Collision Avoidance in the ATM Problem: A Mixed Integer Linear Optimization Approach

A. Alonso-Ayuso, L. F. Escudero, F.J. Martin-Campo

***Abstract*—This paper tackles the collision avoidance problem in ATM. The problem consists in deciding the best strategy for new aircraft configurations (velocity and altitude changes) such that all conflicts in the airspace are avoided; a conflict being the loss of the minimum safety distance that has to be kept between two aircrafts. A mixed 0-1 linear optimization model based on geometric transformations for collision avoidance between an arbitrary number of aircrafts in the airspace is developed. Knowing initial coordinates, angle direction and level flight, the new configuration for each aircraft is established by minimizing several objectives like velocity variation and total number of changes (velocity and altitude), and forcing to return to the original flight configuration when no aircrafts are in conflict. Due to the small computational time for the execution, the new configuration approach can be used in real time by using optimization software.**

***Index Terms*—Air traffic management, collision avoidance, mixed integer linear optimization.**

I. Introduction

ATM is currently based on prefixed routes that pilots have to follow according with a certain flight plan. Next years aim is extending the airspace considering “Free Flight”, where pilots and airlines can decide freely on the control of the flight, keeping in touch with air traffic controllers. To preserve safety in air flights, the Conflict Resolution Problem has been studied deeply from different points of view.

On a recent paper by EUROCONTROL [1], aimed to specify the required capabilities of Medium-Term Conflict Detection (MTCD) for Air Traffic Management Systems, the MTCD system is required to detect and notify the controller about the probable loss of the required separation between two aircrafts, an aircraft penetrating restricted airspace, or an aircraft blocking airspace that might have been used by some other one. That paper considers that, although flight data and trajectories are provided to the MTCD, some uncertainty is likely to be on the trajectories. It distinguishes too between tactical and planned trajectories. Kuchar and Yang (2000) [2] and references therein present a survey of conflict, detection and resolution modeling methods with their own classification. Obstacle avoidance using the linearized constrained Uninhabited Aerial Vehicle (UAV) dynamic has been modeled by Richards and How (2002) [3]. According with these authors the Centralized Model Predictive Control has been widely developed for constrained systems with many results concerning robustness and it has also been applied to the co-operative control of multiple vehicles. By augmenting the system with a binary “target state” indicating whether the target set is reached or not, the authors end up with a hybrid system at hand. Task completion is then guaranteed by imposing a hard terminal equality constraint on the target state. Dell’Olmo and Lulli (2003) [4] describe a model that is solved by using exact optimization software combined with an heuristic approach for large problems. Christodoulou and Costoulakis (2004) [5] propose a Mixed Integer Nonlinear Optimization approach to solve the conflict problem. Their method allows velocity changes and heading angle control to solve all potential conflicts by only using standard optimization software, but it may require more computational effort than what it could be affordable.

The main contribution of this paper is based on the VC (Velocity Changes) model proposed by Pallottino [6] and Pallottino, Feron and Bicchi (2002) [7]. See also [8]. This model considers instantaneous changes. The main extensions to the VC model are as follows: (1) Different safety radius are considered since the safety radius for an aircraft can be adjusted differently. This feature can be applied to include the wind factor in the model, extending the safety radius if bad weather conditions are existing; (2) Altitude changes are allowed to avoid infeasible situations in the VC problem caused by the velocity bounds, or “head to head” conflict situations; (3) All changes in the aircrafts configurations are updated since aircrafts with higher number of changes will be penalized for the equitable distribution of the maneuvers; (4) All aircrafts will be forced to return to the initial configuration when the conflict situations are avoided; and (5) the case discussed in [7] where the denominator is zero, causing physical collisions between the corresponding aircrafts, is avoided in our proposed model. For this purpose a mixed integer linear optimization (MILO) is proposed. The required computing time for optimizing realistic sets of aircrafts in conflict is so small that the approach can be used in real time operations.

This paper is organized as follows: In Section II the general features of the problem required to build the MILO model are described as well as some changes in the VC model. In Section III the formulation of our proposed model is developed. Section IV presents the full problem formulation as well as the dimensions of the model. Section V reports the results of the computational experimentation to verify the efficiency of the proposal and its application in real time. Finally, Section VI presents some conclusions and the main lines of future research.

A. Alonso-Ayuso, L. F. Escudero and F.J. Martin-Campo are with the Department of Statistics and Operations Research, University Rey Juan Carlos, Madrid-Spain. Corresponding author: javier.martin.campo@urjc.es

II. Problem Statement

Aerial sectors and a given number F of aircrafts flying in an aerial sector as well as their configurations are considered. An aerial sector being an airspace portion supervised by an ATC.

Next, all elements concerning velocity and altitude changes model (VAC) are detailed:

Sets

- $\mathcal{F}$, set of aircrafts in the sector $(1, \ldots, F)$.
- $\mathcal{Z}^f$, set of admissible flight levels for aircraft $f \in \mathcal{F}$, $(1, \ldots, Z)$

Parameters

- x_f, y_f, the position (abscissa and ordinate) of aircraft f, for $f \in \mathcal{F}$.
- z_f, initial flight level in the current execution for aircraft f, for $f \in \mathcal{F}$.
- z_f^*, initial flight level configuration for aircraft f, for $f \in \mathcal{F}$.
- v_f, initial velocity in the current execution for aircraft f, for $f \in \mathcal{F}$.
- v_f^*, initial velocity configuration for aircraft f, for $f \in \mathcal{F}$.
- $\widehat{v_f}$, optimal velocity configuration to arrive at the destination sector point at the predicted time for aircraft f, for $f \in \mathcal{F}$.
- t_p, current time for aircraft f, for $f \in \mathcal{F}$.
- $\underline{v_f}, \overline{v_f}$, minimum and maximum velocity allowed for each aircraft f, respectively, for $f \in \mathcal{F}$.
- m_f^*, initial direction of motion in $(-\pi, \pi]$ for aircraft f, for $f \in \mathcal{F}$. See the θ parameter in [7].
- r_f, safety radius for each aircraft, usually 2.5 nautical miles for $f \in \mathcal{F}$. This parameter will be used to consider different safety radii for each aircraft.
- hth_{ij}, 0-1 parameter that determines if there is a “head to head” conflict for aircrafts i, j, for $i < j \in \mathcal{F}$. This parameter is obtained in preprocessing.
- sc_{ij}, 0-1 parameter that determines if two aircrafts i and j have the same coordinates x and y, for $i < j \in \mathcal{F}$. This parameter is obtained in preprocessing.
- pc_{ij}, 0-1 parameter that determines if there is a “pathological case” between aircrafts i, j, for $i < j \in \mathcal{F}$; see below. This parameter is obtained in preprocessing.
- p_{ij}, 0-1 parameter that determines if the intersection point between the aircraft trajectories for aircrafts i, j, for $i < j \in \mathcal{F}$, is less than a fixed distance e. This parameter is obtained in preprocessing.
- r, random angle used if it is necessary a rotation transformation for the airspace.
- n_f^v, n_f^a number of changes in velocity and in altitude in the sector for aircraft f until the new execution, respectively, for $f \in \mathcal{F}$.

Variables

- q_f, velocity variation for aircraft f, for $f \in \mathcal{F}$. This variable is real, and we divide it in two nonnegative variables, say, q_f^+ and q_f^-, such that $q_f = q_f^+ - q_f^-$ as traditional in optimization, where q_f^+ and q_f^- are the positive and negative velocity variation for aircraft f, for $f \in \mathcal{F}$.
- ν_f^z, 0-1 variable that takes value 1 if aircraft f is at altitude level z at the end of the current execution, for $f \in \mathcal{F}$, $z \in \mathcal{Z}^f$ and, otherwise, it is zero.
- a_f, 0-1 variable that takes value 1 if aircraft f changes its velocity at the end of the current execution for aircraft f in the sector, for $f \in \mathcal{F}$ and, otherwise, it is zero.
- b_f, 0-1 variable that takes value 1 if aircraft f changes its altitude at the end of the current execution for aircraft f, for $f \in \mathcal{F}$, and, otherwise, it is zero.
- ρ_f, nonnegative integer variable that shows the number of levels that the aircraft f ascends or descends, for $f \in F$.
- β_f, auxiliary nonnegative continuous variable that model the absolute value of the difference between current velocities and initial velocities as a linear function for aircraft f, for $f \in \mathcal{F}$.
- δ_{ijz}^n auxiliary 0-1 variables to model or-constraint types, for $i, j \in \mathcal{F}$, $z \in \mathcal{Z}^i \cup \mathcal{Z}^j$, $n = 1, ..., 5$.

The problem consists in avoiding all conflicts in a certain aerial sector by using a mixed integer linear optimization formulation. Then, some standard optimization software will be used to solve this problem.

III. Conflict Avoidance Constraints for the VAC Problem

A solution to the conflict problem does not always exist in the VC model, since cases such as “head to head” and others in which the velocity bounds are insufficient to avoid collisions, cannot be solved by only applying velocity changes. To avoid these situations, a extension to the VC problem including altitude changes is proposed, resulting in the VAC (Velocity and Altitude Changes) problem.The VC problem considers that all aircraft safety radii have the same value $r_f = s/2$ where s is 5 nautical miles and, under this assumption builds all α angles based on symmetric geometry. Considering different safety radii constitutes a good approximation to the realistic problem, since each aircraft has a different configuration depending on the aircraft weight, the aerodynamic configuration, the aircraft size, etc. When different aircraft radii are considered, two interior tangent lines to two circumferences have to be computed as well as one of these two straight lines slope.

Now, obtaining the α_{ij} angle is easy by using the archtangent of this slope:

$$\alpha_{ij} = \arctan\left(\frac{r_i + r_j}{\sqrt{d_{ij}^2 - (r_i + r_j)^2}} \right),$$

where d_{ij} is the distance between the aircrafts i and j. If it is preferred, the α angle can be used to calculate the new l angle or the r angle, see [7], by considering that all safety radii are the same. If the wind factor only acts in one direction the previous is also useful to take into account this factor. This calculation is only valid if the two aircrafts distance is greater than $r_i + r_j$. If the distance is equal to $r_i + r_j$, the slope tends to infinity, since there is only one tangent point. If the distance is less than $r_i + r_j$ the two circumferences have a non empty intersection and only exterior tangent straight lines exist.

With the inclusion of altitude levels, the model of the VC problem has to be expanded, since it can happen that two different aircrafts have a conflict in which a velocity change is insufficient to avoid the conflict situation. However, if the aircrafts fly at different altitude levels, then there will be no a conflict. In order to solve these conflict situations, altitude changes will be taken into account and, therefore, some changes in the VC problem will be introduced.

Notice that if altitude changes are considered, the δ variables in the VC problem have to be modified including the dimension of the altitude.

Additionally, a new variable δ^5_{ijz} will be included in the model to avoid infeasible situations that could occur in the sum of the δ variables constraint in the VC model. This new variable will take value 0 if there is a conflict between the aircrafts i and j at the same level z; and it will take value 1 if there is no conflict at the same level between these aircrafts. The use of this new variable is advantageous in the sense that it is able to detect infeasible situations given by the velocity bounds when two different aircrafts are nearby or an aircraft flies faster than other one, for instance.

All VC model constraints must include the dimension z in the δ variables. Also the sum of the δ variables has to be 1. Now, if two different aircrafts fly at different levels, the variable δ^5_{ijz} will be forced to take value 1, i.e., if $\nu^z_i + \nu^z_j \leqslant 1 \Rightarrow \delta^5_{ijz} = 1$, and if $\nu^z_i + \nu^z_j \geqslant 2 \Rightarrow \delta^5_{ijz} = 0$. Thus, the inclusion of the following constraints achieves our purpose:

$$\begin{aligned}\nu^z_i + \nu^z_j &\geqslant 2(1 - \delta^5_{ijz}) \quad \forall i, j \in \mathcal{F} : i < j, \forall z \in \mathcal{Z}^i \cap \mathcal{Z}^j\\ \nu^z_i + \nu^z_j - 1 &\leqslant 1 - \delta^5_{ijz} \quad \forall i, j \in \mathcal{F} : i < j, \forall z \in \mathcal{Z}^i \cap \mathcal{Z}^j.\end{aligned}$$

It is easy to check that if $\nu^z_i + \nu^z_j = 2$, then the two aircrafts fly at the same level and the second constraint forces δ^5_{ijz} to take value 0. On the other hand, if $\nu^z_i + \nu^z_j \leqslant 1$, the two aircrafts fly at different levels and the first constraint forces δ^5_{ijz} to take value 1.

This argument helps to fix the value of some δ^5 variables and strengths the continuous relaxation of the model. This is an important aspect in terms of execution time when using an optimization engine for resolution since it reinforces the model of the VAC problem and this leads to a faster solution.

The model approach presented in this paper forces flying aircrafts to lie at different altitude levels in case of a "head to head" conflict. In a first approach, preprocessing can detect "head to head" conflicts and then, a parameter named hth can be fixed to the value 1, while the rest of the hth parameters shall be fixed to value 0. All "head to head" cases between every pair of aircrafts can be detected in preprocessing and these cases occur when the following conditions are satisfied (see Fig. 1):

$$\begin{aligned}\widehat{\omega}_{ij} - \alpha_{ij} &\leqslant m^*_i \leqslant \widehat{\omega}_{ij} + \alpha_{ij}\\ \widehat{\omega}_{ji} - \alpha_{ji} &\leqslant m^*_j \leqslant \widehat{\omega}_{ji} + \alpha_{ji},\end{aligned}$$

where $\widehat{\omega}_{ij}$ depends in the quadrant on which (x_j, y_j) lies considering (x_i, y_i) centered in the origin. That is, $\widehat{\omega}_{ij} = \omega_{ij}$ if it is on the first or on the fourth quadrant; $\widehat{\omega}_{ij} = \omega_{ij} + \pi$ if is on the second one; and $\widehat{\omega}_{ij} = \omega_{ij} - \pi$ if is on the third one. Then, in this situation, the parameter hth_{ij} is fixed to 1, forcing the two involved aircrafts to fly at different altitude levels, since previously the δ^5 variables are fixed to 1.

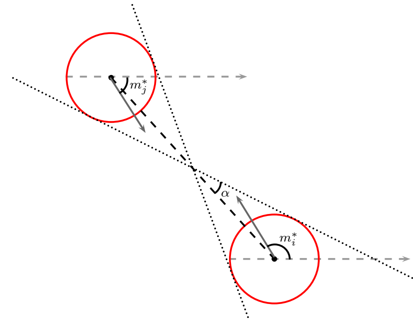

Fig. 1. "Head to head" conflict.

The new altitude configuration will be saved in the ν variables and the following constraint ensures that each aircraft flies at one and only one level:

$$\sum_{z \in \mathcal{Z}^f} \nu^z_f = 1 \quad \forall f \in \mathcal{F}. \tag{3}$$

Now, when a "head to head" conflict occurs, the VAC model forces the two aircrafts in conflict to have different altitude levels as follows:

$$\delta^5_{ijz} = 1 \quad \text{if } hth_{ij} = 1, \forall i, j \in \mathcal{F} : i < j, \forall z \in \mathcal{Z}^i \cap \mathcal{Z}^j.$$

It can occur that two aircrafts have similar coordinates and they must fly at different altitude levels to avoid collisions. The parameter sc_{ij} will take the value 1 if the distance between two different aircrafts is less or equal to $r_i + r_j$, i.e., the safety distance, and it is zero otherwise. The δ^5 variables can be fixed to 1 in preprocessing as follows:

$$\delta^5_{ijz} = 1 \quad \text{if } sc_{ij} = 1, \forall i, j \in \mathcal{F} : i < j, \forall z \in \mathcal{Z}^i \cap \mathcal{Z}^j.$$

These previous constraints are used to fix some δ^5 variables to help speeding up the execution but they are not indispensable for the VAC model, since detecting infeasible situations is autonomous. The two previous constraints can be joined in one:

$$\begin{aligned}\delta^5_{ijz} = 1 \qquad &\text{if } hth_{ij} + sc_{ij} \geqslant 1,\\ &\qquad \forall i, j \in \mathcal{F} : i < j, \forall z \in \mathcal{Z}^i \cap \mathcal{Z}^j.\end{aligned}$$

The altitude changes are the cheapest ones in terms of fuel costs, but are the most expensive ones in terms of passenger comfort. In a first approach, the VAC model could force to aircrafts to climb or descend only one level. To avoid possible changes of more than one altitude level the following constraint is used:

$$\nu^z_f \leqslant 0 \quad \forall f \in \mathcal{F}, \forall z \in \mathcal{Z}^f : z \neq z_f, z \neq z_f \pm 1. \tag{4}$$

The constraint (4) may cause infeasible situations if the airspace is very busy and only one change in altitude level is insufficient to solve the problem efficiently. In a busy airspace, it can occur that the VAC model has infeasible situations when aircrafts can not change more than one level. In these cases, we must allow the aircrafts to change two or more levels, but forcing these changes to be as small as possible. For this aim, a new variable say $\rho_f, \forall f \in \mathcal{F}$ will be included in the model. It will store the number of changes that an aircraft climbs or descends. This number of changes can be greater than one and it will avoid infeasible situations in a busy airspace. The number of levels changed by an aircraft can be modeled as follows:

$$\rho_f = \Big| \sum_{z \in \mathcal{Z}^f} z\nu_f^z - z_f \Big|.$$

The first addend gives the new level in which the aircraft must be after the optimization, since all ν variables take value 0 except the ν variable in $z \in \mathcal{Z}^f$ in which the aircraft flies after the execution. The second addend gives the initial level in which the aircraft flies before the optimization. The absolute value makes the previous expression positive. Notice that this function is not linear so it has to be transformed. For this purpose, the maximum function between the differences $\sum_{z \in \mathcal{Z}^f} z\nu_f^z - z_f$ and $z_f - \sum_{z \in \mathcal{Z}^f} z\nu_f^z$ is taken. That is,

$$\rho_f = \Big| \sum_{z \in \mathcal{Z}^f} z\nu_f^z - z_f \Big| = \max\Big\{ \sum_{z \in \mathcal{Z}^f} z\nu_f^z - z_f, z_f - \sum_{z \in \mathcal{Z}^f} z\nu_f^z \Big\}.$$

The maximum function is also nonlinear but it can be easily modeled with two additional constraints as follows:

$$\rho_f \geqslant \sum_{z \in \mathcal{Z}^f} z\nu_f^z - z_f \qquad \forall f \in \mathcal{F} \tag{5a}$$

$$\rho_f \geqslant z_f - \sum_{z \in \mathcal{Z}^f} z\nu_f^z \qquad \forall f \in \mathcal{F}. \tag{5b}$$

The constraints (5) force ρ_f to be the maximum between the two expressions above, because ρ_f must be higher or equal than the two expressions and ρ_f only can be greater or equal than one expression, except there is no level change. In this case, both (5a) and (5b) are trivially fulfilled, ρ_f being zero. Now, all constraints are linear ones.

In the VAC model, an objective function consists in minimizing the number of velocity and altitude changes executed by each aircraft. Since the algorithm will be iteratively executed updating each aircraft changes number is necessary to balance the number of maneuvers performed by each aircraft. In a first approach the velocity changes number will be counted. With constraints (6), it happens that variable a_f takes value 1 if a velocity change occurs and this occurs if $|q_f| \neq 0$. The absolute value is a nonlinear function and expressing it as a linear one in the traditional optimization way is required: $|q_f| = q_f^+ + q_f^-$ where $q_f^+ = \max\{q_f, 0\}$ is the positive part of q_f and $q_f^- = \max\{-q_f, 0\}$ is the negative part of q_f. These two new variables are positive and $q_f^+ + q_f^- \neq 0$ is equivalent to $q_f^+ + q_f^- > 0$. Thus,

$$M_1(1 - a_f) + \varepsilon \leqslant q_f^+ + q_f^- \qquad \forall f \in \mathcal{F} \tag{6a}$$

$$q_f^+ + q_f^- \leqslant M_2 a_f \qquad \forall f \in \mathcal{F}, \tag{6b}$$

where $M_1 = \underline{v_f} - \overline{v_f}$ is the lower bound of $q_f^+ + q_f^-$; $M_2 = \overline{v_f} - \underline{v_f}$ is the lower bound of $a_f - n_f^v - 1$ and ε is an infinitesimal parameter. It is easy to see that if there is a velocity change for aircraft $f \in \mathcal{F}$, then $q_f^+ + q_f^- \neq 0$, and the second constraint forces a_f to take the value 1. On the other hand, if there is not a velocity change for aircraft $f \in \mathcal{F}$, then $q_f^+ + q_f^- = 0$, and the first constraint forces a_f to take the value 0.

The next step is changing the value of b_f from 0 to 1 if there is an altitude change. With the following constraint, the number of altitude changes is updated:

$$b_f = 1 - \nu_f^{z_f} \quad \forall f \in \mathcal{F},$$

since if aircraft f flies at the same level before and after the current execution, then the variable $\nu_f^{z_f}$ will take the value 1 and the difference will be 0, i.e., the aircraft does not change its altitude level. On the other hand, if the aircraft f flies at different altitude levels before and after the current execution, $\nu_f^{z_f}$ will be 0 and b_f will take value 1, i.e., the aircraft changes its altitude level.

Pathological cases may happen in the VC model, since null denominators may appear when implicitly computing the relative velocity tangent vectors in the VC model. These pathological cases may cause unstable situations in which conflicts between the involved aircrafts cannot be solved.

One of the most relevant contributions of this paper is also the treatment of these pathological cases. The proposed VAC model implicitly detects them in preprocessing. When a pathological case occurs between aircrafts i and j, the conflict is sorted out by turning only the motion angles of the involved aircrafts.

To detect posible pathological cases between two aircrafts in preprocessing a 0-1 parameter is used. This parameter value will be one in case the relative velocity tangent vector tends to infinity and will be zero otherwise. Only pairs of aircrafts with the same abscissa coordinate are considered. The formal definition of the parameter being:

$$pc_{ij} = \begin{cases} 1 & \text{if } |\omega_{ij}| > \frac{\pi}{2} - \alpha_{ij} - 1^{\text{o}} \\ 0 & \text{otherwise.} \end{cases}$$

All conflicts where the abscissa of the positions is the same for both of the two aircrafts will be taken into account since there might be null denominator. In these cases, we turn $\frac{\pi}{2}$ radians the configuration of these two aircrafts i and j but considering the same velocity. This parameter will be included in the VAC model, so that all pathological cases will be avoided, and detecting the existence of a null denominator and solving again will not be required, thus reducing computational costs. When pc_{ij} is equal to 1, the initial configuration parameters depending on some m_f^* angle of aircrafts i and j will be turned $\frac{\pi}{2}$ radians. Both constraints in each case of the VC model, are rewritten as follows:

$$(v_i + q_i)\big(\cos(m_i^*)(1 - pc_{ij}) - \sin(m_i^*)pc_{ij}\big) - (v_j + q_j)\big(\cos(m_j^*)(1 - pc_{ij}) - \sin(m_j^*)pc_{ij}\big) \leqslant M'_m(1 - \delta_{ijz}^n)$$

$$-(v_i + q_i)\big(h_i(1 - pc_{ij}) + \hat{h}_i pc_{ij}\big) + (v_j + q_j)\big(h_j(1 - pc_{ij}) + \hat{h}_j pc_{ij}\big) \leqslant M''_m(1 - \delta_{ijz}^n),$$

where $\cos(m_i^* + \frac{\pi}{2}) = -\sin(m_i^*)$, $M'_m = (\overline{v_i} + \overline{v_j})$ and $\hat{h}_i$ and $\hat{h}_j$ are the new parameters h_i and h_j built by using the new turned angles. The M''_m value in these constraints will be as follows:

$$M''_m = (\overline{v_i}|h_i| + \overline{v_j}|h_j|)(1 - pc_{ij}) + (\overline{v_i}|\hat{h}_i| + \overline{v_j}|\hat{h}_j|)pc_{ij}.$$

Finally, the objective function will be constructed including all factors and different objectives that are relevant for the problem. They are detailed below.

First of all, the first objective is minimizing the velocity changes absolute value, such that velocity changes and an early arrival or an arrival delay to destination point in each aerial aircraft sector is smoothed. To make early arrivals or arrival delays as small as possible, minimizing velocity variations using the absolute value function to avoid high changes in the initial flight plan is proposed. This is done as follows

$$\min \sum_{f \in \mathcal{F}} |q_f| = \min \sum_{f \in \mathcal{F}} (c_f^{q^+} q_f^+ + c_f^{q^-} q_f^-).$$

Different objectives are considered but with different magnitudes. All magnitudes involved in velocity changes expressions are normalized (between 0 and 1) as follows:

$$\min \sum_{f \in \mathcal{F}} \Big(\frac{q_f^+}{\overline{v}_f - \underline{v}_f} + \frac{q_f^-}{\overline{v}_f - \underline{v}_f} \Big).$$

Only q_f^+ or q_f^- will take value greater than zero, and the objective function will make one of these variables equal to zero. Notice that $\overline{v}_f - \underline{v}_f$ is the upper bound of a velocity variation for an aircraft $f \in \mathcal{F}$.

In case of a very congested airspace, an aircraft may need to climb or descend more than one level thus provoking unfeasible situations. A new term is introduced in the objective function to avoid unfeasibility. This option is modeled by adding the following term to the objective function as done with (5); and constraint (3) will be removed,

$$\min \sum_{f \in \mathcal{F}} c_f^j \rho_f.$$

Another aspect to minimize is the weighted number of velocity and altitude changes in each aircraft aerial sector. Notice that the costs must be crescent to penalize several changes in the same aircraft. The function is as follows:

$$\min \sum_{f \in \mathcal{F}} \big(c_f^v (n_f^v + a_f) + c_f^a (n_f^a + b_f) \big).$$

Returning all aircraft configurations to the initial ones, both in velocity and altitude is desirable. A new term can be included so that the aircrafts return again to their initial configurations when they are not in conflict. To model this term, the difference between the current velocity and the initial velocity configuration in the initial flight plan is penalized. A new term can also be included to penalize the difference between the current level and the initial level configuration with the ν variable.

The objective relative to return to the initial velocity configuration consists in forcing the aircrafts to arrive at the destination sector point in the predicted time. For this purpose, some observations about geometric constructions are necessary; take Fig. 2 as support.

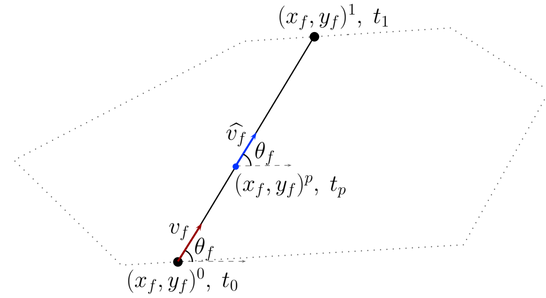

Fig. 2. Return to the initial configuration time

The distance between points (x_0, y_0) and (x_1, y_1) can be known in two ways:

$$dist((x_f, y_f)^0, (x_f, y_f)^1) = v_f(t_1 - t_0)$$
$$dist((x_f, y_f)^0, (x_f, y_f)^1) = \sqrt{(x_f^1 - x_f^0)^2 + (y_f^1 - y_f^0)^2}.$$

Hence,

$$v_f(t_1 - t_0) = \sqrt{(x_f^1 - x_f^0)^2 + (y_f^1 - y_f^0)^2}.$$

If by taking a generic point $(x_f, y_f)^p$ in the aircraft trajectory at time t_p with velocity $\widehat{v_f}$, and substituting it in the previous expressions, it is obtained

$$dist((x_f, y_f)^p, (x_f, y_f)^1) = \widehat{v_f}(t_1 - t_p)$$
$$dist((x_f, y_f)^p, (x_f, y_f)^1) = \sqrt{(x_f^1 - x_f^p)^2 + (y_f^1 - y_f^p)^2},$$

where the velocity an aircraft might have to arrive at the destination point in the predicted time is obtained by combining the previous expressions. Thus,

$$\widehat{v_f} = \frac{\sqrt{(x_f^1 - x_f^p)^2 + (y_f^1 - y_f^p)^2}}{t_1 - t_p}.$$

Therefore, the difference between current and optimal velocities to arrive at the destination point in the predicted time is penalized as follows:

$$\min \sum_{f \in \mathcal{F}} |v_f + q_f - \widehat{v_f}| = \min \sum_{f \in \mathcal{F}} \beta_f.$$

This function is non linear, but it can be modeled as a linear function by using a maximum function like in (5). The addend can be descomposed as $\beta_f = v_f + q_f - \widehat{v_f} = (v_f + q_f^+) - (q_f^- + \widehat{v_f})$, and using it, two additional constraints are necessary:

$$\beta_f \geqslant (v_f + q_f^+) - (q_f^- + \widehat{v_f})$$
$$\beta_f \geqslant (q_f^- + \widehat{v_f}) - (v_f + q_f^+).$$

A term to penalize the difference between the current level and the initial level configuration is added with the ν variable. An aircraft takes its initial level configuration if $\nu_f^{z_f^*} = 1$, therefore, the new objective function will be as follows,

$$\max \sum_{f \in \mathcal{F}} \nu_f^{z_f^*} = \min \sum_{f \in \mathcal{F}} -\nu_f^{z_f^*}.$$

Moreover, the objective function terms and some new costs to model other preferences like fuel consumption can be added. Also, any linear combination of all them is possible.

IV. Problem Formulation

Next, the full VAC model is presented, where all unstable situations are considered and solved.

$$\min c^q \sum_{f\in\mathcal{F}} \frac{q_f^+}{\overline{v}_f - \underline{v}_f} + \sum_{f\in\mathcal{F}} \frac{q_f^-}{\overline{v}_f - \underline{v}_f} + c^{n^j} \sum_{f\in\mathcal{F}} c_f^j \rho_f \tag{11}$$

subject to:

$$\underline{v_f} \leqslant v_f + q_f \leqslant \overline{v_f} \quad \forall f \in \mathcal{F} \tag{12}$$

$\forall i,j \in \mathcal{F} : i < j, \forall z \in \mathcal{Z}^i \cap \mathcal{Z}^j$:

$$\begin{aligned}(v_i+q_i)\big(\cos(m_i^*)(1-pc_{ij}) - \sin(m_i^*)pc_{ij}\big) - \\ (v_j+q_j)\big(\cos(m_j^*)(1-pc_{ij}) - \sin(m_j^*)pc_{ij}\big) \\ \leqslant (\overline{v_i}+\overline{v_j})(1-\delta_{ijz}^1)\end{aligned} \tag{13}$$

$$\begin{aligned}&-(v_i+q_i)\big(h_i(1-pc_{ij}) + \hat{h}_i pc_{ij}\big) \\ &+ (v_j+q_j)\big(h_j(1-pc_{ij}) + \hat{h}_j pc_{ij}\big) \\ &\leqslant \big((\overline{v_i}|h_i| + \overline{v_j}|h_j|)(1-pc_{ij}) \\ &+ (\overline{v_i}|\hat{h}_i| + \overline{v_j}|\hat{h}_j|)pc_{ij}\big)(1-\delta_{ijz}^1)\end{aligned} \tag{14}$$

$$\begin{aligned}(v_i+q_i)\big(\cos(m_i^*)(1-pc_{ij}) - \sin(m_i^*)pc_{ij}\big) - \\ (v_j+q_j)\big(\cos(m_j^*)(1-pc_{ij}) - \sin(m_j^*)pc_{ij}\big) \\ \leqslant (\overline{v_i}+\overline{v_j})(1-\delta_{ijz}^2)\end{aligned} \tag{15}$$

$$\begin{aligned}&(v_i+q_i)\big(k_i(1-pc_{ij}) + \hat{k}_i pc_{ij}\big) \\ &- (v_j+q_j)\big(k_j(1-pc_{ij}) + \hat{k}_j pc_{ij}\big) \\ &\leqslant \big((\overline{v_i}|k_i| + \overline{v_j}|k_j|)(1-pc_{ij}) \\ &+ (\overline{v_i}|\hat{k}_i| + \overline{v_j}|\hat{k}_j|)pc_{ij}\big)(1-\delta_{ijz}^2)\end{aligned} \tag{16}$$

$$\begin{aligned}-(v_i+q_i)\big(\cos(m_i^*)(1-pc_{ij}) - \sin(m_i^*)pc_{ij}\big) + \\ (v_j+q_j)\big(\cos(m_j^*)(1-pc_{ij}) - \sin(m_j^*)pc_{ij}\big) \\ \leqslant (\overline{v_i}+\overline{v_j})(1-\delta_{ijz}^3)\end{aligned} \tag{17}$$

$$\begin{aligned}&(v_i+q_i)\big(h_i(1-pc_{ij}) + \hat{h}_i pc_{ij}\big) \\ &- (v_j+q_j)\big(h_j(1-pc_{ij} + \hat{h}_j pc_{ij})\big) \\ &\leqslant \big((\overline{v_i}|h_i| + \overline{v_j}|h_j|)(1-pc_{ij}) \\ &+ (\overline{v_i}|\hat{h}_i| + \overline{v_j}|\hat{h}_j|)pc_{ij}\big)(1-\delta_{ijz}^3)\end{aligned} \tag{18}$$

$$\begin{aligned}-(v_i+q_i)\big(\cos(m_i^*)(1-pc_{ij}) - \sin(m_i^*)pc_{ij}\big) + \\ (v_j+q_j)\big(\cos(m_j^*)(1-pc_{ij}) - \sin(m_j^*)pc_{ij}\big) \\ \leqslant (\overline{v_i}+\overline{v_j})(1-\delta_{ijz}^4)\end{aligned} \tag{19}$$

$$\begin{aligned}&-(v_i+q_i)\big(k_i(1-pc_{ij}) + \hat{k}_i pc_{ij}\big) \\ &+ (v_j+q_j)\big(k_j(1-pc_{ij}) + \hat{k}_j pc_{ij}\big) \\ &\leqslant \big((\overline{v_i}|k_i| + \overline{v_j}|k_j|)(1-pc_{ij}) \\ &+ (\overline{v_i}|\hat{k}_i| + \overline{v_j}|\hat{k}_j|)pc_{ij}\big)(1-\delta_{ijz}^4)\end{aligned} \tag{20}$$

$$\delta_{ijz}^1 + \delta_{ijz}^2 + \delta_{ijz}^3 + \delta_{ijz}^4 + \delta_{ijz}^5 = 1 - p_{ij} \tag{21}$$

TABLE I
VAC model dimension

Variables q_f^+: F	Variables q_f^-: F
Variables ν_f^z: FZ	Variables a_f: F
Variables b_f: F	Variables ρ_f: F
Variables β_f: F	Variables δ_{ijz}^n: $5Z\frac{(F-1)F}{2}$

(a) Number of variables

C. (12): $2F$	C. (13)-(21): $9Z\frac{F(F-1)}{2}$
C. (22): $Z\frac{F(F-1)}{2}$	C. (23)-(24): $ZF(F-1)$
C. (25): $2F$	C. (26)-(27): $2F$
C. (28): F	C. (29)-(30): $2F$
C. (31)-(32): $2F$	C. (33)-(34): $2F$

(b) Number of constraints

$\forall i,j \in \mathcal{F} : i < j, \forall z \in \mathcal{Z}^i \cap \mathcal{Z}^j$:

$$\delta_{ijz}^5 = 1 \qquad \text{if } hth_{ij} + sc_{ij} \geqslant 1 \tag{22}$$

$$\nu_i^z + \nu_j^z \geqslant 2(1-\delta_{ijz}^5) \tag{23}$$

$$\nu_i^z + \nu_j^z \leqslant 2 - \delta_{ijz}^5 \tag{24}$$

$\forall f \in \mathcal{F}$:

$$\sum_{z\in\mathcal{Z}^f} \nu_f^z = 1 \tag{25}$$

$$(\underline{v_f} - \overline{v_f})(1-a_f) + \varepsilon \leqslant q_f^+ + q_f^- \tag{26}$$

$$q_f^+ + q_f^- \leqslant (\overline{v_f} - \underline{v_f})a_f \tag{27}$$

$$1 - \nu_f^{z_f} = b_f \tag{28}$$

$$\sum_{z\in\mathcal{Z}^f} z\nu_f^z - z_f \leqslant \rho_f \tag{29}$$

$$z_f - \sum_{z\in\mathcal{Z}^f} z\nu_f^z \leqslant \rho_f \tag{30}$$

$$(v_f + q_f^+) - (q_f^- + v_f^*) \leqslant \beta_f \tag{31}$$

$$(q_f^- + v_f^*) - (v_f + q_f^+) \leqslant \beta_f \tag{32}$$

$$(v_f + q_f^+) - (q_f^- + \widehat{v_f}) \leqslant \beta_f \tag{33}$$

$$(q_f^- + \widehat{v_f}) - (v_f + q_f^+) \leqslant \beta_f \tag{34}$$

$\forall f \in \mathcal{F}$:

$$q_f \in \mathbb{R} \tag{35}$$

$$q_f^+, q_f^-, \beta_f \in \mathbb{R}^+ \tag{36}$$

$$\rho_f \in \mathbb{Z}^+ \tag{37}$$

$\forall f,i,j \in \mathcal{F} : i < j, \forall z \in \mathcal{Z}$:

$$\nu_f^z, a_f, b_f, \delta_{ijz}^1, \delta_{ijz}^2, \delta_{ijz}^3, \delta_{ijz}^4, \delta_{ijz}^5 \in \{0,1\} \tag{38}$$

Table I shows the dimension of the VAC model. Notice that model dimension depends on the objective function of choice.

V. Case Studies

The VAC model has been initially tested with the case shown in Fig. 3 by using the objective function introduced above, where velocity variations and the levels that the aircrafts have to change are minimized. In this objective function $c_f^{q^+} = 1$, $c_f^{q^-} = 1$ and $c_f^j = 1$, $\forall f \in \mathcal{F}$. The parameter p_{ij} is not considered, therefore all pairs of aircrafts are taken into

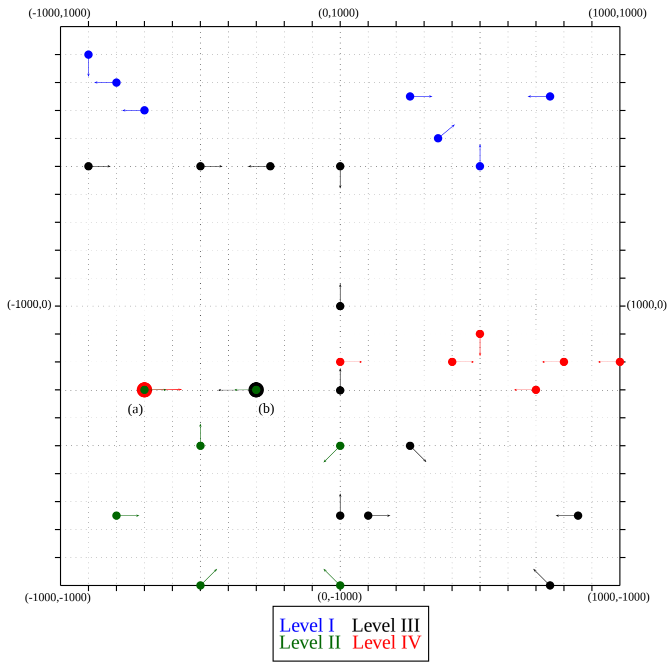

Fig. 3. Testing the VAC model. Initial situation

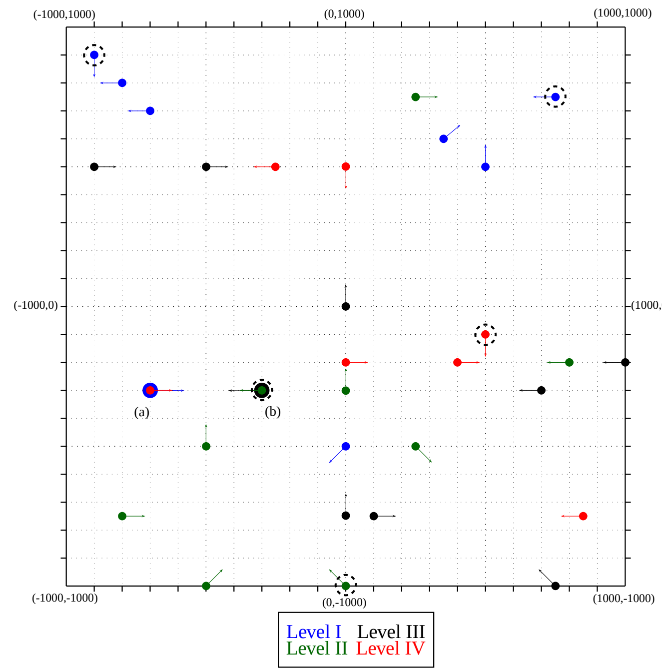

Fig. 4. Results for the VAC model in case of Fig. 3

account. Notice that we are considering $F = 33$ aircrafts in possible conflict. Note: There are two aircrafts in each point (a) and (b).

Fig. 4 depicts the results of applying the VAC model for collision avoidance in the case shown in Fig. 3 as follows:

- 5 velocity changes have been performed. The dotted line circles in the figure denote the positive variation of velocity, and the dashed line circle denotes the negative variation of velocity. In this case, all velocity changes are negative (slowdown).
- The points (a) and (b) still have two aircrafts each, but the conflict has been avoided.
- There are 11 altitude level changes, 4 positive changes and 7 negative changes, one of which descends two levels.
- The number of constraints, and continuous and 0-1 variables in the reduced MIP have been 6850, 33 and 3431, respectively.
- The objective function value is 13.195700.
- The execution time has been 6.77 seconds by using the optimization engine CPLEX v.11.2 [9] (with the default options) in the following HW/SW platform: Intel Core 2DUO P8400, 2.26GHz, 2GB RAM; Microsoft Windows XP Professional SO.

Next, some computational experience for the VAC model is reported. 25 random simulations are performed for each dimensional case, and the result averages are presented. Table II shows the dimensions of the model whereas Table III shows the most important results. The headings are as follows: *Case*: Gives the case configuration: the number of aircrafts and levels that are considered; m: Number of constraints; n: Number of variables; nel: Number of nonzero elements; $density$: The density of the matrix; m^*: Number of constraints after preprocessing by CPLEX; n^*: Number of variables after preprocessing by CPLEX; nel^*: Number of nonzero elements after preprocessing by CPLEX; $density^*$: The density of the matrix after preprocessing by CPLEX; z_{lp}: Value of the objective function in the continuous linear relaxation; z_s: Value of the bound after performing the cut identification and appending at node 0; z_{ip}: Value of the objective function for the optimal solution of the problem; GAP_{lp}: $\frac{z_{ip}-z_{lp}}{z_{ip}}$ %; GAP_s: $\frac{z_{ip}-z_s}{z_{ip}}$ %; nb: Number of times that there is branching; nn: Number of CPLEX branch-and-cut nodes; t_{lp}: Time (secs.) to obtain the z_{lp} value; t_s: Time (secs.) to obtain the z_s value; t_{ip}: Time (secs.) to obtain the z_{ip} value; t_t: Total time (secs.), that is $t_{lp}+t_{ip}$; nc: Total number of cuts performed by CPLEX. Note: The minimum, average and maximum GAPs are reported in Table III.

In a realistic case, the number of aircrafts and levels vary between 15 and 30 and between 1 and 10 levels, respectively. The airspace under consideration has 50x50 squared units leading to a greater number of conflicts while the dimensions are increased.

TABLE II
DIMENSIONS TABLE.

Case	m	n	nel	d	m^*	n^*	nel^*	d^*
C020-05	10650	4970	49110	0.0009	3584.0	1697.1	15273.8	0.0025
C020-07	14830	6910	68610	0.0007	4974.8	2347.4	21297.4	0.0018
C020-10	21100	9820	97860	0.0005	7139.3	3343.2	30827.8	0.0013
C025-05	16750	7775	77325	0.0006	5755.6	2673.4	24744.4	0.0016
C025-07	23350	10825	108075	0.0004	7980.8	3691.3	34470.0	0.0012
C025-10	33250	15400	154200	0.0003	11316.9	5221.6	48990.1	0.0008
C030-05	24225	11205	111915	0.0004	8258.3	3797.4	35466.9	0.0011
C030-07	33795	15615	156465	0.0003	11306.5	5197.2	48440.5	0.0008
C030-10	48150	22230	223290	0.0002	16131.2	7374.8	69361.2	0.0006
C035-05	33075	15260	152880	0.0003	11157.7	5105.0	47762.8	0.0008
C035-07	46165	21280	213780	0.0002	15210.6	6967.2	64873.6	0.0006
C035-10	65800	30310	305130	0.0002	21814.2	9958.0	93415.2	0.0004
C040-05	43300	19940	200220	0.0002	14608.1	6642.0	62579.3	0.0006
C040-07	60460	27820	280020	0.0002	20313.0	9204.3	87217.4	0.0005
C040-10	86200	39640	399720	0.0001	28662.4	12995.6	122937.1	0.0003
C045-05	54900	25245	253935	0.0002	19356.2	8688.6	84169.1	0.0005
C045-07	76680	35235	355185	0.0001	25713.6	11605.7	110313.4	0.0004
C045-10	109350	50220	507060	0.0001	37010.2	16649.2	159640.9	0.0003
C050-05	67875	31175	314025	0.0001	22446.6	10161.2	95649.4	0.0004
C050-07	94825	43525	439275	0.0001	32010.0	14388.7	137661.1	0.0003
C050-10	135250	62050	627150	0.0001	45039.8	20250.4	193319.8	0.0002

The first computational observation that can be made in Table II is the strength of the CPLEX preprocessing by comparing the columns m and m^*, n and n^* and nel and nel^*. Moreover, the dimensions m^* and n^* are still very big. In Table III the very small GAP for all the instances can be observed, showing the tightness of the model. (Notice that nb is small for most of the instances) and, then, the elapsed time is very small.

VI. Conclusion

The so-called VAC model, for the resolution of the Collision Avoidance Problem has been presented. It adds to the VC model new interesting features, the most important being the inclusion of altitude changes to avoid infeasible situations brought by velocity bounds. Also, the VAC model completes the VC model in the sense that it takes into account null denominator appearances. This model looks for an equilibrium in the number of maneuvers for each aircraft, penalizing those ones with many maneuvers to realize. The elapsed times are very small and, then, the model could be applied in real time, helping ATC decision making.

Acknowledgment

The authors are indebted to Lucia Pallottino whose PhD thesis has helped us to go into inside of the problem.

They would like to thank their colleague Ana García Bouso for proof-reading of an earlier version of the paper. She has contributed to improve the quality of the present version.

References

[1] EUROCONTROL, "Fasti atc manual," http://www.eurocontrol.int/fasti last access in October, Tech. Rep., 2009.

[2] J. K. Kuchar and L. C. Yang, "A review of conflict detection and resolution modeling methods," *IEEE Transactions on Intelligent Transportation Systems*, vol. 1, pp. 179–189, 2000.

[3] A. G. Richards and J. P. How, "Aircraft trajectory planning with collision avoidance using mixed integer linear programming," American Control Conference, Anchorage (Alaska), 2002.

[4] P. Dell'Olmo and G. Lulli, "A new hierarchical architecture for air traffic management: Optimization of airway capacity in a free flight scenario," *European Journal of Operational Research*, vol. 144, pp. 179–193, 2003.

[5] M. A. Christodoulou and C. Costoulakis, "Nonlinear mixed integer programming for aircraft collision avoidance in free flight," *IEEE Melecon 2004, Dubrovnik, Croacia*, vol. 1, pp. 327–330, May. 2004, iSBN: 0-7803-8271-4.

[6] L. Pallottino, "Aircraft conflict resolution in "free flight" air traffic management systems: Models and optimal solutions," Ph.D. dissertation, Automation and Industrial Robotics, Università di Pisa (Italy), June 2002.

[7] L. Pallottino, E. Feron, and A. Bicchi, "Conflict resolution problems for air traffic management systems solved with mixed integer programming," *IEEE Transactions on Intelligent Transportation Systems*, vol. 3, no. 1, pp. 3–11, 2002.

[8] J. Clarke, S. Solak, Y. Chang, L. Ren, and A. Vela, "Air traffic flow management in the presence of uncertainty," 8th USA/Europe Air Traffic Management Research and Development Seminar (ATM2009), 2009.

[9] IBM ILOG, *CPLEX v11.2. User's Guide*, 2008.

PLACE PHOTO HERE

Antonio Alonso Ayuso was born in Santander, Spain, in 1968. He received the Msc. in Mathematics in 1992 and the Ph.D. degree in Mathematics in 1997, in University Complutense, Madrid, Spain. He is currently full time professor at the Dep. of Statistics and Operational Research at Rey Juan Carlos University of Madrid. He has been member of several research projects at different Spanish Universities, European Commission (Leonardo Da Vinci program) and National Research Plan (several of them as main research). He has a number of papers in well rated international journal and has collaborate with different firms in Applied Projects.

His main research interests include linear and integer mathematical programming, decision models and stochastic programming applied to combinatorial problems.

Laureano F. Escudero was born in Valladolid, Spain, in 1942. PhD. degree in Economics, Universidad de Deusto, Bilbao, Spain, in 1974,. Currently, he is Professor of Statistics and Operations Research at the University Rey Juan Carlos, Spain. In the period 2003-04 he was the President of EURO (Association of European Operations Research Societies). He has worked at IBM Research, Scientific and Development Centers in Madrid (Spain), Palo Alto (California), Sindelfingen (Germany) and Yorktown Heights (NY), 1972-1991. He taught Mathematical Programming at the Mathematical Sciences School, University Complutense of Madrid, 1992-2000 and Stochastic Programming at the University Miguel Hernández, Spain. He is the author of several books and more than 100 scientific papers published in leading journals.

His main research interest includes different mathematical programming fields (linear, integer, nonlinear, stochastic) and their applications.

Francisco Javier Martín Campo was born in Salamanca, Spain, in 1983. Msc. in Mathematics, University Complutense, Madrid, Spain in 2007. He is currently pursuing the Ph.D. degree in mathematics. At present he is also working in a project based on air traffic flow management and collision avoidance so-called Atlantida from a contract with the company GMV Aerospace and Defence S.A., on the same subject of the thesis.

His main research interests include optimization, mixed integer linear programming models and algorithms, and air traffic management models.

TABLE III
RESULTS TABLE.

Case	z_{lp}	z_s	z_{ip}	GAP_{lp}	GAP_s	nb	nn	t_{lp}	t_s	t_{ip}	t_t	nc
C020-05	0.3218	2.0027	2.0682	0.0000 0.8444 1.0000	0.0000 0.0317 0.4472	3	13.67	0.04	0.23	0.27	0.31	111.0
C020-07	0.0932	1.1695	1.2187	0.0000 0.9235 1.0000	0.0000 0.0403 0.2838	0	0.00	0.05	0.27	0.29	0.34	34.4
C020-10	0.0466	1.0644	1.1329	0.1323 0.9589 1.0000	0.0000 0.0605 0.5857	0	0.00	0.08	0.39	0.46	0.54	24.9
C025-05	0.4716	3.6533	3.7547	0.3333 0.8744 1.0000	0.0000 0.0270 0.1494	1	4.00	0.07	0.44	0.52	0.59	260.1
C025-07	0.4069	2.6195	2.7198	0.4260 0.8504 1.0000	0.0000 0.0369 1.0000	2	6.50	0.09	0.54	0.63	0.72	106.0
C025-10	0.2000	1.7422	1.8004	0.0000 0.8889 1.0000	0.0000 0.0324 0.4877	1	4.00	0.13	0.70	0.76	0.89	77.6
C030-05	0.6495	4.1427	4.3408	0.5665 0.8504 1.0000	0.0000 0.0456 0.2257	4	15.50	0.10	0.76	1.02	1.12	299.6
C030-07	0.4721	2.8227	3.0019	0.2810 0.8427 1.0000	0.0000 0.0597 0.5541	2	15.50	0.14	0.85	1.10	1.23	116.8
C030-10	0.2056	2.3050	2.4116	0.3480 0.9148 1.0000	0.0000 0.0442 0.3113	3	5.67	0.19	1.19	1.41	1.60	125.8
C035-05	0.8585	5.1918	5.3918	0.5543 0.8408 1.0000	0.0000 0.0371 0.1213	2	42.00	0.14	1.28	1.77	1.91	351.3
C035-07	0.8540	4.7520	4.9595	0.6030 0.8278 1.0000	0.0000 0.0418 0.1971	5	44.80	0.19	1.41	1.84	2.03	271.0
C035-10	0.2207	3.2340	3.4276	0.0000 0.9356 1.0000	0.0000 0.0565 0.1739	3	19.00	0.28	1.88	2.34	2.62	141.3
C040-05	1.3807	7.3087	7.7402	0.6049 0.8216 1.0000	0.0000 0.0558 0.2271	13	57.15	0.19	2.10	3.99	4.17	574.2
C040-07	0.9005	5.8852	6.1063	0.5251 0.8525 1.0000	0.0000 0.0362 0.1304	6	23.33	0.26	2.06	2.64	2.90	262.3
C040-10	0.4185	4.5220	4.6666	0.6212 0.9103 1.0000	0.0000 0.0310 0.1723	4	21.00	0.38	2.75	3.31	3.69	288.9
C045-05	1.7200	9.2311	9.8711	0.5458 0.8258 1.0000	0.0016 0.0648 0.1518	17	143.06	0.26	3.15	8.62	8.87	1104.1
C045-07	0.6227	6.8541	7.2115	0.7099 0.9137 1.0000	0.0000 0.0496 0.2224	8	101.13	0.35	3.15	5.26	5.61	280.0
C045-10	0.4042	4.5718	4.7658	0.6041 0.9152 1.0000	0.0000 0.0407 0.3003	4	8.00	0.49	3.89	4.90	5.39	226.9
C050-05	2.0867	10.7150	11.2727	0.5238 0.8149 1.0000	0.0019 0.0495 0.1231	17	183.88	0.32	3.95	13.46	13.79	1333.9
C050-07	1.1475	8.0404	8.5479	0.6659 0.8658 1.0000	0.0000 0.0594 0.2011	8	51.25	0.45	4.59	8.37	8.82	721.9
C050-10	0.3750	6.5023	6.7673	0.7308 0.9446 1.0000	0.0000 0.0392 0.1782	6	14.17	0.62	5.61	7.39	8.01	248.3