Automatic Commercial Aircraft-Collision Avoidance in Free Flight: The Three-Dimensional Problem

Manolis A. Christodoulou, *Senior Member, IEEE*, and Sifis G. Kodaxakis, *Student Member, IEEE*

***Abstract*—In this paper, optimal resolution of air-traffic (AT) conflicts were considered. Aircraft are assumed to cruise within a free altitude layer and are modeled in three dimensions with variable velocity and proximity bounds. Aircraft cannot get closer to each other than a predefined safety distance. The problem of solving conflicts arising among several aircraft that are assumed to move in a shared airspace were considered. For such systems of multiple aircraft, the total flight time by avoiding all possible conflicts were minimized. This paper proposes a formulation of the conflict avoidance problem as a mixed-integer nonlinear-programming problem. In the author's case, only velocity changes are admissible as maneuvers. Nevertheless, in subsequent work, simultaneous velocity and heading angle changes will be checked too. Simulation results for realistic aircraft conflict scenarios are provided.**

***Index Terms*—Conflict resolution, free flight, nonlinear mixed integer programming.**

I. Introduction

A. Free Flight

Today's radar systems used for navigation of commercial airliners are obsolete. The air-traffic (AT) controllers rely on them [1]. The planes are depicted in two dimensions, while the controller has to estimate the plane's trajectories in the actual three-dimensional space. Given the heavy traffic around hub airports, obsolete instrumentation cannot help much in accident avoidance.

In some years from now, the aircraft will be equipped by more sophisticated machinery and electronics. The communications will be converted to digital. This will facilitate three-dimensional (3-D) screen representations for AT controllers. Also, the planes will be equipped by intercommunicating facilities. This process may help by totally avoiding the use of AT controllers, while on the other hand, the planes can be helped through intelligent systems for self-navigation so that the control can be handled by autonomous local-distributed intelligent control systems.

B. Free Flight Advantages

It is almost sure that capacity in the air will be increased dramatically. Free flight methods will help controllers to work more effectively. Thus, the digital era together with the introduction of intelligent systems will help by substantially improving the commercial AT.

C. Previous Work on Free Flight

Work on free flight has been done by Tomlin and coworkers [9]–[24], Sastry and coworkers [25], [26], Bicchi and Pallottino [27], and Feron and coworkers [28]–[33]. Analogous to the present paper has been done in [27] for two dimensions using an optimal control approach. Also, in [25], the authors used the hybrid control system approach to solve the problem of selecting a maximal set of safe initial conditions. Feron and coworkers [6] used the approach of linear mixed-integer programming (MIP) to solve the problem of planar velocity change or heading angle change separately, for two dimensions. Christodoulou and Costoulakis [34], [35] combined the above two approaches in the planar two-dimensional (2-D) motion and formulated the problem as a nonlinear MIP (NMIP). In the present paper, we formulate the velocity change problem (VC problem) as the NMIP in three dimensions for the first time.

II. Proposed Free Flight Scenario

A. Conflict Resolution Strategy

Various methods have been proposed recently for the case of multiple-aircraft-collision avoidance. An extensive review of previous work in the subject has been presented by Kuchar and Yang in [2]. Bilimoria and Lee [3] examined safety features in case aircraft are allowed to fly free in airspace.

In case at least a certain conflict and/or multiple conflicts is/are predicted, aircraft cooperate automatically, to avoid the situation(s). There exist communication lines between all involved aircraft. The cooperation is automatically taking place via an intelligent system, while pilots are not further able to control manually the aircraft. Only as long as the emergency situation is resolved and over, pilots are allowed by the intelligent system to resume manual control.

Each aircraft is surrounded by two virtual spheres, the protected zone and the alert zone, shown in Fig. 1. **A conflict or loss of separation between two aircraft occurs whenever the protected zones of the aircraft overlap.** The radius and the height of the en route protected zone are currently about 2.5 nmi and 2000 ft, respectively. However, it has been proposed in [4] that for a true 3-D free flight, protected zones need to be spheres of radius about 3–5 mi. The size of the alert zone depends on various factors. The alert zone is large so as to allow a comfortable system response but also small in order to avoid unnecessary conflicts.

Manuscript received February 3, 2005; revised September 17, 2005. The Associate Editor for this paper was P. J. Kostek.

The authors are with the Department of Electronic and Computer Engineering, Technical University of Crete, Chania, Crete 73100, Greece (e-mail: manolis@systems.tuc.gr; sifis@systems.tuc.gr).

Digital Object Identifier 10.1109/TITS.2006.874684

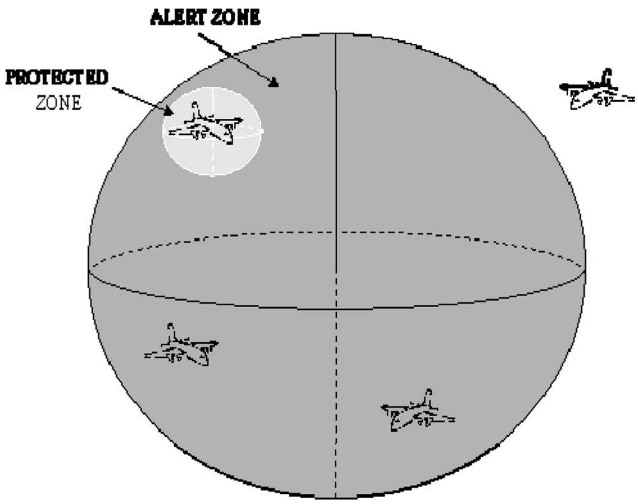

Fig. 1. Aircraft zones.

For the algorithms in this paper, we have considered aircraft randomly distributed on a 3-D spherical alert zone of radius 67.1 nmi and a spherical protected zone of radius $d/2 = 2.8$ nmi.

The next postulates we assume are valid during the following situations.

1) **Free flight in the 3-D space.** Each aircraft is assigned a specific beginning–end intermediate linear route.
2) **There is cooperation among aircraft to avoid possible conflicts.** By the very first moment a possible conflict is predicted, all carriers involved are navigated by an intelligent system, while pilots are prohibited of using manual controls.
3) Here, we allow changes in velocities only and not in heading angles.

The problem becomes the **NMIP** and is automatically solved in real time (when the situation occurs) via CPLEX [5].

We are able to solve the problem for a big number of aircraft involved in the same alert space. We also use geometric constraints in three dimensions in space, in order to generate conflict avoidance constraints for the mathematical optimization problem. Since we may run the problem for many times during the actual situation, we may generate online a database of possible scenarios for collision avoidance and, hence, we are able to select the optimal one.

B. Problem Statement

By looking at the alert zone as in Fig. 1, we assume a finite number of aircraft, occupying the 3-D space enclosed by the surface of the outer sphere. Aircraft are flying freely in the space without initially taking into consideration other's paths. As long as they cross the surface of the alert zone, the intelligent system is activated in order to prevent possible collisions, while on the other hand deviate minimally from the nominal path in order not to increase the flight time and avoiding the use of much more fuel. As we will explain next, a conflict occurs if the distance between two aircraft becomes less than d.

As mentioned before, by considering the aircraft as a moving sphere of radius $d/2$, the condition of nonconflict between

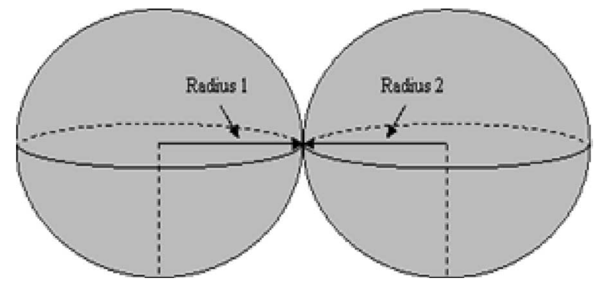

Fig. 2. Spheres in touch.

aircraft is equivalent to the condition of nonintersection of the spheres. To gain a quick understanding of this problem, let us take a look at what happens when two spheres are touching. As we can see in the illustration at Fig. 2, the radius of each sphere now also defines the distance between its center and the opposite sphere's surface. Therefore, given this condition, the distance between the centers would be equal to Radius1 + Radius2. If the distance were greater, the two spheres would not touch. If it were less, the spheres would intersect. In the following, we refer to those spheres as the safety sphere of the aircraft.

Aircraft are identified in a 3-D coordinate system representing the center of the sphere (protected zone position) by three Cartesian coordinates and two angles representing the space direction, in total by a five-dimensional vector as follows:

$$(x, y, z, \theta, \phi) \in R \times R \times R \times S^1 \times S^1$$

Let $(x_i(t),\ y_i(t),\ z_i(t),\ \theta_i(t),\ \phi_i(t))$ be the configuration of the ith aircraft at time t. In a conflict between aircraft, i and j occurs if for some value t

$$\sqrt{(x_i(t) - x_j(t))^2 + (y_i(t) - y_j(t))^2 + (z_i(t) - z_j(t))^2} < d. \tag{1}$$

Since $(\text{radius1} + \text{radius2}) = (d/2 + d/2) = d$.

To avoid possible conflicts, we allow aircraft to change the velocity of flight but the direction of motion remains fixed. We will refer to this case as the VC problem.

At time $t = 0$ the intelligent system takes over. The solution of the NMIP problem provides a deviation path for each aircraft involved in possible future accident. It is assumed, there exist a sufficient amount of time for the intelligent system to take over and provide a solution before any possible accident takes place. Thus, by assumption conflicts do not occur at time $t = 0$ and in a time vicinity of $t = 0$ so as to allow sufficient time for computations.

Assume the ith aircraft changes its velocity by the quantity q_i. The NMIP program solves the problem of finding the optimal q_i. The hard constraints imposed by the problem are the ones that guarantee collision avoidance for all carriers involved. As we will show below, the problem becomes a mixed integer, i.e., it involves continuous as well as Boolean variables. Let us define by q_i the velocity change of the ith aircraft. In Section II-C, we formulate the hard nonlinear constraints that guarantee collision avoidance.

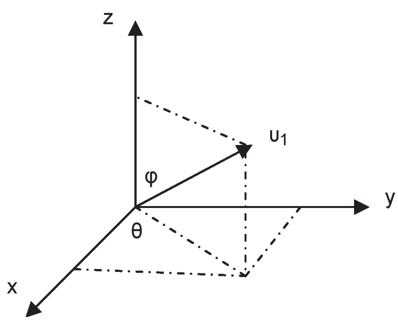

Fig. 3. Velocity vector.

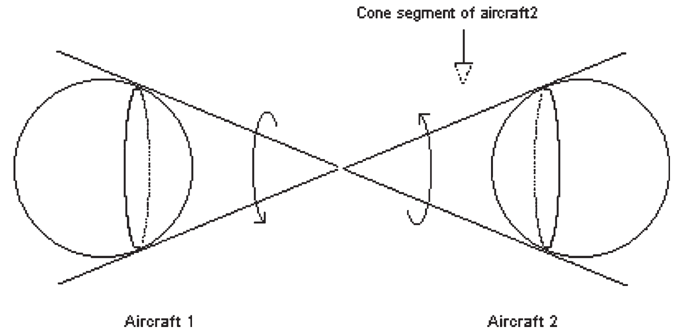

Fig. 4. Cone sections between two moving spheres.

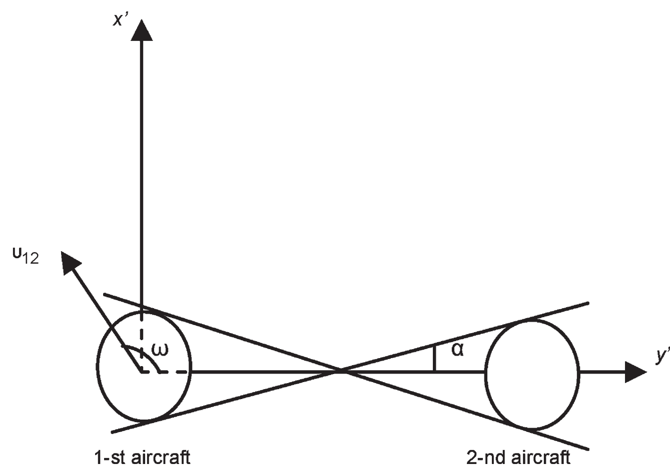

Fig. 5. Two nonparallel straight lines tangent to the safety discs of radius *d*/2 for two aircraft at distance A_{12}.

C. Constraints for the Problem

By working in the 3-D space, and using solid-state geometry, we get the nonlinear hard constraints for our problem. We have aircraft following an assigned path. The particular velocities can be altered only twice, first at time $t = 0$ and then at the end of the trip, during which the aircraft is moving out the surface of the alert sphere zone. The velocity is changed by the quantity q_i only during the flight inside the alert zone, while it is resumed at its nominal value at the time, when the aircraft exits the sphere surface. For every aircraft, there are imposed maximum and minimum values for its velocity as follows: $\upsilon_i : \upsilon_{i,\min} \leq \upsilon_i \leq \upsilon_{i,\max}$. Federal Aviation Administration (FAA) usually considers variations that are of the form: $(\upsilon_{i,\max} - \upsilon_{i,\min})/\upsilon_{i,\min} \leq 0.1$. The system solves a NMIP problem, while the above constraints are satisfied. (For the VC problem in two dimensions, i.e., aircraft flying on a plane, see [6]). According to our assumptions, assuming the new velocities take place, then the following constraints have to be satisfied:

$$\upsilon_{i,\min} \leq \upsilon_i + q_i \leq \upsilon_{i,\max}. \tag{2}$$

Here, we assume the velocity bounds are such that do not allow path changes that will result in, e.g., altitude loss. They are usually provided by FAA.

We will initially examine the case of two aircraft, to obtain conflict avoidance conditions, and then we will consider the case of multiple n aircraft. Consider the aircraft 1 and 2, respectively, and let $(x_i, y_i, z_i, \theta_i, \phi_i)$, $i = 1, 2$ be the aircraft positions and directions of motion and υ_i be the initial velocities.

According to Fig. 3, we define the two velocity vectors

$$\vec{\upsilon}_1 = \begin{bmatrix} (\upsilon_1 + q_1)\sin\phi_1\cos\theta_1 \\ (\upsilon_1 + q_1)\sin\phi_1\sin\theta_1 \\ (\upsilon_1 + q_1)\cos\phi_1 \end{bmatrix} \tag{3}$$

$$\vec{\upsilon}_2 = \begin{bmatrix} (\upsilon_2 + q_2)\sin\phi_2\cos\theta_2 \\ (\upsilon_2 + q_2)\sin\phi_2\sin\theta_2 \\ (\upsilon_2 + q_2)\cos\phi_2 \end{bmatrix}. \tag{4}$$

The relative velocity vector

$$\vec{\upsilon}_1 - \vec{\upsilon}_2 = \begin{bmatrix} (\upsilon_1 + q_1)\sin\phi_1\cos\theta_1 - (\upsilon_2 + q_2)\sin\phi_2\cos\theta_2 \\ (\upsilon_1 + q_1)\sin\phi_1\sin\theta_1 - (\upsilon_2 + q_2)\sin\phi_2\sin\theta_2 \\ (\upsilon_1 + q_1)\cos\phi_1 - (\upsilon_2 + q_2)\cos\phi_2 \end{bmatrix} \tag{5}$$

where $0 \leq \theta_i \leq 2\pi$ and $0 \leq \phi_i \leq \pi$.

The nonparallel straight lines that are tangent to the spheres of both aircraft, localize a segment in the direction on motion of 1 (refer to Fig. 4). We refer to this segment as the **cone** of aircraft 2 along the direction of 1. A conflict occurs if the aircraft 1 and its safety sphere intersect the cone generated by aircraft 2, or vice versa.

We refer to the above case, by examining the motion of two spheres in two dimensions because of symmetry, in the plane defined by vector $\vec{s}$, which represents the straight line segment joining the centers of the two spheres, and by $\vec{\upsilon}_{12}$ the vector of relative speed of motion among the 2 flying aircraft. (Refer to Fig. 5). This means the second aircraft moves with relative velocity $\vec{\upsilon}_{12}$ with respect to the first.

Consider now the two nonparallel straight lines that are tangent to the discs of both aircraft. Let α be the angle between the first straight line and the horizontal axis, and ω be the angle between the vector $\vec{\upsilon}_{12}$ of relative speed and the vector $\vec{s}$ which represents the distance of the two spherical centers.

If ω is the angle between vectors $\vec{\upsilon}_{12}$ and $\vec{s}$, we have

$$\cos\omega = \frac{\upsilon_{12} \bullet s}{|\upsilon_{12}||s|}, \ \vec{s} = \begin{bmatrix} x_2 - x_1 \\ y_2 - y_1 \\ z_2 - z_1 \end{bmatrix}.$$

Since $\tan\omega = \pm\sqrt{1-\cos^2\omega}/\cos\omega$ in the case of positive sign, no conflict occurs if

$$\frac{\sqrt{1-\left(\frac{\upsilon_{12}\bullet s}{|\upsilon_{12}||s|}\right)^2}}{\frac{\upsilon_{12}\bullet s}{|\upsilon_{12}||s|}} \geq \tan(\alpha) \tag{6}$$

$$\text{or} \quad \frac{\sqrt{1-\left(\frac{\upsilon_{12}\bullet s}{|\upsilon_{12}||s|}\right)^2}}{\frac{\upsilon_{12}\bullet s}{|\upsilon_{12}||s|}} \leq \tan(-\alpha). \tag{7}$$

To obtain nonconflict constraints for n aircraft, we need to consider the nonconflict conditions described by (6) and (7) for all possible pairs of aircraft. Let us consider the pair of aircraft $(i,\ j)$. We have to distinguish between two possible cases: 1) $\upsilon_{ij}\bullet s<0$ and 2) $\upsilon_{ij}\bullet s>0$. We also have $\tan(\alpha)=(d/2)/(A_{ij}/2)=d/A_{ij}$, where A_{ij} is the distance between the two aircraft i and j. Therefore, we obtain the following groups of constraints.

Case 1) $\upsilon_{ij}\bullet s<0$ and $\tan\omega$ has "positive sign"

$$\left\{\begin{array}{l}\upsilon_{ij}\bullet s\leq 0\\ |\upsilon_{ij}||s|*\sqrt{1-\left(\frac{\upsilon_{ij}\bullet s}{|\upsilon_{ij}||s|}\right)^2}-\frac{d}{A_{ij}}*(\upsilon_{ij}\bullet s)\leq 0\end{array}\right\} \tag{8}$$

or

$$\left\{\begin{array}{l}\upsilon_{ij}\bullet s\leq 0\\ -|\upsilon_{ij}||s|*\sqrt{1-\left(\frac{\upsilon_{ij}\bullet s}{|\upsilon_{ij}||s|}\right)^2}-\frac{d}{A_{ij}}*(\upsilon_{ij}\bullet s)\leq 0\end{array}\right\}. \tag{9}$$

Case 2) $-\upsilon_{ij}\bullet s>0$ and $\tan\omega$ has "positive sign"

$$\left\{\begin{array}{l}-\upsilon_{ij}\bullet s\geq 0\\ -|\upsilon_{ij}||s|*\sqrt{1-\left(\frac{\upsilon_{ij}\bullet s}{|\upsilon_{ij}||s|}\right)^2}+\frac{d}{A_{ij}}*(\upsilon_{ij}\bullet s)\leq 0\end{array}\right\} \tag{10}$$

or

$$\left\{\begin{array}{l}-\upsilon_{ij}\bullet s\geq 0\\ |\upsilon_{ij}||s|*\sqrt{1-\left(\frac{\upsilon_{ij}\bullet s}{|\upsilon_{ij}||s|}\right)^2}+\frac{d}{A_{ij}}*(\upsilon_{ij}\bullet s)\leq 0\end{array}\right\}. \tag{11}$$

These two groups of constraints will be included in the model as or-constraints. **All constraints obtained are nonlinear in the variable that represents the velocity change** q_i. To conclude with the problem formulation, we must consider the upper and lower bounds in (2) that are already linear in q_i.

As noted earlier, only one set of constraints will be used in our model for each instance. Thus, subject to which of cases (1 or 2) holds true, we use the first set (8)–(10) or the second set (9)–(11) of equations.

In the case of negative sign, for the tangent $\tan\omega$, by using analogous reasoning, no conflict between all n aircraft occurs if we have the following.

Case 1) $\upsilon_{ij}\bullet s<0$ and $\tan\omega$ has "negative sign"

$$\left\{\begin{array}{l}\upsilon_{ij}\bullet s\leq 0\\ -|\upsilon_{ij}||s|*\sqrt{1-\left(\frac{\upsilon_{ij}\bullet s}{|\upsilon_{ij}||s|}\right)^2}-\frac{d}{A_{ij}}*(\upsilon_{ij}\bullet s)\leq 0\end{array}\right\} \tag{12}$$

or

$$\left\{\begin{array}{l}\upsilon_{ij}\bullet s\leq 0\\ |\upsilon_{ij}||s|*\sqrt{1-\left(\frac{\upsilon_{ij}\bullet s}{|\upsilon_{ij}||s|}\right)^2}-\frac{d}{A_{ij}}*(\upsilon_{ij}\bullet s)\leq 0\end{array}\right\}. \tag{13}$$

Case 2) $-\upsilon_{ij}\bullet s>0$ and $\tan\omega$ has "negative sign"

$$\left\{\begin{array}{l}-\upsilon_{ij}\bullet s\geq 0\\ |\upsilon_{ij}||s|*\sqrt{1-\left(\frac{\upsilon_{ij}\bullet s}{|\upsilon_{ij}||s|}\right)^2}+\frac{d}{A_{ij}}*(\upsilon_{ij}\bullet s)\leq 0\end{array}\right\} \tag{14}$$

$$\left\{\begin{array}{l}-\upsilon_{ij}\bullet s\geq 0\\ -|\upsilon_{ij}||s|*\sqrt{1-\left(\frac{\upsilon_{ij}\bullet s}{|\upsilon_{ij}||s|}\right)^2}+\frac{d}{A_{ij}}*(\upsilon_{ij}\bullet s)\leq 0\end{array}\right\}. \tag{15}$$

Only one of these sets of constraints will be used in our model for each instance. Thus, according to which one between the two cases (1 or 2) holds true, we will use the first set (12) and (14) or the second set (13) and (15) of equations.

The minimum time optimization problem then can be defined as

$$\sum_{i=1}^{n} -q_i. \tag{16}$$

Equation (16) defines the performance index of our minimization problem with constraints. The performance index is linear, while the constraints are nonlinear.

The problem sometimes possess several solutions, (same quantity of positive or negative q_i), and some times does not have a solution as is the case of opposite linear motions of two aircraft or "almost linear opposite motions."

D. Formulation of the Problem

The hard constraints obtained in the above sections are nonlinear in the decision variables q_i for the problem. Because we have a very large number of constraints that increases exponentially with the number of aircraft involved, it is imperative that we use a software optimization package in order to solve it. There are, indeed, many such tools available. We have chosen the General Algebraic Modeling System (GAMS) software package (www.gams.de), which is essentially a front end for solvers such as CPLEX, dicopt, etc. Its friendly interface and accessibility make it an ideal tool for the user who does not wish the full processing power of professional high-end products.

We now show how the above set of constraints should be recast as mixed-integer nonlinear constraints suitable for standard optimization software such as CPLEX. We assume that the reader is familiar with the basics of linear and nonlinear optimization problems.

As any other optimization package, GAMS, requires that the constraints present for any problem are all satisfied simultaneously (together with the constraints). In other words, GAMS is able to solve optimization problems of the form

$$\min f(x) \tag{17}$$

such that

$$g(x) \leq 0 \tag{18}$$

where $f(x)$ is a function of n real variables $x = (x_1, x_2, \ldots, x_n) \in R^n$ and is subject to a set of inequality constraints $g(x) \leq 0$ $(g_j(x) \leq 0, j = 1, 2, p)$. This means that the constraints $g_j(x)$ must be all valid at the same time (g_1 and g_2 and $\ldots$ and g_p). Clearly in our case, where we have or-constraints, a reformulation is necessary. We therefore shall have to introduce new Boolean variables to convert these or-constraints to and-constraints. A simple example will be presented for comprehensive purposes.

Let us assume that we have the following sets of constraints similar to the conflict avoidance constraints described in the previous sections.

$$\begin{cases} c_1 \leq 0 \\ \text{and} \\ c_2 \leq 0 \end{cases} \tag{19}$$

or

$$\begin{cases} c_3 \leq 0 \\ \text{and} & (20) \\ c_4 \leq 0 \\ \text{and} & (21) \\ c_5 \leq 0 \end{cases}$$

where $c_i, i = 1, \ldots, 5$, are nonlinear constraints in the problem variables.

Transformation of the or-constraints into and-constraints is done via Boolean variables [7]. Assume f_k with $k = 1, 2$, is a binary number that becomes zero when one of the or-constraint is active and 1 otherwise (i.e., $f_1 = 0$ if constraints c_1 and c_2 are active, $f_1 = 1$ otherwise). Letting M be a large arbitrary number, the previous set of constraints is equivalent to

$$\begin{aligned} c_1 - Mf_1 &\leq 0 \\ c_2 - Mf_1 &\leq 0 \\ c_3 - Mf_2 &\leq 0 \\ c_4 - Mf_2 &\leq 0 \\ c_5 - Mf_2 &\leq 0 \\ f_1 + f_2 &\leq 2. \end{aligned} \tag{22}$$

The above constraints are all and-constraints, therefore we have overcome the previous difficulty. It is however obvious that now we are faced with a so-called MIP problem [8], because we have two different kinds of variables: normal variables that can take any value and binary variables (f_1, f_2) that can only take the values 0 or 1. MIP problems are considerably more complex than both the pure integer programming problems (where the decision variables can only take binary values) and the classic linear programming (LP) or nonlinear programming (NLP) problems. Because the formulation is as a NLP problem, it appears to be very fast and computations can be performed in real time, at least for a small number of ten aircraft, before a conflict takes place.

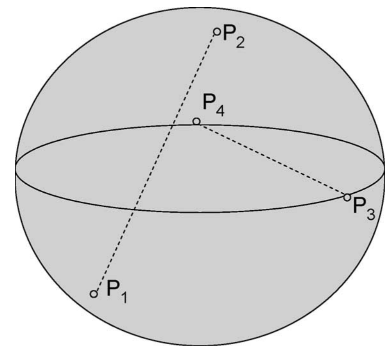

Fig. 6. Initial and final configuration points for two aircraft into the control sphere.

III. Simulation and Case Studies

A. Introduction

In Section II, we provided the mathematical solution for the automatic conflict avoidance problem between n aircraft. The problem was formulated as a mixed-integer NLP. Here, we will use the CPLEX software in simulation cases. We consider aircraft randomly distributed on a sphere of radius 67.1 nmi in nonsymmetrical cases.

The initial configuration of every aircraft consists of its velocity and its two heading angles at the point of entry in the sphere, while its final configuration consists of its velocity and its two heading angles at the point of exit from the sphere. Such kind of points, have been chosen randomly for a more realistic scenario.

Basically, **collision detection** is a system program that determines whether two objects will collide inside the sphere. If there are more than two objects, then we consider all possible pairs of objects.

If a collision occurs, the motion of objects should be reformulated. Perhaps they should move just enough to touch each other (spherewise). Our calculations take the following order:

1) future position computation;
2) possible collision detection;
3) collision handling.

In collision detection, we usually want to know whether two objects intersect. Consider now, the prediction problem as a dynamic one. Two objects are moving relative to one another. Their positions are functions of time. We want to know exactly when they collide, if it so happens. What we want is to define some sort of relationship between the two objects that changes as a function of time. Here is what this kind of prediction problem would involve: Referring to Fig. 6, the initial configuration points for aircraft 1,2 are $P_1(x_1, y_1, z_1)$ and $P_3(x_3, y_3, z_3)$, respectively, while the final configuration points are $P_2(x_2, y_2, z_2)$ and $P_4(x_4, y_4, z_4)$, respectively. Hence, the parametric equations for line "l" through P_1 and P_2 are the following:

$$\begin{cases} x = x_1 + a_1\lambda \\ y = y_1 + a_2\lambda \\ z = z_1 + a_3\lambda \end{cases} \tag{23}$$

where, $a = (a_1, a_2, a_3) = (x_2 - x_1, y_2 - y_1, z_2 - z_1)$ and "λ" is a variable.

Since aircraft 1 is moving in a straight line with a standard velocity υ_1, we have a linear equation. In time "t," aircraft 1 "travels" an interval "s," so we have

$$s = \upsilon_1 t. \tag{24}$$

Using the fact that in the same time "t," we have

$$s = \sqrt{(x - x_1)^2 + (y - y_1)^2 + (z - z_1)^2}. \tag{25}$$

Applying (23) and (25) in (24), we have

$$\lambda = \frac{\upsilon_1 t}{|a|} \tag{26}$$

where $|a| = \sqrt{a_1^2 + a_2^2 + a_3^2}$.

Therefore, (23) now can be rewritten with respect to (26) as follows:

$$\begin{cases} x = x_1 + \left(\frac{a_1 \upsilon_1}{|a|}\right) t \\ y = y_1 + \left(\frac{a_2 \upsilon_1}{|a|}\right) t \\ z = z_1 + \left(\frac{a_3 \upsilon_1}{|a|}\right) t. \end{cases} \tag{27}$$

Hence, each time instant, we know at exactly which point of a straight line the aircraft 1 is located. We work similarly for all others. If the distance between the centers of the safety spheres of the aircraft is smaller than the sum of their radius at time t, this means that the aircraft collide. Thus, by choosing equal small time intervals, we detect during each one whether a collision occurs. When the aircraft are found in the control sphere, by using the above method, we check whether they collide. If the answer is negative, we do not use the algorithm and do not interfere with the aircraft motion. In case a collision is detected, we apply the collision avoidance algorithm.

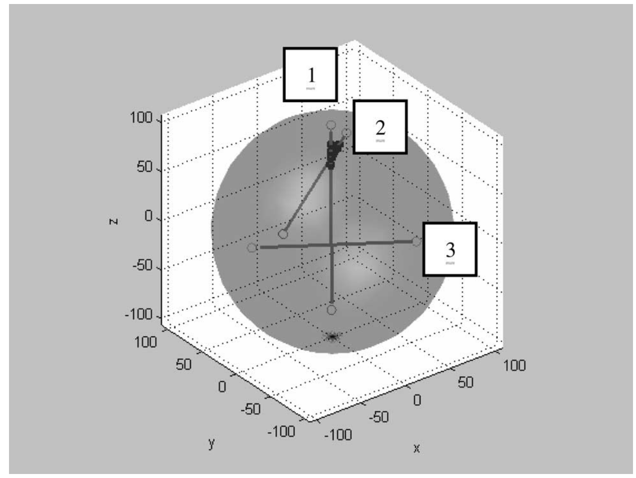

Fig. 7. Three randomly distributed aircraft and their projected trajectories before conflict resolution. With blue color, we present the area where the conflict is detected.

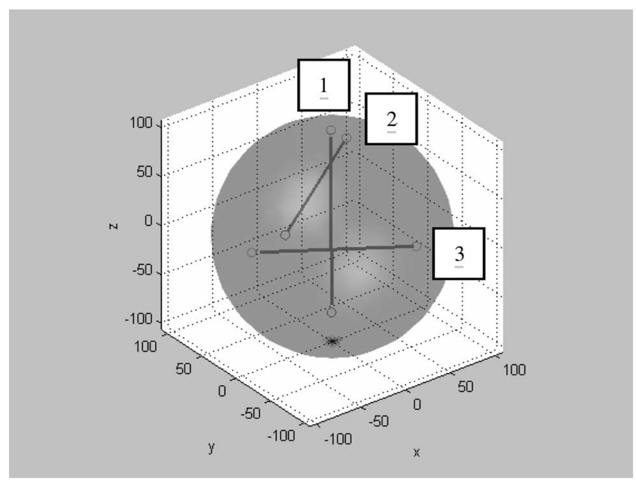

Fig. 8. Three randomly distributed aircraft and their projected trajectories after conflict resolution. No conflict occurs after the velocity changes.

B. Case Study 1: Three Randomly Distributed Aircraft

In the following simulations, all aircraft are assumed crossing the control volume with the same speed. The control volume has a radius of 67 nm or 108 km and the minimum safety distance has been set to 5.6 nm or 9 km. Two plots are presented for each case study, one that shows the aircraft configuration and their projected trajectories before maneuvers are made to avoid possible conflicts and the next shows the corresponding situation after the various speed maneuvers. Each case study is accompanied by a table that shows the velocities of the aircraft before and after the conflict resolution in order to compare the various cases.

In Fig. 7, we see three randomly distributed aircraft that are all headed inside the control volume. The final configuration points of them are also presented. In this paper, there is one conflict between aircraft's 1 and 2, which is resolved by velocity maneuvers of all aircraft's 1, 2, and 3; all interacting aircraft cooperate towards optimization of a common goal, as agents in the same team.

In Fig. 8, we see the aircraft and their trajectories after the maneuvers for conflict resolution. It is important to remember that an aircraft does not change its trajectory in order to avoid a conflict; it just merely changes its speed (accelerate or decelerate). This means that in all the following plots, identical trajectories between two consecutive plots do not imply absence of maneuver in general, but rather absence of heading angle change. In the specific example that we study, the conflicts were resolved only by velocity changes. The values of the velocities of the aircraft before and after the conflict resolution are shown in Table I.

We notice that cases like head-to-head conflicts can be easily solved with a heading-angle-change maneuver, but not only with velocity change maneuver. Also conflicts that occur in time near $t = 0$ cannot be solved always by the VC problem because the upper and lower bounds on the velocity υ_i, are not adequate in order to avoid the conflict.

All the changes in velocities after conflict resolution are optimal with respect to the common goal of aircraft. This means that for aircraft 1, the velocity change $q_1 = -0.703$ is the minimum acceptable deceleration that could be applied in order to avoid conflict with aircraft 2.

TABLE I
TABLE SHOWING THE VELOCITIES FOR THREE RANDOMLY DISTRIBUTED AIRCRAFT. CR STANDS FOR CONFLICT RESOLUTION

	Aircraft 1	**Aircraft 2**	**Aircraft 3**
Velocity before C.R. (km/min)	15	15	15
Change in Velocity after C.R. (km/min)	-0.703	0.99	0.99

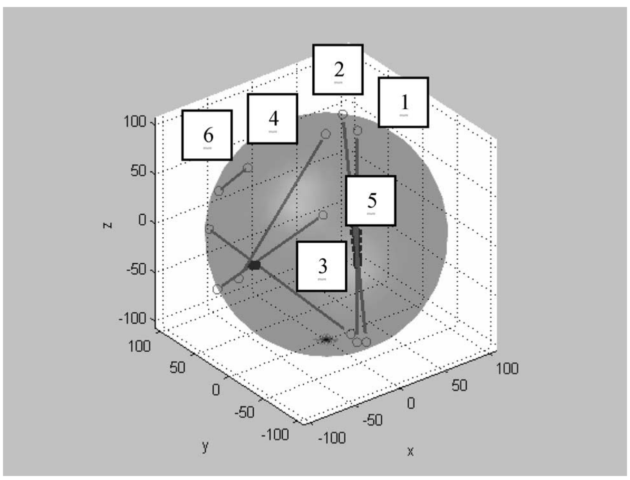

Fig. 9. Six randomly distributed aircraft with two conflicts and their projected trajectories before conflict resolution. Blue color shows the areas where conflict is detected.

C. Case Study 2: Six Randomly Distributed Aircraft With Two Collisions

In the random configuration of Fig. 9, there are two conflicts: one between aircraft's 1 and 2, which is resolved with velocity maneuvers by all aircraft and one between aircraft's 4 and 5, which is resolved in a similar manner, as seen in Fig. 10. The values of the velocities of the aircraft before and after the conflict resolution are shown in Table II.

All the changes in velocities after conflict resolution are optimal with respect to the common goal of aircraft. This means that for aircraft 1, the velocity change $q_1 = 0.364$ is the maximum acceptable acceleration that should take place in order to avoid conflict with aircraft 2. Moreover, for aircraft 4 the velocity change $q_4 = 0.136$ is the maximum acceptable acceleration that should take place in order to avoid conflict with aircraft 5.

In all examples, we used a Pentium IV PC station and the solution was obtained in less than a second, which is acceptable for online purposes.

It is stressed out that in all random positions of aircraft in the control sphere, presented in the examples, a solution exists. A solution is impossible only in the case the two aircraft travel in opposite directions entering a diameter in the control sphere, since there is no way to avoid collision by only velocity change. Here, we need also heading angle change.

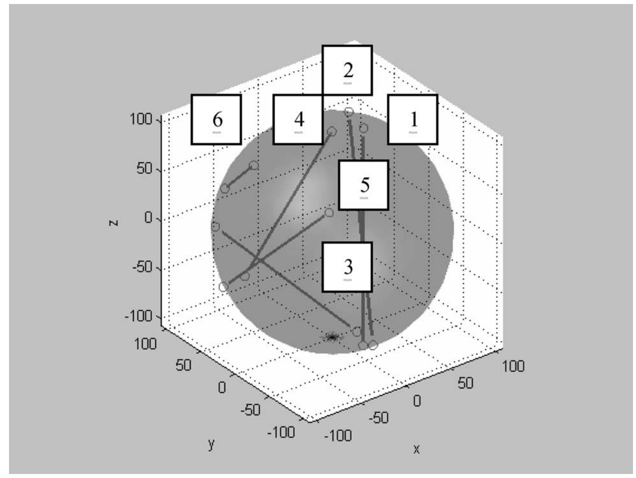

Fig. 10. Six randomly distributed aircraft with two conflicts and their projected trajectories after conflict resolution. No conflict occurs after the velocity changes.

TABLE II
TABLE SHOWING THE VELOCITIES FOR SIX RANDOMLY DISTRIBUTED AIRCRAFT WITH TWO CONFLICTS. AIRCR STANDS FOR AIRCRAFT, BCR ARE THE VELOCITIES BEFORE CONFLICT RESOLUTION, AND ACR ARE THE VELOCITY CHANGES AFTER CONFLICT RESOLUTION

Aircr	**1**	**2**	**3**	**4**	**5**	**6**
B.C.R.	15	15	15	15	15	15
A.C.R.	0.364	0.66	0.66	0.136	0.66	0.66

IV. CONCLUSION

We obtained an integer optimization problem with a linear performance index and nonlinear constraints. CPLEX software is used for numerical solution. The problem provided real time online solutions for many aircraft, at least less than ten.

Future investigations of the optimal maneuvers (velocity change and heading angle) in the 3-D space, in terms of flight time, are part of future work. Due to the nonlinearity that follows from considering heading angle and velocity variation, a future work is to consider other variables and formulate the problem as MIP. In another direction, more case studies should be examined, with more complex configuration patterns and a greater number of aircraft so as to gain more insight in the algorithm and the way it works, although the situations examined here are realistic.

REFERENCES

[1] *Air Traffic Control*. [Online]. Available: http://www.stasys.co.uk/transport/atm_airports/default.htm

[2] J. K. Kuchar and L. C. Yang, "A review of conflict detection and resolution modeling methods," *IEEE Trans. Intell. Transp. Syst.*, vol. 1, no. 4, pp. 179–189, Dec. 2000.

[3] K. D. Bilimoria and H. Q. Lee, "Properties of air traffic conflicts for free and structured routing," presented at the AIAA Guidance, Navigation, Control Conf., Montreal, QC, Canada, Aug. 2001, Paper 2001-4051.

[4] J.-H. Oh and E. Feron, "Fast detection and resolution of multiple conflicts for 3-dimensional free flight," in *Proc. 36th IEEE Conf. Decision Control*, Dec. 1997, pp. 66–74.

[5] *ILOG CPLEX Users Guide*, ILOG, France, 1999.

[6] L. Pallottino, E. M. Feron, and A. Bicchi, "Conflict resolution problems for air traffic management systems solved with mixed integer

programming," *IEEE Trans. Intell. Transp. Syst.*, vol. 3, no. 1, pp. 3–11, Mar. 2002.
[7] F. S. Hillier and G. J. Lieberman, *Introduction to Operations Research*, 6th ed. New York: McGraw-Hill, 1995.
[8] C. A. Floudas, *Non-Linear and Mixed-Integer Optimization*. London, U.K.: Oxford Univ. Press, 1995.
[9] G. Inalhan, D. Stipanovic, and C. Tomlin, "Decentralized optimization, with application to multiple aircraft coordination," in *Proc. 41st IEEE Conf. Decision Control*, Las Vegas, NV, Dec. 2000, pp. 1147–1155.
[10] M. Oishi, I. Mitchell, A. Bayen, C. Tomlin, and A. Degani, "Hybrid verification of an interface for an automatic landing," in *Proc. 41st IEEE Conf. Decision Control*, Las Vegas, NV, Dec. 2002, pp. 1607–1613.
[11] D. Stipanovic, G. Inalhan, and C. Tomlin, "Decentralized overlapping control of a formation of unmanned aerial vehicles," in *Proc. 41st IEEE Conf. Decision Control*, Las Vegas, NV, Dec. 2002, pp. 2829–2835.
[12] R. Teo, D. Stipanovic, and C. Tomlin, "Multiple vehicle control over a lossy datalink," in *Proc. 4th Asian Control Conf.*, Singapore, Sep. 2002, pp. 2063–2068.
[13] A. Bayen, P. Grieder, and C. Tomlin, "A control theoretic predictive model for sector-based air traffic flow," presented at the AIAA Guidance, Navigation, Control Conf., Monterey, CA, Aug. 2002, Paper 2002-5011.
[14] A. Bayen, P. Grieder, H. Sipma, G. Meyer, and C. Tomlin, "Delay predictive models of the national airspace system using hybrid control theory," in *Proc. Amer. Control Conf.*, Anchorage, AK, Jun. 2000, pp. 762–772.
[15] J. Evans, G. Inalhan, J. S. Jang, R. Teo, and C. Tomlin, "Dragonfly: A versatile UAV platform for the advancement of aircraft navigation and control," in *Proc. 20th IEEE Digital Avionics Syst. Conf.*, Oct. 2001, pp. 1C3/1–1C3/12.
[16] R. Teo and C. Tomlin, "Provably safe evasive maneuvers against blunders in closely spaced parallel approaches," presented at the AIAA Guidance, Navigation, Control Conf., Montreal, QC, Canada, Aug. 2001, Paper 2001-4293.
[17] J. S. Jang and C. Tomlin, "Autopilot design for the stanford dragonfly UAV: Validation through hardware-in-the-loop simulation," presented at the AIAA Guidance, Navigation, Control Conf., Montreal, QC, Canada, Aug. 2001, Paper 2001-4179.
[18] N. Shishido and C. Tomlin, "Ellipsoidal approximation of reachable sets for linear games," in *Proc. 39th IEEE Conf. Decision Control*, Sydney, Australia, Dec. 2000, pp. 999–1004.
[19] R. Teo and C. Tomlin, "Computing provably safe aircraft to aircraft spacing for closely spaced parallel approaches," in *Proc. 19th IEEE Digital Avionics Syst. Conf.*, Sep. 2000, pp. 2D2/1–2D2/9.
[20] R. Ghosh and C. Tomlin, "Nonlinear inverse dynamic control for mode-based flight," presented at the AIAA Guidance, Navigation, Control Conf., Denver, CO, Aug. 2000, Paper 2000-4066.
[21] ——, "Manoeuvre design for multiple aircraft conflict resolution," in *Proc. Amer. Control Conf.*, Chicago, IL, Jun. 2000, pp. 672–676.
[22] M. Oishi and C. Tomlin, "Switching in nonminimum phase systems: Applications to a VSTOL aircraft," in *Proc. Amer. Control Conf.*, Chicago, IL, Jun. 2000, pp. 487–491.
[23] ——, "Switched nonlinear control of a VSTOL aircraft," in *Proc. 38th IEEE Conf. Decision Control*, Phoenix, AZ, Dec. 1999, pp. 2685–2690.
[24] J. Evans, W. Hodge, J. Liebman, C. Tomlin, and B. Parkinson, 'Flight tests of an unmanned air vehicle with integrated multi-antenna GPS receiver and IMU: Towards a test bed for distributed control and formation flight," in *Proc. ION-GPS Conf.*, Nashville, TN, Sep. 1999. [Online]. Available: http://www.sun-valley.stanford.edu/~tomlin/papers/Iongps99.pdf
[25] C. Tomlin, G. Pappas, and S. Sastry, "Conflict resolution for air traffic management: A case study in multi-agent hybrid systems," *IEEE Trans. Autom. Control*, vol. 43, no. 4, pp. 509–521, Apr. 1998.
[26] G. J. Pappas, C. Tomlin, J. Lygeros, D. N. Godbole, and S. Sastry, "A next generation architecture for air traffic management systems," in *Proc. 36th IEEE CDC*, San Diego, CA, Dec. 6–8, 1997, pp. 2405–2410.
[27] A. Bicchi and L. Pallottino, "On optimal cooperative conflict resolution for air traffic management systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 1, no. 4, pp. 221–232, Dec. 2000.
[28] Z. H. Mao, E. Feron, and K. Bilimoria, "Stability of intersecting aircraft flows under decentralized conflict avoidance rules," presented at the AIAA Guidance, Navigation, Control Conf., Denver, CO, 2000, Paper 2000-4271.
[29] D. Dugail, Z. H. Mao, and E. Feron, "Stability of intersecting aircraft flows under centralized and decentralized conflict avoidance rules," presented at the AIAA Guidance, Navigation, Control Conf., Exhibit, Montreal, QC, Canada, Aug. 6–9, 2001, Paper 2001-4296.
[30] Z. H. Mao and E. Feron, "Stability and performance of intersecting aircraft flows under sequential conflict resolution," in *Proc. Amer. Control Conf.*, Arlington, VA, Jun. 25–27, 2001, pp. 722–729.
[31] L. Pallotino, A. Bicchi, and E. Feron, "Mixed integer programming for aircraft conflict resolution," presented at the AIAA Guidance, Navigation, Control Conf. Exhibit, Montreal, QC, Canada, Aug. 6–9, 2001, Paper 2001-4295.
[32] T. Schouwenaars, B. De Moor, E. Feron, and J. How, "Mixed integer programming for multiple-vehicle path planning," in *Proc. Eur. Control Conf.*, Porto, Portugal, Sep. 4–7, 2001, pp. 2603–2608.
[33] A. Richards, J. How, T. Schouwenaars, and E. Feron, "Plume avoidance maneuver planning using mixed integer linear programming," presented at the AIAA Guidance, Navigation, Control Conf. Exhibit, Montreal, QC, Canada, Aug. 6–9, 2001, Paper 2001-4091.
[34] M. A. Christodoulou and C. Costoulakis, "Nonlinear mixed integer programming for aircraft collision avoidance in free flight," in *Proc. MELECON Conf.*, Dubrovnik, Croatia, May 12–15, 2004, vol. 1, pp. 327–330.
[35] ——, "Nonlinear mixed integer programming for aircraft collision avoidance in free flight: Minimum time index," in *Proc. MED, IEEE Conf.*, session MoA1-B.4, Aydin, Turkey, Jun. 6–9, 2004, p. 24.

Manolis A. Christodoulou (S'78–M'82–SM'89) was born in Kifissia, Greece, in 1955. He received the Diploma degree from the National Technical University of Athens, Greece, in 1978, the M.S. degree from the University of Maryland, College Park, in 1979, the Engineer degree from the University of Southern California, Los Angeles, in 1982, and the Ph.D. degree from the Democritus University, Thrace, Greece, in 1984.

In 1988, he joined The Technical University of Crete, Greece, where he is currently a Professor of control. He has been a Visiting Professor at Georgia Institute of Technology, Syracuse University, the University of Southern California, Tufts University, Victoria University, and the Massachusetts Institute of Technology. He is the Organizer of various conferences and sessions of IEEE and International Federation of Automatic Control (IFAC) and Guest Editor in various special issues of international journals. He is managing and cooperating on various research projects in Greece, in the European Union, and in collaboration with the United States. He has held many administrative positions such as the Vice Presidency of the Technical University of Crete, as Chairman of the office of Sponsored research, and as a member of the Board of Governors of the University of Peloponnese. He is a member of the Technical Chamber of Greece. He has authored and coauthored more than 170 journal articles, book chapters, books, and conference publications in the areas of control theory and applications, robotics, factory automation, computer integrated manufacturing in engineering, neural networks for dynamic system identification and control, in the use of robots for minimally invasive surgeries, and recently in systems biology. He is cited more than 330 times in the *International Journal Science Citation Index* and more than 1000 times in all journals, books, as well as conference proceedings.

Dr. Christodoulou has been active in the IEEE CS society as the Founder and first Chairman of the IEEE Control Systems Society Greek Chapter, which received the 1997 Best Chapter of the Year Award, and as the Founder of the IEEE Mediterranean Conference on Control and Automation, which became an annual event. He also received the Mediterranean Control Association (MCA) Founders award in 2005. He has been a member of the Board of Governors of the MCA since 1993.

Sifis G. Kodaxakis (S'05) was born in Chania, Crete, Greece, in 1976. He received the Diploma and the M.S. degrees from the Technical University of Crete, Chania, Greece in 2000 and 2004, respectively. He is currently working towards the Ph.D. degree at the Technical University of Crete.

He has been a Teaching Assistant at the same department since 2002. His main research interests are in the area of neural networks for doping detection in athletes and recently in air-traffic (AT) algorithms for intelligent control of free flying aircraft.