Clustering and Complexity Estimation for Air Traffic Flow Management

Joana Costa
joana.p.costa@tecnico.ulisboa.pt

Instituto Superior Técnico, Universidade de Lisboa, Portugal

January 2021

Abstract

Air traffic control is usually considered one of the most demanding and stressful jobs in the world, as they are responsible for the safety of thousands of passengers. To support their tasks, several computer-aided systems have been developed, helping them to identify typical aircraft routes they have to control, and measuring air traffic congestion and complexity. Such tools help them to anticipate the aircraft's trajectory and alert them if there is a conflict along its path or provide more time to process the viable solutions.

In this context, this thesis focuses on identifying and clustering typical aircraft routes and on computing the air space complexity. To achieve these objectives, a new method is proposed to evaluate and compare different clustering techniques applied to traffic datasets, by sorting them according to the best clustering result. On the other hand, it is also proposed a new method to estimate the controller's workload based on traffic data and airspace volume configuration. This last method can also suggest the best configuration for a time interval of interest.

The conducted experimental evaluation showed that the OPTICS clustering algorithm, allied with a preprocessing phase based on a 2-components PCA, is the best combination at clustering the traffic dataset. It was also possible, to define a capable tool to identify the corresponding clusters to a trajectory in real-time, without having to re-process considerable amounts of data. In what concerns the controller's workload, the obtained results showed to accurately estimate and predict the airspace complexity, allowing to anticipate a sector configuration change for the time intervals of interest.
Keywords: Air Traffic Flow Management (ATFM), Clustering, Typical Flows, Complexity Indicators, CAPAN parameters, Air Traffic Controller Workload, Sector Configuration.

1. Introduction

Air traffic control (ATC) is a crucial and unquestionable pillar in aviation, to ensure and maintain airspace safety and control.

Considering that most of European countries are moving to free route airspace (FRA) giving aviation companies more freedom to plan their flight routes, it is now important to identify typical flows and routes to improve the efficiency of Air Traffic Controllers (ATCO). This efficiency is directly related with capacity optimisation through sector management.

The main objective of this research is to first find typical flows, by computing trajectory clusters. Then, use those clusters to calculate sector capacity and complexity considering the controller working experience.

Different density based clustering methods with and without pre-processing mechanisms are analysed through qualitative parameters, improved and compared.

The improvement is performed over merging clusters which present to be very similar in heuristic computation.

The search for the best technique is done by comparing the different techniques in a variety of quality indicators, addressing a value for each (technique, indicator) given how well they perform in comparison to the others. Afterwards, results are correlated into one unique performance metric to rank all the clustering methods.

This work also focus on delivering an accurate complexity estimator. This same relies on ATCOs workload, through the computation of a set of tasks defined to describe any situation the controller may experience. This set of tasks were defined by EUROCONTROL [1], its time averages were identified from studies performed over many European Countries (in 2015 more than half of the European Air Control Centers (ACCs) have performed at least one study). The goal at this research is to create an algorithm that estimates complexity in three to six hours in advance to inform ATCOs about hotspots and the FMP in order to take more informed decisions on sector configurations to apply in the next hours.

2. Background

2.1. Air Traffic Complexity

Complexity is a topic that has been mentioned and studied since 1960. Schmidt [2] in 1978 studied the qualitative relations between workload (strain, fatigue) and the performance of the men.

Hurst and Rose [3] were the first in measuring the correlation of expert workload ratings with traffic density having come up with a correlation value of 53%.

Stein [4] used Air Traffic Workload Input Technique (ATWIT) in which controllers report workload levels during simulation. After a regression analysis they were able to explain 67% of the variance. Four factors were considered: localised traffic density, number of handoffs outbound, total amount of traffic, number of handoffs inbound.

2.1.1 Relationship Complexity-Workload

Over the years, a variety of researches showed a strong relationship between complexity metrics and controller workload. Hurst and Rose [3] measured the correlation of expert workload ratings with traffic density with a result of 53% of correlation. Stein [4] used Air Traffic Workload Input Technique(ATWIT) for four metrics (localised traffic density, number of handoffs outbound, total amount of traffic, number of handoffs inbound) having a regression correlation of 67%. Laudeman [5] introduces Dynamic Density as a combination of "both traffic density and traffic complexity".

Mogford et al. [6] reviewed a number of studies examining effects of ATC complexity on workload and performance. They created a model relating "source factors" and mediating factors resulting in controller workload, Fig. 1.

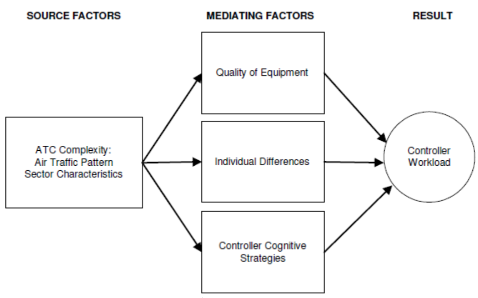

Figure 1: Relationship ATC Complexity and controller workload [7]

2.1.2 CAPAN tasks

Complexity Estimation depends on the sector configuration, weather and 4D air traffic, airport problem, restricted airspaces and air traffic controllers. At this analyses an approach based on controllers workload is proposed. The CAPacity ANalyser (CAPAN) parameters are 38 tasks, very detailed and specific, defined by EUROCONTROL who tries to table all kinds of tasks the controllers need to perform [8]. This table also tells the amount of time each controller spends on each task, given that he/she is EC or PC. Usually, sector are controlled by two controller one occupying the Executive Controller (EC) position and the other working as the Planning Controller (PC).

2.2. Clustering - define the typical Routes

There are five main clustering methods which told by order of importance: partitioning method, hierarchical method, fuzzy clustering, density based clustering, model based clustering. Nonetheless, at this analyses the focus stays in density-based and hierarchical clustering mechanisms.

Table 1 shows the main advantages and disadvantages of some clustering methods. k-means (partitioning method) is a well-known clustering technique, reliable a still widely used. However, it will be tested on the experimental results because the density-based and hierarchical algorithms pose the great benefit of not having to define apriori the number of clusters required to identify.

Pre-processing algorithms, applied before clustering, may improve final results. Principal component analysis (PCA) [20, 21] is a well-known pre-processing technique, but focused on global similarity, PCA does not work well on non-linear state spaces. To tackle those challenges t-distributed stochastic neighbour embedding (t-SNE) [22, 23] a more recent algorithm, focused on keeping the local similarities can be better than PCA.

Gallego et al. [14] introduce and applied two evaluation methods to compare four clustering techniques, those are the Silhouette Width Criterion (SWC), proposed by Rousseeuw [24], and the Density Based Clustering Validation (DBCV) metric, presented by Moulavi [25].

Having acknowledged CAPAN tasks as useful elements to estimate complexity and talked about some clustering methods, pre-processings, which could improve results, and evaluation techniques applied at clustering, in next chapter a new evaluation and comparison mechanism is proposed as well as the explanation of how the CAPAN parameters were inserted in an algorithm that by looking at traffic data can define air traffic controller workload at a given time resolution.

Table 1: Comparison of density based clustering methods.

Clustering Method	Advantages	Disadvantages
k-means [9, 10]	Easy, simple computation.	Non-generic, it requires that the user has some knowledge of the airspace. Does not detect outliers.
DBSCAN [11, 12]	Generic, does not demand any previous knowledge allowing it to be applied in any airspace. Detects outliers.	Does not adapt to the data since it has a fixed epsilon.
R-DBSCAN [13, 14]	Generic, adapts to the dataset (low density, high density), detects outliers.	Computational Time.
HDBSCAN [15, 14]	Only Demands hyper-parameter (MinClusterSize). Outlier detection. Generic.	Computational Time.
OPTICS [14, 16, 17]	Demands hyper-parameter MinClustSize. Outlier detection. Generic.	Computational Time.
Automatic Hierarchical clustering [18, 19]	No need to provide MinClustSize.	Suggests an outlier removal method before clustering.

3. Proposed System

At Figure 2 is the big picture of the the proposed system.

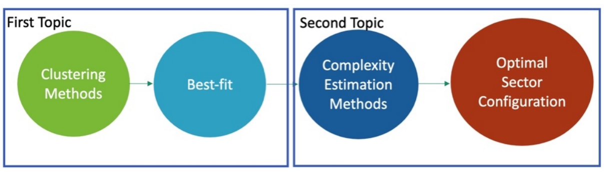

Figure 2: The general view of the proposed system

A selection of clustering methods is tested and the clusters of the technique whose results are the best are saved. At the best-fit algorithm it is possible to see to which new cluster a trajectory fits or a set of flights.

Having the traffic which we are interested in analysing assigned to their respective clusters, it is possible to predict the complexity with a time resolution within the traffic situation.

With the complexity estimation an an application is suggested, the automatic selection of the optimal sector configuration.

3.1. Main flows determination - clustering

Figure 3 refers to the main steps of the clustering algorithm.

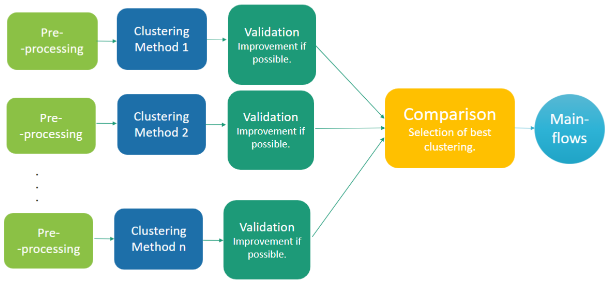

Figure 3: Scheme Clustering Implementation

The creation of a new clustering technique was not in the scope of this analyses, but the study and possible enhancement of already existing clustering methods is. The system here propose has the ultimate objective of validation, enhancement, determining certain identified quality indicators and lastly compare and define the best clustering results. It is the combination of machine learning outcomes with deterministic benchmarking.

Some of the most mentioned clustering techniques, on articles in air traffic sector [13, 26, 15, 27, 14, 28] are here tested and the objective is to create an air traffic data focused technique to qualify, evaluate and compare their results. The input is traffic data:

- flight_id, callsign, icao24
- grounspeed
- altitude, latitude and longitude
- track
- vertical rate

with positions updated at time rate 10 seconds.

First, there is the integration or not of pre-processing PCA with number of components equal to 2, 4 and 6 or t-SNE [29] algorithm with dimension of 2. These pre-processing changes the original data to a smaller dimension state-space.

Then, any of clustering technique from: DBSCAN [30], R-DBSCAN [13], R-DBSCAN*, a transformation of the R-DBSCAN which only accepted cluster above k, it was was changed to accept cluster with $\geq$ k, HDBSCAN [27] and OPTICS [30].

The outputs are the assignment of each trajectories to one cluster or as belonging to an atypical route and the centroids of each cluster. The centroid of each cluster is the trajectory which best suits to be in the center.

3.1.1 Validity

There are two types of validity intra- and inter-cluster, these two base on the following validation methods:

- Lateral validation
  - distance similarity
  - heading similarity
- Vertical validation
  - vertical evolution similarity

For a cluster to be considered intra-cluster valid 70% of the trajectories must verify the 2D lateral distance validity, the heading validity and also the vertical evolution validity, i.e., the three constraints must be verified, for the cluster to be considered *intra-cluster*.

By means of simplification from this page on, valid cluster means a cluster that verified intra-cluster validity.

For a cluster to be considered inter-cluster valid, it is enough that only one of the three constraints is not complied.

With these validation parameter, quality indicators can be taken:

1. *number of cluster* (NC);
2. *trajectory assigned to clusters* (TAC);
3. *trajectories assigned to valid cluster* (TAVC);
4. *clusters intra- valid(%)* (CIV);
5. *Mean flights per valid cluster* (MF_VC);
6. *standard deviation number of flights over valid cluster* (SDF_VC);
7. *percentage of inter-clusters valid(%)* (PIV);
8. *mean of vertical validity among valid clusters(%)* (MVS_VC);
9. *mean of lateral validity among valid clusters (%)* (MLS_VC);
10. *mean of vertical validity among non-valid clusters (%)* (MVS_NVC);
11. *mean of lateral validity among non-valid clusters (%)* (MLS_NVC).

3.1.2 Enhancement Merge

An enhancement is introduced at the methods which changes also the quality indicators result. The hypothesis is raised of merging the clusters which tested to have any similarity, the new centroid is computed and the intra-cluster validity of the hypothetical new, larger one is computed. In case, the hypothetical cluster is intra-valid, the merge is accepted and the other clusters are eliminated.

3.1.3 Comparison

Each value (quality_indicator, clustering_technique) suffers a transformation which takes into account their rank in indicator i and how far from the mean value at quality indicator i (Equation 1). The transformed values at the same clustering mechanism are summed into one single value. Having a single value assign to each clustering technique, it is possible to sort the clustering techniques from the one with better qualitative results (lowest single value) to the worse one (Equation 2).

$$\text{comp\_value}_{i,j} = \text{rank}_{i,j} \pm \frac{\text{qualt\_mean}_i \; - \; \text{qualt\_value}_{i,j}}{100} \tag{1}$$

$$R_j = \Sigma_{i=1}^{N} result_{i,j} \tag{2}$$

3.2. Best-fit Cluster

This algorithm was created to be used on a flight that changed trajectory or to define the cluster each flight belongs to from a set that is not large enough to perform clustering.

Succinctly, the three steps of Best-Fit-Cluster:

- Perform Lateral Evaluation (2d lateral distance and heading difference) and Vertical Evaluation (vertical evolution) from one trajectory to all existing clusters;
- For each trajectory, there are three possible cases, the trajectory only belongs to one case:
  - **Case 1** the trajectory has *min_percentage_of_samples_valid* on all three constraints at one or, possibly, more clusters;
  - **Case 2** the trajectory only has *min_percentage_of_samples_valid* at the lateral constraints, but not at the vertical evaluation, at one or more clusters; it means the trajectory could not comply all the three constraints with not even one cluster;
  - **Case 3** the trajectory does not even comply Lateral Evaluation with any cluster;
- Case 3 is the only one which inserts the trajectories inside the anomalies designation, if the trajectory fits in Case 1 or 2 it will be assign to a cluster.

3.3. Optimal Sector Configuration

This study focus on the building of a reliable algorithm that through air traffic trajectories (past, present and near future) can determine (or predict) the complexity of an air traffic situation.

At Figure 4 is the principal scheme of the complexity estimation system's organization.

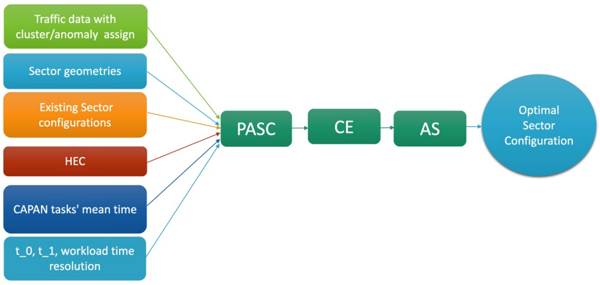

Figure 4: Proposed system for complexity estimation and application

There are a lot of inputs to be able to extract a valuable output. The traffic data of interest already assign to its cluster or as an anomaly, the existing sector configurations in the air control center (ACC) where it applies and the sectors' geometries and the time resolution to extract the complexity.

PASC means Pre-selection of the Airspace Sector Configuration, which with the analysis of the Hourly Entry Count (HEC) will decide the set of sector configurations feasible to perform a more extended analyses based on complexity. At the next stage, CE stand for Complexity Estimation where an algorithm performs the necessary complexity prediction calculations, and finally at AS, airspace sectorisation, an objective function decides which sector configuration is the best for the time interval of analysis.

3.3.1 Pre-selection

Usually the tabled HEC are compared with the actual HEC to identify hotspots. Figure 5 is a representation of the Peak and Sustain values, which in this example are constant, and the actual HEC values calculated every 20 minutes.

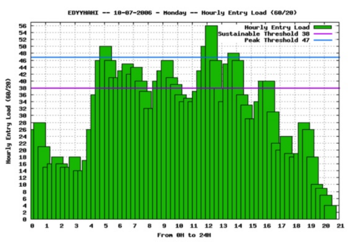

Figure 5: Graph demonstrating HEC between 0 to 24H, it seems from 21 to 24H there was not any traffic

3.3.2 Complexity Estimation

Complexity estimation is based on the CAPAN parameters, a list of tasks EUROCONTROL created [8] with their time means was used at this stage.

The tasks can be identified on two main groups:

- Tasks that only refer to an isolated aircraft, such as the entry, the exit, the report on reaching a specific flight level (FL), instruction to climb, etc.;
- Tasks that refer to the interaction of two flights; usually flight that get to a distance which demands levels of cautiousness or possibly intervention.

Figure 6 exemplifies different tasks that can happen inside a sector. Tasks related to flight x (flx) are not written at the Figure not to fill the picture with lot of information. All the tasks related to Flight y (fly)are written.

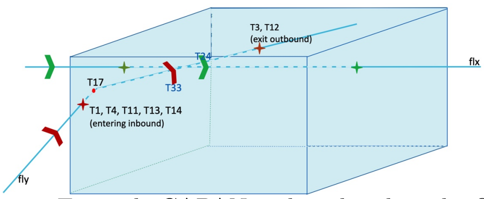

Figure 6: Example CAPAN tasks related to the flights according to where they come from, in which flight phase are they, if they change flight phase, the time they spend in sector, the trajectory and the existence of conflict.

When flight y comes in at climb from the inbound, 5 tasks are associated: receiving flight information (T1), receive time and level estimation from same ACC (T4), first call with aircraft that comes from same ACC (T11), conflict search to establish initial level clearance for flight entering the sector in climb or descent (T13) and, considering that the only flight in sector is flx, by the the time fly passes, conflict search to establish sector planning clearance (T14).

When fly changes from climb to cruise phase, task 17 happens, report on an aircraft (AC) on reaching a specified level has to be performed. Task 33 (radar intervention two aircraft on crossing tracks, both in cruise at same FL, needs to be called when the flights are that close, as it is drawn inside sector, their distance is bellow the minimum. Some time after, task 24 is called to supervise the two AC, at crossing tracks, same FL.

3.3.3 Airspace Sectorisation

Complexity is a function of time and tasks performed. When a task is confirmed to happen it is also assign to the time it happens. Just like a dictionary, where the keys are the time when tasks are suppose to happen, more precisely, the epoch [1]

[1] unix epoch started at 00:00:00 UTC 1^{st} January 1970

and the value, assigned to the key, is a cumulative sum of the time means associated to the tasks that happened at the same epoch.

Whenever the algorithm finds a task that is predicted to happen, according to a flight trajectory or a conflict, it also determines when it happens. Hence, the epoch to when it is predicted to happen is searched in the dictionary and, if there is a key with the same epoch, the time mean associated to the task is added to the value already there (a simple sum). Otherwise, a new key is defined, assigned to the time mean related to the task.

After having computed the complexity at the sector configurations, pre-selected at stage one, the airspace Sectorisation phase is designated to compute the *Complexity* at a time resolution between t_0 and t_1 as shown in Equation 3.

$$Complxty(\%) = \frac{\begin{bmatrix}\text{total workload time} \\ \text{by the sum of} \\ \text{tabled\_tasks\_mean\_time} \\ \text{occurred inside } [t_i,\ t_i + \\ time\_workload\_resolution]\end{bmatrix}}{time\_workload\_resolution} \times 100 \quad (3)$$

3.3.4 Objective function

Finally, the delta function is the cumulative sum of the excess of complexity given by to sectors of each sector configuration being tested.

The delta function is computed for configurations with the same number of active sectors. Then, the sector configuration with the smallest delta is saved in a short term memory, as the best_delta$_t$. After having the best_delta$_{n+1}$, which corresponds to opening of one more sector, the deltas are compared. If, Equation 4 is verified, the search stops and the sector configuration chosen is the one related to best_delta$_n$. Otherwise, the search continuous until the next best_delta does not considerably improve the results to justify the opening of one more sector.

$$best\_delta_n \leq best\_delta_{n+1} + 20\% \quad (4)$$

4. Experimental Results

This chapter presents the results of the conducted evaluation to validate the proposed *main-flows determination* and *complexity estimation* methods.

In order to provide meaningful results, an input dataset with a significant representation of real life traffic has to be used preferably, with typical challenges that air traffic controllers face everyday. Therefore, from the available datasets, the flights crossing the upper Switzerland airspace, from August 1^{st} 2018 was selected. This airspace is divided into two ACCs (Geneva ACC and Zurich ACC), which is relevant for computing the workload between different ACCs and analyse how it translates in terms of complexity.

4.1. Main Flows - Clustering

This section presents the testing results for the clustering techniques already discussed. Before the introduction of clustering the original traffic dataset (Figure 7) is difficult to understand and make some sense out of.

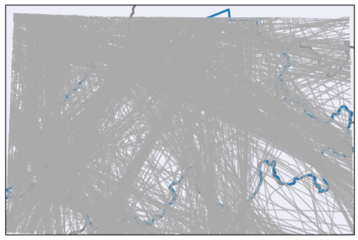

Figure 7: Daily Traffic 1^{st} August 2018

4.1.1 Evaluation and Comparison Techniques

With the five techniques and all the possible cases with and without the pre-processing methods PCA and t-SNE, there are 23 different clustering mechanism to compare. At Table 2 are the top 6 clustering methods.

Table 2: Results Comparing Techniques

Rank	Method	Qualification
1	OPTICS PCA 2	53.16
2	DBSCAN t-SNE	54.38
3	DBSCAN PCA 2	57.04
4	RDBSCAN* t-SNE	57.20
5	RDBSCAN t-SNE	58.12
6	RDBSCAN PCA 2	58.98

Looking at Table 2, it is concluded that the best technique is OPTICS with PCA 2. It was thought that DBSCAN would occupy the lower ranks at the Ranking Table 2 because of the *epsilon* static nature. Nevertheless, DBSCAN with t-SNE gets the second place at this Table and DBSCAN with PCA 2 gets the third.

There's no significantly differences between R-DBSCAN and R-DBSCAN*, which accepts smaller clusters. They are almost always hand in hand.

Hence, it is clear that the use of a pre-processing method improves their place on the rank. Many researchers discuss that PCA and t-SNE can bias the data, but, at least here, it created better results. However, it is not possible to say which method creates the best results, t-SNE and PCA with 2 components occupy the first places. Figure 8 synthesizes the main quality indicator results of the top 4 techniques.

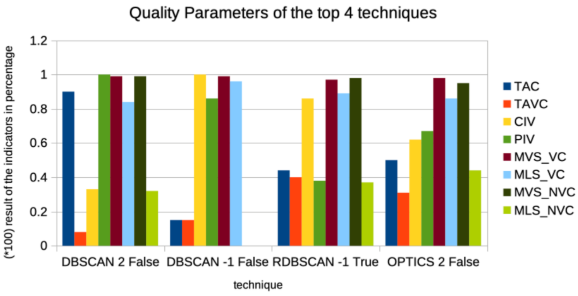

Figure 8: Results of most relevant quality indicators of the top 4 clustering techniques.

4.1.2 Results of Best-clustering Technique

Figure 9 presents the obtained cluster centroids at the best technique (OPTICS with PCA 2), and trajectories related to them. 24 clusters were identified, here the clusters were organized by six per graph to facilitate the understanding. The top six clusters are represented at the top left of Figure 9 with a number of trajectories per cluster between 65 and 32.

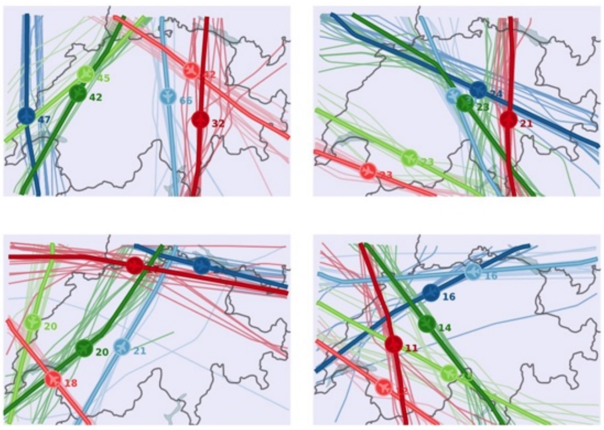

Figure 9: The resulting best clustering technique OPTICS PCA 2 represented using sectflow library [31].

The six main clusters (upper left) of Figure 9 show that there are at least two flows from South heading North in each corner of Switzerland upper airspace (the one more to the left may be heading France and the other may be heading Austria, Germany or Czech Republic). Another two are coming from Northeast (from German Airspace) to Southwest (heading South of France). There is also one, the most common one (with 66 flights), coming from North (maybe Germany, Austria or Poland) to South in direction to Italy or North Africa. Another identical cluster with 32 routes, is doing the opposite route heading North of Europe. At last, there is one in pink (42 flights) heading Hungary, Slovenia or Croatia. It is very interesting to find out how just one day of analysis finds such patterns.

4.1.3 Best-fit-cluster applied between 9a.m. and 10 a.m.

The considered methodology to analyse if a flight belongs to a cluster using the independent Best-fit Cluster approach, is deterministic, as explained at Section 3.2. As an example, flights taken between 9 a.m. and 10 a.m. are submitted to best-fit-cluster algorithm with the cluster centroids obtained from OPTICS with PCA 2 (the best qualified method at ranking 4.1.1). Figure 10 illustrates the traffic between 9 a.m. and 10 a.m. (128 trajectories) before running the best-fit-cluster, Figure 11 presents the results of the best-fit 88 trajectories (68.8% of the total) were tagged as typical routes. The most used routes during this time interval are at the top left, eight flights (pink route) are coming from South of France heading Germany, other route is doing exactly the opposite (light green), nine flights. Another common route does the trajectory from South to North France crossing Switzerland (dark blue). 12 flights are crossing Switzerland Northeast entering Austria's Airspace.

Figure 10: Traffic between 9 a.m. and 10 a.m. without tagging

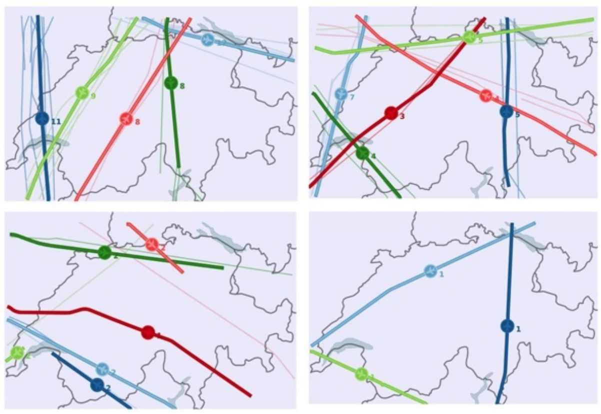

Figure 11: Typical routes identified at the traffic Figure 10 with the best-fit-cluster

This provides one example of the importance and potential of clustering the trajectories and finding the cluster that best-fits in it. Seeing the clusters that are going to be present in a specific time interval helps visualizing and understanding better the conflict and consequences before they take place, also it is very helpful for the FMP to choose the best sector configuration according to the main traffic.

4.2. Complexity Estimation

It was defined by EUROCONTROL [32] that the workload should not exceed 70%. A study of the best sector configuration for the time interval 5 a.m. to 6 a.m. and 9 a.m. to 10 a.m. is performed it was observed that the best sector configurations were U3K and U4J, respectively. There were still some peaks but some sectors were already as small as possible others did not presented a considerable improvement by opening one more sector. Here we present the evolution of complexity (as explained in section 3.3) as well as HEC with time, at resolution 3 minutes.

4.2.1 Results between 5 a.m. and 6 a.m.

The optimal sector configuration obtained for the time interval 5 a.m. to 6 a.m. is *U3K*. A simple representation of the sector configuration *U3K*, is provided at Figure 13. Sectors LSGL14C and LSGL67C are represented at Figure 12, they do not suffer any excess of complexity.

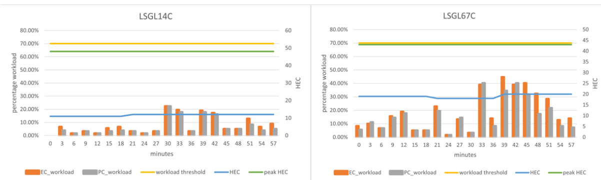

Figure 12: Sector complexity LSGL14C and LSGL67C for both EC and PC and HEC, between 5a.m. and 6 a.m.

At Figure 13 is the evolution of complexity and HEC at sector LSGL5C this sector has a hotspot at [05:21, 05:25[with workload 124.58% for the EC controller and 40.78% for the PC controller, the considerable discrepancy between the workload of the two controllers stress out that it is probably due to conflicts. The EC has the greater responsibility, therefore spending more time, on conflicting tasks than the PC controller. It is not possible to decrease or cease the hotspot by changing sector configuration because sector LSGL5C is already the smallest as it gets, the next best solution is to re-route or level-cap the flight that is causing a lot of conflicting tasks.

4.2.2 Results between 9 a.m. and 10 a.m.

At time interval 9 a.m. to 10 a.m. the suggested sector configuration by the algorithm is

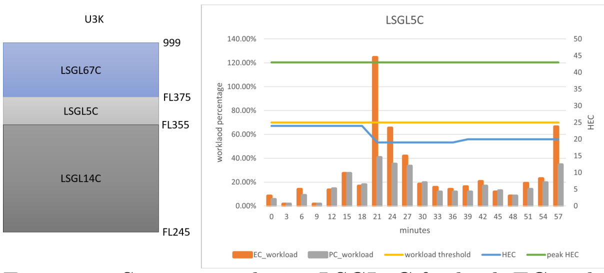

Figure 13: Sector complexity LSGL5C for both EC and PC and HEC, between 5 a.m. and 6 a.m. and well as simple representation of *U3K* sector configuration

U4J, it has four active sectors, LSGL14C (figure 14), LSGL5C (figure 15), LSGL6C (figure 16) and LSGL7C. LSGL7C is not here represented but it does not have any hotspot. LSGL14C has two

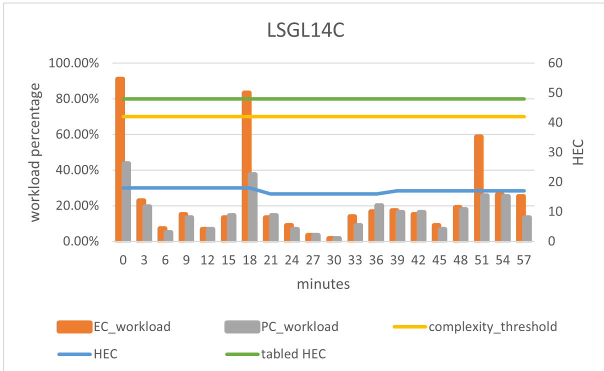

Figure 14: Sector complexity LSGL14C for both EC and PC and HEC

hotspots one at first interval and another at [09:18, 09:21[. LSGL5C has five hotspots, among which there is one particularly high (above 200%, at [09:39, 09:41[. LSGL6C has also five hotspots for the EC controller but the highest one is below 140%.

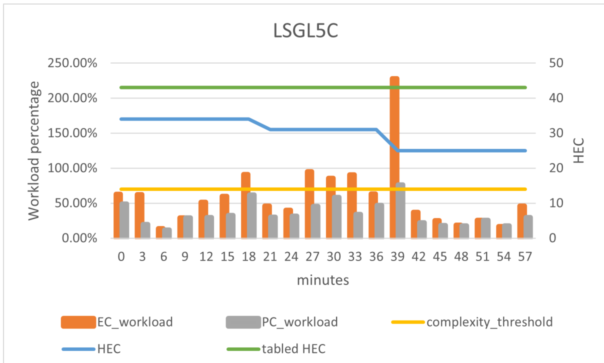

Figure 15: Sector complexity LSGL5C for both EC and PC and HEC

The high hotspot at LSGL5C is mainly due to a conflict of one flight with two other flights, at different times but very closely (between 09:39 and 09:40).

LSGL14C could be split in more sectors in order

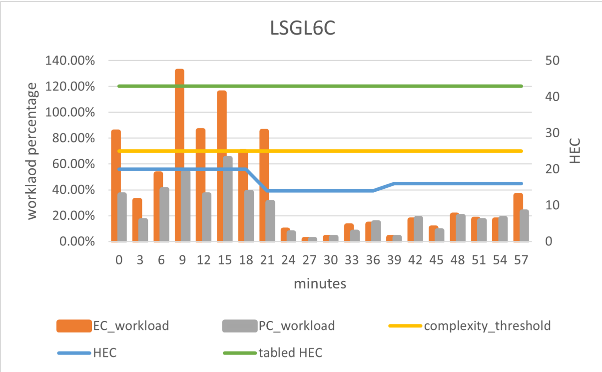

Figure 16: Sector complexity LSGL6C for both EC and PC and HEC

to avoid the two workload of EC controller above the *safety threshold*, but it was observed that the other sector did not have such considerable amounts of workload (the highest peak was at LSGL3C below 20% at 09:36 to 09:39). LSGL4C the smallest sector as it gets still has one peak at first time interval (09:00 to 09:03) at 90% even tough the peak at 09:18 to 09:21 decreases below 30%, however tha algorithm considers it is better to have the sectors joined into one and suffer an additional peak of 83% (below 100%).

The analysis is also performed per flight such that, when a flight is found causing many conflicts, it is easy to change its trajectory and compute again the sectors complexity. Usually it is always better to avoid complexity hotspots through optimal sector configuration, but when it still does not remove the excess of controllers workload, short-term ATFCM measures (STAM) are mostly adequate.

5. Conclusions

Pre-processing techniques did improve the clustering results, specially PCA with 2 components and t-SNE, with OPTICS with PCA 2 as the top clustering method. Also best-fit-cluster happened to be a technique that can rapidly and without needing as many data define which flight trajectories are typical routes and with are not.

Concerning, complexity estimation the algorithm is capable of predicting the workload of each controller just by analysing traffic data, it can also define the time intervals of highest demand to posteriorly study which flight is causing this increase in complexity.

In the future, it would be advisable to use the three constraints (lateral 2d distance, heading difference, vertical evolution) to disregard the trajectories that were wrongly tagged to a cluster by the machine learning mechanism. Another suggestion would be to take some complexity indicators and perform the correlation between them having as reference the complexity value determined at the algorithm here developed or to teach a neural network.

Acknowledgements

The author would like to thank Eng. Delfim Rego and Prof. Nuno Roma.

References

[1] T Chaboud, R Hunter, J Hustache, S Mahlich, and P Tullett. Investigating the air traffic complexity: Potential impacts on workload and costs. *Brussels, Belgium: Eurocontrol, NATS*, 2000.

[2] David K Schmidt. Queuing analysis of air traffic controller's workload. Technical report, 1978.

[3] Michael W Hurst and Robert M Rose. Objective job difficulty, behavioural response, and sector characteristics in air route traffic control centres. *Ergonomics*, 21(9):697–708, 1978.

[4] ES Stein. Air traffic controller workload: An examination of workload probe (report faa/cttn90/60). *Atlantic City, New Jersey: FAA*, 1985.

[5] Irene Vincie Laudeman, Stephen G Shelden, R Branstrom, and CL Brasil. Dynamic density: An air traffic management metric. 1998.

[6] Richard H Mogford, JA Guttman, SL Morrow, and P Kopardekar. The complexity construct in air traffic control: A review and synthesis of the literature. Technical report, CTA INC MCKEE CITY NJ, 1995.

[7] Brian Hilburn. Cognitive complexity in air traffic control: A literature review. *EEC note*, 4(04), 2004.

[8] Kim Schickel. Balancing sector volume workload by using air traffic controller click inputs. 2013.

[9] Anil K Jain. Data clustering: 50 years beyond k-means. *Pattern recognition letters*, 31(8):651–666, 2010.

[10] Greg Hamerly and Charles Elkan. Learning the k in k-means. In *Advances in neural information processing systems*, pages 281–288, 2004.

[11] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd*, volume 96, pages 226–231, 1996.

[12] Erich Schubert, Jörg Sander, Martin Ester, Hans Peter Kriegel, and Xiaowei Xu. Dbscan revisited, revisited: why and how you should (still) use dbscan. *ACM Transactions on Database Systems (TODS)*, 42(3):1–21, 2017.

[13] Thomas Dubot. Clustering aircraft trajectories with recursive dbscan, October 2019.

[14] Christian Eduardo Verdonk Gallego, Victor Fernando Gómez Comendador, Francisco Javier Saez Nieto, and Miguel Garcia Martinez. Discussion on density-based clustering methods applied for automated identification of airspace flows. In *2018 IEEE/AIAA 37th Digital Avionics Systems Conference (DASC)*, pages 1–10. IEEE, 2018.

[15] Leland McInnes and John Healy. Accelerated hierarchical density based clustering. In *2017 IEEE International Conference on Data Mining Workshops (ICDMW)*, pages 33–42. IEEE, 2017.

[16] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[17] Mihael Ankerst, Markus M Breunig, Hans-Peter Kriegel, and Jörg Sander. Optics: ordering points to identify the clustering structure. *ACM Sigmod record*, 28(2):49–60, 1999.

[18] JAS Almeida, LMS Barbosa, AACC Pais, and SJ Formosinho. Improving hierarchical cluster analysis: A new method with outlier detection and automatic clustering. *Chemometrics and Intelligent Laboratory Systems*, 87(2):208–217, 2007.

[19] Adria Segarra Torne. Route clustering for strategic planning in air traffic management. Master's thesis, UC Irvine, 2015.

[20] Ian T Jolliffe. Springer series in statistics. *Principal component analysis*, 29, 2002.

[21] Jonathon Shlens. A tutorial on principal component analysis. *arXiv preprint arXiv:1404.1100*, 2014.

[22] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

[23] Laurens Van Der Maaten. Accelerating t-sne using tree-based algorithms. *The Journal of Machine Learning Research*, 15(1):3221–3245, 2014.

[24] Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65, 1987.

[25] Davoud Moulavi, Pablo A Jaskowiak, Ricardo JGB Campello, Arthur Zimek, and Jörg Sander. Density-based clustering validation. In *Proceedings of the 2014 SIAM international conference on data mining*, pages 839–847. SIAM, 2014.

[26] Ricardo JGB Campello, Davoud Moulavi, and Jörg Sander. Density-based clustering based on hierarchical density estimates. In *Pacific-Asia conference on knowledge discovery and data mining*, pages 160–172. Springer, 2013.

[27] Leland McInnes, John Healy, and Steve Astels. hdbscan: Hierarchical density based clustering. *Journal of Open Source Software*, 2(11):205, 2017.

[28] Maxime Gariel, Ashok N Srivastava, and Eric Feron. Trajectory clustering and an application to airspace monitoring. *IEEE Transactions on Intelligent Transportation Systems*, 12(4):1511–1524, 2011.

[29] Dmitry Ulyanov. Multicore-tsne. https://github.com/DmitryUlyanov/Multicore-TSNE, 2016.

[30] Lars Buitinck, Gilles Louppe, Mathieu Blondel, Fabian Pedregosa, Andreas Mueller, Olivier Grisel, Vlad Niculae, Peter Prettenhofer, Alexandre Gramfort, Jaques Grobler, Robert Layton, Jake VanderPlas, Arnaud Joly, Brian Holt, and Gaël Varoquaux. API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pages 108–122, 2013.

[31] Luis Basora. sectflow, June 2019.

[32] EUROCONTROL Operations Planning Raffaele Russo. Capan methodology sector capacity assessment. Air Traffic Services System Capacity Seminar/Workshop, June 2016. https://www.icao.int/ESAF/Documents/meetings/2016/CAPAN_Sector_Capacity_Assessment.pdf (available on June 2020).