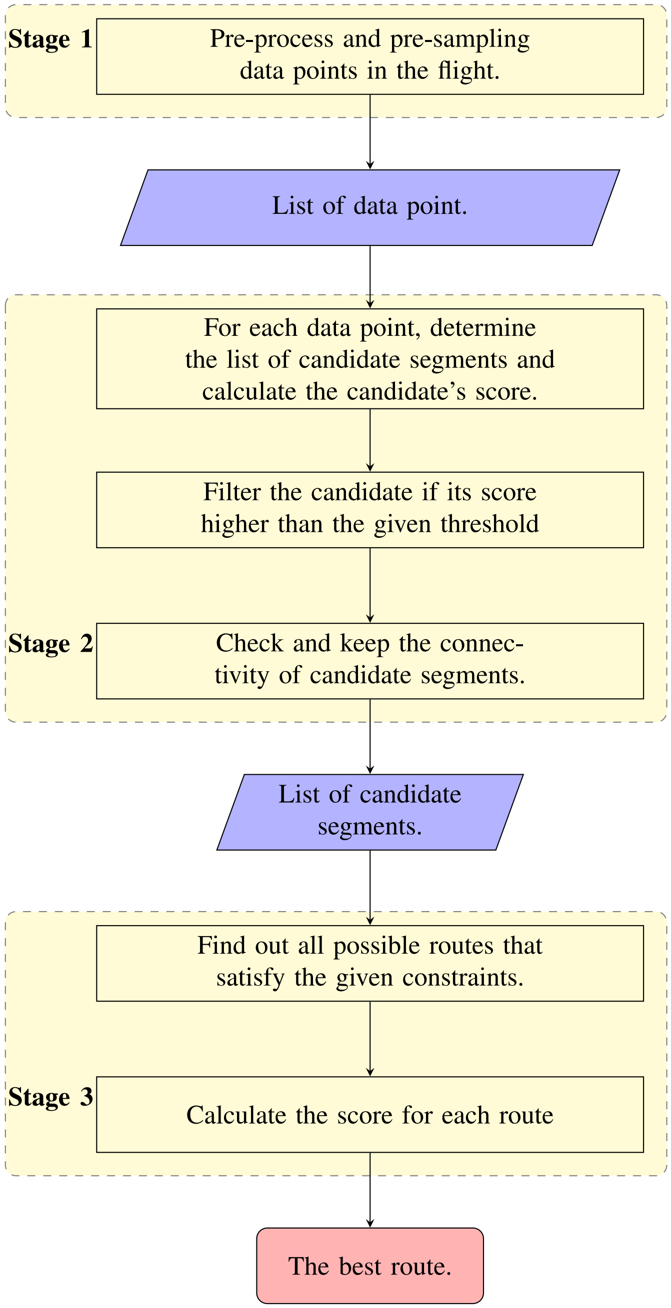


Fig. 6: The flowchart of proposed map matching algorithm

III. Proposed methods

A. Map-matching

1) Proposed map-matching algorithm: The proposed map-matching algorithm work on flight data in an offline situation that means from the beginning, the algorithm can access the whole sequence of tracked points where each point contains the coordinates and speed of the aircraft at a specific time.

The proposed method has three stages, as shown in Fig. 6. In the first stage, we re-sample the coordination (latitude and longitude) of tracked points by the fixed distance between two consecutive points. This re-sampling step is very useful in reducing the density of tracking points in the dense area.

In stage 2, for each tracked data point, we determine the list of candidate segments by getting all the segments that contain at least one of three closest nodes to the current point. Motivate by the work of [11] we define the score of each candidate from three criteria: the distance of current tracked point to candidate segment, the direction different of the candidate and the last two tracked points, the direction different of the candidate and the vector of current and the next tracked point — the detail formula is shown in formula 1. The smaller the score, the better it is. The candidate with the score higher than the given threshold is filtered out of list candidates. We also compute the confidence for each tracked point. If one tracked point has a confidence level below the confidence threshold, it is considered as noise and be filtered out to flight data. As mentioned above, when the distance difference is huge in the dense segments network, the candidate segments of one point often lack the connectivity with the candidate segments of the previous point. To solve this problem, we find all combinations of segments from the candidate list of two consecutive points. If they are not connected, we fill the gap by the shortest path between them and add the new segments to the candidate list.

$$\begin{aligned} Score_i = {} & W_{dist}\frac{\text{distance to segment}_i}{\Sigma_{j=1}^{N}\text{distance to segment}_j} \\ & + W_{\text{prev dir}}\frac{\text{direction difference from segment}_i}{\Sigma_{j=1}^{N}\text{direction difference from segment}_j} \\ & + W_{\text{next dir}}\frac{\text{direction difference from segment}_i}{\Sigma_{j=1}^{N}\text{direction difference from segment}_j} \end{aligned} \tag{1}$$

Where:

$$W_{dist} = \text{Constant}$$

$$W_{\text{prev dir}} = \begin{cases} 1, & \text{if travel distance} \geq \text{Given threshold} \\ \frac{\text{travel distance}}{\text{distance threshold}}, & \text{otherwise} \end{cases}$$

In stage 3, we list out all possible routes that can go through the list of candidate segments we get from stage 2. In practical, the airport has many operation constraints that can be used to narrow down the list of the valid route. Different kinds of vehicles can have different constraints. By separating the flow of the algorithm into different stages, our proposed algorithm can easily impose a different set of constraints based on operation requirements. In this study, we focus on mapping the movement of aircraft at the airport only. Several aircraft movement's constraints are implemented, such as the angle between two consecutive segments must be greater than 90 degrees and do not have any node be traveled more than one time. Finally, the score for each route is calculated by the total number of the tracked point assigned to that route, the assignment count when the best segment of each tracked point appear in the route. Opposite to the score for tracked points, the bigger the route score, the better it is. The route with the highest score and lowest number of nodes is chosen as the final map matching result.

B. Matching time and Speed

The time and speed are essential information to analyze the operation of the airport. Thus, we not only apply map-matching algorithms to find the best route for given tracked points but also matching the right time and speed when the aircraft is moving on every segment. The input of matching time and speed module is the sequence of tracked points, and the route from the map matching method described above. In the first step, we projected every tracked point in the flight to the trajectory on the graph. In the second step, we resample the new points by a fixed distance and match the node begin and end of each segment to the nearest points. The travel time of one segment is approximating by the time of the nodes begin and end of this segment. From this information, we can calculate the duration and speed the aircraft move on each segment. The final results are recorded in the trajectory information.

C. Map-matching results

We use the point-to-point map-matching algorithm proposed in [14] as the baseline model. We preprocess flight data by filtering out the tracked points that have extremely low or extremely high velocity. The dataset in this experiment contains 23090 flights.

To access the performance of the map matching algorithm, we use percentage error between the distance of the flight and the distance of the matching route. Where the distance of flight is calculated approximately by the sum of the distance of every tracked point and the distance of trajectory is the sum of all edge lengths. We also tried using the Frechet distance that is commonly used to measure the distance between two curves. However, from our experiment, the Frechet distance is not a useful metric in this case. The wrong segment assignment of the small segments, especially in the intersection, is a severe error, but it just makes a slightly different in Frechet distance value. While the mismatch between the last point of the flight and the last node of the trajectory (in the right segment) causes a big different in Frechet distance value, but it often does not affect the quality of the map-matching algorithm.

The Fig. 7 show map matching percentage error of baseline model and proposed models. The performance of the proposed model is significantly better than the baseline model. The mean percentage error on the overall dataset of the proposed model is 8.13% compare with 15.84% of the baseline mode.

In some cases, the baseline algorithm can deal well with noisy data because, as a point-to-point algorithm, it only considers the points near the nodes of the graph, so it is not affected by noise in other areas. However, in the case of tremendous noise in data, as shown in Fig. 8, the baseline algorithm still matches incorrectly. Meanwhile, the proposed algorithm by combining with the constraint of movement of the aircraft has eliminated unreasonably trajectory, and it returns the correct result.

In the high distance cases, as Fig. 9, the baseline algorithm is often insufficient to accurately match the coordinate point to the nearest node on the graph, resulting in incomplete trajectory. Instead of point-to-point matching, the proposed algorithm matches the point to the segment so it can overcome this drawback.

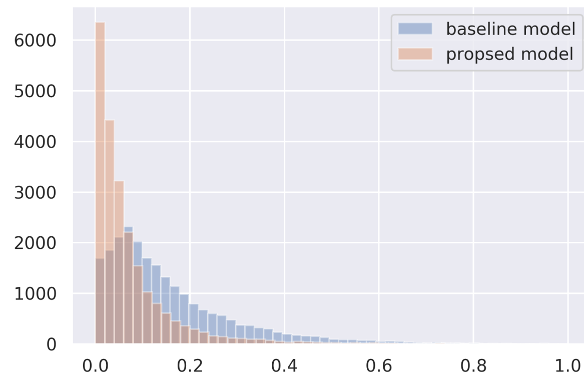


Fig. 7: Percentage error distribution of baseline model and proposed model


(a) Baseline model (b) Proposed model

Fig. 8: Map-matching results in the noise case

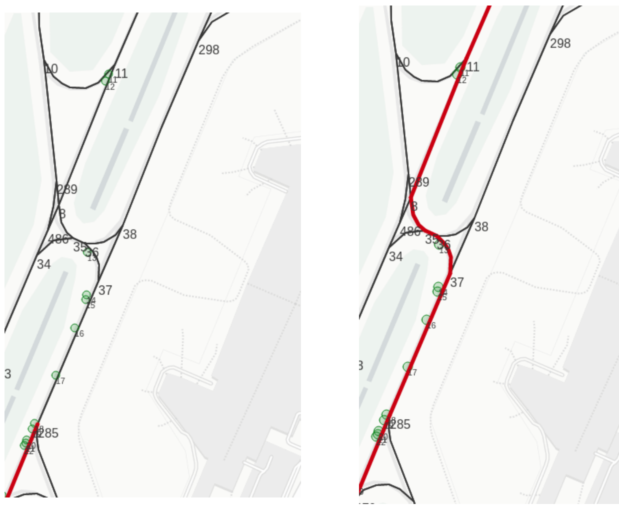
IV. Data Analysis Using Constructed Graph

A. Patterns extraction using clustering on graph

The standard method to analyze the trajectory is to cluster them to similar groups [7]. After the map matching process, we can represent the trajectory as a sequence of nodes that is suitable for applying many different clustering methods.

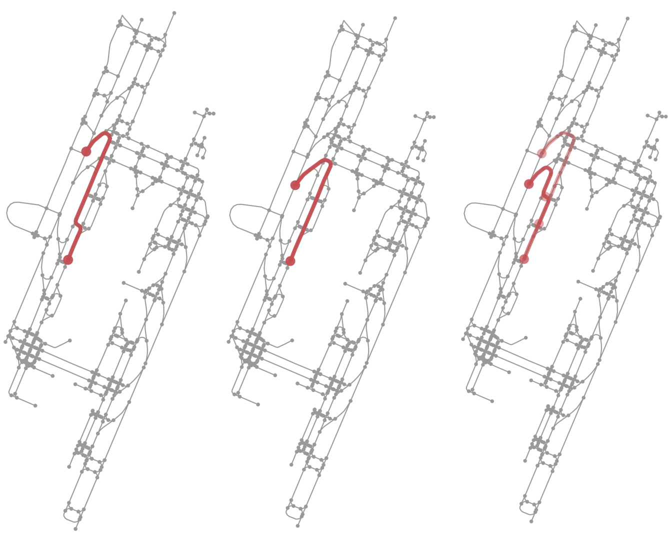
1) Controller's options on O-D pairs: Similar to the work of [14], we can extract the pattern in assigning the taxiway of controllers by applying the clustering method on the trajectory group by 0-D pairs. By applying DBSCAN (Density-Based Spatial Clustering of Applications with Noise) [19] algorithm, we can extract the patterns of trajectory and identify the noise case in on pair of O-D data. The Fig. 10 show the clustering result of the arrival flights from runway 02L/20R to gate B3. It detects the two groups of trajectories 10a and 10b that share the same pattern and one group 10c that consider as the rare trajectories cause by noise or uncertainty event.

This approach help in automatic detection of anomalous trajectories from historical data. It can also work in real-time to alert the anomalous movement of aircraft or ground vehicles



(a) Baseline model (b) Proposed model

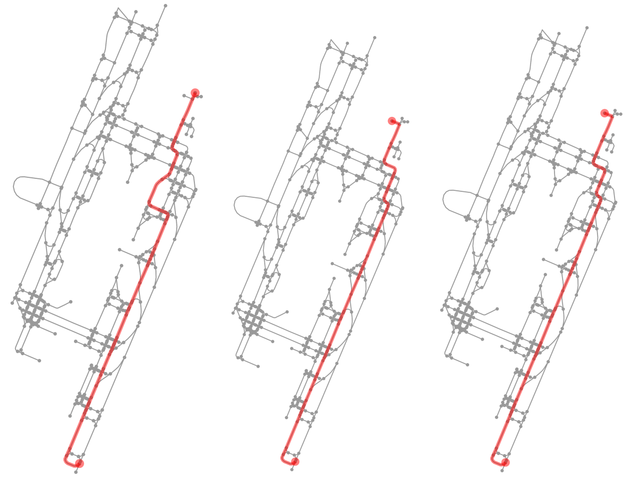
Fig. 9: Map-matching results in the high distance difference case



(a) Pattern 1 (b) Pattern 2 (c) Noise

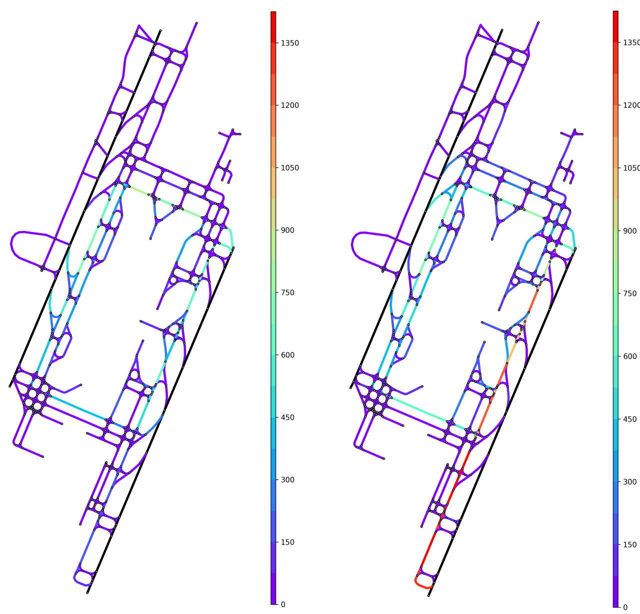
Fig. 10: Clustering result of arrival flights from runway 02L/20R to gate B3

to support the controller in situation awareness for maintaining airport-airside safety. Fig. 11 show the case the aircraft moves from the East cargo apron gate 604 to the runway C02 where clustering model extracts two normal patterns using taxiway NC1 or NC2 go to taxiway A7 then go to EP. The anomalous trajectory is detected as using taxiway A6 instead of A7.



(a) Anomalous trajectory (b) Normal trajectory (c) Normal trajectory

Fig. 11: Comparison between anomalous trajectory and the historical patterns

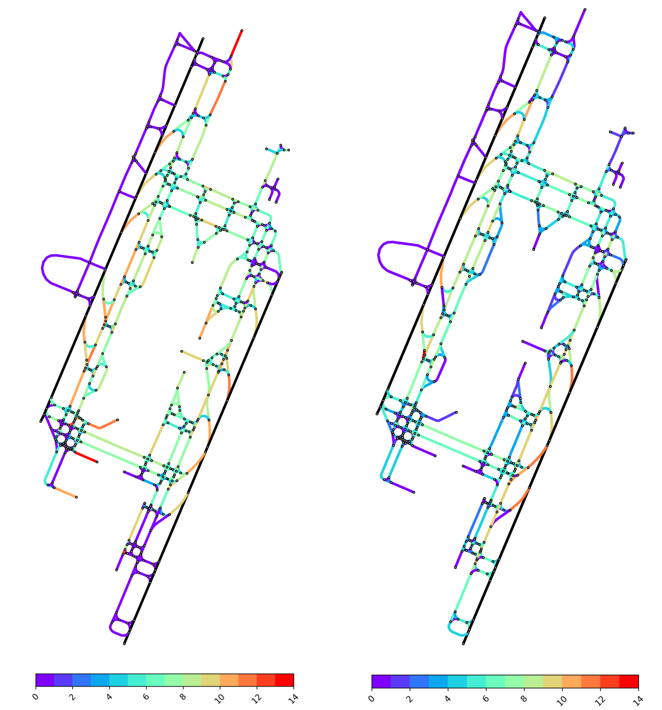


(a) Traffic from 5 am to 8 am (b) Traffic from 10 pm to 1 am

Fig. 12: Airport traffic at different time

B. Airport traffic

Representing the trajectory on the graph makes retrieving information about airport traffic straightforward. As an example, the Fig. 12 shows the traffic of Singapore Changi airport at off-peak hours from 5 am to 8 am and the peak hours from 10 pm to 1 am. The color of each edge identifies by the number of flight travel over that edge. The significant difference of Singapore Changi Airport between the normal demand state and high demand state is the usage of EP taxilane, while EP taxilane is used rarely in off-peak hours, it is the most used taxilane in peak hours. Such information may help the controller to better plan the ground movement/Taxing operation, especially at large airports with complex taxiway intersections.



(a) Edge Speed of arrival flights (b) Edge Speed of departure flights

Fig. 13: Aircraft Speed heatmap

C. Aircraft Ground Movement Speed Profile

Aircraft ground movement speed can only be synthesized by representing the trajectory on the graph. Speed information can be used to support training predictive models for predicting aircraft speed during taxiing. Fig. 13b and Fig. 13a show the average speed of arrival flights and departure flight. In general, the average speed of arrival flights is higher than departure flights, especially in the taxiways near the parking slots. It is understandable, because arrival flights have momentum from landing phases while departure flights often move slowly at the taxiway close to parking slots. From the aircraft speed heat map, we can identify which taxiways are high speed like E6, E4, E6,... It is useful for analyzing the characteristics of the airport taxi network at each link and intersections for identifying potential risks.

In Fig. 14, we combine the speed information with the clustering result to show the speed profile for one cluster. The flights with the same taxi route have similar speed on the sequence of segments. It can be a useful hypothesis for detecting abnormal movement of the flight when they move too fast or too slow compared to the historical speed of the flights within the same group.

V. Conclusion

In this study, we have proposed the simplified representation of surface trajectories as a sequence of nodes from airport network graph. A map-matching algorithm is proposed for

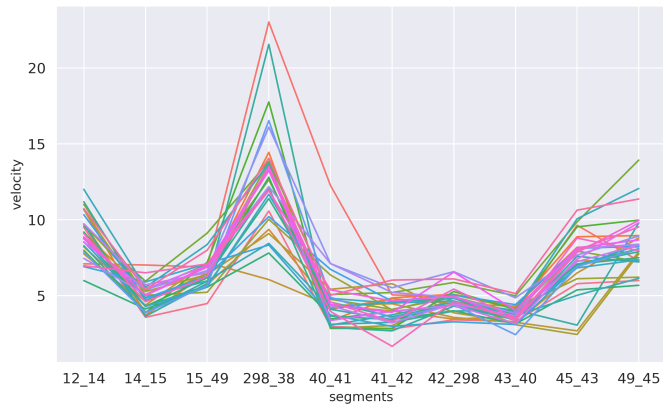


Fig. 14: The speed profile of the flights in the same group.

surface movement data to convert trajectory data from xy-coordinates to sequence of nodes. The algorithm is tailored for airport surface movement data by considering its unique characteristics and airport structure. The algorithm can deal with the problem of noise and missing data in high-density segments area at airports. It achieves 8.15% mean percentage error which is far improved compared to the similar algorithm [14] (15.85%) on the similar dataset. Our simplified representation opens new perspectives for analysing surface movement data such as understanding ground movement controller's taxi-route planning patterns, detecting anomalous trajectories or taxi-ways usages, etc. For the next step, we will extend this approach for other moving vehicles at the airport.

VI. Acknowledgement

This research / project* is supported by the Civil Aviation Authority of Singapore and Nanyang Technological University, Singapore under their collaboration in the Air Traffic Management Research Institute. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of the Civil Aviation Authority of Singapore."

References

[1] I. A. T. Association, "Press release no.: 62." [Online]. Available: https://www.iata.org/pressroom/pr/Pages/2018-10-24-02.aspx

[2] K. G. Zografos, M. A. Madas, and K. N. Androutsopoulos, "Increasing airport capacity utilisation through optimum slot scheduling: review of current developments and identification of future needs," *Journal of Scheduling*, vol. 20, pp. 3–24, 2017.

[3] A. M. Churchill and M. Bloem, "Clustering aircraft trajectories on the airport surface," 2019.

[4] M. Z. Li and M. S. Ryerson, "Novel terminal arrival airspace robustness metrics via topological density clustering," in *ICRAT2018 conference, Barcelona, Spain*, 2018.

[5] 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.

[6] M. C. R. Murça and R. J. Hansman, "Identification, characterization, and prediction of traffic flow patterns in multi-airport systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 5, pp. 1683–1696, 2018.

[7] Y. Zheng, "Trajectory data mining: an overview," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 6, no. 3, p. 29, 2015.

[8] P. Newson and J. Krumm, "Hidden markov map matching through noise and sparseness," in *Proceedings of the 17th ACM SIGSPATIAL international conference on advances in geographic information systems*. ACM, 2009, pp. 336–343.

[9] D. Bernstein, A. Kornhauser *et al.*, "An introduction to map matching for personal navigation assistants," 1996.

[10] N. Tradišauskas, J. Juhl, H. Lahrmann, and C. S. Jensen, "Map matching for intelligent speed adaptation," *IET Intelligent Transport Systems*, vol. 3, no. 1, pp. 57–66, 2009.

[11] M. Hashemi and H. A. Karimi, "A weight-based map-matching algorithm for vehicle navigation in complex urban networks," *Journal of Intelligent Transportation Systems*, vol. 20, no. 6, pp. 573–590, 2016.

[12] M. A. Quddus, W. Y. Ochieng, and R. B. Noland, "Current map-matching algorithms for transport applications: State-of-the art and future research directions," *Transportation research part c: Emerging technologies*, vol. 15, no. 5, pp. 312–328, 2007.

[13] C. Gong, "Kinematic airport surface trajectory model development," in *9th AIAA Aviation Technology, Integration, and Operations Conference (ATIO) and Aircraft Noise and Emissions Reduction Symposium (ANERS)*, 2009, p. 7076.

[14] D. Pham, M. Ngo, N. Tran, S. Alam, and V. Duong, "A data-driven approach for taxi-time prediction: a case study of singapore changi airport," in *ENRI International Workshop on ATM/CNS*, 2019.

[15] O. de l'aviation civile internationale, *Advanced Surface Movement Guidance and Control Systems (A-SMGCS) Manual*. ICAO, 2004.

[16] OpenStreetMap contributors, "Planet dump retrieved from https://planet.osm.org ," https://www.openstreetmap.org , 2017.

[17] G. Boeing, "Osmnx: New methods for acquiring, constructing, analyzing, and visualizing complex street networks," *Computers, Environment and Urban Systems*, vol. 65, pp. 126–139, 2017.

[18] Y. Lou, C. Zhang, Y. Zheng, X. Xie, W. Wang, and Y. Huang, "Map-matching for low-sampling-rate gps trajectories," in *Proceedings of the 17th ACM SIGSPATIAL international conference on advances in geographic information systems*. ACM, 2009, pp. 352–361.

[19] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise." AAAI Press, 1996, pp. 226–231.