

Smoothed traffic complexity metrics for airspace configuration schedules

David Gianazza, DSNA, Toulouse, France

Email : {lastname}@recherche.enac.fr

Abstract—This paper is a continuation of previous research on optimal airspace configuration. It is expected to improve the predictability and the flexibility of the airspace management process by computing realistic predictions of the sectors opening schedules in *En-route* ATC centers. In previous papers, we selected relevant complexity metrics to predict the controllers workload, using neural networks trained on recorded airspace configurations. We also introduced new algorithms to build optimally balanced airspace configurations, exploring all possible combinations of elementary sectors.

As a result of this previous work, we were able to compute realistic schedules on a whole day of traffic, using complexity metrics that were computed from recorded radar tracks. The raw metrics, however, showed high variations in time which caused a "configuration switching" phenomenon. Although the number of control sectors in the computed schedule stayed globally close to the recorded number of sectors, the airspace was reconfigured much more often than in reality. The present paper shows how the input metrics can be smoothed in order to avoid this problem, and what may be the subsequent problems caused by the smoothing strategy.

INTRODUCTION

Over the years, and in a context of increasing air traffic demand, there has been a growing need to increase the capacity of the Air Traffic Management system. Improving the *predictability* of the system's response to the traffic demand is also a crucial issue, as it would allow a better use of the existing resources and an earlier anticipation of future congestions.

The work presented in this paper is the continuation of previous research on airspace configuration schedules ([1], [2], [3]) and air traffic complexity metrics ([4], [5]) previously led at the Global Optimization Laboratory (CENA/ENAC) and now continued within the Planification, Optimization, and Modeling team of DSNA/DTI-R&D. The initial aim of this research is to compute realistic sectors opening schedules for *en-route* air traffic control centres, given an input traffic demand on a chosen day.

The current FMP/CFMU working method to build airspace configuration schedules relies on pre-defined sectorization scenarios, where the incoming traffic flows¹ are matched against the sector capacities² to detect

¹The metric used is the "incoming flow", also called "flight counts" or "traffic-volume" in some Eurocontrol documentations ([6]) or "traffic load" in the CFMU handbook. For a sector, it is the number of flights that will enter the sector within the next 60 minutes (or any other chosen period of time).

²The sector capacity is defined as a threshold value on the number of flights that may enter the sector in a chosen period of time.

potential overloads. Although it may prove effective in practice as it relies on the FMP/CFMU operators experience, this method is not grounded on a solid assessment of the actual controllers workload. Consequently, implementing any strategy to optimize the airspace schedule on this basis may lead to unexpected results (see [1]). Another drawback of the current method is that only a small subset of all possible airspace configurations is used.

In [3], new algorithms were proposed, using more relevant complexity metrics to assess the controllers workload, and exploring all possible combinations of elementary sectors to build optimal airspace configurations. As a result of this previous work, we were able to compute realistic airspace configuration schedules on a whole day of traffic, using raw complexity metrics computed from recorded radar tracks. The raw metrics, however, showed high variations in time which caused a "configuration switching" phenomenon. Although the number of control sectors in the computed schedule stayed globally close to the recorded number of sectors, the airspace was reconfigured much more often than in reality.

The present paper shows how the input metrics can be smoothed in order to avoid this problem. The next section first provides a short overview of the current research on airspace configuration and air traffic complexity. Section II describes the algorithms used to predict the sector status and to build airspace configurations, mainly focusing on the few improvements that were made since [3] was published. The experimental procedure applied to select the best smoothing parameters is described in section III. Results are provided in sections IV and V. Section VI concludes this paper.

I. OVERVIEW

A. Airspace configuration

Current research on airspace configuration is manifold and may deal with strategic airspace partitioning (see [7] and included references, [8]), pre-tactical sectors opening schedules ([9]), or tactical airspace management ([13]). In this paper, we are mainly concerned with pre-tactical airspace configuration schedules, although some of the proposed algorithms may also be used in tactical applications, provided the complexity metrics being used are relevant in that context.

The FMP/CFMU working method to build sectors opening schedules was shortly described in the introduction. Current research led by Eurocontrol proposes

short-term improvements of the Flow Management process, mainly by avoiding unnecessary regulations when building sectors opening schemes ([9],[6], [11]). One of the main concerns is the *network effect* observed in ATM regulations ([12]). These studies still use incoming flows and sector capacities, and a small number of pre-defined configurations.

In the United States, the main concern seems to be the dynamic adjustment of the airspace structure to the traffic flows reroutings caused by severe weather conditions. It is expected that more flexible boundaries would allow a more efficient use of airspace and increase the overall capacity. In [13], pre-defined scenarios of airspace sectorizations associated to traffic rerouting scenarios are proposed as a short-term improvement to the current practice.

A more dynamic resectorization with flexible boundaries is envisioned in future operational concepts ([14], [15], and some SESAR *Operational Improvement* steps). It is expected that moving the sector's boundaries in real-time to adapt to the traffic demand would increase the capacity and the efficiency of the ATM system. The actual capabilities and potential benefits of this new operational paradigm are still largely unknown at this early stage, however. There is also some concern that unlimited flexibility in the sectors boundaries would lead to a loss of situational awareness by the air traffic controllers (see discussion and literature review in [16]).

The work presented in this paper is more medium-term research, trying to improve the predictability and the flexibility of today's airspace management in Europe. The idea is to find the optimal combination of elementary (or modular) sectors that will provide the maximum capacity to a given input traffic, and balance the controllers workload as best as possible among the control sectors.

This airspace partitioning problem would be difficult to solve without choosing a heuristic if every combination of sectors was possible. The partitioning of the whole ATCC's airspace into control sectors is highly combinatorial ([2]), even with relatively few elementary sectors. Hopefully, the list of possible *control sectors* (either elementary or collapsed sectors) that can be operated in an air traffic control center is relatively small³, as not all combinations of elementary sectors are operationally valid⁴. So we may explore all valid airspace configurations, which may be built with operationally valid control sectors only, using classical tree search methods ([1], [3]).

These algorithms are applied to the prediction of airspace configuration schedules, optimally balancing the workload among the control sectors. Consequently, we need a way to assess the controller's workload, and it was proposed to use relevant air traffic complexity metrics to that purpose ([4], [5]).

³The list of control sectors is available from the ATCC's database

⁴One usually does not merge sectors which are not geographically connex, for example.

B. Air traffic complexity

A multitude of air traffic complexity metrics have been proposed in the literature (see [17] and [18] for a review), and many studies tried to correlate some of these metrics to the controllers workload, using various methods: linear ([19]) or logistic ([20]) regression, cross-sectional time series analysis ([21]), neural networks ([22]),... Many ways to quantify the controller's workload have also been tried: physical activity ([23], [21]), physiological indicators ([24], [25]), simulation models of the controller's tasks ([26], [27]), subjective ratings ([19], [22], [20]). The reader may refer to [4] for a discussion on these variables. Let us just say that, in addition to being subject to noise and biases⁵, most of the above dependent variables require relatively heavy experimental setups to collect the data, usually with the active participation of controllers. Databases are often small and might exhibit low variability, which may in turn harm the statistical relevance of the results.

In order to avoid some of these drawbacks, we proposed a new dependent variable for which a large amount of data is available from the ATCC databases, and which does reflect an operational reality. The basic idea, introduced in [29], is that the decisions to split (resp. merge) a sector are mostly taken when the controller is close to overload (resp. under-load). So the sector status (merged, operated, or split) is directly related to the controller's workload and may therefore provide an acceptable dependent variable. In [4] and [5], neural networks were trained on recorded patterns of metrics and sector statuses⁶ to select the most relevant metrics for our airspace configuration problem.

The proposed method allowed to select a subset of only 6 relevant indicators among the initial 28 chosen from [19], [22], [30], [31] and other sources. The airspace configuration schedules obtained with these metrics as input were quite realistic ([3]) when computed from recorded radar tracks. In this previous work, however, the input metrics were not smoothed, and a "configuration switching" phenomenon was observed. Let us now see, after a short description of the algorithms, if smoothed metrics provide better results.

II. ALGORITHMS

Our aim is to build a realistic schedule of the airspace configuration throughout the day. To that purpose, one needs first a correct assessment of the workload generated by the traffic throughput in a control sector, and second an algorithm exploring all possible airspace configurations to find out the optimal one, with respect to the workload balance over control sectors.

A. A neural network to predict the sector status

Neural networks are used to issue sector status probabilities for each control sector of a candidate configura-

⁵such as the subjective ratings recency effect denounced in [25], or raters errors in the case of "over-the shoulder workload ratings" [28]

⁶In our case, a pattern is a vector of complexity metrics measured at a time t in a given sector, together with the sector status (*merged*, *normal*, or *split*) that was recorded at this time.

tion. Beyond the similarities with the biological model, an artificial neural network may be viewed as a statistical processor, making probabilistic assumptions about data ([32]). A *training* set of *patterns* is used to determine a statistical model of the process which produced this data. Once correctly trained, the neural network uses this model to make predictions on new data. The reader may refer to [33] and [34] for an extensive presentation of neural networks for pattern recognition.

In our case, the neural network is trained on recorded airspace configurations, considering the actual status of each control sector : *merged* when the sector is collapsed with other sectors to form a larger sector (low workload), *normal* when the control sector is opened (normal workload), or *split* into smaller sectors operated separately (high workload)⁷. The input variables are the relevant complexity metrics, or any candidate subset of metrics, normalized by subtracting the mean value and by dividing by the standard deviation. The output of the neural network is a triple of sector status probabilities ($p_{merge}, p_{normal}, p_{split}$).

The network is unable to make complex recommendations such as to split the sector's volume in several parts and then to merge each of these parts with other sectors. It only recommends to merge the sector when the workload is low, or split it when the workload is high, or operate it normally when the workload is acceptable. As we are necessarily in one of the above three cases, the sum of the three probabilities $p_{merge}, p_{normal},$ and p_{split} is always 1.

More details on neural networks applied to sector status prediction, in the context of airspace configuration, can be found in previous works ([3]). How these networks were used to select the most relevant metrics is described in [4] and [5]. The same network's topology and training algorithms are used in the work presented here to select the most relevant smoothing strategy for the input metrics.

The software implementation is different, though. In previous works, the `nnet` R package developed by Pr. Ripley was used. As it is envisioned in a near future to try other types of neural networks, more suited to time series, some new software⁸ was developed. A backpropagation method⁹ and a BFGS¹⁰ quasi-Newton optimization method were implemented in Ocaml language. The same stopping parameters as in previous works with `nnet` were used.

⁷Irrelevant statuses, such as when a part of the initial sector is merged with one control sector, and the other part with another control sector, were discarded in the neural network's training.

⁸ANNiML (Artificial Neural Networks in ML) is written in Ocaml and should be made available soon, probably under GNU Lesser General Public License.

⁹Backpropagation of the output error through the network's layers allows to approximate the partial derivatives of the error function with respect to the weights

¹⁰BFGS (Broyden-Fletcher-Goldfarb-Shanno) is an iterative local optimization method, starting from an initial point (weights values in our case) and using an approximate hessian and the gradient of the objective function to find a local optimum. Note that different initial points may lead to different local optima.

B. Tree search algorithms for well-balanced sector configurations

As previously told, the neural network cannot issue complex recommendations on how to reconfigure several control sectors. A tree search algorithm was used to that purpose, exploring all possible combinations of elementary sectors, to find out the optimal one.

An optimal configuration is one for which the workload among the control sectors is balanced as best as possible, while using the less possible resources, and satisfying operational constraints such as a maximum number of available working positions for example.

Once again, we used the same algorithm as in [3] to compute optimal airspace configurations, with a few improvements that shall be detailed later in this section, and with the aim to study the influence of the smoothing strategy on the computed opening schedule.

Let us just describe the main features of this algorithm. Starting at time $t=0$ with a configuration where all elementary sectors are assigned to a single controller's working position, the situation is reconsidered every minute of the day, using the status probabilities ($p_{merge}, p_{normal}, p_{split}$) of each control sector in the current configuration to decide if the airspace should be reconfigured or not.

The decision criterion may be straightforward (taking the action corresponding to the highest probability), or it may propose to take an action only when the corresponding probability is close enough to 1, and when the difference between the two highest probabilities is sufficient. The first, straightforward, decision criterion was called D1 in [3], and the second was name D2, with decision parameters η (threshold on the difference between the two highest probabilities, for merging decisions), α (proximity of p_{merge} to 1) and β (proximity of p_{split} to 1). Figure 1 illustrates criterion D2, showing the evolution of the sector status probabilities just before a "split" decision, when p_{split} reaches $1 - \beta$.

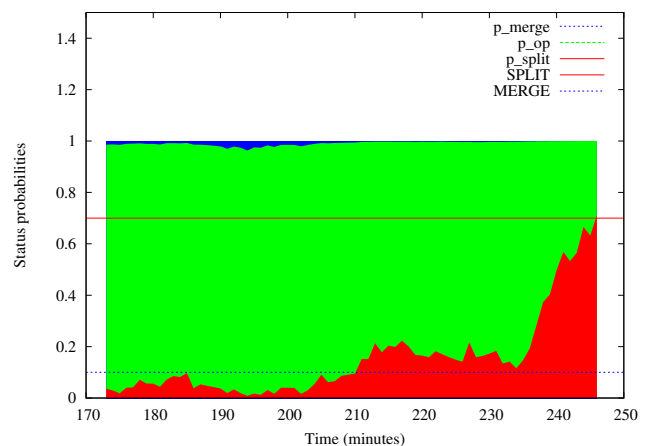


Fig. 1. Example of sector status probabilities in NGA sector (Brest ATCC) just before the algorithm decides to split the sector.

Once a decision to reconfigure some control sectors is taken, the corresponding elementary sectors are recombined, exploring all possible partitions of this set.

Some drawbacks of this local recombination method were highlighted in [3], for example in the case where the decision criterion triggers a "merge" action for two control sectors which are not neighbours. This is typically a case where the local recombination leads to no change, because the airspace should be reconfigured on a larger scale. A solution to this problem is to reconfigure the whole airspace in such cases. However, exploring exhaustively the whole tree of possible configurations by computing all of them becomes very rapidly computationally intensive even with a relatively small number of elementary sectors.

So the previous algorithm was improved as follows. Local recombinations are made as before when the control sectors that need to be reconfigured are geographically connected. If this is not the case, a full airspace reconfiguration is triggered, using a *Branch & bound* algorithm to explore all possible combinations. The detailed description of this algorithm will be the subject of a next publication, but the reader may refer to [1], [2], and [29] where a very similar *Branch & bound* algorithm is detailed.

A second improvement introduced in this paper is about the cost function allowing to compare the candidate airspace configurations. A more simple and more understandable cost function was designed, where the cost depends on the number of control sectors and the maximum probability in each category (*merge*, *normal*, *split*).

An "ideal" configuration should have $(p_{merge}, p_{man}, p_{split}) = (0, 1, 0)$ for all its control sectors. This is not always possible, so we need to take account of overloaded or underloaded sectors, and ill-balanced configurations. The cost of a configuration c , with a vector x of complexity metrics measured at time t is expressed as follows:

$$cost(c, x, t) = \underbrace{xx}_{k_1} \underbrace{xxx}_{k_2} \underbrace{xx}_{k_3} \underbrace{xxx}_{k_4} \underbrace{xx}_{k_5} \underbrace{xxx}_{k_6}$$

where we have assigned:

- k_1 digits to the number of overloaded sectors,
- k_2 digits to the maximum value of p_{split} among the overloaded sectors, where the probability is suitably scaled to the allowed number of digits,
- k_3 digits to the number of under-loaded sectors,
- k_4 digits to the maximum value of p_{merge} among the under-loaded sectors,
- k_5 digits to the number of normally loaded sectors,
- k_6 digits to the maximum value of $1 - p_{normal}$ among the normally loaded sectors,

With this cost, the first priority is to have the less possible overloaded sectors, and if there still remains some then the maximum probability p_{split} among these sectors should be as small as possible. The same explanation stands for underloaded sectors. For normally loaded sectors, we still want to use the less possible resources, but workload should be balanced as well as possible among the sectors. So the minimum value of p_{normal} among the normally loaded sectors should be

as high as possible. This is why we use the *maximum* of $1 - p_{normal}$ in the cost, so that minimizing this cost will lead to more desirable configurations.

III. EXPERIMENTAL PROCEDURE

Each complexity metric x_i may be smoothed by taking its average value over a period of time $[t - \delta_1, t + \delta_2]$, where we may try different values for δ_1 and δ_2 for each metric.

For now, the metrics are computed on past data (recorded radar tracks). In future applications, they may be computed either from simulated trajectories following flight plans, in the context of airspace configuration schedules, or from real-time radar tracks and trajectory predictions for tactical airspace management purposes. For real-time applications, one may prefer to smooth the metrics on a time window $[t - \delta, t]$, considering only the past positions of the aircraft. We decided to try this strategy first, which may be applied also to simulated trajectories for airspace schedules.

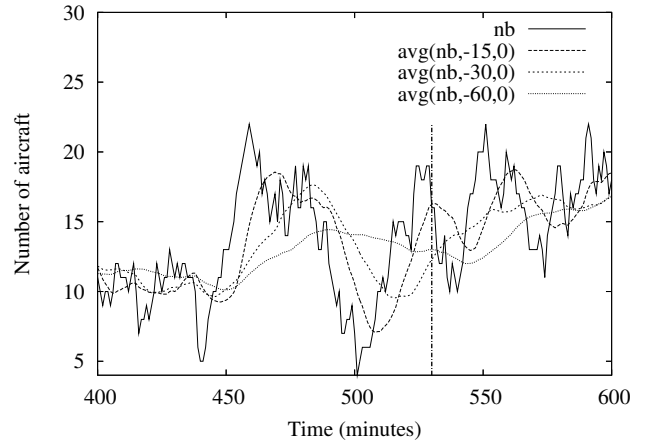


Fig. 2. Raw and smoothed number of aircraft in N sector (Brest ATCC). The splitting decision was taken at $t=530$ (vertical line).

As an illustration of the effect of smoothing on the metrics, figure 2 shows the number of aircraft within sector N (Brest ATCC), with different values of δ . We may notice the high variations in the raw aircraft count. The vertical line shows when the decision to split the sector into two smaller sectors was taken.

A. Testing different smoothing strategies

We would like to find out which combination of metrics and smoothing parameters is the best. This is a model selection problem. The main difficulties in model selection are the choice of a search strategy (how to explore the possible subsets of explanatory variables, knowing that the number of combinations is usually too large for an exhaustive search), and also the assessment of each model's performance (quality criterion, ability to generalize to fresh data).

In this paper, we will consider different values for the size of the smoothing window: 3, 5, 10, 15, 30, or 60 minutes. Ideally, we should make the same study as in [4], [5] but applied to the 27 complexity metrics with

all smoothing possibilities, which means 190 variables¹¹. The forward strategy that was used in previous works to explore different combinations of variables would take too much computation time, so it was decided to focus on the 6 most relevant variables found in our previous studies. These were the sector volume V , the number of aircraft within the sector Nb , the average vertical speed avg_vs , the incoming flows with time horizons of 15 minutes and 60 minutes (F_{15} , F_{60}), and the number of potential crossings with an angle greater than 20 degrees ($inter_hori$).

As a first approach, and keeping in mind that it is a fairly restrictive search strategy, it was decided to try different smoothing values, *applying the same smoothing window to all variables in the set of relevant raw metrics*. The reference set of variables is $REF = \{V, Nb, avg_vs, F_{60}, F_{15}, inter_hori\}$. The other combinations that were tested are **SM3**, **SM5**, **SM10**, **SM15**, **SM30**, **SM60**, which contain the same complexity metrics, smoothed respectively using time windows of 3, 5, 10, 15, 30, or 60 minutes.

B. Model selection and performance assesment

In our previous works ([4], [5]), the mean AIC¹² (averaged over the sample's size) was used to compare the performance of a given neural network on data samples of different sizes (a training set and a test set), and the mean BIC¹³ was used to compare different neural networks, trained on candidate subsets of complexity metrics (models of different sizes). In this paper, we will also use the mean BIC to compare the candidate models and assess the improvements provided by smoothing the metrics values, and the AIC to assess the generalization performance.

Once trained on past data, it important to check if the neural network also provides good predictions of the sector status when feeded with new inputs. This is generally done by splitting the initial data set in two samples: a train set and a test set. This *split-sample* (or hold-out) procedure is generally satisfying on large data samples, but may be prone to overfitting¹⁴ problems

¹¹27 metrics multiplied by 7 smoothing values (counting a zero value for the raw metrics), plus the sector volume.

¹²Akaike's "An Information Criterion" $AIC = 2\lambda - 2\ln(L)$, where λ is the number of unadjusted parameters of the model (i.e. the number of weights and biases of the network), and $\ln(L)$ is the log-likelihood error. When used for model selection with neural networks, AIC tends to overfit (see discussion in [34], p. 61), leading to select bigger models. The Schwartz's Bayesian Information Criterion is usually preferred.

¹³Schwartz's Bayesian Information Criterion $BIC = 2\lambda \ln(N) - 2\ln(L)$, where N is the size of the data sample. The BIC criterion gives a higher penalization than AIC to big models, but varies with the size of the data sample, so it may not be used to compare the performances of a neural network on samples of different sizes. Note that AIC and BIC are not absolute criteria: their evaluation is specific to the underlying "true" model, and only the relative differences in the criterion's value is useful.

¹⁴*Overfitting* occurs when the statistical model fits very well the data from which it was derived, but cannot generalize well on fresh data. The number of parameters in the model (network's weights for example) and few data samples may both cause overfitting problems. A neural network with too few weights may not be able to capture all the variations of the response to the input x , whereas a network with too many weights will more likely be subject to overfitting (see [32]).

on small samples. It was used in [4] and [5] with good results, but one may argue that the selected models may only be fit to the chosen train and test sets, although some tests on a second test set (another day of traffic) proved also satisfying.

So is was decided to apply a more sophisticated procedure, using first a *k-fold cross-validation* method for the model selection, and second a *split-sample* method (or hold-out validation) to assess the generalization performance of the best model. The initial data set is randomly split in two samples. The first one (*training set*) is again divided in k sub-samples and used for an iterative k -fold cross-validation allowing to select the best smoothing parameter. Then, the neural network is trained on the whole training set, and the generalization performance is checked on the the *test set*.

In our case, we applied a 10-fold cross-validation, iteratively holding out one of the 10 sub-samples of the training set to assess the candidate model, and training the neural network on the 9 remaining sub-samples. The Schwartz's Bayesian Information Criterion (BIC) is computed on the sample that was not used to train the network. The BIC is averaged on the 10 runs for each model. The best model is found by comparing the average BIC.

Once we have found the best model, the neural network is trained on the whole training set (the 10 samples). The generalization performance of the trained network is assessed by comparing the AIC value found for the training set to the AIC of the test set. As the training method is an iterative local optimization (BFGS) which may fall into local optima depending on the chosen initial weights vector, ten training runs are made with different random values of the initial weights¹⁵.

C. Comparison of airspace configurations schedules

So far we have only detailed how to compare different statistical models allowing to predict the sector status from smoothed complexity metrics. Our final goal, however, is to build realistic airspace configuration schedules. So we also need to consider the influence of the smoothing strategy on the overall airspace configuration.

Ideally, the computed schedule should reproduce the actual configurations recorded that day. However, there is a high variability in the decisions made by control room managers on *how* to reconfigure the airspace, which comes in addition to the variability of decisions on *when* to reconfigure. We may hope that our sector status prediction could give an indication on when to trigger a reconfiguration and allow to build realistic configurations, but our algorithms may not compute exactly the same configuration as in reality.

We will assess the realism of the computed schedule by comparing the number of control sectors to the actual number of sectors that were opened that day. The Pearson's correlation coefficient may give an indication of the linear correlation between the computed and the

¹⁵Note that the ten runs of the cross-validation were also made with different random initial weights

real number of control sectors. However it may not be always reliable¹⁶ so we will also compute an *ad-hoc* "dissimilarity measure" which is the surface delimited by the two curves, divided by the surface of the real schedule. With this measure, two identical curves shall have a dissimilarity 0 if they are exactly superposed. In addition, we will also consider the number of recon-figurations throughout the day, which should be close enough to the real one.

So we don't have a unique quantified measure of similarity between airspace configuration schedules for now: the influence of the smoothing parameter on the opening schedule is assessed by considering both the number of control sectors and the number of configuration changes.

But before looking how smoothing the complexity metrics may change the overall airspace configuration schedule, let us show some results on the influence of the smoothing parameter on the prediction of the sector status.

IV. INFLUENCE OF SMOOTHED METRICS ON SECTOR STATUS PREDICTION

The results of the 10-fold cross-validation with different values of the smoothing window are presented in tables I, II, and III.

Set	mean BIC	BIC std dev
REF	1.163	$2.7E - 2$
SM3	1.156	$3.0E - 02$
SM5	1.141	$2.9E - 02$
SM10	1.117	$2.4E - 02$
SM15	1.114	$2.4E - 02$
SM30	1.059	$2.6E - 02$
SM60	1.046	$3.5E - 02$

TABLE I
MEAN BIC VALUES AND STANDARD DEVIATIONS FOR THE
CROSS-VALIDATION

Table I shows the mean value and the standard deviation of the BIC criterion over the 10 runs of the cross-validation, for each candidate model. A somewhat surprising result is that SM60 (smoothing over 60 minutes) seems to provide the best results if we look only at the mean BIC. However, considering the standard deviation, it is not obvious that there is a true statistical difference between SM30 and SM60. Note also that the model with the lowest mean BIC is the one with the highest standard deviation.

Tables II and III show the mean correct classification rates and their standard deviations, over the ten runs, for all classes (Global) and also for each sector status class. Let us notice that the main improvement, when smoothing the input metrics, is made for the class corresponding to the normal domain of operation.

Let us now assess the generalization performance of the models. As explained in the previous section, the

¹⁶The correlation coefficient between two equal variables x and $y = x$ will be 1. Let us note however that this coefficient is not sufficient to actually measure how close we are to equality: the correlation coefficient between a variable x and another variable $y = x + d$, where d is a constant offset, will also be 1.

Set	Global	Merged	Normal	Split
REF	82.074%	88.353%	62.799%	90.322%
SM3	82.428%	88.451%	63.691%	90.587%
SM5	82.673%	88.545%	64.279%	90.75%
SM10	83.454%	89.401%	64.918%	91.548%
SM15	83.784%	89.365%	66.066%	91.699%
SM30	84.947%	89.910%	68.036%	93.147%
SM60	85.811%	90.698%	69.798%	93.242%

TABLE II
CORRECT CLASSIFICATION RATES

Set	Global	Merged	Normal	Split
REF	0.446%	0.616%	1.543%	0.572%
SM3	0.662%	0.586%	1.853%	0.641%
SM5	0.662%	0.852%	1.275%	0.556%
SM10	0.574%	0.714%	1.419%	0.384%
SM15	0.696%	0.771%	1.822%	0.636%
SM30	0.648%	0.568%	1.667%	0.794%
SM60	0.601%	1.091%	1.488%	0.794%

TABLE III
STANDARD DEVIATIONS OF THE CORRECT CLASSIFICATION RATES

neural network is trained again on the whole training set of patterns (instead of 9 sub-samples in the cross-validation). Ten runs were made with different random initial weights. The difference with cross-validation is that there are all made on the same training set.

Set	Training set		Test set	
	mean AIC	AIC std dev	mean AIC	AIC std dev
REF	0.765	$1.2E - 02$	0.781	$1.5E - 02$
SM3	0.750	$1.3E - 02$	0.757	$1.3E - 02$
SM5	0.743	$1.6E - 02$	0.751	$1.5E - 02$
SM10	0.733	$2.1E - 02$	0.744	$2.3E - 02$
SM15	0.710	$2.2E - 02$	0.727	$2.2E - 02$
SM30	0.681	$1.5E - 02$	0.697	$1.3E - 02$
SM60	0.644	$1.6E - 02$	0.662	$1.8E - 02$

TABLE IV
GENERALIZATION PERFORMANCE: MEAN AIC VALUES AND
STANDARD DEVIATIONS FOR THE TRAINING SET AND THE TEST SET

Table IV shows the mean AIC values and the standard deviation, over the 9 best results out of the 10 runs¹⁷, for each smoothing parameter. The neural networks performances on the training and test sets are quite close, with any smoothing parameter. All models seem to generalize well, and show little differences in that respect (see figure 3).

V. INFLUENCE OF SMOOTHED METRICS ON OPENING SCHEDULES

The last section was dedicated to the influence of the smoothing strategy on the performance of the sector status prediction. Now, let us see how it modifies the resulting opening schedules, comparing the different models on a same day and for a chosen air traffic centre (Brest ATCC, 2003, June 1st). The same algorithms

¹⁷In two cases, it happened that the choice of the random initial weights and the training process relying on a local optimization led to significantly less performing networks. So it was decided to remove the ten percent less performing networks from the results.

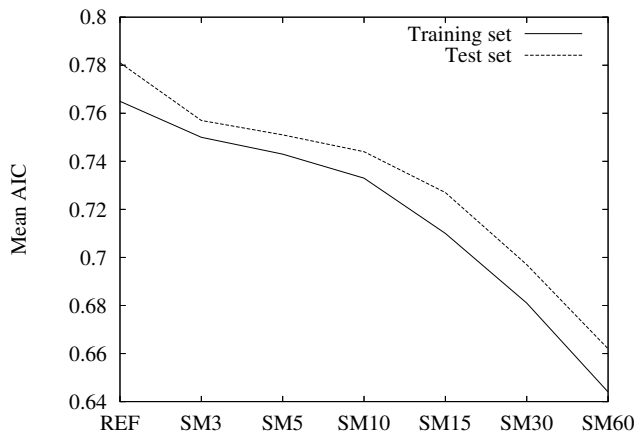


Fig. 3. Mean AIC for the training set and the test set.

were used for all models, with the same values for the split/merge decision parameters ($\eta = 0.2, \alpha = 0.1$, and $\beta = 0.3$).

Set	Correlation coeff.	Dissimilarity	Nb. config.
REF	0.9443	0.1169	101
SM3	0.9196	0.1489	202
SM5	0.9142	0.1510	179
SM10	0.9038	0.1783	125
SM15	0.9065	0.1681	140
SM30	0.9471	0.1094	34
SM60	0.9101	0.1426	23

TABLE V
CORRELATION COEFFICIENT AND NUMBER OF AIRSPACE CONFIGURATIONS FOR EACH MODEL

Table V shows the correlation coefficient, the dissimilarity measure, and the number of configurations for each model. All models show a good correlation, above 0.9 to the recorded number of control sectors. The number of reconfigurations is fairly high when smoothing on less than 15 minutes, showing a lot of "configuration switching", whereas SM30 and SM60 are much closer to the 28 airspace configurations that were actually used that day. Considering the dissimilarity measure and the number of configurations, SM30 seems to be the model that is most similar to reality. Let us now have a closer look at each computed schedule.

Figure 4 shows the reference situation, for Brest ATCC (2003, June 1st). The number of control sectors computed by our algorithm, using raw complexity metrics, can be compared to the actual number of control sectors that were opened, for each minute of this day. The evolution of the number of aircraft within the center is also displayed, above the two other curves. Let us remind that the number of aircraft is not sufficient to explain the number of control sectors, as other complexity metrics are also involved in the explanation of the sector status, and as the traffic load may not be equally dispatched among the sectors. It is still a good indication of the overall traffic load, however.

We may notice that, while the computed output stays globally close the recorded number of control sectors,

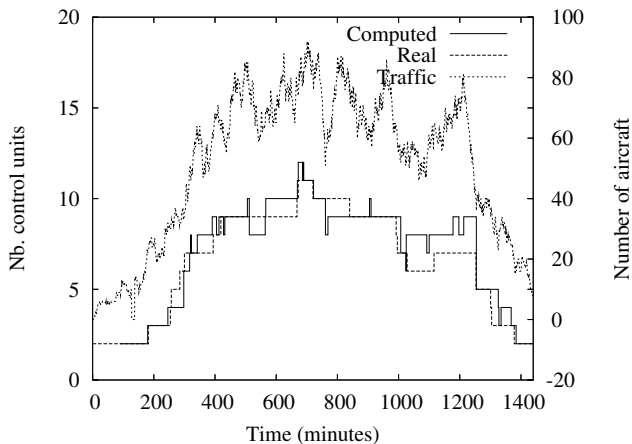


Fig. 4. Number of control sectors (computed schedule, real configurations), and traffic for Brest ATCC (2003, June 1st), with REF setup

it also shows many variations around the actual curve, more or less following the traffic trends on that day. Notice the peak of traffic around 20:00 UTC (1200 minutes after 00:00), where the curve of the computed schedule apparently better follows the traffic trend than the actual configurations (we shall see later that this depends on the chosen smoothing parameter).

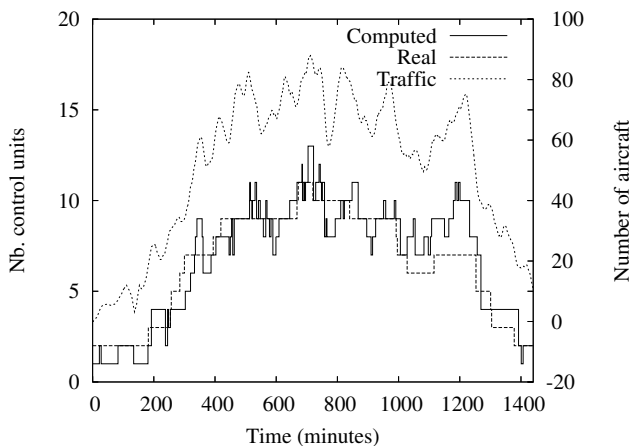


Fig. 5. Number of control sectors (computed schedule, real configurations), and traffic for Brest ATCC (2003, June 1st), with smoothed metrics (SM15)

Figures 5, 6 and 7 show the airspace schedule computed with smoothed metrics, using a smoothing window of 15, 30, or 60 minutes respectively. The traffic load's curve displayed on each figure shows the smoothed number of aircraft, using the smoothing window corresponding to each model.

At this point, when comparing figures 4, 5, 6, and 7, we may notice two phenomena which are not quantified by the measures of correlation and the number of reconfigurations. First of all, considering the peak of traffic around 20:00 UTC (1200 minutes after 00:00), we can see that the more you smooth the metrics, the less the computed number of control sectors reflects this peak of traffic. In fact, it becomes closer to the actual number of control sectors.

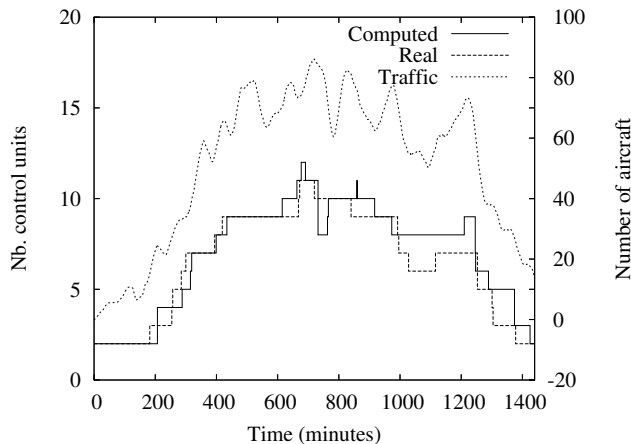


Fig. 6. Number of control sectors (computed schedule, real configurations), and traffic for Brest ATCC (2003, June 1st), with smoothed metrics (SM30)

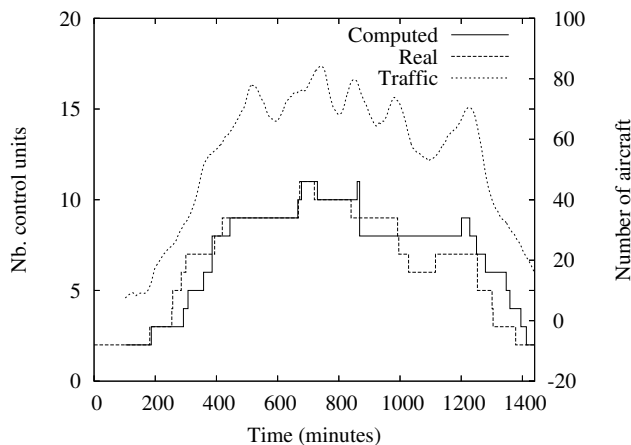


Fig. 7. Number of control sectors (computed schedule, real configurations), and traffic for Brest ATCC (2003, June 1st), with smoothed metrics (SM60)

The second conclusion that may be drawn from these figures is that smoothing the input metrics leads to delay the decisions to reconfigure the airspace. This is most visible on figure 7 (SM60) where the "climbing steps" corresponding to the split decisions in the morning and the "descending steps" of the merge decisions towards the end of the day are both on the right of the actual curve. In other words, the sector status prediction seems more performant *on average* when smoothing over 60 minutes¹⁸, but smoothing too much leads to take late split/merge decisions, thus delaying the moments at which the reconfigurations should be triggered.

All these experiments were made using the same decision parameters ($\eta = 0.2$, $\alpha = 0.1$, and $\beta = 0.3$) for all models. These parameters also have an influence on the moment at which reconfigurations are triggered. Some other parameter values were tried ($\eta = 0$, $\alpha = 0.5$, and $\beta = 0.5$), with the aim to improve the reactivity of the reconfiguration algorithm. For SM60, the reconfigurations were triggered slightly

¹⁸Although it was not such a clear-cut in deciding which of SM30 or SM60 was the best model, in table I

earlier but still the same phenomenon was observed, and the number of configurations increased to 38 configurations. Other trials were made, mixing metrics smoothed over 60 minutes and metrics smoothed over 10 minutes, with similar results.

So, smoothing the metrics over 15 minutes or less allows a higher reactivity to the traffic variations, but with much more reconfigurations than observed in real life. Among the models that were tested, SM30 (smoothing the input metrics over 30 minutes) seems the best compromise, considering the performance of the sector status prediction, but also the realism of the computed airspace configuration schedule. It seems to better capture the moments at which the reconfigurations are triggered, than when smoothing over 60 minutes.

VI. CONCLUSION AND PERSPECTIVES

The opening schedule computed with metrics smoothed over 30 minutes showed a number of reconfigurations close to reality, and with a number of control sectors well correlated to the actual configurations. It seems the best compromise among the models tested so far with the chosen neural network topology.

In a pre-tactical context, smoothing over relatively long periods of time may have positive consequences. The model should be more robust to uncertainties on aircraft trajectories when the complexity metrics will be computed from flight plans instead of past radar tracks.

In regard to the instant workload of a controller operating a sector at a time t , this smoothing strategy seems too drastic and may lead to miss the exact moments at which reconfigurations should be triggered, if this model was to be used for tactical purposes in a dynamic airspace management tool. An explanation is that only snapshots of the traffic situation – i.e. metrics values measured at time t – were used to predict the sector status. We may expect better results by considering the input metrics as time series, and by using recurrent neural networks instead of simple feed-forward networks. Provided this approach proves successful, the airspace configuration algorithms may prove useful for tactical purposes: flow managers may issue *what-if* requests and get some feedback on the resulting sectorization and workload balance among the control sectors.

Further works shall address both issues: improve the statistical model by using time series and recurrent networks to better capture the instant workload, and test the current model on simulated traffic, using flight plans as inputs, in order to predict the airspace opening schedule for the next day. Other smoothing strategies may also be tried, with different smoothing parameters for each metric for example, or with smoothing intervals centered on the current time.

REFERENCES

- [1] D. Gianazza and J. M. Alliot. Optimization of air traffic control sector configurations using tree search methods and genetic algorithms. In *Proceedings of the 21st Digital Avionics Systems Conference*, 2002.

- [2] D. Gianazza, J. M. Alliot, and G. Granger. Optimal combinations of air traffic control sectors using classical and stochastic methods. In *Proceedings of the 2002 International Conference on Artificial Intelligence*, 2002.
- [3] D. Gianazza. Airspace configuration using air traffic complexity metrics. In *7th USA/Europe Seminar on Air Traffic Management Research and Development*, 2007. best paper of "Dynamic Airspace Configuration" track.
- [4] D. Gianazza and K. Guittet. Evaluation of air traffic complexity metrics using neural networks and sector status. In *Proceedings of the 2nd International Conference on Research in Air Transportation*. ICRAT, 2006.
- [5] D. Gianazza and K. Guittet. Selection and evaluation of air traffic complexity metrics. In *Proceedings of the 25th Digital Avionics Systems Conference*. DASC, 2006.
- [6] C. Verlhac and S. Manchon. Improved configuration optimizer methodology to use a decision support tool. Technical note EEC Note No. 10/05, Eurocontrol Project NCD-F-FM, June 2005.
- [7] A. Klein. An efficient method for airspace analysis and partitioning based on equalized traffic mass. In *Proceedings of the 6th USA/Europe Air Traffic Management R & D Seminar*, 2005.
- [8] C.E. Bichot and N. Durand. A tool to design functional airspace blocks. In *7th USA/Europe Seminar on Air Traffic Management Research and Development*, 2007.
- [9] C. Verlhac and S. Manchon. Optimization of opening schemes. In *Proceedings of the fourth USA/Europe Air Traffic Management R&D Seminar*, 2001.
- [10] P. Flener, J. Pearson, M. Agren, C. Garcia-Avello, M. Celiktin, and S. Dissing. Air-traffic complexity resolution in multi-sector planning using constraint programming. In *7th USA/Europe Seminar on Air Traffic Management Research and Development*, 2007.
- [11] C. Verlhac, A. Schweitzer, E. Dumont, and S. Manchon. Improved configuration optimizer technical documentation. Technical note EEC Note No. 11/05, Eurocontrol Project NCD-F-FM, June 2005.
- [12] Brankica Pesic Le Foll. Network effect: A possible model to highlight interdependencies between flow management regulations. Master's thesis, Faculty of Transport and Traffic Engineering of Belgrade, 2006.
- [13] A. Klein, P. Kopardekar, M. D. Rodgers, and H. Kaing. Airspace playbook: Dynamic airspace reallocation coordinated with the national severe weather playbook. In *Proceedings of the 7th AIAA Aviation Technology, Integration and Operations Conference*, 2007.
- [14] Global air traffic management operational concept. Technical report, International Civil Aviation Organization, 2005.
- [15] H. Swenson, R. Barhydt, and M. Landis. Next Generation Air Transportation System (NGATS) Air Traffic Management (ATM)-Airspace Project. Technical report, National Aeronautics and Space Administration, 2006.
- [16] E. S. Stein, P. S. Della Rocco, and R. L. Sollenberger. Dynamic resectorization in air traffic control: A human factors perspective. Technical report DOT/FAA/TC-TN06/19, Atlantic City International Airport, NJ: Federal Aviation Administration William J. Hughes Technical Center, 2006.
- [17] Cognitive complexity in air traffic control, a literature review. Technical report, Eurocontrol experimental centre, 2004.
- [18] P. Kopardekar. Dynamic density: A review of proposed variables. FAA WJHTC internal document. overall conclusions and recommendations, Federal Aviation Administration, 2000.
- [19] P. Kopardekar and S. Magyarits. Measurement and prediction of dynamic density. In *Proceedings of the 5th USA/Europe Air Traffic Management R & D Seminar*, 2003.
- [20] A. J. Masalonis, M. B. Callahan, and C. R. Wanke. Dynamic density and complexity metrics for realtime traffic flow management. In *Proceedings of the 5th USA/Europe Air Traffic Management R & D Seminar*, 2003.
- [21] A. Majumdar, W. Y. Ochieng, G. McAuley, J.M. Lenzi, and C. Lepadetu. The factors affecting airspace capacity in europe: A framework methodology based on cross sectional time-series analysis using simulated controller workload data. In *Proceedings of the 6th USA/Europe Air Traffic Management R & D Seminar*, 2005.
- [22] G.B. Chatterji and B. Sridhar. Measures for air traffic controller workload prediction. In *Proceedings of the First AIAA Aircraft Technology, Integration, and Operations Forum*, 2001.
- [23] I. V. Laudeman, S. G. Shelden, R. Branstrom, and C. L. Brasil. Dynamic density: An air traffic management metric. Technical report, 1999.
- [24] J.H. Crump. Review of stress in air traffic control: Its measurement and effects. *Aviation, Space and Environmental Medicine*, 1979.
- [25] P. Averty, S. Athènes, C. Collet, and A. Dittmar. Evaluating a new index of mental workload in real ATC situation using psychological measures. Note CENA NR02-763, CENA, 2002.
- [26] A. Yousefi, G. L. Donohue, and K. M. Qureshi. Investigation of en route metrics for model validation and airspace design using the total airport and airspace modeler (TAAM). In *Proceedings of the fifth USA/Europe Air Traffic Management R&D Seminar*, 2003.
- [27] ACT-540 NAS Advanced Concepts Branch. An evaluation of dynamic density metrics using RAMS. Technical report (draft) DOT/FAA/CT-TN, Federal Aviation Administration, April 2001.
- [28] C. Mannings, S. Mill, C. Fox, E. Pfeleiderer, and H. Mogilka. The relationship between air traffic control events and measures of controller taskload and workload. In *Proceedings of the 4th Air Traffic Management Research & Development Seminar*, 2001.
- [29] D. Gianazza. *Optimisation des flux de trafic aérien*. PhD thesis, Institut National Polytechnique de Toulouse, 2004.
- [30] D. Delahaye and S. Puechmorel. Air traffic complexity: towards intrinsic metrics. In *Proceedings of the third USA/Europe Air Traffic Management R & D Seminar*, 2000.
- [31] P. Averty. Conflict perception by ATCS admits doubt but not inconsistency. In *Proceedings of the 6th Air Traffic Management Research & Development Seminar*, 2005.
- [32] M. I. Jordan and C. Bishop. *Neural Networks*. CRC Press, 1997.
- [33] C. M. Bishop. *Neural networks for pattern recognition*. Oxford University Press, 1996. ISBN: 0-198-53864-2.
- [34] B. D. Ripley. *Pattern recognition and neural networks*. Cambridge University Press, 1996. ISBN: 0-521-46086-7.

KEYWORDS

Air traffic complexity, airspace configuration, neural networks.

BIOGRAPHY

David Gianazza is currently researcher in the Planning, Optimization, and Modelling team¹⁹ of the DSNA/DTI R&D domain. He received his engineer's degrees (IEEAC in 1986, IAC in 1996) from the french civil aviation academy (ENAC) and his M.Sc. (1996) and Ph.D. (2004) in computer science from the "Institut National Polytechnique de Toulouse" (INPT). A more detailed biography may be found here: www.recherche.enac.fr/~gianazza/

¹⁹The POM team was formerly part of the Global Optimization Laboratory LOG CENA/ENAC