An Agent Based Model of Air Traffic Management

C. Bongiorno[1], S. Miccichè[1], R. N. Mantegna[1,2]
[1]Dip. di Fisica e Chimica, Univ. di Palermo
Viale delle Scienze, 90128 Palermo, Italy
[2]Center for Network Science and
Dep. of Economics,
Central European University,
Nador u. 9, 1051 Budapest, Hungary

G. Gurtner[3], F. Lillo[1,3,4], L. Valori[3]
[3]Scuola Normale Superiore di Pisa
P.zza dei Cavalieri 7
56126 Pisa, Italy,
[4]Santa Fe Institute
1399 Hyde Park Rd.,
Santa Fe NM 87501, USA.

M. Ducci[5], B. Monechi[5], S. Pozzi[5]
[5]Deep Blue s.r.l.
P.zza Buenos Aires 20
00100 Roma, Italy.

Disclaimer - *This work is co-financed by EUROCONTROL acting on behalf of the SESAR Joint Undertaking (the SJU) and the EUROPEAN UNION as part of Work Package E in the SESAR Programme. Opinions expressed in this work reflect the authors' views only and EUROCONTROL and/or the SJU shall not be considered liable for them or for any use that may be made of the information contained herein.*

***Abstract*—The WP-E ELSA project aims at developing an empirically grounded agent based model that describes some of the stylized facts observed in the Air Traffic Management of the European airspace. The model itself has two main parts: (i) The strategic layer, focused on the interaction between the Network Manager and the Airline Operators and (ii) the tactical layer, focused on aircraft and controllers behaviour in Air Traffic Control (ATC) sectors.**

The preliminary results for the strategic layer show that when we have a mixing of re-routing and shifting companies, the overall satisfaction can even increase together with the number of flights, which is an effect not observed when only one type of companies is present. The preliminary results for the tactical layer indicate that when shocks in the system are confined in small areas, the interplay between the re-routing and change of flight level strategies may even lead to trajectory modifications that give smaller average delays as long as the number of shocks increases.

I. Introduction

In the future of Air Traffic Management (ATM) it is expected to observe an increase of traffic demand and new business challenges that will bring the current ATM system to its capacity limits within the 2013-2015. As a consequence, an overall productivity improvement is urgently needed [1], [2], [3], [4]. The structure of ATM system, as it is known today, will therefore change in many aspects. One of the key enabler to the productivity and efficiency shift foreseen by SESAR will be the business-trajectory concept [1]. In the future SESAR scenario airspace users will not fly along structured routes. On the contrary, they will be able to fly a 4D trajectory selected on the basis of their own business and efficiency needs. Within this major change not only the ATM productivity should be drastically enhanced, but consequently also the ATM system safety and resilience standards will have to be improved.

The path to a deep understanding of how these aspects will impact the future air traffic management procedures goes through a better understanding of the actual air traffic system and its management procedures. To this end the WP-E ELSA project aims at developing an empirically grounded agent based model that describes some of the stylized facts observed in the Air Traffic Management of the European airspace [5]. The model itself has two main parts, a Strategic layer and a Tactical layer, which aims at emulating the two main steps relevant in the ATM: (i) The strategic layer, focused on the interaction between the Network Manager and the Airline Operators [6], [7] and (ii) the tactical layer, focused on aircraft and controllers behavior in Air Traffic Control (ATC) sectors [8], [9]. This model will then be used as a scenario simulator in order to understand which benefits the trajectory based scenario will bring to the ATM world. Here we will discuss the features of the model only within the current ATM scenario.

The paper is organized as follows: in section II we will discuss the main features of the strategical and tactical layers of the model. In section III we will summarize the main inputs we have gathered from operational experts about some of the ATM features to be implemented in the model. In section IV we will show the main preliminary results of the model and in section V we will finally draw our conclusions.

II. The Model

As mentioned above, the model is organized in two main layers: a Strategic layer and a Tactical layer. In the current implementation of the model these two layers are still independent from each other. However, they are logically connected in the sense that the results of the strategic layer should feed the tactical layer. Such integration stage is currently ongoing.

A. The Strategic layer

The strategic layer of the ABM aims at modeling the events taking place from the planning of the flight plan by the Airline Operators to the final acceptance by the network manager. Its real temporal scale goes from months down to a few hours before departure. On the other hand, its spatial extension concerns the whole ECAC space down to the sectors. The main aim of the strategic layer is to model how and why the airspace is filled such as it is. The "final state" is a result of different actors, but also of the structure of the airspace itself: sectors, national airspaces, etc. In the following, we focus on the role of the former, the airspace itself being kept fixed, either by using an artificially generated network of sectors or the real one.

There are two main types of agents in the system. On one hand, Airline Operators try to get the best trajectories for their flights. This best trajectory has two components: the geometrical length (in 3 dimension, and taking into account dominant winds) and the times of departure/arrival. Different Airline Operators, which may have different goals, are competing one with each other for the "best" slots and trajectories, based on business considerations and constrained by the structure of the airspace. On the other hand, the network manager tries to fill the airspace as best as possible, its main concern being to avoid to overload the airspace, in order to guarantee safety [10]. In the current scenario, the network manager is very passive, takes only propositions from the Airline Operators and tries to fill the airspace. In the future SESAR scenario, the network manager might have more proactive behavior, for instance submitting counter-propositions to the Airline Operators. This is a feature we plan to add to the modeling, which currently only describes the current ATM scenario.

The main object of the model is the flight plan. It is defined as a pair $fp = (t, \mathbf{p})$, where t is the time of departure and $\mathbf{p}$ is a vector containing the list of sectors followed by the aircraft. t is a real number, hence our model is a continuous time model.

The aircraft are bound to travel on a network, whose nodes represent sectors and links exists if two sectors share a common boundary. Moreover, a weight is associated with the links: it represents the time of travel between two sectors. These different times are clearly related one to each other in reality – because of the physical extension of the sectors. However in this paper, we draw weights as independent normal random variables with mean τ and standard deviation σ, in order to be able to perform semi-analytic calculations. A second metric is associated to the nodes themselves: an integer representing the capacity of the sector. This capacity is here defined as the maximum number of aircraft that can be simultaneously present in a sector[1]. In reality, it is related to many features of the sector, like its area, its volume, etc. Here we chose a constant capacity (equal to 5) for each sector.

We used mainly a simulated network of sectors for the simulations, but we tested our models also on real airspace networks, obtaining similar results (not shown here). The airspace is built by using a Voronoi tessellation of a set of 90 points randomly drawn on the plane, see also [11], [12]. After the tessellation, we connect each node with its geometrical neighbors – thus building the Delaunay triangulation. Then we choose the distribution of weights on the network (normal), the capacities (constant) and the airports (randomly chosen, here we pick only two airports). The structure of this network, even though it is planar, has some strong similarities with the real network of sectors, as we show in section IV-A.

[1]We are aware that this not the standard definition of capacity, defined as the average number of aircraft per unit of time present in a sector. We are planning to implement the standard definition of capacity within the model in the next releases.

In the simulations we performed, we have several agents of type AO (Airline Operator). Each of them has a unique form for the cost function for its flights, which is

$$c(t, \mathbf{p}) = \alpha|\mathbf{p}| + \beta(t - t_0),$$

where t_0 is the desired time of departure, $|\mathbf{p}|$ the weighted length of the path on the network and α and β two parameters defining the main characteristics of the company. Please notice that flights are only shifted ahead in time, therefore $t \geq t_0$. A high value of β simulates a company eager to have its flights on time. On the other hand, a high α simulates a company more preoccupied by the length of the trajectory.

Each AO begins with the generation of k flight plans for each "flight", defined by a pair of airports and a desired departure time t_0. They are generated by finding the k (k is set to 10 in the simulations) best flights plans, ranked by increasing cost, among all the possible paths $\mathbf{p}$ connecting the two airports and times of departure t. Among them, the best one is the one with the shortest path $\mathbf{p}_{sp}$ on the network and the desired time ($t = t_0$), with associated cost $c_{best} = \beta|\mathbf{p}_{sp}|$. More specifically, the AO can shift the flight plan in time by a constant increment, that we fix always equal to the average time of travel τ between sectors, hence giving the natural time scale of the system.

Once all the flight plans have been generated, a company is randomly drawn and it submits the k flight plans of one flight to the second type of agent, the network manager (NM). Following this queue, the NM tries to fill the airspace. For each flight, it takes the best flight plan and tries to fill it on the network. If one sector or more overreaches its capacity, the flight plan is rejected. Then the next flight plans, with higher costs, are tried, until one is accepted or all are rejected. The NM checks the flight plans of a given AO in the precise order given by the AO. This is done in order to mimic the iteration process that in reality occurs between NM and AO.

A key parameter of the model is the desired departing time t_0. We tested several patterns for the density of desired departing times by all the companies, ranging from a totally uniform distribution to very peaked distributions. The results in terms of occupation of the space and satisfaction of the Airline Operators are very different, as we show in the results part (see IV-A).

Finally, we define a metric to measure the satisfaction of the system. First, we choose to define the satisfaction of a single flight by computing $s_f = c_{best}/c(fp_{accepted})$, where c_{best} is the cost of the best flight plan (hence the one with the smallest cost) and $c(fp_{accepted})$ is the cost of the accepted flight plan. We choose also to take $s_f = 0$ if all its flight plans have been rejected. We define also the satisfaction of the overall system as the average satisfaction over all flights:

$$S = \frac{1}{N_f} \sum_f s_f,$$

where N_f is the total number of flights.

We implemented the model in Python, using the library `networkx`.

B. The Tactical layer

The agents of the tactical layer of the ELSA agent-based model are aircraft/pilots and controllers who are active at the level of ATC sectors. In this layer of the agent based model we model and simulate the events that make a planned flight plan, recorded in the so-called M1 files [13], transform into an actual one, recorded in the so-called M3 files [13]. The aim is that of investigating the issues that affect the predictability[2] of the last filled flight-plan within the ATM system. The specific scientific questions we are investigating are:

- What are the issues that affect the predictability of the last filled flight-plan within the ATM system? How is the predictability affected by these issues?
- Can sectors capacity be improved by a more efficient management of conflicts?

1) General features of the model: The interaction between the agents is needed in order to manage the tactical changes occurring in the system due to unforeseen events, i.e. weather events, congestions, limitation of sectors capacity, etc. Moreover, the ATC sectors are the places where flight trajectories are made conflict free. In the current version of the model the aircraft/pilots have a limited intelligence. We are also giving these agents the opportunity of modifying the aircraft velocity in order to solve safety events, even though this strategy is actually used by controllers only in specific cases.

The model takes into account that M1 trajectories are not conflict free. Thus one main task to be performed within the model is to deconflict trajectories. The model works either for solving conflicts due to non-intersecting trajectories and for conflicts due to both head-on and converging traffic. Moreover, we simulate shocks in the system and see how the system reacts to it. Specifically we simulate a shock in an area around a navigation point. We assume that the shock lasts for a certain time window. Operatively, this means that for a certain time window a certain area of the ATC sector can not be crossed by flights. This might correspond to a situation where an extreme weather event occurs as well as to a situation when a certain area is highly congested and therefore the air traffic must be deviated [14], [15]. As a result, another task of the model is to change one or more flight trajectories in order to avoid the shocked areas. The way we model this step is to deviate the flight trajectories along new navigation points that are external to the restricted area and with the constraint that (i) we want to minimize the length of the deviated trajectory and (ii) the deviated trajectory must be conflict free. We will perform different simulation experiments changing the statistical features of the shocks.

Based on the inputs from the italian ENAV operational experts, we have introduced the feature such that when a trajectory is deviated, then it is not sent back to its planned trajectory. Rather, it is sent to the planned exit navigation point of the sector. This helps us in implementing the fact that airplanes are given directs within a sector.

2) Implementation of the Model: The code that implements the model is written in Python [16]. However, some modules have been written in C [17] in order to improve the computational efficiency of the ABM. Below we describe the modules that compose the tactical layer of the Agent based model.

In the current version, the tactical layer of the model works at the level of a single ATC sector. Having that, in the current version of the model we consider shocked areas which are totally included in the sector and that do not involve navigation points on the boundaries. In the current implementation the planned trajectories are assumed to be existing. Specifically we will consider the flight trajectories are recorded in the DDR (Demand Data Repository) M1 files we have access to [13].

Navigation Points module - Given the sector, we populate it with navigation points. On one hand, part of the navigation points selected are real ones. On the other hand, other nav-points are generated randomly from an uniform distribution. We have considered only those falling inside the sector. These new navigation points could be seen as temporary points (!-points) in the M3 flight plan. We have not generated navigation points on the boundaries because these are sensible points and need to be treated separately.

Flight List module - Once the sector has been populated with navigation points, we create a list $FL_k = \{f_1^{(k)}, f_2^{(k)}, \cdots\}$ of flights active in the k-th time-step in the considered sector. Such list will be reshuffled in the next time-step. Within this list we check whether the flights are crossing a shocked area and whether or not they conflict with other trajectories. Specifically, the i-th aircraft in the list will be checked against all other $j < i$ flights. When a trajectory modification is needed, it will affect the i-th flight. The reason for shuffling the list at each time step is in order to avoid that the trajectory modifications are always applied to the same aircraft.

Collision module - In order to check for collision between two flights, we use a data structure that considers the aircraft localized inside the trajectory segments travelled within the given time-step Δt. For this purpose, we introduce a finer subdivision of the time-step into N *elementary time increments* δt and compute the real space-time position of the aircraft at each elementary time increment, by assuming a constant velocity. The collision algorithm will have to simply calculate the positions of the aircraft for each of the elementary time increments and then compute the distances between the two aircraft at these positions. Suppose we are now checking if the i-th flight trajectory is conflicting with all other trajectories, with $j < i$. We are therefore considering a number i of flight trajectories. For each of them we have an array $\mathcal{P}_j$, $j = 1, \cdots i-1$ of positions computed according to the algorithm illustrated above. For each of the elementary time-increments, and for the i-th flight we compute an array whose elements are the minimum distances $d_{ij}^{(k)}$ between the i-th

[2]Predictability is here intended as a comparison of the actual flight arrival time to the scheduled flight arrival time.

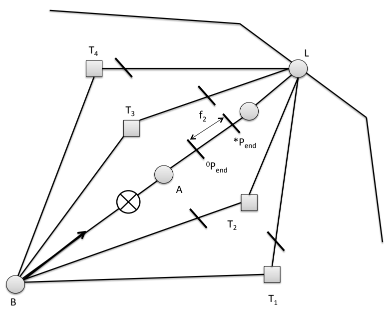

Fig. 1. The figure illustrates the techniques of rerouting in the de-conflicting module. The crossed circle indicates the shocked area. The squares indicate the temporary navigation points. The circles indicate the navigation points in the trajectory to be modified.

aircraft and all the other aircraft $f_j^{(k)}$ in the list FL_k with $j < i-1$. From such an array it is possible to estimate a *fitness value* to be maximized. In fact, by assuming that the safety distance threshold is d_{thr}, we consider the subset of distances $X_i^{(k)} = \{\cdots, d_{ij}^{(k)}, \cdots\}$ where $d_{ij}^{(k)} \leq d_{thr}$. The fitness value is then defined as $\mathcal{F}_1^{(i,k)} = \sum_{q \in X_i^{(k)}} (d_{thr} - d_q^{(k)})$. If this value is different from zero then there is a conflict and the algorithm proceeds to the next module that performs the de-conflicting of trajectories. As a result the computational time increases linearly with the number of aircraft.

De-conflicting module - After the check for collision has been done, this module searches for a new conflict-free trajectory. It is conceived as a three-step algorithm that acts on the velocities of the aircraft, the search of a new trajectory (re-routing) and the change of flight level, in case conflict exists. The order by which the three steps are applied might be changed. Here we describe them.

The first step of the module we present here is the one that performs the re-routing. The procedure is illustrated in Fig. 1. We first identify the two navigation points B and A which are before and after the collision (crossed circle in the figure), respectively. The idea is to (i) keep B, (ii) substitute A and (iii) eliminate all the other subsequent navigation points but the last one L. To do that, we take the previously generated temporary navigation points T_k (squares in the figure) and we order them with respect to the angle that the segment connecting B and T_k forms with the original trajectory. We select the temporary navigation point that have the smallest angle. We admit a maximum angle of $45°$. Having this new navigation point we compute again all the distances with the $j < i$ trajectories and compute again $\mathcal{F}_1$. If $\mathcal{F}_1 = 0$, then we select this navigation point, otherwise we go to the navigation point with the second smallest angle. This procedure is iterated until we find $\mathcal{F}_1 = 0$. If we find only navigation points with an angle larger than $45°$, the algorithm exits this module and go the the next one.

The second step of this module involves changes of flight level. The model implements three possible flight levels. All flights are initially considered to be active in the central flight level. Therefore they can move upwards or downwards whenever the re-routing is not feasible. The choice of the new level is done by considering the one where there is less probability of having conflicts. This is assessed by computing the sum of the $\mathcal{F}_1$ functions for all the flights in the two external flight levels.

The third step of the module would be based on a genetic algorithm [18] for changing the flight velocity. Due to limitation of space we will not present this part of the model.

3) Expected results: For each simulated flight we will monitor two variables. First we will consider the number $A_{c,f}$ of actions that any controller has performed for each flight $f = 1, \cdots, F$ and for each of the three types of possible actions (re-routing, flight-level change, velocity change), i.e. $c = 1, 2, 3$. This will simply be the number of changes (velocity, position, ...) operated on the planned trajectory for each type of action. We will also consider positive actions $A_{1,f}^{(+)}$ as the ones when the controller gives a direct. We will consider negative actions $A_{1,f}^{(-)}$ all the others. As a second variable we will consider the exit time $T_{3,f}^{(0)}$ of each flight $f = 1, \cdots, F$ from the sector. This will be compared with the planned exit time $T_{1,f}^{(0)}$.

III. Validation Activities

A. Operational Input for the Strategic Layer

The main operational inputs for the tactical layer have been collected during interviews with Alitalia Flight Dispatchers that work at the Alitalia Operation Center (OCC). They are the professional figures in charge of defining the flight plans and monitoring the flight execution phase. The Alitalia Operation Center is responsible of coordinating and managing almost 700 flight per day, of which around 70 are long-haul flights. For each of these flights a flight plan has to be produced by the OCC and then submitted and approved by the CFMU. Long-haul flights' planning is handled manually and starts 6 hours before the scheduled departure time while short and medium-haul flights are handled using an automatic procedure. Dispatchers have to intervene only if the system flags an exception. This process starts 2 hours before the scheduled departure time. In both cases flight dispatcher make use of a dedicated software tool called LIDO Flight.

The planning phase starts by collecting information about the flight such as weather at destination and on the route or the aircraft performances and possible limitations and failures on board. On the basis of the information collected a flight plan is prepared by optimizing the overall cost of the flight and by ensuring at the same time the safe execution of the flight. For example, the occurrence of a weather perturbation is considered to be an unsafe event and it will always be avoided even at the cost of travelling a longer route. The costs taken into account always include fuel and ATC fees. Costs related to delays are not taken into account by the software tool but can be evaluated on a case-by-case basis by the flight dispatcher. At this stage no information about other flight trajectories is taken into account. As a result, flight trajectories might be not conflict free. After the flight plans are prepared, manually or automatically, they are submitted according to the ICAO format to the CFMU through a dedicated system (SITA). The ICAO format contains the take-off and landing times and a list of navigation points with the related flight level. The CFMU recalculates the flight plan using their own models. These models differ from those used by Airline Operators. In fact they do not consider the differences in performance that aircraft of the same type may have and they manage the vertical profile of the trajectory in a different way. If the flight plan is rejected the dispatcher is noticed; CFMU gives the reason of the rejection but they do not suggest an alternate solution. Moreover the flight dispatcher is unaware of what other companies are doing. When a flight plan is rejected there is no bargaining between CFMU and the dispatcher. The dispatcher simply submits an updated flight plan and waits for its approval. This process is iterated until the dispatcher has a flight plan approved. Communication between CFMU and the dispatcher takes place almost exclusively through the SITA system. In some case the final flight plan can be discussed with CFMU through a phone call. A schematic representation of the process is shown in figure 2. The information collected

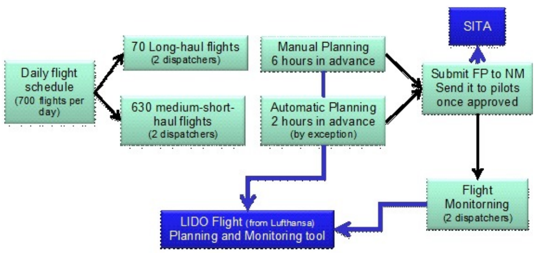

Fig. 2. The process of flight planning and monitoring in the Alitalia OCC.

that are more relevant for the development of the strategic ABM are mainly related to: (a) the timeframe of the flight plan definition process (6 hours in advance for long-haul flights and 2-hours in advance for medium and short-haul flights); (b) the costs taken into account for the flight plan optimization; (c) the interactions between the Flight Dispatcher and the Network Manager and the fact that the flight dispatcher is unaware of other companies strategies; (d) the flight plan submission process, how flight plans are rejected and submitted again for the final approval; (e) the criticalities related to the planning phase such as the exceeding of capacity of one or more sector, bad weather avoidance, partial or total closure of destination airport or unpredictable events like strikes, big events, wars.

B. Operational Input for the Tactical Layer

The sector chosen for the calibration of the tactical layer was the sector "LIRROV" in the Rome Area Control Center (ACC), presented in figure 3.

The sector is crossed by North-South and East-West over-flight traffic as presented in figure 3. The flights inside the sector are mainly commercial, with few exceptions of military flights that however behave like commercial flights. As a result of these traffic pattern, several critical areas emerge from the crossing of these traffic flows, highlighted by red circles in figure 3.

The sector can operate in two main configurations presented in table I. During summer the traffic load is usually high so the sector operates with the configuration "B" in which it is vertically split in order to increase its capacity. On the other hand, during winter when the traffic is lower, it usually operates in configuration "A" where it is composed by just one volume and lesser capacity. The strategies used to avoid conflicts are both horizontal and vertical. In the first case one of the aircraft involved in a possible conflict is deviated from it original route to achieve horizontal separation, while in the other case a small variation in flight level of just 10 FL is used to achieve vertical separation. Combinations of these two strategies are also possible. Despite the fact that horizontal deviations are more convenient in terms of fuel consumption, a small vertical deviation is usually preferred. Moreover in order to reduce the amount of traffic to be managed and the delay generated by their action, controllers usually send aircraft directly to the exit point of the sector after any deviation and whenever possible. Another possibility for the controllers to reduce the traffic load of the sector is to apply a "direct", i.e. to send an aircraft directly to a point in the next sector of its flight plan. However, since directs require the coordination of the controllers in the involved sectors and thus an increase of their workload, they are considered unlikely events. The tactical layer implements all these finding regarding the traffic patterns inside the sector. Main routes and critical areas are reproduced and also the seasonality of traffic has been considered in the calibration.

Adverse weather conditions occur on a daily basis and are not a negligible effect inside the sector and the system in general. This kind of events does not represent a challenge for the controllers that are always supposed to be capable of handling them. We have been able to identify two major classes of perturbations depending on their dimension:

1) Small shocks ($\approx 5\ NM$ of radius), with a fast dynamics and a short lifetime ($\approx\ 1\ h$), usually occurring during summertime;

2) Large shocks (around $60\ NM \times 20\ NM$), which can be considered static and with a lifetime that goes from $8\ h$ to $10\ h$.

This kind of perturbations represents big storms occurring during winter. While the only possible way to manage a small shock is to avoid it, it is possible that the biggest one could be crossed by an aircraft instead of being avoided and thus generating a small delay instead of a large one.

These kinds of shocks has been implemented into the tactical layer, following the discussion with the operational expert. The correlations between size, lifetime and dynamics of the events has been introduced, so that both small dynamical shocks and large static ones are present. In both cases shocks can be avoided using horizontal deviations as well as vertical ones if the considered shock does not affect all the flight levels. Since aircraft might fly through a large perturbation instead of completely avoiding it, a probability of being crossed is assigned to each large perturbation. Moreover in order to simulate correctly the seasonality of the disturbances, small shocks are more likely to occur when simulating summertime while large shocks occur more frequently when simulating winter. After any redirection, due to separation or adverse weather conditions, controllers try to sent the aircraft directly towards its exit point in the sector.

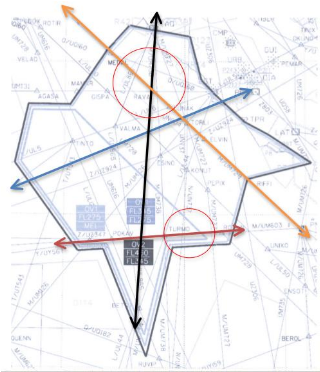

Fig. 3. Projection of the LIRROV sector on a map including the navigation points. Major routes and their directions are indicated by the blue, red, orange and black arrows while the critical areas emerging from their intersection are marked with red circles.

TABLE I
OPERATION CONFIGURATIONS OF SECTOR LIRROV. MEL STANDS FOR MINIMUM EN-ROUTE LEVEL.

Configuration	Sector/s	Heights	Capacity
A	$OV1 + OV2$	$MEL - 460\ FL$	52
B	$OV2$	$350 +\ FL$	44
B	$OV1$	$MEL - 345\ FL$	44

IV. RESULTS

A. The Strategic layer

In this section we present some of the results obtained in the strategic layer. We explore the output of the model by modifying three parameters independently.

The first is the ratio β/α, defining the behavior of the AO. In particular, we use the two extreme configurations: $\beta/\alpha \gg 1$, in which the company care mostly about punctuality and takes any path that guarantees the desired departing time, and $\beta/\alpha \ll 1$, for which the company cares mostly about the length of the path, thus taking the shortest one, possibly shifted in time. In the following, we will call the first type of company "R" (for rerouting) and the second one "S" (for shifting). From an operational point of view the "S" companies mimic the low-cost companies while the "R" companies mimic the hub-based traditional ones.

Secondly, we tested different patterns of the desired times of departure. Specifically, we use a "wave" structure, as it is the case currently in the air traffic. The wave itself has a duration equal to the average time τ of crossing between sectors, which is also the increment of shifting of the flight plan for the AO. We denote with Δt the time between the end of a wave and the beginning of the next one, using τ as unit. Moreover, we choose 24τ as the interval of possible departure times, meaning that the waves need to occur between $t = 0$ and $t = 23\tau$.

Finally, the last parameter is simply the total number of flights submitted to the NM.

1) Pure population: We begin by showing the results of simulations with "pure" populations. This means that each AO has the same cost function, i.e. the same ratio β/α within each simulation.

The top panel of Figure 4 shows how the total satisfaction of the system varies with the number of flights, for different values of the ratio β/α, with $\Delta t = 23$ (only one peak at $t = 0$). Although possibly unrealistic, this is an extreme value that will help us in better understanding how the model works. As expected, we see that the satisfaction is monotonically decreasing with the total number of flights. Moreover, we see that there are some differences between different values of β/α. Indeed, the initial plateau is more extended for small values of the ratio. Overall, company of type S, with small ratio, tend to be more satisfied that company of type R.

This is actually due to the pattern of desired departing times. On the bottom panel of figure 4, we present the satisfaction against the ratio β/α, for different values of Δt. As one can see, for big values of Δt, i.e. few and well separated waves, the satisfaction decreases monotonically with the ratio. Company S is always doing better than company R. On the contrary, when there are many waves (small Δt), the satisfaction is increasing with the ratio, i.e. companies R are better off. This is the case because in this situation the companies which shift in time find other companies ahead: it is thus better to change the route instead. It is also interesting to see that there is an intermediate situation ($\Delta t = 1$) for which none of the extremes is better: the companies doing a compromise between length of path and time of departure have a higher satisfaction.

2) Mixed population: Now we consider a system where there are two types of companies competing with different cost functions. A fraction of the flights, called "S", are operated by AOs with $\beta/\alpha = 10^{-3}$. On the other hand, flights called "R" are operated by AOs with $\beta/\alpha = 10^3$.

Figure 5 shows the dependance of the satisfaction of companies from the fraction of "S" flights. This figure, which should be compared to figure 4 (bottom) is strikingly different. In fact, it is clearly not trivial to find a pure population with a behavior similar to the mixing of two extreme companies. In particular, the satisfaction is hardly monotonic now, except for very small values of Δt. We note that the overall tendency is that a uniform distribution of departing time increases the satisfaction, except for very pure populations, as we saw before. Moreover, even a high value of Δt does not strongly favor S companies. In fact, the plot is almost entirely flat for $\Delta t = 23$ and thus the result is insensitive to the fraction of S or R companies.

In figure 6 we show more in details how the satisfaction of each type of company depends on the mixing of the population. In the top panel, the satisfaction of company R

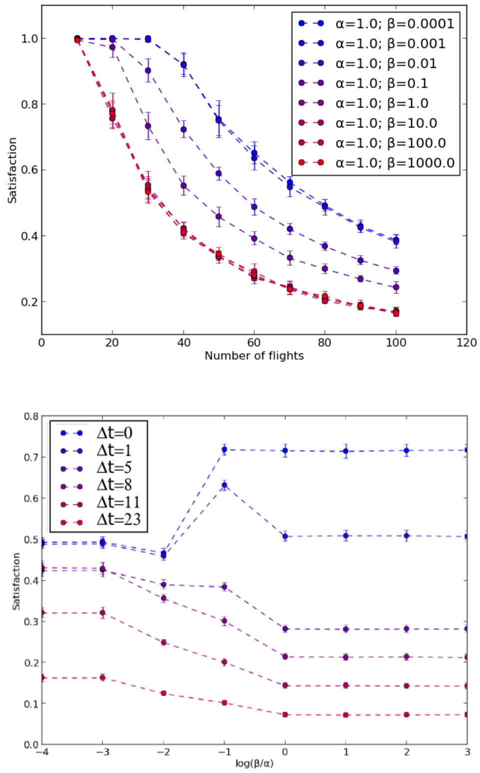

Fig. 4. Top: Satisfaction against number of flights, for different ratios β/α with $\Delta t = 23$. Bottom: Satisfaction against ratio β/α, for different value of Δt and 240 flights. Each point is the results of the average over 100 simulations, with standard errors taken as error bars.

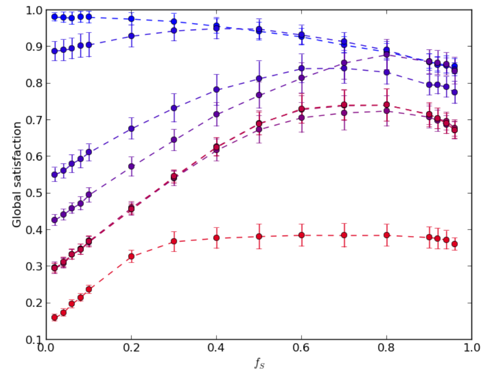

Fig. 5. Total satisfaction against the fraction of flight "S" for different values of Δt and 120 flights. The color legend is on the bottom panel of figure 6.

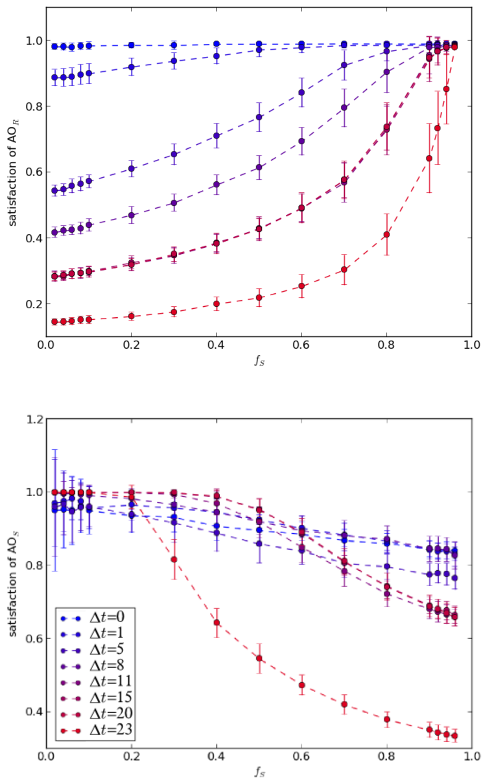

Fig. 6. Top: Average satisfaction of companies R against the fraction of S-flights. Bottom: Average satisfaction of companies S against the fraction of S-flights. The total number of flights is equal to 120. The same color legend (on the bottom) applies to both plots.

is quite simple. First, it is maximum when f_S is close to 1, thus displaying a "the loner, the better" effect: a company gets a higher satisfaction when it is surrounded by companies of the other type, rather than its own. We can understand this result by considering that if everybody wants to shift the flight plan, it is better for a company to reroute, because the secondary routes will be left free. In the bottom panel, we see that the satisfaction of company S displays the same type of behavior: the satisfaction is maximum when the company is alone. However, the variations are more complex here. As one can see, the different curves corresponding to different values of Δt are crossing each other. This means that for different values of f_S, it is better sometimes to have a small Δt, and sometimes a big Δt. For instance, for $f_S = 0.4$, it is better to have $\Delta t = 20$ rather than to have $\Delta t = 1$, which is perfectly normal for S companies. On the other hand, for $f_S = 0.8$, the contrary happens, and suddenly it is better for S companies to have a more uniform distribution of departing times. The whole picture is even more complicated by the fact that this behavior is not monotonic with Δt. With the same example, with $f_S = 0.4$, it is better for S to have $\Delta t = 20$ rather than 1, but it is much worse to have $\Delta t = 23$. This non trivial effect of mixing different companies gives a rich behavior in terms of optimization of the total satisfaction, as noted above.

B. The Tactical layer

We model the trajectories of the flights within a single specific sector, i.e. LIRROV, which is an ATC sector located close to the Fiumicino airport in central Italy. Moreover, we also consider real M1 trajectories taken from the DDR data already available within the project. We are considering all flights in this sector active on day 06 May 2010. We have a total number $F = 172$ of flights during this day, above the flight level threshold of 200. We will simulate three main flight levels in the sectors. Initially we will populate the central one. We will then allow aircraft to change upwards or downwards their flight levels in case of conflict.

As mentioned above, flights can change their flight level and undergo re-routings. In some special case, when the previous two strategies fail, they can also change velocities. Due to the feedback from operational experts, in the current version of the model the way by which controllers try to solve possible conflicts or avoid shocked areas by firstly do a re-routing, then by changing the flight level and finally, if necessary, by changing the aircraft velocity.

1) Preliminary Results: In the current version of the model shocks are still totally randomly occurring within the sector. Although un-realistic, this is the simplest assumption that allows us to investigate how the delays in the flights depend on the number and size of shocks.

In a preliminary set of simulations we considered a situation when we have an average number N_S of shocks per time step and the shocked area is inversely proportional to N_S. We also assume that when $N_S = 1$ then the radius is 5 NM. Each time-step is $\Delta t = 300s$ and for each simulation experiment we perform $N_E = 5000$ runs. In this way the part of sector not available to aircraft is essentially the same as long as N_S increases. This allows us to test whether delays are depending from the available sector space or from the fact that there are many shocked areas. Again, in all performed simulations the change of velocity module did not operate.

Table II shows that as long as N_S increases, the percentage of aircraft that change their flight level increases. However, surprisingly, the percentage of aircraft that are re-routed follows a non-monotonic pattern, although it shows an overall increment. The first point $N_S = 0$ corresponds to the case when we do not have shocked areas in the model.

In the top panels of Fig. (7) we show the average (top-left) and the standard deviation (top-right) of the delays experienced by the F flights in the N_E simulations. Average and standard deviations are computed on the re-routed (and therefore delayed) aircraft only. The non-monotonic behavior previously observed is again visible. As a general trend it seems that larger values of N_S here give rise to a smaller delay. A possible explanation for the observed effect is that, when the same amount of not available space is fractioned into smaller pieces, the model will change the flight level more frequently and therefore the ones that experience a re-routing will suffer an overall smaller delay. A similar situation is shown in the bottom panels of the figure, were we show the same variables for the case when the radius of the shocked areas is doubled.

TABLE II
PERCENTAGE OF AIRCRAFT THAT ARE RE-ROUTED OR CHANGE THEIR FLIGHT LEVEL. THESE SIMULATIONS ARE PERFORMED ASSUMING THAT THE SHOCKED AREAS IS INVERSELY PROPORTIONAL TO N_S AND SUCH THAT WHEN $N_S = 1$ THE RADIUS IS 5 NM.

N_S	re-routed	flight level changes
0	0.159343	0.0977733
$\frac{1}{8}$	0.218005	0.166307
$\frac{1}{4}$	0.166115	0.187858
$\frac{1}{3}$	0.171248	0.191772
$\frac{1}{2}$	0.177301	0.199638
1	0.189807	0.216972
2	0.210333	0.241837
3	0.222334	0.258345
4	0.227505	0.267824
8	0.274455	0.326434

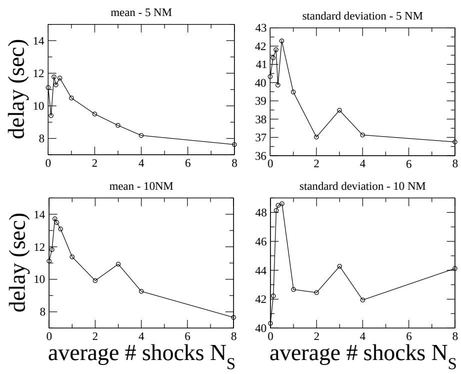

Fig. 7. In the figure we we show the average (top-left) and the standard deviation (top-right) of the delays experienced by the F flights in the N_E simulations. These simulations are performed assuming that the shocked areas have a radius inversely proportional to N_S and such that when $N_S = 1$ the radius is 5 NM in the top panels and 10 NM in the bottom panels.

Further investigations are needed in order to better characterize the non-monotonic behavior observed above. In particular, it is expected that when there are numerous shocking areas they might overlap, thus making the sector space available to aircraft larger than one would have expected if they were not overlapping. Another feature worth of further investigation is the role of the directs that again improve the capacity of the considered flight level.

V. CONCLUSIONS AND FUTURE WORK

In this paper we have described the main features and the preliminary results of an agent based model aiming at modeling the management procedure of the air traffic system in the current ATM scenario. The model mainly consists of two layers, the strategic and the tactical one. The strategic layer aims at modeling the interactions between the Airline Operators and the network manager in the process that leads to the generation of the flight plans. The tactical layers aims at describing the interactions between air traffic controllers and aircraft/pilots in the process that lead to the actual flights trajectories.

The preliminary results for the strategic layer show that the proto "low-cost" and "traditional" Airline Operators have different advantages depending on the departure pattern and the level of mixture. Even if it is always better for companies to be surrounded by companies of the other type – hence displaying a "the loner, the better" effect, the dominant strategy might also depend on the distribution of departure times, in a non trivial way. For instance, on the contrary of the general case, it is sometimes better for "S" companies to have a more uniform distribution. Finally, we find that the advantage of "S" companies over "R" companies increases in a constrained environment, leading to a higher resilience for the type of shocks we considered. The preliminary results for the tactical layer indicate that when shocks in the system are confined in small areas, the interplay between the re-routing and change of flight level strategies may even lead to trajectory modifications that give smaller average delays as long as the number of shocks increases.

Future work will be needed to integrate the two layers into a truly single model: the output of the strategic layer would be a set of flight plans detailed at the level of navigation points to be fed into the tactical layer.

ACKNOWLEDGMENT

This work is co-financed by EUROCONTROL on behalf of the SESAR Joint Undertaking in the context of SESAR Work Package E - ELSA research project.

REFERENCES

[1] SESAR, "Definition of the future atm target concept - d3," 2007.

[2] "Commission regulation (EU) no 691/2010," 2010.

[3] EUROCONTROL, "Final report on european commissions mandate to support the establishment of functional airspace blocks (fabs)," 2005.

[4] Complex World - SESAR WP-E Research Network for the theme "Mastering Complex Systems Safely" , *The ComplexWorld Position Paper* , Version: August 2012, (Restricted audience)

[5] B. Chen, and H. H. Cheng. "A review of the applications of agent technology in traffic and transportation systems." Intelligent Transportation Systems, IEEE Transactions on 11.2, 485-497 (2010).

[6] A. Mota, J.M. Castro, L. P. Reis. Recovering from airline operational problems with a multi-agent system: a case study. *Progress in Artificial Intelligence* , pp. 461-472. (Springer Berlin, Heidelberg, 2009)

[7] G. Clare, A. Richards, J. Escartin, D. Martinez, J. Cegarra, L. J. Alvarez. Air Traffic Flow Management Under Uncertainty: Interactions Between Network Manager and Airline Operations Centre. In: Proceedings of Second SESAR Innovation Days, Braunschweig, 27-29 November 2012, Germany).

[8] K. M. Feight, A. R. Prichett, A.P. Shah, S. A. KAlaver, A. Jadhav, D. M. Holl, R. C. Bea, A. Z. Gilgur. Analyzing Air Traffic Management Systems Using Agent-based Modeling and Simulation. In: Proceedings of 6^{th} USA/Europe ATM R&D Seminar. (Baltimore, 27-30 June 2005, USA)

[9] A. K. Agogino, K. Tumer. A multiagent approach to managing air traffic flow. Auton Agent Multi-Agent Syst **24**, 1-25 (2012).

[10] S. R. Conway An agent-based model for analyzing control policies and the dynamic service-time performance of a capacity-constrained air traffic management facility In: Proceedings of ICAS 2006 - 25^{th} Congress of the International Council of the Aeronautical Sciences; (Hamburg, 3-8 September 2006, Germany).

[11] H. Trandac, P. Baptiste, V. Duong. Optimized sectorization of airspace with constratints. RAIRO - Operations Research **39**, 105-122 (2005).

[12] R. Ehrmanntraut, S. McMillan. Airspace design process for dynamic sectorization. In: Proceedings of: 26^{th} DASC 2007, (Dallas, 2007, USA)

[13] F. Lillo, S. Miccichè, R.N. Mantegna, V. Beato, and S. Pozzi. Elsa project: Toward a complex network approach to atm delays analysis. In: Proceedings of the SESAR Innovation Days, (Toulouse, November 2011, France).

[14] A. Agogino, K. Tumer. Regulating Air Traffic Flow with Coupled Agents. In: Proceedings of 7^{th} Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2008). (Estoril, 12-16 May 2008, Portugal).

[15] S. R. Wolfe, P. A. Jarvis, F. Y. Enomoto, M. Sierhuis. A Multi-Agent Simulation of Collaborative Air Traffic Flow Management. In: Edited Collection on Multi-Agent Systems for Traffic and Transportation (2009).

[16] G. Van Rossum; F. L. Drake Jr. *Python reference manual*. (Centrum voor Wiskunde en Informatica, Netherlands, 1995).

[17] B. W. Kernighan; D. M. Ritchie. *The C Programming Language* . (Prentice Hall, England, 1988).

[18] D. E. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. (Addison-Wesley Professional, 1 edition, 1989).