

# Occupancy Peak Estimation from Sector Geometry and Traffic Flow Data

Luis Basora, Valentin Courchelle, Judicaël Bedouet, Thomas Dubot  
ONERA / DTIS, Université de Toulouse  
Toulouse, France

**Abstract**—This paper describes a methodology to estimate the peak threshold of the occupancy count metric by using machine learning techniques on operational sector geometry and traffic flow data. We propose a clustering approach to identify the traffic flows crossing a sector from a sample of trajectories representative of the periods when the sector is operationally deployed. A dataset for the Bordeaux Area Control Center (ACC) is built from metrics computed on the traffic flows and sector geometry (e.g. number of crossing flows, sector volumes) and used to train a model able to predict the peak value. The model can be used to estimate the initial peaks for new designed sectors as well as to help assess the validity of the current values for the existing operational sectors. Even though the results are accurate for Bordeaux ACC, the applicability of the method to different ACC or its adaptation to estimate other capacity metrics' thresholds should be further investigated.

**Index Terms**—Sector capacity, traffic flows, traffic complexity, machine learning, trajectory clustering, DBSCAN, random forest

## I. INTRODUCTION

An Area Control Centre (ACC) provides Air Traffic Control (ATC) services to the airspace under its jurisdiction. The ACC airspace is made up of a set of elementary sectors some of which can be combined together into control sectors. A sector configuration is the set of sectors deployed within a certain time period according to the daily Sector Configuration Plan (SCP) established by the Flow Management Position (FMP). Each deployed sector in a sector configuration is tactically operated by a controller team and monitored by the FMP to ensure that the traffic demand can be effectively managed by the controllers.

The FMP can use several tactical indicators or metrics for monitoring the traffic load in a sector, such as the Entry Count (EC) or the Occupancy Count (OCC) [1]. In order to properly represent the use of the capacity in a sector, these metrics have a number of associated parameters and thresholds to be fine-tuned. These thresholds are currently estimated based on the judgement and experience of the FMP and can be regularly adjusted to take into account the feedback of the controllers as well as to adapt to any relevant operational change.

The OCC is a tactical indicator calculated at each defined time step (usually every minute) as the number of flights inside a sector during a period of time called OCC count duration. If the OCC is higher than the defined peak threshold, the FMP will consider taking some Demand and Capacity Balance (DCB) measures in order to resolve the traffic overload situation, e.g. creating a regulation or splitting the sector into

two. In order to avoid unnecessary and costly DCB actions, the peak thresholds should be determined according to the volume of traffic that can be effectively managed by the controller teams in the corresponding sectors.

For instance, Figure 1 shows an example of the OCC time series for sector X4 in Bordeaux ACC, with the peak threshold set to 20 by the operational teams (horizontal red line). We can observe a traffic overload situation between about 10:30 and 11:10 requiring DCB action. As X4 is an elementary sector which can't be split, a cherry-picking measure to keep some flights in sector X3 (below X4) or regulation may be needed, inducing delays and costs for the airspace users.

On the other hand, with the SESAR Dynamic Airspace Configuration (DAC) concept [2], it is expected that a wider range of sector geometries will be designed to generate additional and more modular sector configurations, which will be deployed in a further dynamic and flexible way in reaction to changes in the traffic demand, weather and controller workload. This creates the need for support tools to determine the capacity thresholds such as the OCC peak for the new sector designs.

This paper presents an approach based on Machine Learning (ML) techniques to automatically estimate the OCC peak threshold for either a new sector or an existing sector for which we want to assess the validity of the existing peak value. Even though the methodology described here is applied to the specific case of the OCC peak estimation, it should be possible to adapt it to similarly determine other traffic monitoring values, e.g. Hourly Rate for the EC or the Sustain Value for the OCC (see definitions at [1]).

The dataset for the ML process consists of a set of geometric

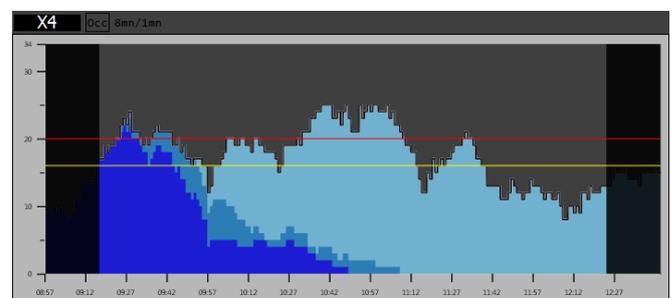


Figure 1. Occupancy values for sector LFBBX4 with the peak threshold represented by the red line.

and traffic complexity features computed on the existing sectors along with the OCC peak values determined by the FMP. In order to select the relevant geometric features, we have considered some of the variables used in the analytical model developed in [3] to assess sector capacity. As for the traffic complexity metrics, we have found relevant the research in [4], where the authors propose a list of traffic complexity metrics based on a set of identified sector flow patterns.

A traffic flow can be defined as a group of similar trajectories, where the notion of similarity can be based on either some operational criteria (e.g. flights concerning a specific city-pair) or distance metric between trajectories when clustering techniques are used to automatically identify them.

We believe like the authors in [5] that complexity metrics based on traffic flows could be a better way to account for traffic complexity than metrics such as the dynamic density [6], which are instead based on instantaneous measures of individual aircraft positions and intent. Firstly, flow patterns are more robust to traffic uncertainties because of their aggregated nature and higher-level view of traffic characteristics. Secondly, flows provide a more natural representation of the mental picture of the operational teams when assessing complexity, which is useful for model validation.

Therefore, we have developed a specific algorithm to identify the sector traffic flows from the set of trajectories crossing the sector when operationally deployed, which can be determined by exploiting the dataset of historical SCP. Based on previous methods to determine traffic flows from trajectories by using clustering techniques [7] [8] [9], our algorithm is designed to be relatively simple to use as it does not require the definition of specific parameters per sector.

We describe here the application of our methodology to the Bordeaux ACC, but we believe the generalisation to any other ACC should be possible. However, this should be further investigated as should the feasibility of combining data from multiple ACC into a single dataset by considering the specificities of the operational environments of each ACC. When the study is limited to a single ACC, the small amount of sectors available for the training of the OCC peak model is a real challenge. Fortunately, research in the area of ML with small data exists in the clinical/biomedical domain [10] that can be applied to our case.

This paper is organized as follows. Section II describes the Bordeaux ACC scenario and associated datasets. Section III presents the methodology including the clustering algorithm to identify the sector traffic flows as well as the selection of the relevant sector features. A ML model is then introduced to estimate the OCC peak value from geometrical and traffic flow features. Section IV presents the results obtained from the Bordeaux ACC operational data. Finally, section V summarizes the main achievements and limitations and identifies some ideas for future work.

## II. SCENARIO AND DATASETS

The scenario used in this study is illustrated in Figure 2. Figure 2a shows the horizontal and vertical split of Bordeaux

ACC in terms of elementary sectors whereas in Figure 2b an example of a sector configuration is presented.

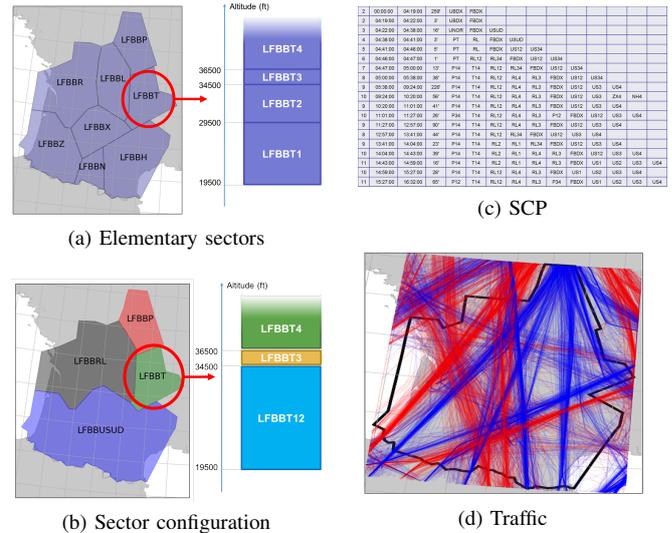


Figure 2. On the left, the Bordeaux ACC elementary sectors and an example of a sector configuration. On the right, an example of a daily SCP and the traffic between July 13th and 19th 2017 in blue (resp. in red) southward (resp. northward) trajectories.

Sector data (AIRAC 1711<sup>1</sup>) including geometry and OCC peak values as well as the SCP datasets were provided by the Bordeaux operational team. In total, we have a dataset of 75 sectors with OCC peaks ranging from 16 to 30.

We downloaded the trajectories for each of the 75 sectors from the OpenSky ADS-B network [11] according to the deployment timeslots of the sector configurations specified in the SCP dataset (Figure 2c shows an example of a daily SCP in the dataset). The main period considered was the week of operations between July 13th and 19th 2017 in the Bordeaux ACC (see Figure 2d). Even though this was the week with the highest traffic volume in 2017, some of the sectors still had an insufficient number of trajectories (under 400) to be representative of the traffic operated in the sectors. For these sectors, we also downloaded the traffic for their deployment slots during the rest of the days in July and the first semester of 2017. Thus, the total number of trajectories per sector ranges from 481 for sector T123 to 4965 for sector US34, with an average of 2523 trajectories per sector.

## III. METHODOLOGY

Figure 3a presents an overview of the methodology, which will be detailed in the next sections. The trajectory, sector and SCP datasets are the main inputs to the traffic flow identification algorithm (see Figure 3b). The dataset for the ML process is generated from the computed sector geometry and flow metrics, the operational peaks and some other contextual features.

<sup>1</sup>In fact, we could have used any 2017 AIRAC cycle as no significant changes were introduced in the ACC sector geometry data in 2017.

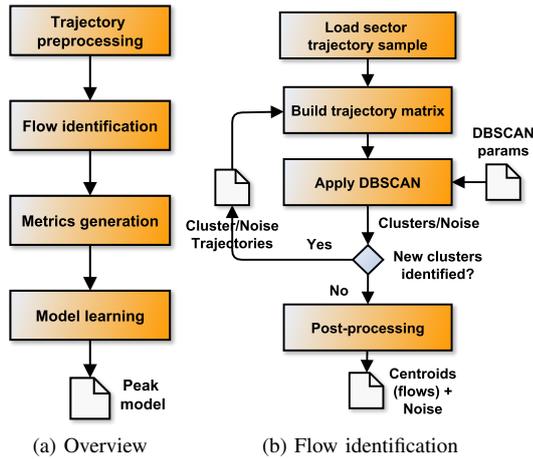


Figure 3. Methodology to build a model for OCC peak estimation with a focus on the sector flow identification process.

### A. Trajectory preprocessing

Data preprocessing of the ADS-B trajectories was necessary to extract the trajectories for the area of interest (Bordeaux sectors), filter out the noise, project their geographic coordinates in order to facilitate distance calculations and finally simplify them with the Douglas Peucker [12] algorithm.

For each sector, a representative trajectory sample was extracted from the downloaded ADS-B traffic by clipping the trajectories crossing the sector when operationally deployed (opening slots defined in the SCP dataset). Finally, we re-sampled the extracted trajectories with the PCHIP algorithm [13] [14] to smooth each trajectory with 50 data points regularly distributed, which is only useful to compute the cluster centroids (see next section for further details).

### B. Traffic flow identification

Traffic flow identification is required to compute the traffic complexity metrics which will be part of the ML dataset.

One way to identify the sector flows is by clustering similar trajectories together, each cluster representing a traffic flow. The trajectories should be representative of the sector traffic for the periods when it is operationally deployed, which can be determined thanks to the SCP dataset. A number of methods in the literature [7] [15] [8] [9] [16] are specifically designed to cluster flight trajectories into flows, most of them based on density-based clustering algorithms such as DBSCAN [17].

A general framework for flow analysis is developed in [7] based on the HDBSCAN clustering algorithm [18] and two trajectory distances. We tried to apply this method with the Euclidean distance, but the need to fine-tune the parameters of the clustering algorithm to adapt to the specificities of each sector made the approach impractical for our use case in spite of obtaining sometimes reasonably good results.

A much simpler approach for flow identification is presented in [4], where a flow is represented by a triplet of sector designators: entry sector, current sector and exit sector. However, as pointed out by the authors, an issue in the way

the flows are identified is that some flows contain a mix of ascending and descending flights that should actually be separated into two different flows. A related issue is that the lateral dispersion of trajectories associated with a flow can sometimes be considerable due to the fact that the location of the trajectory entry and exit points in the sector are not taken into account.

In order to overcome some of the issues of these two approaches, we propose here a new method. The idea is to represent a trajectory within a sector only by its first and last points corresponding to the entry and exit points of the flight in the sector. The reason is that, after experimentation, we realised that the intermediate trajectory points were actually superfluous in terms of the quality of the resulting clustering and that by ignoring them we could develop a clustering approach more appropriate to our problem than the one proposed in [7]. On the other hand, flows' homogeneity should be improved compared to [4] since trajectories are associated with flows based on the precise entry and exit points of the trajectory in the sector and not simply by a triplet of sector identifiers.

The proposed clustering method is further detailed in Figure 3b. First, for each sector, a matrix is built from the sector trajectory sample generated in the pre-processing phase, where each line of the matrix is a vector of six dimensions representing the trajectory entry and exit points in the sector, i.e. the two geographic coordinates and altitudes of the entry and exit points.

The logarithmic function is applied to the altitude. Thus two stable (levelled off) cruise trajectories at high levels (e.g. FL330 and FL370) will be relatively closer than a potentially evolving (climbing or descending) and a stable trajectory interacting at lower levels (e.g. FL290 and FL330), which should help the algorithm to separate the two types of trajectories. Also, all matrix values are standardized to have zero mean and unit variance. The resulting matrix is the input to the DBSCAN algorithm which identifies the initial set of clusters (flows) and outliers, i.e. the set of trajectories identified by the algorithm as not belonging to any of the clusters (noise).

After the first application of DBSCAN, certain clusters seem to represent more than one flow (see Figure 4). A possible solution would have been to set specific DBSCAN parameters per sector, which would negatively impact the usability of our approach. Also, we could have included the aircraft track as an additional feature in the clustering, but this would have not solved completely the issue, which is especially frequent when two flows have the same entry and exit coordinates in the sector, but one is stable and the other evolving. In this situation, the trajectories in both flows can be relatively very close as only their altitude differs.

However, if we apply the same steps recursively to the subset of trajectories in a cluster (see Figure 3b), we can increase the distance between trajectories as an effect of data standardization being applied to a subset of more homogeneous data. In the same way, additional flows can also be identified in the subset of outlier trajectories by providing sub-

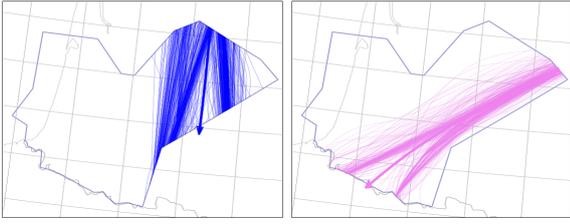


Figure 4. Examples of clusters containing several flows after the first application of DBSCAN for trajectories in sector ZX4.

cluster level DBSCAN parameters facilitating the creation of new clusters.

In fact, the idea is to use a progressive clustering approach [8] [9], where the same steps are applied recursively for each of the identified clusters in order to refine them. This refinement can be achieved thanks to the effect of data standardization which will transform data differently at cluster and sub-cluster level, as well as by fine-tuning at each level the DBSCAN parameters  $\epsilon$  and  $MinPts$  determining the size and density of clusters.

In the post-processing phase of the clustering process, a representative trajectory (centroid) is computed for each cluster as the average of the trajectories belonging to the cluster, where each trajectory is represented by a sample of 50 data points. However, because both the number of outliers and flows were excessive, a fusion step was needed to compensate for the effect of over-clustering caused by the recursive application of DBSCAN.

Thus, we added the following two steps to the post-processing:

- 1) Among the outliers, a trajectory that is close to a cluster centroid is added to the corresponding flow and the centroid is recomputed.
- 2) If two centroids are close, we merge them and compute the centroid of the resulting cluster.

For a trajectory to be considered close to a centroid, both need to be of the same nature (stable or evolving) and their entry points as well as their exit points have to be separated by less than 30NM. The same conditions apply for two centroids to be considered close, but the maximum distance is in this case 15NM. Finally, a flow is evolving (or in evolution) when the altitude difference between its entry and the exit points is greater than 4000ft.

In order to find the suitable values for the clustering parameters (see Table I), several trials were performed with different settings. Although Davies-Bouldin Index and Silhouette Index [19] [20] were computed to statistically assess the quality of the clustering, results were validated mainly by visual inspection, when possible with the help of the operational team in Bordeaux ACC (see conclusions in section V for further details).

Figure 5 shows the results for sector T4, where nine flows are identified with only 6.4% of the trajectories classified as outliers. The descending flows in the 3D figure are a

TABLE I  
CLUSTERING PARAMETERS

Name	Value
DBSCAN $\epsilon$	Cluster: 0.4
	Sub-cluster/noise: 0.5
DBSCAN $MinPts$	Cluster: 1% $N$
	$N$ : total number of trajectories
	Sub-cluster/noise: $\max(1\%N, 3)$
Minimum number of trajectories to form a flow	2% of the global number of trajectories
Distance to merge an outlier trajectory and a centroid	<30NM
Distance to merge two centroids	<15NM
Evolving if altitude difference	>4000ft

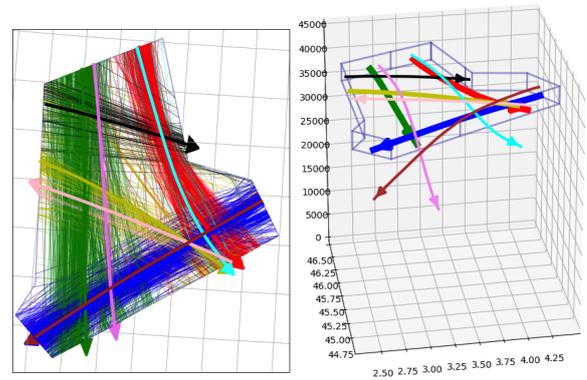


Figure 5. Clustering results for sector T4 displayed in 2D and 3D.

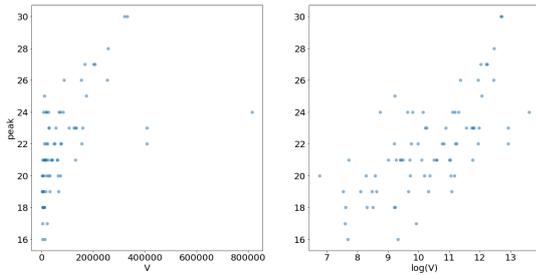
good illustration of flows not being recognized during the first application of DBSCAN and justifying the progressive clustering approach.

### C. Sector metrics generation

The OCC peak and more generally the capacity of a sector depends on a number of geometric and traffic features, but it is difficult to determine beforehand which ones are the most significant for prediction. Thus, from analysis of previous research and available operational expertise from Bordeaux, a set of potentially relevant sector metrics have been generated as input to the ML model for OCC peak estimation. A preliminary assessment of metric relevance has also been performed by computing the standard correlation coefficient with the target variable OCC peak.

1) *Geometry metrics*: Table II lists the metrics we have considered based on the study in [3], where the analytical relationship between sector geometry and capacity is analysed in detail. In particular, the study points out the non-linear relationship between sector volume  $V$  and capacity (see Figure 6). If we create a new feature  $\log(V)$ , we obtain the strongest correlation coefficient of all computed metrics (0.682).

Features  $l_l$  and  $l_u$  should provide some indication of the

Figure 6. OCC peak versus  $V$  and  $\log(V)$ .TABLE II  
SECTOR GEOMETRY METRICS

Name	Description
$V$	Sector volume (NM <sup>3</sup> )
$A$	Sector area (NM <sup>2</sup> )
$H$	Sector floor-to-ceiling height (ft)
$l_l$	Sector lower limit (sector floor altitude) (ft)
$l_u$	Sector upper limit (sector ceiling altitude) (ft)

traffic complexity in a sector, since sectors in the lower airspace have more traffic in evolution than those in the upper airspace. However, they can only take five different values (sectors in Bordeaux are vertically structured in four layers) when considered together, which is rather limited to properly discriminate between the different OCC peak values corresponding to a certain flight level. Feature  $l_u$  has nevertheless a correlation of about 0.54 whereas the  $l_l$  one is considerably weaker ( $< 0.3$ ).

Feature  $H$  has a correlation similar to the one of  $l_u$  and shares the same poor variance issue with  $l_l$  and  $l_u$  preventing both of them from being good discriminants if used on their own. However, the surface  $A$ , which alone has a weak correlation coefficient ( $< 0.3$ ), could be used in combination with  $l_l$  and  $l_u$ . In particular, the derived feature  $\log(A * l_l * l_u)$  has a correlation coefficient of 0.675, almost matching the  $\log(V)$  one.

2) *Traffic flow metrics*: The study in [4] proposes a list of traffic flow metrics to assess traffic complexity in order to dynamically predict the capacity of a sector. Based on this list, we have computed from the identified flows in each sector the complexity indicators in Table III.

For the characterization of the distribution of stable and evolving flows in a sector, we tested several possibilities and

TABLE III  
SECTOR TRAFFIC METRICS

Name	Description
$N_{flo}$	Total number of flows
$N_{stb}$	Number of stable flows
$N_{evol}$	Number of evolving flows
$N_{conv}$	Number of converging flows
$N_{cros}$	Number of crossing flows

we found that  $N_{stb}$  and  $R_{evol} = \frac{N_{evol}}{N_{flo}}$  had the strongest correlation coefficients with 0.434 and  $-0.349$  respectively. The latter negative correlation reflects indeed the fact that the capacity of a sector (OCC peak) should decrease when the rate of evolving traffic ( $R_{evol}$ ) goes up.

The metric  $N_{conv}$  counts the number of converging flow pairs, i.e. entering a sector via two different entry points to eventually merge into a single flow within the sector. For a flow pair to be considered as converging, we check that the distance between their exit sector points is less than 15NM and the difference between their directions when leaving the sector is not greater than  $40^\circ$ . The latter condition is to filter out flow pairs crossing rather than converging at the proximity of the sector border. By computing the rate of converging flow pairs, we obtain a new variable  $R_{conv}$  with a better correlation coefficient of  $-0.469$ .

Converging flows is a source of complexity for the controllers because they need to monitor that the separation minima is satisfied by aircraft when reaching their convergence points. Similarly, crossing flows are sources of potential conflicts that should increase complexity. This is the reason for computing the metric  $N_{cros}$ , the number of crossing flow pairs, and the derived metric  $R_{cros}$ , the latter presenting a stronger negative correlation coefficient of  $-0.366$ .  $R_{cros}$  is  $N_{cros}$  divided by the maximum number of potential crossing pairs ( $\frac{N \times (N-1)}{2}$ ), where  $N$  is the total number of flows in the sector. A crossing flow pair is identified when the minimum distance between the two flows is less than 5NM, excluding the case when a flow pair shares the same entry point but then the flows split into two different routes. For the sake of simplification, the distance is computed by considering each flow as a straight line.

#### D. Model learning

Our ML problem falls into the category of supervised learning, since for each sector in the dataset, we have the set of sector metrics/features described in the previous section along with the defined target OCC peak value as a label, which is an integer between 16 and 30. We dismissed addressing the problem as a classification task due to the important number of classes (15) relative to the small size of data and use instead a regression model.

On the other hand, since our data for this study is limited to the Bordeaux ACC sectors, we have a very small dataset with only 75 sectors. Therefore, in order to increase the chances of a better generalization, we started by testing some ML algorithms with high bias and low variance such as a simple linear regression model or Support Vector Machines (SVM). However, we obtained the best results with a Random Forest (RF) model, which also proved to be successful with small data in the field of clinical tests [10]. To be more specific, we selected a variant of RF called Extremely Randomized Tree (Extra-Tree) [21] as it slightly outperformed the RF in our experiments and it is considerable faster to run.

The RF model was trained with 80% of the data and tested with the remaining 20%. Because of the small size of the

dataset, the split was not done randomly but by ensuring that the training set was representative enough to cover both the Bordeaux ACC airspace and the different peak values as uniformly as possible. Thus, 60 sectors were used for training and the remaining 15 sectors were included in the test set: FNOR, H3, L4, N34, NH34, R12, RL12, RL1234, T123, USUD, US4, X1, Z2, ZNH4 and ZX34. Unfortunately, there is only enough data to cover 86.6% of the peak values in the training set (only values 28 and 29 are not included) and 53.3% in the test set (values included are: 18-23, 25-26, 28, 30).

The RF hyper parameters' maximum number of trees (MNT) and maximum tree depth (MTD) have been tuned using a 5-fold cross-validation (CV) method and the coefficient of determination  $R^2$  score. A parameter grid has been defined with the MNT ranging from 5 to 145 in steps of five and the MTD ranging from 2 to 17 in steps of one, that is 29 and 16 different values respectively. The CV assessment process has been performed uniquely on the training dataset and for each pair of the grid parameters (464 candidates), which in a 5-fold CV resulted in a total of 2320 assessments. The best settings found during this process for the MNT and MTD were of 100 and 7 respectively and the resulting RF model was the one selected for final evaluation on the testing set (see results in the next section).

The features (metrics) pre-selected to be part of the training and test sets were the ones presenting the strongest linear correlations with the peak. However, the final selection (features listed in Table VI) was done during the selection process of the RF hyper parameters by testing in particular whether to include  $\log(A * l_l * l_u)$ ,  $\log(V)$  or both. Thus, in spite of its strong correlation with the peak, feature  $\log(V)$  was finally excluded as we realised it actually had a slightly negative impact on the performance of the model.

For illustration purposes, we can see in Figure 7 a plot of one of the RF decision trees showing the different splits at each node and the leaves at the rightmost level. Unfortunately, this kind of visualisation does not make the results more interpretable, because the final predicted peak is computed as the average of the predictions of the set of decision trees forming the RF and rounded to the nearest integer number. To get a better meaning of a RF model, it is more helpful to look at the list of the most significant features (see Table VI).

#### IV. RESULTS

We use the coefficient of determination  $R^2$  to assess the performance of the RF model. This coefficient is of 0.979 for the training set indicating that the predictions are in general very accurate. Thus, the peak has not been correctly predicted for ten of the sectors or 17% of the training set (see Figure 8a and Table IV), out of which nine present an error of one unit.

If we analyse these errors further, we realize that eight of the problematic sectors are elementary sectors rarely used (less than 50 times in three years<sup>2</sup>) on a sector configuration

<sup>2</sup>From 2014 to 2017

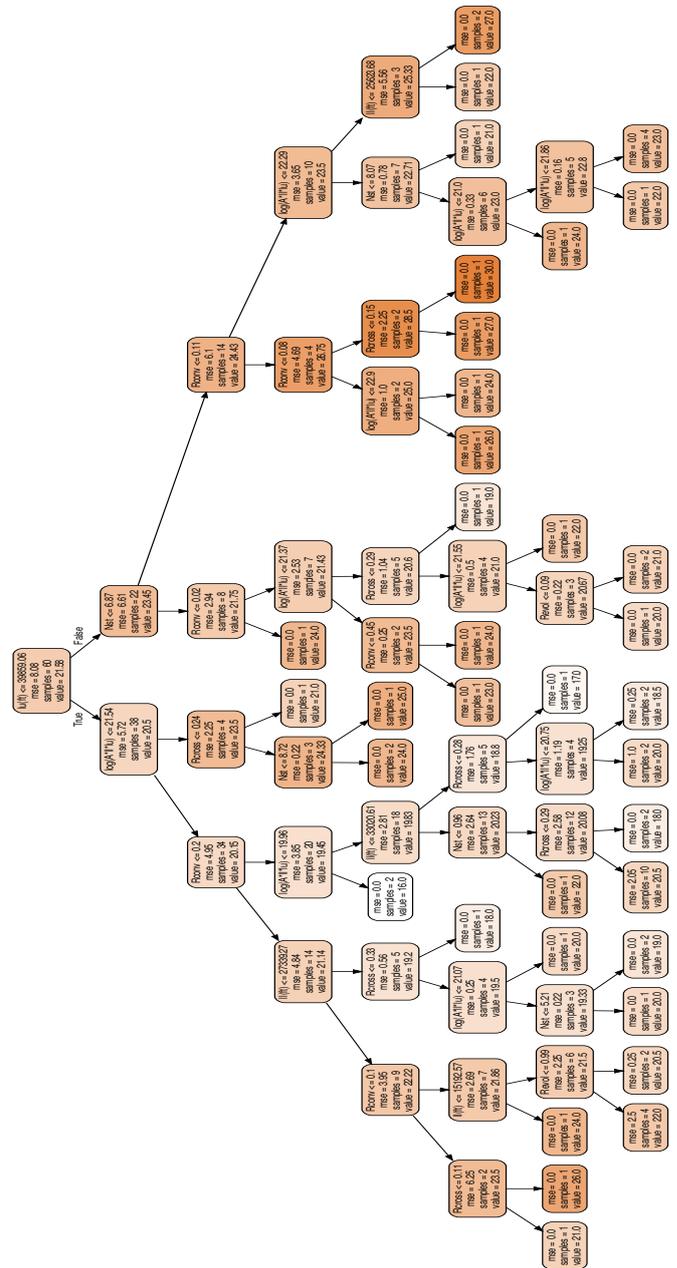


Figure 7. Example of a RF decision tree (50th tree in the RF).

with the exception of L2. Consequently, the feedback from the operational teams on the peak of these sectors is less frequent and their values potentially less reliable. On the other hand, trying to increase the tree depth parameter in the RF model to further reduce these errors may lead to overfitting the model to the training data.

As for the test set, the prediction results are displayed in Table V and Figure 8b. The coefficient  $R^2$  for the test set is of 0.838 reflecting also a good level of accuracy. Thus, for 80% of the test sectors, the error is either non-existent or of just one unit. For the three sectors with errors of two units, in the same three years, X1 has never been opened and NH34

TABLE IV  
RESULTS FOR THE TRAINING SET (ONLY SECTORS WITH ERRORS IN THE  
OCC PEAK PREDICTION ARE DISPLAYED)

Sector	Real peak	Prediction	Error	Abs. error
X2	18	19	1	1
X3	17	18	1	1
H2	22	21	-1	1
T12	24	23	-1	1
R3	18	19	1	1
L1	16	17	1	1
L12	20	21	1	1
L3	21	20	-1	1
L2	19	20	1	1
R1	17	19	2	2

TABLE V  
OCC PEAK PREDICTIONS FOR THE SECTORS IN THE TEST SET

Sector	Real peak	Prediction	Error	Abs. error
H3	20	20	0	0
ZX34	25	25	0	0
ZNH4	28	28	0	0
T123	22	22	0	0
FNOR	22	22	0	0
Z2	21	20	-1	1
NS34	23	22	-1	1
US4	30	29	-1	1
USUD	24	23	-1	1
R12	19	20	1	1
L4	20	21	1	1
RL12	22	21	-1	1
X1	18	20	2	2
NH34	23	25	2	2
RL	26	24	-2	2

only eight times with an accumulated time of 208 minutes in operation. Thus, the errors may be again a consequence of the limited use of the sector and the resulting poor operational feedback.

However, this is not the case of sector RL which has been used 3610 times in the same three years with a total operation time of 63,184 minutes. Since we were unable to explain the error in this case, we checked with the operational team in Bordeaux. They indicated to us that this sector was usually opened as an intermediate step to switch between two sector configurations. In fact, RL is not usually opened for a long time since it is quickly split into sectors RL12 and RL34. To be more precise, RL is actually useful as a transition between UBDX (355,826 minutes of accumulated time) and sector pair RL12/RL34 (269,285 and 142,629 minutes respectively). Therefore, poor operational feedback is again to blame, because full capacity in RL is actually never reached.

An advantage of an RF model is its capability to perform automatic feature selection, which is helpful to understand how the model makes the predictions. In Table VI, we list the features and their relative importance as computed by our

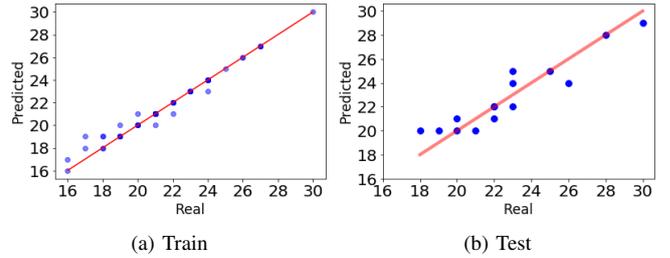


Figure 8. Results with test and train datasets.

TABLE VI  
RANDOM FOREST FEATURE SELECTION

Feature	Importance
$\log(A * l_l * l_u)$	32.6%
$l_u$	23.6%
$R_{conv}$	12.7%
$l_l$	11.0%
$R_{cross}$	8.7%
$N_{st}$	6.6%
$R_{evol}$	4.8%

RF model.

We can observe the general pre-eminence of the geometric features over the traffic ones with  $\log(A * l_l * l_u)$  being in the top. A possible conclusion is that the geometric features already offer on their own a good approximation to the peak prediction, whereas the traffic complexity metrics play only a secondary role in refining the prediction for the sectors which are geographically similar. Nevertheless, we were expecting the traffic metrics to be more significant, especially  $N_{st}$  and  $R_{evol}$ , which may be explained in part by the fact that features  $l_u$  and  $l_l$  contain already in themselves some information concerning the stability or evolving nature of the traffic.

## V. CONCLUSIONS

This paper has presented a methodology based on ML techniques aimed at predicting the sector OCC peak threshold from a dataset of sector geometry and traffic complexity metrics. We have developed a specific clustering method adapted to the identification of the relevant sector flows from a sample of trajectories representative of the traffic in the sector by exploiting the SCP. The methodology has been applied to Bordeaux ACC with the help of the operational teams who provided us with the necessary data and support.

The results show that an RF model can be trained to accurately predict a sector OCC peak from a small dataset. However, additional tests and assessments from the operational experts should be performed to ensure that some of the assumptions and choice of parameters are optimal and to further confirm that the model generalizes well when applied to new sector designs.

The fact that the traffic complexity metrics are less significant than expected in the predictions of the model may be explained as mentioned before by the fact that features  $l_l$  and

$l_u$  already characterize in part the nature of the traffic. But it may also be due to shortcomings in our method for flow identification or the way metrics are computed.

We asked an operational expert to draw the flows for sector RL12. Only one of the flows identified by the operational expert was not found by our algorithm and after analysis we realized that it had been merged with another flow. Further efforts are needed to assess the sensitivity of our peak estimation model when this happens and if this could be improved by setting specific clustering parameters for the problematic sectors. Ideally, the same operational validation should be performed for the 75 sectors. After discussion with the operational team in Bordeaux and in order for the clustering algorithm to better identify the sector flows, it may be useful to consider the notion of flight city pairs as we realized it was for the operational experts a natural way to think of flows.

Concerning the complexity metrics, the operational team in Bordeaux suggested a list of improvements to be tested in a future version:

- To distinguish the evolving flows entering/leaving from/through the sector floor or ceiling as they are more complex than the rest.
- To consider only the parts of the flows within the 3D volume rather than 2D border of a sector in order to avoid for instance counting crossings actually happening outside the sector.
- To distinguish crossings between stable flows from crossings where at least one of the flows is in evolution since the complexity of the latter is more significant.
- To create an additional feature linked to the number and location of points of interest in a sector, i.e. the areas of high complexity. The bigger the number of these hotspots and their dispersion in the sector, the higher the complexity should be.
- To consider the time dimension since metrics on converging or crossing flows are computed in a purely geometric way without regard to the possibility that two flows may actually never exist at the same time in the sector.

As for the improvement of the training process, the following ideas could be worth investigating:

- To extend the dataset with sector data from other ACC in France or even in Europe. This would need to be further researched as the operational environment and the way capacity is assessed may be significantly different between ACC. Also, if our methodology is to be applied to other ACC, geometric features may need to be adapted to the fact that some sectors may be vertically composed of prisms with different surface areas, which was not the case in Bordeaux.
- To focus the training on the sectors where the peak is more reliable, i.e. the sectors being regularly used. This has the inconvenient of reducing even more the data available for training, in particular by excluding a significant part of the elementary sectors.

## ACKNOWLEDGMENT

The authors would like to thank the operational team in Bordeaux and Xavier Olive from ONERA for their contributions to the present work.

## REFERENCES

- [1] M. Dalichampt and C. Plusquellec, "Hourly entry counts versus occupancy count—relationship, definitions and indicators," 2007.
- [2] "SESAR Solution 08.01 SPR-INTEROP/OSED for V2 - Part1, Edition 00.01.00, D2.1.020," Tech. Rep., 2017.
- [3] J. D. Welch, J. W. Andrews, B. D. Martin, and B. Sridhar, "Macroscopic workload model for estimating en route sector capacity," in *Proc. of 7th USA/Europe ATM Research and Development Seminar, Barcelona, Spain*, 2007, p. 138.
- [4] L. Song, C. Wanke, and D. Greenbaum, "Predicting sector capacity for tfm decision support," in *6th AIAA Aviation Technology, Integration and Operations Conference (ATIO)*, 2006, p. 7827.
- [5] E. Salaun, M. Gariel, A. E. Vela, and E. Feron, "Aircraft proximity maps based on data-driven flow modeling," *Journal of Guidance, Control, and Dynamics*, vol. 35, no. 2, pp. 563–577, 2012.
- [6] I. V. Laudeman, S. Shelden, R. Branstrom, and C. Brasil, "Dynamic density: An air traffic management metric," 1998.
- [7] L. Basora, J. Morio, and C. Mailhot, "A trajectory clustering framework to analyse air traffic flows," in *SIDs 2017, 7th SESAR Innovation Days*, 2017.
- [8] G. Andrienko, N. Andrienko, G. Fuchs, and J. M. C. Garcia, "Clustering trajectories by relevant parts for air traffic analysis," *IEEE transactions on visualization and computer graphics*, vol. 24, no. 1, pp. 34–44, 2018.
- [9] S. Rinzivillo, D. Pedreschi, M. Nanni, F. Giannotti, N. Andrienko, and G. Andrienko, "Visually driven analysis of movement data by progressive clustering," *Information Visualization*, vol. 7, no. 3–4, pp. 225–239, 2008.
- [10] T. Shaikhina, D. Lowe, S. Daga, D. Briggs, R. Higgins, and N. Kovanova, "Decision tree and random forest models for outcome prediction in antibody incompatible kidney transplantation," *Biomedical Signal Processing and Control*, 2017.
- [11] M. Schäfer, M. Strohmeier, V. Lenders, I. Martinovic, and M. Wilhelm, "Bringing up opensky: A large-scale ads-b sensor network for research," in *Proceedings of the 13th international symposium on Information processing in sensor networks*. IEEE Press, 2014, pp. 83–94.
- [12] D. H. Douglas and T. K. Peucker, "Algorithms for the reduction of the number of points required to represent a digitized line or its caricature," *Cartographica: The International Journal for Geographic Information and Geovisualization*, vol. 10, no. 2, pp. 112–122, 1973.
- [13] F. N. Fritsch and R. E. Carlson, "Monotone piecewise cubic interpolation," *SIAM Journal on Numerical Analysis*, vol. 17, no. 2, pp. 238–246, 1980.
- [14] D. Kahaner, C. Moler, and S. Nash, "Numerical methods and software," *Englewood Cliffs: Prentice Hall*, 1989, 1989.
- [15] A. T. Nguyen, "Identification of air traffic flow segments via incremental deterministic annealing clustering," Ph.D. dissertation, 2012.
- [16] M. C. R. Murça, R. DeLaura, R. Hansman, R. Jordan, T. Reynolds, and H. Balakrishnan, "Trajectory clustering and classification for characterization of air traffic flows," *AIAA Aviation*, 2016.
- [17] M. Ester, H.-P. Kriegel, J. Sander, X. Xu *et al.*, "A density-based algorithm for discovering clusters in large spatial databases with noise." in *Kdd*, vol. 96, no. 34, 1996, pp. 226–231.
- [18] L. McInnes, J. Healy, and S. Astels, "hdbscan: Hierarchical density based clustering," *The Journal of Open Source Software*, vol. 2, no. 11, mar 2017. [Online]. Available: <https://doi.org/10.21105/joss.00205>
- [19] O. Arbelaitz, I. Gurrutxaga, J. Muguerza, J. M. Pérez, and I. Perona, "An extensive comparative study of cluster validity indices," *Pattern Recognition*, vol. 46, no. 1, pp. 243–256, 2013.
- [20] P. J. Rousseeuw, "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis," *Journal of computational and applied mathematics*, vol. 20, pp. 53–65, 1987.
- [21] P. Geurts, D. Ernst, and L. Wehenkel, "Extremely randomized trees," *Machine learning*, vol. 63, no. 1, pp. 3–42, 2006.