Airspace configuration using air traffic complexity metrics

David Gianazza

Airspace configuration using air traffic complexity metrics

David Gianazza, DSNA, Toulouse, France
Email : {lastname}@recherche.enac.fr

***Abstract*— Flow regulation is a critical process in air traffic management, ensuring that the incoming traffic does not exceed the ability of air traffic controllers to handle it safely and efficiently. Currently, the european Flow Management Positions (FMP) use flight counts and sector capacities to assess the traffic load and build predicted opening schemes. These schemes, made of pre-defined airspace configurations, are used to detect potential overloads. Some past research undertaken at the Global Optimization Laboratory led to think that this process was not grounded on solid scientific notions, as concerns the quantification of the controllers workload. Consequently, it is proposed to stop using flight counts and sector capacities to predict this workload, and to use relevant air traffic complexity metrics instead. Another proposal is to explore all the possible combinations of elementary sectors, instead of the small subset of pre-defined configurations currently being used, so as to offer the maximum capacity to the incoming traffic.**

In previous works, we assessed the relevance of complexity metrics by comparing their relative influence on the sector status prediction (*merged*, *manned*[1], or *split*) made by a neural network. Real sector statuses issued from filed configurations were used to train the neural network[2]. A fairly simple relationship between the relevant metrics and the sector status was found. The main contribution of this paper is to use the relevant metrics and the sector status prediction to build realistic airspace configurations. The computed configurations are compared to the actual configurations archived by the ATC centers, and to the FMP opening schemes.

INTRODUCTION

Over the last years and with the increasing traffic demand, there has been a growing need for a better understanding of how the current air traffic management system is operated and how it could be improved. Quantifying the complexity of a given air traffic situation has become an issue for the assessment of ATC/ATFM performances, the benchmarking of new airspace structures, the design of new control tools, and other activities related to air traffic management. This paper focuses on the airspace configuration schedules made by the Flow Management Positions (FMP) of each Air Traffic Control center (ATCC), and sent to the european Central Flow Management Unit (CFMU) two or three days before the traffic actually enters the centre's airspace.

The work presented here is the continuation of previous research. In [1], [2], and [3], we proposed several algorithms combining elementary sectors so as to offer the maximum capacity while making the best possible use of the available ressources. These algorithms used the same variables (flight counts in a period of time) and thresholds (sector capacities), as well as the same constraints (number of controllers on the duty schedule) than the FMP. However, the optimized schedules were still far from realistic when compared to real configurations. In [4] and [5], we used the real sector status (*merged*, *manned*[3], or *split*) to assess the relevance of several air traffic complexity metrics found in the litterature, assuming that the decision to split or merge sectors is somewhat related to the actual contoller's workload,. Neural networks were used for that purpose.

We will now see how the relevant air traffic complexity metrics and the sector status prediction made by the neural network can be used to produce more realistic schedules. In order to be closer to the real decision process than in our previous works ([1],[2]), the computed airspace configuration will be dynamically re-assessed every minute of the day.

If we were to implement this process within an existing decision support tool, we would have to compute the complexity metrics from the flight plans filed for the considered day, as the radar tracks are not yet available, two or three days before D-day. This issue is not adressed here and will be a further step of our research. In this paper, the metrics are computed from recorded radar tracks, in order to compare the computed configurations to the real configurations recorded by the ATCC. This is a first and necessary step in the validation of realistic airspace configurations schedules.

The current paper is organized as follows. Section I defines some vocabulary and gives the context and the motivation of our study, as well as some background on air traffic complexity. Section II describes the raw input data used to compute the air traffic complexity metrics. The neural network that predicts the sector status using the metrics values as input is shortly described in section III. Section IV introduces the algorithm building realistic airspace configuration schedules, according to the sector status predictions. Some results on the french air traffic centres are provided in section V, and section VI concludes the paper and gives the perspectives of future

[1]A control sector may be *manned* (or operated, or armed) by a team of two controllers when the workload is normal, or *split* , when this is possible, into smaller sectors operated separately if the traffic is high, or *merged* (or collapsed) with other sectors to form a larger sector when the traffic is low.

[2]Irrelevant 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.

[3]From now on, we will prefer the term *manned* instead of *armed* that was used in previous works, to denote that the sector is in its normal domain of operation.

research.

I. Background

A. Vocabulary

Before all, let us define a few terms of vocabulary used in this paper. The airspace of each center is made of several *elementary sectors* that may be merged (or collapsed) into larger sectors. In the rest of the paper, the term *control sector*, or simply *sector*, is used to denote a volume of airspace made either of one elementary sector or several elementary sectors merged together, and controlled by a team of usually two controllers (radar, and planning).

A configuration is a mapping of the set of elementary sectors onto a smaller number of control units, operated each by a radar and a planning controller. An airspace configuration schedule, also called opening scheme, shows the successive configurations for each time of the day.

We call *incoming flow* the number of aircraft entering the control sector within a period of time (30 or 60 minutes, for example). The incoming flows are sometimes called flight counts ([6]), or traffic load in the CFMU handbook. The *sector capacity* is a threshold value on the number of flights that may enter the sector in a chosen period of time.

B. Context and motivation of the study

The current method used by the european FMP/CFMU to build airspace configuration schedules, or opening schemes, is fairly simple. A set of static airspace configurations defined by the FMP staff is filed in a database. When building a schedule, the FMP operator chooses for each time step (usually 30 or 60 minutes) the best configuration among those he or she thinks may be the most adequate for this time of the day. The subset of possible configurations at each time step is chosen according to the number of controllers listed on the duty schedule at that time of the day. Each configuration is evaluated by comparing the incoming flows to the sector capacities. This choice of the best configuration is subjective and based on the past experience of the operator. Of course, once the final schedule is built, there may remain some sectors for which the predicted incoming flow is higher than the capacity. This may be considered as a potential overload problem and signaled to the CFMU. Some regulations may be requested for one or several traffic flows crossing some sector boundaries or overflying some beacons[4].

The Eurocontrol NCD (Network Capacity and Demand management) Research Area team has been working for several years on the optimization of this process (see [7],[6], [8]), with the aim to propose new decision support tools for the flow management operators in a short-term future. The proposed methodology is strongly grounded on the current working method: it relies very much on the operators experience, and also uses predefined configurations as well as incoming flows and sector capacities (see [6], p. 12) to assess when a control sector is overloaded. One of the main concerns in this work seems to be the network effect affecting the set of regulations ([9]), and how it is connected with the building of opening schemes in several centres belonging to a same area of interest.

Our research, which started at the Global Optimization Laboratory (LOG[5] CENA/ENAC) some years ago, is a fairly different approach. We were initially interested in how to combine elementary sectors so as to offer a maximum capacity for a given trafic demand. To do that, we chose to explore the whole space of possible configurations that can be built from the set of elementary sectors, rather than the relatively small subset of predefined configurations that are currently being used by the FMPs. In [1] and [2], we proposed several algorithms (tree search methods and stochastic methods) to compute optimal airspace configurations, using the same variables and thresholds that are used in the operational field (incoming flows and sector capacities).

In [2], the potential profits provided by the optimized schedule showed a decrease of 69% of the cumulated delays, while using 20% less ressources, considering only one air traffic centre on one day of traffic. These results – too good to be true – together with the simple comparison of the registered schedule actually sent by the FMP to the CFMU with the real airspace configurations recorded the same day, led us to think that the incoming flows and sector capacities were really inadequate, or at least not sufficient for our purpose. In other words, the traffic load expressed as the number of flights entering the sector in the next M minutes may be useful for smoothing the traffic flows, but it is not enough to assess the actual controller's workload operating the sector. Strangely, this well-known fact seems completely forgotten in the current working method used to build opening schemes.

A general conclusion that can be drawn from these past studies is that the current traffic regulation process *is not grounded on solid scientific notions*, but highly depends on the operators experience (and of the feedback of experience from past situations): you know how to smooth the traffic using flow regulations, but you are unable to actually quantify the level at which it should be tuned. So, in order to build more realistic airspace configuration schedules, we propose to *stop using the incoming flows and sector capacities to assess the controllers workload* and to use relevant metrics instead. In order to be actually related to the controller's workload, such metrics should take account of the air traffic complexity within the sector.

C. Air traffic complexity

When considering air traffic complexity, we must be aware that it may be quantified in different ways, depending on the time and geographic scale involved, and on the

[4] such regulations are called "traffic-volume" in some Eurocontrol documentations ([6])

[5] http://www.recherche.enac.fr/opti/

intended application. It is one thing to compare several service providers over a year of traffic, taking account of aggregated complexity factors, and a very different thing to evaluate the controller's workload every few seconds, taking account of the traffic's complexity within the control sector. So there is no single universal complexity measure and all we may expect is a set of complexity metrics, useful and relevant only in a particular context and for a particular purpose. As a consequence, it is crucial to assess the relevance of the chosen metrics to the final goal.

In our study, the chosen time scale is one measure every minute, the geographic area is the set of control sectors usually operated in an ATC center, and the intended application is the airspace configuration process, with the final aim to find a realistic prediction of the opening scheme. The actual configuration of the control sectors depends on the controllers workload: overloads may lead to a decision to split a sector, and underloads lead to "merge" decitions. As we would rather avoid using (and tuning) a multitude of sector-specific parameters (like capacity values), we need to know how the actual controllers workload is related to the complexity of the sectors they are operating.

Several studies tried to find how the controller's workload is related to air traffic complexity. To do this, one usually tries to maximize the correlation of a set of complexity metrics with a quantifiable variable (called *dependant variable*) assumed to represent the actual controller's workload. Various methods have been used: linear ([10]) or logistic ([11]) regression, cross-sectional time series analysis ([12]), neural networks ([13]),... A multitude of metrics have been proposed in the literature. The reader may refer to [14] and [15] for a review.

Whatever method is used, the choice of the dependent variable is crucial to determine how well complexity is actually measured. Many ways to quantify the controller's workload have been used in past studies: physical activity ([16], [12]), physiological indicators ([17], [18]), simulation models of the controller's tasks ([19], [20]), subjective ratings ([10], [13], [11]). 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 biases[6], 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 in [4] and [5], 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 [3], 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, manned, or split) is directly related to the controller's workload and may therefore provide an acceptable dependent variable. This data may also be biased or noisy, as we may not be sure that the splitting and merging decisions are always directly related to the workload. Other factors might distort data, such as the training of unexperienced controllers, technical incidents, real-traffic experimentations... However, we assume that the impact of these phenomena on the accuracy of the results is limited particularly because of the kind of complexity we are looking at here[7]. Moreover, we selected our data in a period of high traffic, so we may expect that most configuration changes were caused by the controllers workload.

The selection of the relevant metrics was done in collaboration with the Laboratory of Economy and Econometry of Air transport (LEEA[8]), within the S2D2 project[9]. In [4] and [5] we proposed an original method to assess the metrics relevance, using neural networks and sector statuses obtained from the actual airspace configurations recorded by each ATC center. This method allowed us to select a subset of only 6 relevant indicators among the initial 28 chosen from [10], [13], [25], [26] and other sources.

As a result of [5], we also obtained a fairly simple equation (the trained neural network) allowing to predict the sector status from the six relevant metrics, which 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*). We now propose to use these metrics and the neural network's prediction to build a realistic schedule of the airspace configurations operated in the ATC centre's control room.

II. Input data

The relevant metrics are computed every round minute of the day, using recorded radar data and environnement data (sector description) from the french ATC centers. Radar data is available in several forms: records made by each center, with one position every twelve seconds, in average, and a global record of the five french centers, with one position every three minutes. Several months of global records were available, whereas the centers local records were not readily available, at least for a sufficiently long period of time. So we used the global records (made by the IMAGE system), and interpolated the aircraft positions in order to get one position per minute. As many trajectory changes may occur within three minutes of flight, the computed positions are not highly accurate, and this may introduce a bias in the indicators values. However, this bias is most probably of small importance in our problem: we just want to

[6] such as the subjective ratings recency effect denounced in [18], or raters errors in the case of "over-the shoulder workload ratings" [21]

[7] Pre-tactical applications do not ask for as much details as studies of instantaneous workload would. On the opposite, benchmarking of ATC centers would require an even coarser granularity, as indicators are averaged on wide temporal and geographical horizons ([22], [23],[24])

[8] http://www.enac.fr/recherche/leea

[9] http://www.recherche.enac.fr/opti/S2D2/

predict when a sector will be merged into another one, or split in several smaller sectors. We are not considering the instant workload, which may require a higher level of accuracy on the aircraft positions, speed, and so on.

Several months of recorded traffic are available. We have restricted our choice, at least for the moment, to two days of traffic. We chose the 1st June 2003, that was previously used in [4] and [5] to select the relevant metrics and train and test the neural network providing the sector status prediction. We also tried another day (June 2nd, 2003) that was not previously used.

Airspace configurations are built from elementary sectors that may be merged together into larger control sectors. So an airspace configuration is a partition of the set of elementary sectors. However, these partitions should only contain control sectors that are operationally valid. For example, a control sector should generally not be made of two sectors that are not neighbours. Once again, we referred to the ATC centers databases, which provide the list of elementary sectors of each center, and also the groups of sectors that can be merged together into a control sector.

III. Sector status prediction

Artificial neural networks are used to predict the sector status from the relevant complexity metrics. In previous works ([4], [5]), we trained such networks on real sector statuses in order to assess the relevance of a set of complexity metrics.

In the work presented here, we use the trained network and the most relevant metrics to predict the sector status on fresh input data. This allows us to decide when a given control sector is normally loaded, overloaded, or under-loaded, and when an airspace configuration should be recombined.

The reader unfamiliar with neural networks may refer to the the fairly wide literature on the subject. Let us just say that an artificial neural network is an algorithm inspired from the biological neurons and synaptic links, that may be represented as a graph with vertices (neurons, or units) and edges (connections) between vertices. There are many types of such networks, associated to a wide range of applications: pattern recognition (see [27] and [28]), control theory,...

Beyond the similarities with the biological model, an artificial neural network may be viewed as a statistical processor, making probabilistic assumptions about data ([29]). Some *train* data 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.

For our application, we use a fully connected feed-forward neural network with three unit layers (or two weight layers), as shown on figure 1. For details on how this network was trained and tested, the reader may refer to [5]. We will just summarize its main features and give the equation of the relationship between the input and the output of the network.

The input variables are the relevant complexity metrics $\{V, Nb, avg\_vs, F_{60}, F_{15}, inter\_hori\}$, normalized

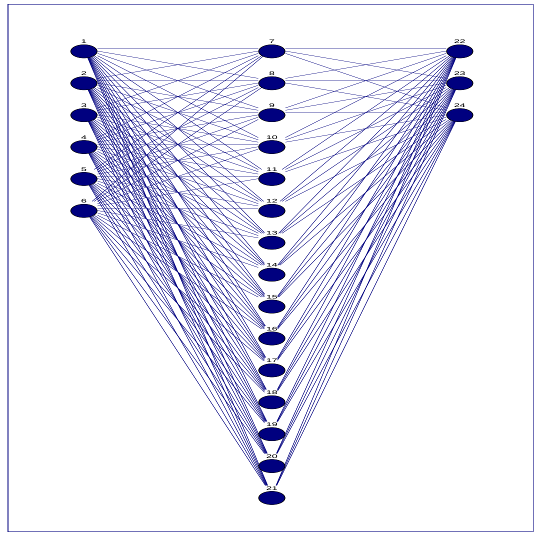

Fig. 1. Feed-forward network for sector status prediction

by substracting the mean value and dividing by the standard deviation. Let us denote $x = (x_1, ..., x_i, ..., x_6)^T)$ this input vector.

The neural network's equation is written as follows:

$$(y_1, y_2, y_3)^T = \Psi(\sum_{j=1}^{15} w_{jk} \Phi(\sum_{i=1}^{6} w_{ij} x_i + w_{0j}) + w_{0k}) \tag{1}$$

where the output vector $y = (y_1, y_2, y_3)^T$ is a triple of posterior probabilities. This means that y_1 is the probability to fall in the *merged* class when the input vector is $(x_1, ..., x_6)^T$, and the same for y_2 and y_3, with the *manned* (i.e. normal domain of operation) and *split* classes respectively. Ψ is the softmax function:

$$\Psi(z_k) = \frac{e^{z_k}}{\sum_{m=1}^{3} e^{z_m}} \tag{2}$$

Φ is the sigmoïd logistic function:

$$\Phi(z) = \frac{1}{1 + e^{-z}} \tag{3}$$

The adjusted weights w of the neural network are provided in the appendix[10], together with some R code allowing to make the prediction.

It must be underlined that the output of the neural network is a triple of values that can be considered as probabilities, with each output y_k comprised between 0 and 1. To emphasize this fact, we will denote $(p_{merge}, p_{man}, p_{split})$ the output of the neural network in the next section.

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

[10] although not with the full range of decimals, due to presentation constraints.

we are necessarily in one of the above three cases, the sum of the three probabilities y_1, y_2, and y_3 is always 1.

Let us note that, although these probabilities can be used to determine if a sector is normally loaded, overloaded, or under-loaded, they are not actually a numeric quantification of the workload.

IV. Building airspace configurations

A. General description of the process

Our aim is to compute at each time step of the day a realistic airspace configuration, so that each control sector has the highest possible probability to be in its normal domain of operation.

To do so, we start at time t=0 with a configuration where all elementary sectors are assigned to a single controller's working position. The situation is then reconsidered every minute of the day. The current configuration is re-assessed by considering the status prediction of each control sector.

Some of these sectors will remain unchanged: those for which the status is *manned* with a high probability. The others – which need either to be split or merged according to the neural network's prediction – will be entirely recombined as follows. We build every possible partition[11] from the elementary sectors composing these sectors. The best partition is then chosen (we shall see later what criteria are used to assess the partitions), and it replaces the initial set of sectors that needed to be recombined.

Let us now give a few details, first on the decision criteria which allows to decide when the airspace configuration must be changed, second on the algorithm used to explore all the combinations of sectors that need to be reconfigured, and third on the evaluation criterion used to select the best configuration.

B. Decision criteria

The decision to reconfigure sectors is driven by the prediction made by the neural network. The neural network's output is a triple $(p_{merge}, p_{man}, p_{split})$ of probabilities to belong to one of the classes (*merged*, *manned*, or *split*), with $p_{merge} + p_{man} + p_{split} = 1$ (see section III).

The most straightforward decision criterion is to choose the class with the highest probability. Let us call $D1$ this decision criterion. This criterion may lead to many sectors reconfigurations: the highest probability may be only slightly higher than the others, and for short periods of time.

The actual sector configurations exhibit few fluctations in time. So we tried another decision criterion $D2$, considering how close the highest probability is from 1, and also the difference between the two highest probabilities. For example, a triple of probabilities $(0.1, 0.1, 0.8)$ is a stronger incitation to split the sector than $(0.1, 0.44, 0.46)$.

[11] A partition is a mapping of the elementary sectors on a number of controller's working positions.

Let us consider our three probabilities sorted by decreasing values p_1, p_2, p_3. The new decision criterion, that we will call $D2$, is expressed as follows, depending on which probability is the highest:

- if p_1 is the probability p_{merge} to merge the sector: the decision to recombine the sector is taken only if $1 - p_1 < \alpha$ and $p_1 - p_2 > \eta$, where α and η are chosen parameters.
- if p_1 is the probability p_{split} to split the sector: we decide to split only if $1 - p_1 < \beta$, where β is a chosen parameter.

In the other cases, the sector remains unchanged. We choose different criteria for the 'split' and 'merge' decisions, because we may need to be more reactive when the workload is increasing than when it is dropping down.

The decision criteria will provide us with a list of sectors that need to be recombined. The decision criterion $D2$, as described above, is bound to find a smaller list of sectors than $D1$, which may reduce the choices when recombining the sectors. A typical example is when only one sector needs to be merged, or several sectors that are not neighbours. So we will use $D2$ to decide when a configuration must be recombined, and use $D1$ to build the larger list of sectors that will be recombined.

The algorithm described in the next section will use this list as input, by building the subset of elementary sectors from which the initial control sectors are made, in order to compute all the possible partitions from these sectors. Let us now describe how this is done.

C. Searching sector configurations

The algorithm used to build all the valid sector configurations from an initial set of elementary sectors is a classical tree search algorithm. Figure 2 shows how to build all the partitions of a set of 5 elements: starting with element 1 in a single group, we have the choice to add the second element in the same group, or create a new group, and so on until all elements have been considered.

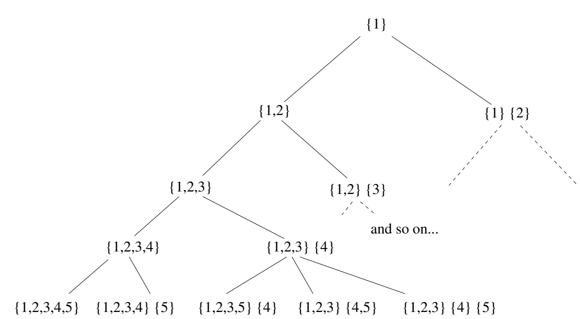

Fig. 2. Searching the partitions of a set of 5 elements

We can use this algorithm to build all the partitions of a set of sectors. However, many of these partitions contain groups of sectors which are not valid, from the operational point of view. For example, it is most unlikely that a controller will handle a group of sectors which does not form a contiguous portion of airspace.

Hopefully, the airspace description found in the ATCC databases provides us, together with the list of elementary sectors, with the list of operationally valid groups of sectors. We just need to introduce this restriction in our tree search algorithm, as shown in figure 3.

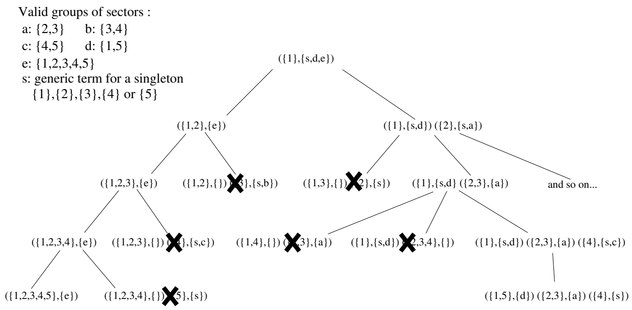

Fig. 3. Searching all valid airspace configurations

In this tree search, each node is a list of couples. The first item of a couple is a subset of sectors, and the second is the list of valid groups containing this subset, but which contain no sector from the other subsets of the node. When a list of valid groups is found to be empty, the branch is not explored: it will never lead to a valid configuration. The leaves of the tree are partitions made only of valid groups.

Let us now consider an hypothetic ATC centre of 20 elementary sectors for example. Let us imagine a situation at a time t where the airspace is configured in 4 large sectors, and with an increasing workload requiring that 3 of them be split in smaller sectors. Let us say that the 3 sectors are made of a combination of 15 elementary sectors. However steep the workoad's increase is, it is most unlikely in real life that the traffic will require to man all the 15 elementary sectors on separate control units at time $t+1$. Real traffic doesn't exhibit such drastic variations in one minute.

So it may be useless to compute all the partitions. We may focus on the partitions for which the number of groups does not exceed a chosen maximum. Typically, if p is the number of control sectors that need to be split, and q the number of sectors that need to be merged according to our decision criterion, we will search all the partitions comprising at most $2p + q + 1$ groups. This will avoid to explore all possible partitions, without compromising our aim to find a realistic configuration.

D. Evaluation criterion

We have shown how to compute valid sector partitions that will replace the initial sectors that need to be split or merged. The tree search algorithm explores all possible airspace configurations, among which we must choose only one. Let us define the evaluation criterion used to select the best configuration.

Ideally, a good configuration is one for which all sectors (groups of elementary sectors) will have a probability to be in the *manned* status as close as possible to 1, according to the neural network's prediction.

However, we must also be able to evaluate configurations with overloaded sectors, for which the neural network issues a *split* recommendation, as well as underloaded sectors (*merge* recommendation).

Our evaluation criterion should be designed so as to penalize overloads more than underloads, which should themselves be more penalized than configurations with only normally loaded sectors.

When comparing two configurations with underloads or two configurations with normal workoad, we would rather choose the one with the less sectors, if they are otherwise equivalent. So the number of control sectors should also be included in the evaluation criterion.

Considering *config(t)* the configuration of the airspace at time t, let us call $p_{merge}(x,t)$, $p_{man}(x,t)$, and $p_{split}(x,t)$ the probabilities computed by the neural network for each sector x of the configuration, and $N_{sect}(x)$ the number of elementary sectors grouped together to form the sector x.

Let us define $\delta_{merge}(x,t)$ the function such that $\delta_{merge}(x,t) = 1$ if $p_{merge}(x,t)$ is higher than the two other probabilities, and is equal to 0 otherwise. Two other functions $\delta_{man}(x,t)$ and $\delta_{split}(x,t)$ are similarly defined. In our evaluation criterion, we shall use:

- $N_{units}(t)$ the number of airspace units in the configuration at time t,
- $C_+(t) = \sum_{x \in config} \delta_{split}(x,t).p_{split}(x,t).N_{sect}(x)$,
- $C(t) = \sum_{x \in config} \delta_{man}(x,t).(1 - p_{man}(x,t))$,
- $C_-(t) = \sum_{x \in config} \delta_{merge}(x,t).p_{merge}(x,t).N_{sect}(x)$.

We could have assigned a cost to the overloads, normal loads, and underloads, simply by cumulating the values of the probabilities in each case. For example, we could have chosen a cost $\sum_{x \in config} \delta_{split}(x,t).p_{split}(x,t)$ for the overloads, and similar costs for the under-loads and normal loads. With such cost functions, however, a highly overloaded configuration with only one sector[12] may well have a cost close to 1, as the probability $p_{split}(x,t)$ is likely to be high, while always remaining between 0 and 1. Considering a configuration where we split this single sector into two sectors, and supposing that these two sectors are still highly overloaded, the cost of such a configuration is likely to be close to 2, that is to say more than for the previous configuration.

This undesirable feature, due to the fact that we use status probabilities, and not actually a numeric quantification of the workload, is avoided by introducing the number of elementary sectors in the costs, so that an overloaded configuration with few control units controlling big sectors will have a higher cost than configurations with more control units with sectors of smaller sizes.

Finally, the evaluation criterion used to select the best

[12]All elementary sectors merged as a single control unit.

configuration is:

$$eval_{config} = \begin{array}{l} 10^{k_2+k_3+k_4} \times N(k_1, C_+(t)) \\ +10^{k_3+k_4} \times N(k_2, N_{units}(t)) \\ +10^{k_4} \times N(k_3, C_-(t)) \\ +N(k_4, C(t)) \end{array} \quad (4)$$

where N is a function such that

$$N(k,c) = \lfloor max(0, (10^k - 1) - c) \rfloor$$

and the exponents k_1, k_2, k_3, and k_4 are chosen so that the term $(10^k - 1)$ is an upper bound of the corresponding cost c.

This criterion is designed so that the k_1 digits of higher order represent the cost of overloads, the k_2 next digits are related to the number of units in the configurations, the k_3 next digits are assigned to the underloads cost, and the k_4 digits of lower order are related to the cost of normally loaded configurations.

V. Results

Let us now present some results, and compare the configuration schedule computed with the complexity metrics to the real configurations recorded by the ATCC on the same day, and also to the FMP opening scheme.

Note that the comparison with the FMP opening scheme is not entirely fair: our configuration schedule is computed using recorded radar tracks, whereas the FMP scheme was built using the filed flight plans available two or three days before D-day. Still, it gives a good hint of the potential benefits of the method we propose, and the comparison with the real configurations remains fair: the actual merge/split decisions were taken with the same radar data as input.

A. A full example

As a first example of the results provided by the algorithms proposed in section IV, let us consider Brest Air Traffic Control Centre, on the 1st june, 2003. The decision criterion is $D2$, with the following parameters values: $\alpha = 0.5$, $\eta = 0.2$, and $\beta = 0.5$.

Figure 4 shows an example of the final ouput of our algorithm. The time t is given in minutes after 00h00. Only a short period of time is presented, roughly between 04h20 and 07h00. The airspace configurations actually used on the same day for the same period are shown on figure 5.

```
... t= 258 [AOUS FZX RNG RQJ]
t= 261 [AOUS RFX RNG RQJ Z]
t= 265 [AOUS FZX RNG RQJ]
t= 266 [AOUS RFX RNG RQJ Z]
t= 281 [AOUS J Q RFX RNG Z]
t= 283 [AOUS FBRT J Q RNG RZX]
t= 304 [AOUS FBRT J Q RNG X Z]
t= 317 [AOUS J Q RFX RNG Z]
t= 324 [A FBRT J O Q RNG RZX]
t= 328 [A FBRT J O Q RNG X Z]
t= 332 [AOUS FBRT J Q RNG X Z]
t= 335 [AOUS FBRT G J N Q X Z]
t= 337 [AOUS J Q RFX RNG Z]
t= 338 [AOUS FBRT G J N QS QU X Z]
t= 342 [AOUS FBRT J QS QU RNG X Z]
t= 343 [AOUS J QS QU RFX RNG Z]
t= 344 [AOUS FBRT J QS QU RNG X Z]
t= 347 [AOUS FBRT J QS QU RNG ZXIU ZXS]
t= 354 [AOUS FBRT J Q RNG ZXIU ZXS]
t= 356 [AOUS FBRT J Q RNG XIU ZIU ZXS]
t= 360 [AOUS FBRT J Q RNG X ZIU ZS]
t= 363 [AOUS FBRT J Q RNG X Z]
t= 370 [A FBRT J O Q RNG X Z]
t= 381 [AOUS FBRT J Q RNG X Z]
t= 394 [AOUS FBRT J Q RNG X ZIU ZS]
t= 409 [A FBRT J O Q RNG X ZIU ZS]
t= 426 [AOUS FBRT J Q RNG X ZIU ZS
... and so on
```

Fig. 4. Example of computed airspace configurations for Brest ATCC (2003, June 1st)

```
t= 254 [NGA ROQ RFJ RZX]
t= 257 [NGA ROQ J FBRT RZX]
t= 285 [N RGA ROQ J FBRT RZX]
t= 300 [N RGA ROQ J FBRT ZXU ZXSI]
t= 394 [N RGA ROQ J FBRT ZXU ZXI ZXS]
t= 419 [N A G ROQ J FBRT ZXU ZXI ZXS]
```

Fig. 5. Example of real airspace configurations for Brest ATCC (2003, June 1st)

When comparing these two schedules, we observe some differences in the choice of the control sectors composing the configurations. This is not a surprise. For a same traffic situation, there are often several ways to split the airspace in order to balance the workload among the control sectors. And in a same situation, different control room's managers may make different arbitrary choices. So it is not surprising that we do not find exactly the same configurations.

From the author's past experience in Brest ATCC, the configurations of the computed schedule seem realistic, although this would have to be confirmed by experienced FMP operators.

At a more macroscopic level, we may compare the number of control units in the computed schedule and in the real configurations. It is also interesting to compare these two schedules with the FMP opening scheme made two days before D-day. Figure 6 shows the evolution of the number of control units for the three configuration schedules. The horizontal axis shows the time, expressed in minutes after 00h00, and the vertical axis the number of control units.

On this figure, we see that the prediction using complexity metrics shows more variations in time than the real configurations. The reason is that, with our algorithms, some overloads or under-loads lasting only a few minutes will lead to configuration changes, whereas in the real world, they are disregarded and the resulting curve is smoother. If we go a little further in the explanation, it may be that, apart from the incoming flows which use an anticipation window of 15 or 60 minutes, most of the metrics used as input of our neural network give only a snapshot of the situation at time t. These metrics are subject to high variations in time, which may account for the many changes we observe in the number of configurations. It may have been useful to smooth the metrics values on a chosen time window

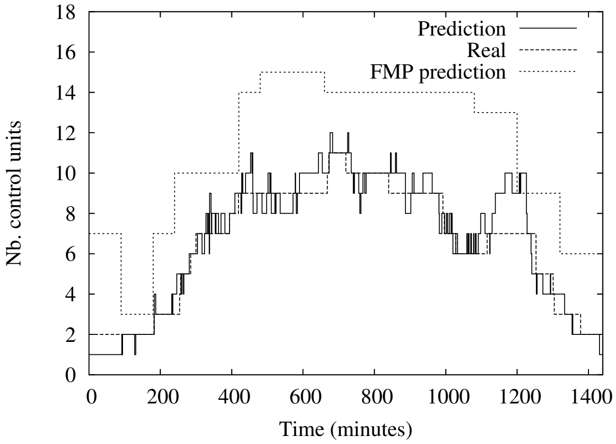

Fig. 6. Number of control sectors (computed schedule, real configurations, and FMP opening scheme) for Brest ATCC (2003, June 1st), with decision criterion $D2$

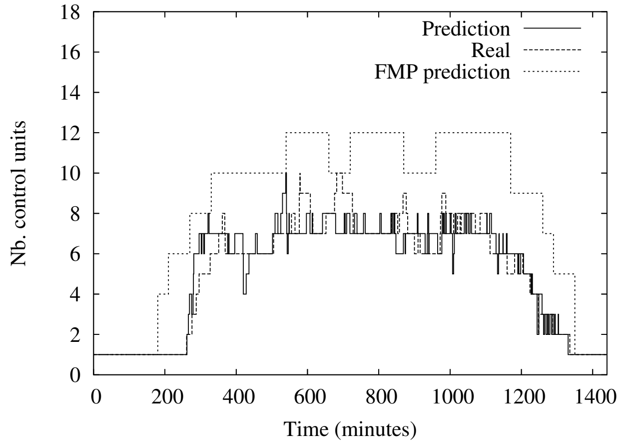

Fig. 7. Number of control sectors (computed schedule, real configurations, and FMP opening scheme) for Reims ATCC (2003, June 2nd), with decision criterion $D1$

before using them as input, or to use the anticipated values of these metrics, predicted for a short period ahead the current time t.

Another and more simple way to get a more realistic airspace configuration schedule would be to introduce an anticipation window in the process described in IV. This would allow to detect and discard the configuration changes that last only a few minutes.

Despite these variations, our prediction stays fairly close to the real configurations. There is however an exception between 1110 minutes (18h30) and 1240 minutes (20h40), where our computation shows between two and three more control sectors than what was actually manned. The detailed outputs of our algorithms showed several consecutive configurations where two or three non-adjacent sectors (Q, ZX, and GS for example) were given a "merge" decision by the neural network, and the others had a "no change" decision. The list of sectors to recombine was only made of sectors that were not neighbours, so it was not possible to merge them, and they remained opened.

We tried to avoid this problem when designing the decision criterion $D2$ but it seems this was not enough, and we will have to improve our algorithm by extending the recombination to neighbouring sectors, even if it was not initially decided to merge or split them.

Although there remains a few points that need to be improved, the results shown on figure 6 are already fairly good, especially if we compare with the FMP opening scheme. Of course, in order to be in the same context as the FMP, we should use the flight plans of the final traffic demand as input instead of the recorded radar tracks. We could then check if our prediction would remain so close to reality. The work presented here was only intended to show that the use of complexity metrics allows to simulate how a control's room is operated, with a high degree of realism. This is a necessary step towards realistic predictions that could be used in the FMP/CFMU flow regulation process.

B. Influence of the decision criterion

Figure 7 shows the evolution of the number of control units for Reims ATCC, on june the 2nd, 2003, using the decision criterion $D1$.

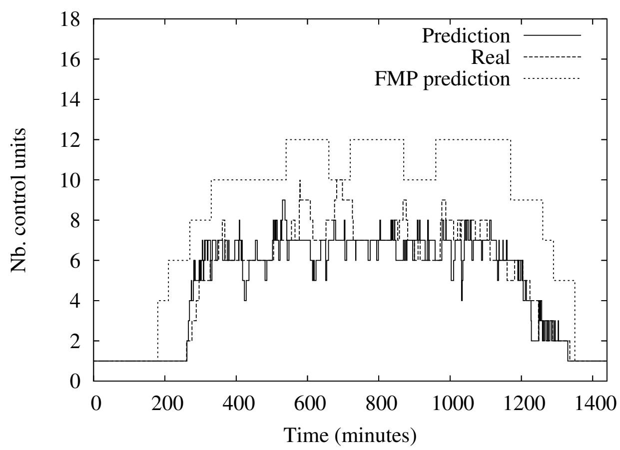

Fig. 8. Number of control sectors (computed schedule, real configurations, and FMP opening scheme) for Reims ATCC (2003, June 2nd), with decision criterion $D2$ ($\alpha = 0.5$, $\eta = 0.2$, and $\beta = 0.5$)

Figure 8 shows the results for the same day and the same center, but using criterion $D2$ with $\alpha = 0.5$, $\eta = 0.2$, and $\beta = 0.5$. The curve of the computed configurations is slightly different from figure 7. This is not surprising, as the sequence of merge/split decisions is different. However, the curve still exhibits many variations. The use of criterion $D2$ with the chosen values of α, η and β does not succeed in smoothing the variations in the number of control sectors.

We tried different values of these parameters. Figure 9 shows the best compromise that we found, with $\alpha = 0.10$, $\eta = 0.2$, and $\beta = 0.3$. These values were chosen so that our algorithms is more reactive when the workload increases than when it drops down. This seems to reflect the actual controllers behaviour.

The results may again be improved through more extensive parametric tests to tune the criterion $D2$, by minimizing the quantified difference between the com-

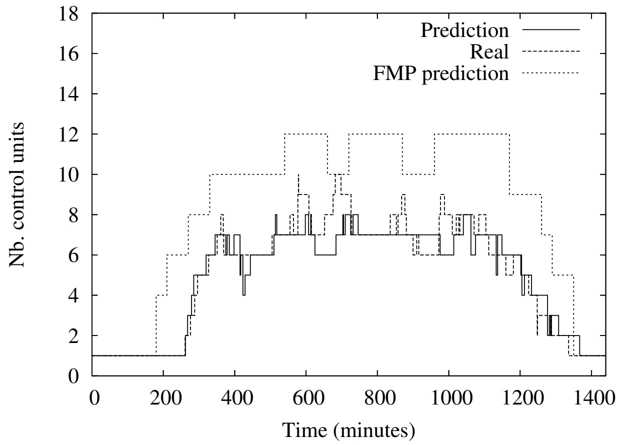

Fig. 9. Evolution of the number of control sectors (computed schedule, real configurations, and FMP opening scheme) for Reims ATCC (2003, June 2nd), with decision criterion $D2$ ($\alpha = 0.1$, $\eta = 0.2$, and $\beta = 0.3$)

puted schedule and the real configurations. There could be other alternative methods than the use of criterion $D2$ to smooth the schedule. We could smooth the values of the metrics taken as input, and use metrics that better reflect the anticipation of future workoad. We could also introduce an anticipation window in the decision process.

On the real configurations of that day, we observe four peaks in the number of control sectors which are not found in the computed schedules, whatever the chosen decision criterion. The first two are in the morning from 09h37 (t=577) to 10h08 (t=608), and from 11h16 (t=676) and 12h06 (t=726). The two others are in the afternoon, from 14h28 (t=868) to 14h37 (t=877), and from 16h13 (t=973) to 16h50 (t=1010). There may be several explanations to this. The population of controllers is not homogeneous, and some may accept more complex and heavy traffic than others whereas the neural network provides only an averaged prediction.

However, these peaks are more probably caused by "split" decisions which are not related to the controllers workload. On the opposite, the decisions to split sectors in the early morning are clearly related to a workload increase, and the correlation of the computed schedule with the real configurations is fairly good.

VI. Conclusion

To conclude, let us summarize the main ideas that were developped during our research on airspace configuration schedules and complexity metrics, with the aim to ground the FMP/CFMU flow regulation process on more objective and quantified variables.

First, we introduced algorithms exploring the whole space of possible configurations that can be built from the center's elementary sectors, rather than the small subset of pre-defined configurations currently used by the FMP.

Second, we proposed to stop using flight counts and sector capacities to assess the overloads or underloads, and to use relevant air traffic complexity metrics instead.

The original idea for selecting these relevant metrics was to use the real sector status (*merged*, *manned*, or *split*) of the control sectors as a new dependent variable to assess the actual controller's workload. This data, issued from the real airspace configurations recorded by the ATCCs, is available in large quantities and does not require heavy experimental setups to be collected. In addition, it does reflect an objective operational reality, directly related to the intended application.

In this paper, we showed how the selected metrics could be used to build airspace configuration schedules. The results presented in section V are an important step towards a realistic prediction of airspace configurations. When using recorded radar data as input, our algorithms compute opening schemes which seem realistic, with a number of control sectors close to the real configurations recorded by the ATC center.

However, before considering a possible future use of our method in the operational context, it remains to be checked if such a realistic prediction could be made with the flight plans of the final traffic demand as input, instead of the radar tracks. It may be so, considering that the relevant complexity metrics are not so sophisticated, and that smoothed values of these metrics may be even more relevant. If the existing flight plan data is not accurate enough, we may expect that future 4D-trajectory planning requirements will allow to compute a good prediction of the opening scheme, two or three days before D-day.

Apart from the FMP/CFMU opening schemes, there are many potential applications to the neural network's sector status prediction. As it is representative of how the current control sectors are being operated today, it could be used, for example, in real-time simulations to benchmark new working methods, new control tools, or changes in the airspace structure.

A more extensive testing and tuning of our algorithms is required first. In future research, we shall explore several ideas that could improve our method, like smoothing the computed schedule by introducing an anticipation window, or recombining sectors on a larger scale when merging decisions are issued. We could also try other complexity metrics that would exhibit less variations than the ones we used, as in real life the decision to split or merge sectors is not taken so often in a day. Such metrics could be obtained by averaging existing metrics over a short period of time. It may be useful to reflect the anticipation of future workload by using the predicted values of some metrics as input. Other types of neural networks, or other techniques, more adapted to dynamic discrete choices could also be tried.

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. *Optimisation des flux de trafic aérien*. PhD thesis, Institut National Polytechnique de Toulouse, 2004.

[4] D. Gianazza and K. Guittet. Evaluation of air traffic complexity metrics using neural networks ans 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] C. Verlhac and S. Manchon. Optimization of opening schemes. In *Proceedings of the fourth USA/Europe Air Traffic Management R&D Seminar*, 2001.

[8] 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.

[9] 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.

[10] 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.

[11] A. J. Masalonis, M. B. Callaham, 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.

[12] 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.

[13] 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.

[14] Cognitive complexity in air traffic control, a litterature review. Technical report, Eurocontrol experimental centre, 2004.

[15] P. Kopardekar. Dynamic density: A review of proposed variables. FAA WJHTC internal document. overall conclusions and recommendations, Federal Aviation Administration, 2000.

[16] I. V. Laudeman, S. G. Shelden, R. Branstrom, and C. L. Brasil. Dynamic density: An air traffic management metric. Technical report, 1999.

[17] J.H. Crump. Review of stress in air traffic control: Its measurement and effects. *Aviation, Space and Environmental Medecice*, 1979.

[18] 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.

[19] 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.

[20] 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.

[21] C. Mannings, S. Mill, C. Fox, E. Pfleiderer, 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 & Developpment Seminar*, 2001.

[22] G. M. Flynn, C. Leleu, and L. Zerrouki. Traffic complexity indicators and sector typology analysis of U.S. and european centres. Technical report, Eurocontrol, 2003.

[23] Note de synthèse sur l'indicateur de complexité PRU. Technical report, DTI/SDER (ex CENA), 2005.

[24] ACE Working Group on Complexity. Complexity metrics for ANSP benchmarking analysis. Report commissioned by the performance review commission, Eurocontrol, April 2006.

[25] 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.

[26] P. Averty. Conflit perception by ATCS admits doubt but not inconsistency. In *Proceedings of the 6th Air Traffic Management Research & Developpment Seminar*, 2005.

[27] C. M. Bishop. *Neural networks for pattern recognition*. Oxford University Press, 1996. ISBN: 0-198-53864-2.

[28] B. D. Ripley. *Pattern recognition and neural networks*. Cambridge University Press, 1996. ISBN: 0-521-46086-7.

[29] M. I. Jordan and C. Bishop. *Neural Networks*. CRC Press, 1997.

KEYWORDS

Air traffic complexity, airspace configuration, flow management, neural networks.

BIOGRAPHY

David Gianazza is currently researcher in the Planning, Optimization, and Modelling team[13] 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/

APPENDIX A: FULL DESCRIPTION OF THE NEURAL NETWORK USED FOR SECTOR STATUS PREDICTION

This appendix details the adjusted weights of the neural network that was found in [5], and also provides some R code allowing to predict the sector status. Due to presentation constraints, the full range of decimals is not written.

The R code providing the triple of probabilities (y_1, y_2, y_3) (see equation III) is written on figure 10. In this code, `Mh` and `Mo` are the matrices of parameters (weights and biases) assigned to the connections between the input layer and the hidden layer, and between the hidden layer and the output layer, respectively. Biases are handled in the network by adding special units returning a constant value 1. The biases are the weights assigned to the connections between such an unit and the next layer.

```
f.softmax <- function(z)
   {exp(z)/sum(exp(z))};
f.logistic <- function(z)
   {1/(1+exp(-z))}

f.predict <- function(input) {
   f.softmax(Mo %*% c(1,
   f.logistic(Mh %*% (c(1,input)))))
}
```

Fig. 10. R code for the prediction of sector status

Table I details the values of `Mh`. The first column contains the biases w_{0j}. The second gives the weights of the connections between the input x_1 (volume V) and the 15 hidden units, and so on with the other columns.

Table II describes the transpose matrix $\mathtt{Mo}^T$ (the matrix has been transposed in order to fit in the page). The first column of `Mo` also contains biases, and the 15 others give the weights of the connections between the hidden units and the output units.

[13]The POM team was formerly part of the Global Optimization Laboratory LOG CENA/ENAC

	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	[,7]
[1,]	2.6700	0.6035	−3.2242	1.5125	2.84665	0.17517	1.99546
[2,]	1.3664	5.5459	1.2125	−0.5923	−2.27209	0.69194	−1.43059
[3,]	0.3902	0.6342	1.2399	−0.0718	0.54612	−0.03906	−0.92772
[4,]	0.5531	0.2000	0.3522	−0.0218	0.21848	−0.00494	−0.17895
[5,]	12.7531	−12.9649	3.8234	−0.1762	−0.05529	0.15634	−2.54624
[6,]	−2.5944	−0.5692	3.0737	−1.5888	−2.70640	−0.20288	−1.86718
[7,]	−62.6510	−112.4401	0.8557	0.3602	2.27203	−0.58279	−1.27146
[8,]	248.0508	696.6207	1.4739	−9.8654	0.63017	−0.11527	−4.27899
[9,]	−16.6421	−43.5024	−3.2625	−0.5521	−0.95558	−0.08815	3.29767
[10,]	−24.9896	138.5643	2.9702	−1.5819	0.66453	−0.50509	−4.00290
[11,]	−14.1631	14.4098	−3.6297	0.1643	0.17842	−0.16329	2.39568
[12,]	1.3726	5.4969	1.0101	−0.5810	−2.41572	0.70270	−1.25898
[13,]	−54.6044	−146.4793	−1.4978	−0.8585	−0.74717	−0.26718	2.03936
[14,]	−10.0972	0.6481	−4.0344	0.2354	−5.43306	0.54850	3.24417
[15,]	2.3865	1.9788	−0.3361	0.0353	0.20662	−0.02091	0.50016

TABLE I

MATRIX Mh OF WEIGHTS AND BIASES FOR THE CONNECTIONS BETWEEN THE INPUT LAYER AND THE HIDDEN LAYER

	[,1]	[,2]	[,3]
[1,]	75.33756	195.42101	−269.39478
[2,]	46.33752	35.77978	−83.11712
[3,]	−77.57276	−52.77819	129.61192
[4,]	67.61727	−3.31538	−64.50745
[5,]	−200.23995	18.92759	181.49234
[6,]	−58.88296	−210.66951	270.42091
[7,]	47.64013	34.81970	−82.17057
[8,]	3.90553	0.69906	−4.47156
[9,]	−11.08421	−6.53394	17.31517
[10,]	7.13780	−5.45104	−2.09130
[11,]	−4.38653	2.88566	1.17121
[12,]	−59.72134	−210.62732	269.54483
[13,]	77.51679	51.23199	−127.99521
[14,]	−14.74073	−3.60412	18.72713
[15,]	240.72415	110.87175	−350.40952
[16,]	38.76275	−25.18119	−14.04946

TABLE II

TRANSPOSE OF THE MATRIX Mo OF WEIGHTS AND BIASES FOR THE CONNECTIONS BETWEEN THE HIDDEN LAYER AND THE OUTPUT LAYER