AIAA 2003–5341

A differential game formulation of alert levels in ETMS data for high altitude traffic

Alexandre M. Bayen
Stanford University, Stanford, CA 94305

Shriram Santhanam
Stanford University, Stanford, CA 94305

Ian Mitchell
University of California at Berkeley, Berkeley, CA 94720

Claire J. Tomlin
Stanford University, Stanford, CA 94305

GNC Conference
11–14 August, 2003/Austin, Texas

A differential game formulation of alert levels in ETMS data for high altitude traffic*

Alexandre M. Bayen †
Stanford University, Stanford, CA 94305

Shriram Santhanam ‡
Stanford University, Stanford, CA 94305

Ian Mitchell §
University of California at Berkeley, Berkeley, CA 94720

Claire J. Tomlin ¶
Stanford University, Stanford, CA 94305

This paper applies a differential game formulation of a two-vehicle collision avoidance problem to the derivation of an alerting logic for conflicts in high altitude air traffic. Using computational methods based on level sets, the three dimensional viscosity solution of the Hamilton-Jacobi-Isaacs equation describing this game is calculated. This solution is employed to define unsafe regions for each pair of aircraft in the relevant airspace, and these regions are used as a metric to indicate if loss of separation could occur. This metric is evaluated using ETMS data, by comparing the alert result with behavior observed in the current Air Traffic Control system. The possible use of this method for human-in-the-loop decision support tools is discussed. A key advantage of this technique over previously developed collision detection methods is that it does not assume anything about the type of blunder that may occur — rather it computes worst case blunders from the aircraft kinematic configuration.

Keywords: differential games, Hamilton-Jacobi-Isaacs equation, high altitude traffic, ETMS, human in-the-loop.

Introduction

The *National Airspace System* (NAS) is a large scale, nonlinear dynamic system, with a control authority which is organized hierarchically. A single *Air Traffic Control System Command Center* (ATCSCC), in Herndon VA, supervises the overall traffic flow, and this is supported by 22 (20 in the continental US or CONUS) *Air Route Traffic Control Centers* (ARTCCs, or simply, Centers) organized by geographical region and controlling the airspace up to 60,000 feet.[10,14,15,17,21,23,24] Each Center is sub-divided into about 20 sectors, with at least one air traffic controller responsible for each sector. Each sector air traffic controller (ATC) may talk to 20-25 aircraft at a given time (the maximum allowed number of aircraft per sector depends on the sector itself). The controller guides the aircraft through the sector using a set of standard commands (over voice channels). In general, the controller has access to the aircraft's flight plan and may revise the altitude and provide temporary heading assignments, amend the route, speed, or profile in order to maintain safety and attempt to optimize the flow. One of the most important and time consuming controller tasks is to prevent a *loss of separation* (LOS), between aircraft; for high altitude sectors (above 29,000 ft), this means that the controller must keep each pair of aircraft in the sector separated by more than 5 nautical miles (nm) horizontally, and 1000 feet vertically. The terminology *protected zone* is used to represent the 5 nm radius, 2000 foot high cylinder around an aircraft, that another aircraft must not penetrate. For any pair of aircraft, the relative configuration or *state* (relative position and orientation) is referred to as *unsafe* if there is a *rational* process of actions which leads one aircraft to penetrate the protected zone of another.

A method for numerically computing unsafe states for pairs (a, b) of aircraft has been developed in previous work.[20,27] In this work, the action of one of the aircraft (aircraft b) is assumed to be uncertain

*Research supported by NASA under Grant NCC 2-5422, by ONR under MURI contract N00014-02-1-0720, by DARPA under the Software Enabled Control Program (AFRL contract F33615-99-C-3014) and by a Graduate Fellowship of the Délégation Générale pour l'Armement (France).

†AIAA student member, Ph.D. Student, Hybrid Systems Laboratory, Aeronautics and Astronautics, Stanford University.

‡M.S. student, Hybrid Systems Laboratory, Mechanical Engineering, Stanford University.

§Research Associate, Electrical Engineering and Computer Sciences, University of California at Berkeley.

¶AIAA member, Assistant Professor, Aeronautics and Astronautics, Hybrid Systems Laboratory, Stanford University.

but bounded within known bounds, and is treated as an opposing player to the control action of the first aircraft (aircraft a). From these assumptions, using the framework of differential game theory (the celebrated pursuit evasion game of Isaacs[12]), two regions may be calculated and used to determine LOS threats. These regions are: *(i)* the set of states from which aircraft b can cause a LOS of separation with aircraft a, regardless of the control action of aircraft a; *(ii)* the set of states from which aircraft a can find a safe escape maneuver to prevent conflict, for all possible actions of aircraft b. This scenario could model the situation in which one aircraft blunders, due to navigation error, human error, or communication loss, yet ATC maintains communication and control of the second aircraft. Variations of this framework could be used to treat other scenarios, such as the case in which communication is lost with both aircraft, yet still only one blunders, as well as the true worst case, in which communication is lost and the aircraft appear to coordinate to cause a collision. This third case is extreme, yet it could be used to assess the true worst case: a tragic recent example is the midair crash in July 2002 above Überlingen, caused by communication of erroneous maneuvers.

Recently, with the help of new computational methods[19,20] for calculating these two classes of states relatively fast and efficiently, it has become feasible to apply these methods to problems in ATC.

In the current paper, our goal is to evaluate this tool as a possible online ATC advisory for assessing the LOS *alert level* in high altitude traffic. By alert level, we mean a metric which can be used practically to indicate when ATC should modify the aircraft's trajectory. As a testbed for this work, we use a set of *Enhanced Traffic Management System* (ETMS) data for high altitude traffic in several sectors of the Oakland Center airspace. The ETMS data file used for this work is from several years ago. This paper is to the best of our knowledge the first application of Isaacs' *Game of two identical vehicles* to ATC with real data (in the original work of Isaacs[12] and Merz,[18] specific cases are investigated, with dummy numerical parameters).

Using the classification of Kuchar and Yang,[16] the *state propagation* of our method is *worst case*, the *state dimension* is two (even if our system enables three dimensional conclusions, as will appear), the definition of *detection* is one of the points of this work, and the *conflict resolution* is *optimized*. We attempt to answer some of the questions raised by Kuchar.[16] In particular, we show that a worst case approach does not always provide false alarms and might be a very appropriate metric for short term conflicts, and we also discuss the issue of the time horizon.

Finally, using the ETMS data, we demonstrate a good agreement between the decisions advised by our tool, and the human ATC decision currently observed in the ARTCC (and observable in ETMS data through comparison of actual trajectory with original flight plan). Thus, we believe that the tool presented in this paper could provide a useful advisory to ATC.

Problem statement and formulation

In the current ARTCC monitoring system, human Air Traffic Controllers have a visual display showing all airborne aircraft in their airspace (sector). Numerous information can be superimposed on the symbol corresponding to each aircraft, such as: heading vector, speed, altitude, filed flight plan, flight plan based anticipated positions of the aircraft in the current sector. Based on this information, the controller makes a decision regarding which flight plans to alter to avoid potential LOS. The controller decision is based on anticipation of the positions of each pair of potentially conflicting aircraft. The set of commands applied to the aircraft to achieve certain goals (spacing, conflict avoidance) has been well studied.[1,2,11] The method presented in this article aims at making this anticipation systematic and providing a guarantee on the relative separation of the two aircraft, through the generation of an automated tool to aid the decision-making of the controller:

> **Problem:** Consider a pair of aircraft (a, b) in the en route airspace, each aircraft having known bounds on its control actions. Compute the set of relative positions and orientations of a and b from which for all possible control action of aircraft a (resp. b), there exists an uncertainty in the control input of aircraft b (resp a) rendering the pair susceptible to a LOS.

This problem captures, for example, the situation in which one aircraft blunders, and ATC maintains communication with the second aircraft. In addition, two other problems are investigated, which are relevant for situations with communication loss. First, given an aircraft a following its flight plan and an aircraft b, what is the set of relative configurations of a and b from which there exists an uncertainty in the input of b which could cause a LOS? Second, given two aircraft a and b, what is the set of relative positions of a and b from which the two aircraft can collaboratively (unwillingly) create a LOS?

Reachable set formulation of the collision avoidance problem

The collision avoidance scenario used for each pair of aircraft is modeled using the *Game Of Two Identical Vehicles*, introduced in Isaacs.[12] This differential game was solved analytically by Merz for a particular choice of numerical parameters[18] and has been a focus of research.[22] Mitchell and al.[20] provides a mathematical method to solve the problem (and other differential

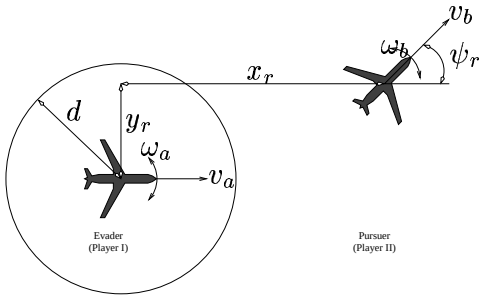

Fig. 1 Relative coordinate system. Origin is located at the center of the evader.

var.	meaning
x_r	rel. position in direction of evader's flight
y_r	rel. position perp. to direction of evader
ψ_r	rel. heading ($0 \leq \psi_r < 2\pi$)
z	state vector ($z = [x_r, y_r, \psi_r]^T$)
ω_a	angular vel. and input of evader ($\omega_a \in \mathcal{A}$)
ω_b	angular vel. and input of pursuer ($\omega_b \in \mathcal{B}$)
v_a	speed of evader
v_b	speed of pursuer
d	minimum safe separation distance ($d = 5$nm)
$\mathcal{G}_0$	LOS set $\{(x_r, y_r, \psi_r) \mid x_r^2 + y_r^2 \leq d^2$

Table 1 Variables for the two vehicles game.

games) numerically in the general case, based on the initial formulation of Tomlin and al.[27] Our work uses this formulation, extended to the case of non-identical vehicles, having different capabilities, which we now summarize.

The two aircraft system is modeled with a commonly used, very simple kinematic system. The state of each vehicle is represented by a location in the $x-y$ plane and a heading ψ relative to the x-axis. The evolution of these states is governed by the vehicle's forward velocity v and rotational velocity ω:

$$\frac{d}{dt}\begin{bmatrix} x \\ y \\ \psi \end{bmatrix} = \begin{bmatrix} v\cos\psi \\ v\sin\psi \\ \omega \end{bmatrix} \tag{1}$$

The linear velocity v of each aircraft is assumed fixed. The angular velocity (turn rate) ω is allowed to vary, and is considered here the input of the aircraft. Even if in practice the aircraft could also change their velocities, most of the conflicts of the type investigated here are solved horizontally by modifying the heading of one of the vehicles, which makes the assumption of constant velocity valid. Conflict resolutions involving speed changes are also observed in practice (mostly in converging traffic[2]), but for these cases, conflict resolution is very tightly linked with scheduling[3] which is a fundamentally different problem and uses other mathematical resolution techniques. The two vehicles have the same kinematics (1) with different values of v and ranges of ω. Using differential game terminology,[12] we denote the evader by the subscript a and the pursuer by the subscript b. The turn rate of each aircraft is limited by the performance of the aircraft. We denote $\mathcal{A} = [\underline{\omega}_a, \overline{\omega}_a]$ the range of possible turn rates of aircraft a and similarly $\mathcal{B} = [\underline{\omega}_b, \overline{\omega}_b]$ for aircraft b. The values of $\underline{\omega}_a$, $\overline{\omega}_a$, $\underline{\omega}_b$ and $\overline{\omega}_b$ will be derived later.

We say that a LOS has occurred if the two vehicles come within distance d of one another. Our goal is to determine the set of states from which the pursuer can cause a LOS to occur. Translating into reachability terms, we define $\mathcal{G}_0$ as the set of all states at which the two vehicles are within d units of one another. Denote by $\mathcal{G}(\tau)$ the set in which the pursuer can cause a LOS in the next τ time units despite the best efforts of the evader. This set is variously referred to as the *reachable set*,[20] *victory domain*,[12] or *discriminating kernel*.[6] We can simplify the two-aircraft system to three dimensions by working in relative coordinates. Furthermore, because the variable x has special meaning in the horizontal plane, throughout this section we will denote the state vector as $z \in \mathbf{R}^3$. We fix the evader at the origin, facing along the positive x_r axis (see Figure 1 and Table 1). Then, the pursuer's relative location and heading are described by the following dynamical system:

$$\dot{z} = \begin{bmatrix} \dot{x}_r \\ \dot{y}_r \\ \dot{\psi}_r \end{bmatrix} = \begin{bmatrix} v_b\cos\psi_r - v_a + \omega_a y_r \\ v_b\sin\psi_r - \omega_a x_r \\ \omega_b - \omega_a \end{bmatrix} = f(z, \omega_a, \omega_b) \tag{2}$$

where v_a and ω_a are the velocity and turn rate of the evader, v_b and ω_b, of the pursuer. The meaning of the variables above is explained in Table 1.

Tomlin and al.[27] derived a general algorithm for computing $\mathcal{G}(\tau)$ for hybrid systems; the details of the computation of $\mathcal{G}(\tau)$ for a single mode has been developed by Mitchell and al.[20] We summarize these results[20, 27] here. Since a collision can occur at any relative heading, the target set $\mathcal{G}_0$ depends only on x_r and y_r and includes any state within distance d of the planar origin:

$$\mathcal{G}_0 = \{z \in \mathbf{R}^3 \mid x_r^2 + y_r^2 \leq d^2\} \tag{3}$$

which can be converted into a signed distance function

$$\phi_0(z) = \sqrt{x_r^2 + y_r^2} - d \tag{4}$$

The set $\mathcal{G}_0$ is given by $\mathcal{G}_0 = \{z \in \mathbf{R}^3 \mid \phi_0(z) \leq 0\}$. We proved[20] that the set $\mathcal{G}(\tau)$ is given by $\mathcal{G}(\tau) = \{z \in \mathbf{R}^3 \mid \phi(z, -\tau) \leq 0\}$, where $\phi(\cdot,\cdot)$ is the viscosity solution[7, 8] of the following modified *Hamilton-Jacobi-Isaacs partial differential equation* (HJI PDE).

$$\frac{\partial\phi(z,t)}{\partial t} + \min[0, H^*(z, \nabla\phi(z,t))] = 0 \tag{5}$$

for $t \in \mathbf{R}^-$, with terminal conditions

$$\phi(z, 0) = \phi_0(z) \tag{6}$$

In (5), $\nabla\phi$ represents the gradient of ϕ, and H^* is the Hamiltonian of the system. For the problem defined at the beginning of this section, in which one aircraft blunders, H^* is defined as:

$$H^*(z,p) = \max_{\omega_a \in \mathcal{A}} \min_{\omega_b \in \mathcal{B}} H(z,p,\omega_a,\omega_b) \qquad (7)$$

with H given by: $H(z,p,\omega_a,\omega_b) = p^T \cdot f(z,\omega_a,\omega_b)$:

$$H(z,p,\omega_a,\omega_b) = \begin{array}{l} -p_1 v_a + p_1 v_b \cos\psi_r + p_2 v_b \sin\psi_r \\ +\omega_a (p_1 y_r - p_2 x_r - p_3) + \omega_b p_3 \end{array} \qquad (8)$$

In the previous formula, the costate is $p := [p_1, p_2, p_3]$. The optimal input ω_a^* and worst disturbance ω_b^* achieving H^* in (7) can be easily computed from (8):

$$\omega_a^* = (\underline{\omega}_a + \overline{\omega}_a)/2 + \mathrm{sgn}(p_1 y_r - p_2 x_r - p_3)(\overline{\omega}_a - \underline{\omega}_a)/2$$
$$\omega_b^* = (\underline{\omega}_b + \overline{\omega}_b)/2 - \mathrm{sgn}(p_3)(\overline{\omega}_b - \underline{\omega}_b)/2$$

Equation (5) has to be solved from time $t = 0$ backwards to time $t = -\tau \leq 0$. $\mathcal{G}(\tau)$ is shown in Figure 2 left and right for $\tau = 30$sec. and $\tau = 120$sec. The sets $\mathcal{G}(\tau)$ can be interpreted the following way. If the relative positions of two aircraft are such that $(x_r, y_r, \psi_r) \in \mathcal{G}(\tau)$, then the pursuer can cause a LOS in the next τ time units. If $(x_r, y_r, \psi_r) \notin \mathcal{G}(\tau)$, then no matter what the pursuer does, the evader can always avoid a LOS. As can be seen in Figure 2 center and right, the growth of $\mathcal{G}(\tau)$ is not isotropic: Figure 3 shows slices of the set $\mathcal{G}(\tau)$ for various τ, obtained at $\psi_r = 90°$. The slices extend in the $(1,-1)$ direction (to the bottom right on the plot): the conflict is most difficult to avoid when the relative position of the two aircraft points in this direction (the two aircraft are heading towards the same point). The set $\mathcal{G}(\tau)$ converges (i.e. stops growing) after $\tau = 150$sec. This fact, which is proved analytically for $\mathcal{A} = \mathcal{B}$,[18] will have practical significance to our implementation, in which we will demonstrate numerical convergence of the set even for non identical vehicles. A possible heuristic interpretation of this fact is that a can "always" escape by turning, even if b is faster. Similar observations have been reported by Saint-Pierre for an analogous acoustic capture problem.[6] Note that the shapes of the two dimensional slices of $\mathcal{G}(\tau)$ change as the heading changes.

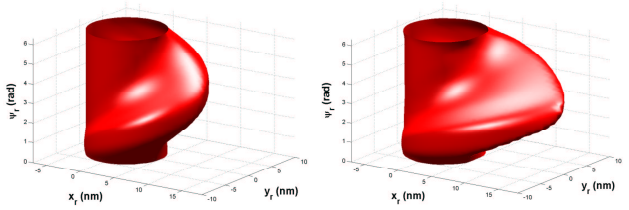

Fig. 2 Left: Set $\mathcal{G}(\tau)$ for $\tau = 30$sec. Right: Set $\mathcal{G}(\tau)$ for $\tau = 120$sec. (the set has converged). For this case, both aircraft are cruising at 500kts. $\mathcal{A} = \mathcal{B} = [-1.3, 1.3]$ °sec^{-1}.

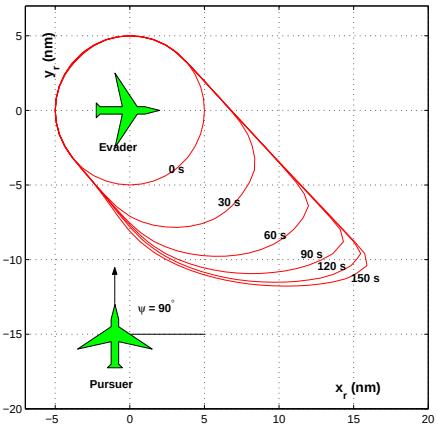

Fig. 3 Slice of $\mathcal{G}(\tau)$ from Figure 2 right, for $\psi_r = 90°$, for $\tau = 30,\ 60,\ 90,\ 120,\ 150$ sec. The set extends in the direction $(1,-1)$: the conflict is most difficult to avoid when the pursuer is initially to the bottom right of the evader on this plot (the aircraft are heading towards the same point). Numerical convergence to a fixed point is observed for $\tau \geq 150$sec.: $\mathcal{G}(\tau)$ stops growing.

Extraction of parameters from ETMS data

In the following sections of this article, we use the method summarized above. We perform reachability computations (i.e. computing the set $\mathcal{G}(\tau)$ from the set $\mathcal{G}_0$) for given choices of v_a, v_b, and ranges $\mathcal{A}$ and $\mathcal{B}$ of input turning rates ω_a and ω_b. We now explain how to choose these values from ETMS data.

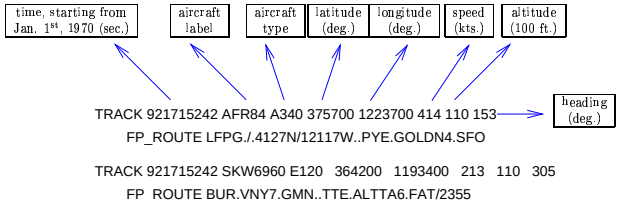

Fig. 4 Format of the ETMS data subset used for the reachability computations.[5] The data is available approximately every 3 min. We only use a subset of the data (two lines per airborne aircraft per time tag, whereas in the full data set, ETMS also contains lists of jetways and centers). The first line gives the flight information for one airborne aircraft in the NAS: time, flight number, aircraft type, latitude, longitude, speed, altitude and heading. The second line is the filed flight plan and can be used to determine the intent of the aircraft. Each of the acronyms in this line corresponds to a navaid, a fix, a jetway or an arrival.[1,2]

The ETMS database contains all flight plan information for flights in the NAS. Data are collected from the entire population of flights in the NAS with filed flight plans. ETMS data is sent from the Volpe National Transportation System Center to registered participants via the Aircraft Situation Display to Industry electronic file server. The Federal Aviation Administration (FAA) uses these data to monitor the effectiveness of its National Route Program, in which the user community is offered flexible, cost-effective routing options as an alternative to published ATC preferred routes. A subset of the data used is shown in Fig-

ure 4. Let us consider any pair of aircraft described in this ETMS set, and arbitrarily assign one as the evader (aircraft a) and one as the pursuer (aircraft b). From ETMS data, we can extract the velocity v_a of the evader aircraft; from this, we determine the range $\mathcal{A}$ of available turning rates ω_a as follows. A first order approximation of the dynamics of aircraft a provides the following equations for lift L, weight $W = mg$, bank angle α, mass m and turning radius R_a:

$$L\cos\alpha = W \quad \text{and} \quad L\sin\alpha = mv_a^2/R_a \tag{9}$$

from which we can compute R_a:

$$R_a = v_a^2 \;/\; g\tan\alpha \tag{10}$$

The previous result provides a lower bound on R_a, given by $\underline{R}_a = \frac{v_a^2}{g\tan\alpha_{\max}}$, where $\alpha_{\max}$ is the maximal bank angle, which for this study is $\alpha_{\max} = 45°$. The turning radius can be related to the maximal turning rate by: $\overline{\omega}_a = \frac{v_a}{\underline{R}_a}$, and symmetrically, $\underline{\omega}_a = -\frac{v_a}{\underline{R}_a}$, which provides $\mathcal{A} = \left[-\frac{v_a}{\underline{R}_a}, \frac{v_a}{\underline{R}_a}\right]$ and similarly for $\mathcal{B}$. Using this relationship, the parameters of the relative kinematics (2) of the two aircraft are completely defined. In order to solve the corresponding HJI PDE (5), we need to prescribe ϕ_0: $\mathcal{G}_0$ is chosen to be a cylinder of radius $d = 5$nm according to FAA regulations. In this article, we use one 24 hour subset of ETMS data in several sectors of the Oakland airspace.

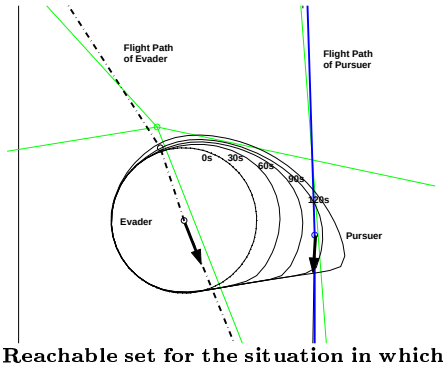

Fig. 5 Reachable set for the situation in which aircraft a (dash dotted flight plan) and aircraft b (solid flight plan) have intersecting flight plans. $\mathcal{G}(\tau)$ is shown for $\tau = 30,\ 60,\ 90,\ 120$sec. Aircraft b is in $\mathcal{G}(\tau = 90)\backslash\mathcal{G}(\tau = 60)$, which means that aircraft b is a potential threat to aircraft a in the next 90sec, but aircraft a is safe for at least 60sec. The conflict would not have been detected if the time horizon of the computation had been 60sec. For the rest of the computations, we will take a time horizon of 150sec., in order to take into account points at the extremity of the reachable set.

Choice of the time horizon, computational time

We need to determine the time horizon τ for the computation of $\mathcal{G}(\tau)$. This choice is determined by the time scale of the physical problem of interest: a conflict avoidance maneuver is on the order of several minutes; the anticipation ability of a human ATC is also on the order of several minutes. Therefore, we choose a time horizon of $\tau = 3$ min. The distance flown at 500kts. in 3min. is 25nm., which enables one to capture potential threats with aircraft 50nm from each other. Even if 50nm. might seem excessive, it is necessary to choose at least 3min., as is illustrated by the growth of the reachable set in Figure 3. One can see in this figure that the shape of $\mathcal{G}(\tau)$ changes significantly in the interval $[0, 3\,\text{min}]$.

Alt. (100ft)	310	350
num. of measur.	2097	2515
Min. v (kts)	414	410
Max. v (kts)	514	480
Mean v	454	441
Std. dev.	16	12
Min. ω (deg·s^{-1})	2.12	2.27
Max ω (deg·s^{-1})	2.63	2.66
Mean ω (deg·s^{-1})	2.41	2.47
Std. dev.	0.079	0.067

Table 2 Speed and extremal turning rate distribution for two altitudes of traffic (31,000ft and 35,000ft), over a 24 hour period in the Oakland ARTCC. The number of measurements used for the computation of these statistics is reported in the first row. Each aircraft can lead to several measurements (if an aircraft appears more than once at the corresponding altitude).

In practice, numerical convergence was observed for $\tau \leq 3$ min for all the examples treated here (see for example Figures 3 and 5). This confirms the relevance of our choice: numerical convergence for $\tau \leq 3$ min implies that the duration of a conflict avoidance maneuver (obtained when a chooses ω_a^* to be its input during the maneuver) is of that order.

The computation of $\mathcal{G}(\tau)$ can not yet be performed online. It takes approximately 5 minutes to compute $\mathcal{G}(3\,\text{min})$ (on a standard laptop) for a case similar to Figure 2. It is therefore impossible for an online implementation of our method to perform this computation in real time for a given pair of $\mathcal{A}$ and $\mathcal{B}$. However, it is possible to create a library of $\mathcal{G}(3\,\text{min})$ for different $\mathcal{A}$ and $\mathcal{B}$. Since this method is a horizontal conflict detection method — and therefore applies to aircraft within 2000 ft of each other, we can use the fact that the range of v_a, v_b is relatively small (and therefore the same is true for the ranges of possible $\mathcal{A}$ and $\mathcal{B}$). Table 2 justifies the fact that velocities at a given altitude have a relatively small range. The range in turning rates is accordingly small.

In computing the statistics of Table 2, we accounted for the fact that the speeds might change over time. Therefore, the average quantities in Table 2 are computed using one entry for each occurrence of an aircraft in the 24 hour ETMS data set which we use. For example, if an aircraft appears 12 times at altitude 35,000ft

before it changes altitude, we will account for the 12 speed records. We used a database of $\mathcal{G}(3\,\text{min})$ for our implementation and believe that a similar database could be used for an online implementation.

Choice of the strategy

The differential game formulation (5),(6) enables the investigation of multiple scenarios, which are all relevant for ATC. Figure 6 shows three possible applications of this methodology:

1. *Communication loss with a blunderer.* This situation is labeled DG (differential game). It is modeled by the differential game setting developed previously. Blunders can happen in ATC because of navigation or human errors. This scenario models the situation in which one aircraft blunders and ATC communicates with the other aircraft (and prescribes a corresponding conflict resolution maneuver). This setting thus encompasses the worst case blunderer, heading directly towards the other aircraft.

2. *Communication loss with both aircraft in presence of a blunderer and a "blind aircraft".* This situation is labeled BB (blunderer + blind). ATC loses communication with both aircraft. In general, when there is a communication loss, aircraft are supposed to follow their original paths until communication is reestablished. The present scenario models a situation in which one aircraft follows its original path while the other blunders (because of navigation or human error, for example). The worst case for this scenario is thus: the pursuer tries to cause LOS directly, while the evader blindly stays on its course. This is modeled by setting $\mathcal{A} = \{0\}$ (in other words, the evader has no input, i.e. no turning ability).

3. *Collaborative collision strategy.* This situation is labeled CC (collaborative collision). Because of misunderstanding in communication with ATC or failure to follow procedures, both aircraft might collaborate (unwillingly) to a LOS. The worst case scenario for collaborative collision is thus when both aircraft head directly at each other from their initial position. It can be modeled by replacing the Hamiltonian (7) by $H^*(z,p) = \min_{\omega_a \in \mathcal{A}} \min_{\omega_b \in \mathcal{B}} H(z,p,\omega_a,\omega_b)$. In the previous formula, the two aircraft contribute to the LOS, as can be seen in the min-min instead of the max-min.

Conflict avoidance maneuvers typically fall into category 1 (the no blunder case is encompassed by the worst blunder case). However, a case such as the midair crash above Überlingen (Germany) between a DHL Boeing 757-200 (DHX 611) and a Bashkirian Airlines Tupolev 154 (BTC 2937) on July 1st, 2002, was caused by the simultaneous action of the two pilots. The DHX Boeing followed a command issued by TCAS (onboard), while the BTC Tupolev followed the (erroneous) orders of ATC, leading to a collaborative collision rather than conflict avoidance. It therefore could fall in categories 2 or 3, depending on how the communication records of the accident are interpreted.

The three scenarios are of equal interest for ATC, and will obviously lead to different reachable sets. It is intuitive to see that the following inclusion holds: $\mathcal{G}_{\text{scenario 1}} \subseteq \mathcal{G}_{\text{scenario 2}} \subseteq \mathcal{G}_{\text{scenario 3}}$. Figure 6 illustrates the growth of $\mathcal{G}(\tau)$ with τ for the three scenarios, for two cases where the relative headings of the pursuer and the evader are respectively $\psi_r = 90°$ and $180°$. The fast growth of $\mathcal{G}(\tau)$ for scenarios 2 and 3 can clearly be seen from these figures.

For the rest of the study, we will focus on scenario 1: it is assumed that ATC can communicate correctly with at least one aircraft. We use this scenario to create a metric for conflict detection.

Alert levels in ETMS data

Conflict detection methodology

We apply the previous method to high altitude traffic in ATC, using prerecorded ETMS data. Since our goal is to develop a technique which can serve as an advisory tool to ATC, the method is designed to work in real time with information provided at a given rate. For the present case, the update rate in the ETMS data is 3 min., but using precomputed sets, our method works as well with higher rates, such as 15 sec., which is on the order of the current monitoring display at a sector level in the ARTCC. Our methodology is as follows:

1	At time t, select all aircraft pairs which have a relative vertical separation of within 2000ft.
2	For a given pair, extract v_a, v_b, compute $\mathcal{A}$, $\mathcal{B}$ using (9) and (10).
3	Compute $\mathcal{G}(\tau)$ for a desired time horizon τ.
3'	Relabel aircraft a and b. Compute new $\mathcal{G}(\tau)$.
4	Compute x_r, y_r, ψ_r from the ETMS data at t. For $\mathcal{G}(\tau)$ of 3 and 3', check if $(x_r, y_r, \psi_r) \in \mathcal{G}(\tau)$.
5	If $(x_r, y_r, \psi_r) \in \mathcal{G}(\tau)$ for either 3 or 3', a potential LOS is detected with τ time units.
6	Go back to 2 until all pairs have been tested.
7	Wait until next data update and return to 1.

We now describe the steps of the method above.

1. The aircraft selection can be done manually or automatically. A manual selection would enable ATC to directly pick out an aircraft pair and test for a LOS threat. It is also very easy to automate the procedure, to do an exhaustive test of all aircraft pairs and display the threats only. Technically all pairs of aircraft should be selected iteratively. It is easy however to introduce heuristics based on separation and heading, in order to discard a large portion of them (for example if two aircraft are separated by more than 50 nm

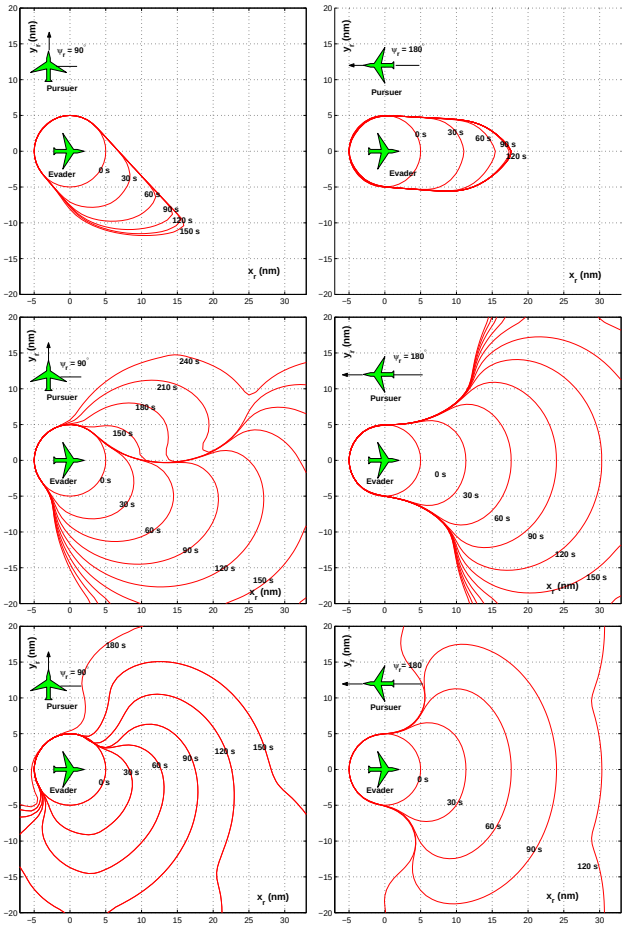

Fig. 6 **Growth of the reachable set $\mathcal{G}(\tau)$ for various values of τ, for a given initial relative heading $\psi_r = 90°$ (left), and $\psi_r = 180°$ (right) of the two aircraft. First row: DG setting. $\mathcal{G}(\tau)$ is bounded, and extends in the $(1,-1)$ – resp. $(1,0)$ direction. For points in the $(1,-1)$ – resp. $(1,0)$ direction outside the reachable set, the evader can avoid the LOS by going around the pursuer on both sides. Second row: BB setting. For the $\psi = 90°$ case, $\mathcal{G}(\tau)$ extends first in the $(1,-1)$ direction: the pursuer can cause a *direct* collision by heading towards the evader if it is below and to the right of the evader on this plot. For $t = 150s$, $\mathcal{G}(\tau)$ grows for $y_r > 0$. The corresponding (x_r, y_r) are positions for which the pursuer has to turn right by $90°$ and intercept the evader (*indirect collision*). For the $\psi = 180°$, the situation is entirely symmetrical. Third row: CC setting. Both aircraft collaborate to a LOS, therefore, $\mathcal{G}(\tau)$ grows much faster, and for the same values of τ, all initial positions in $\mathcal{G}(\tau)$ correspond to a direct collision (both aircraft are heading at each other). For the three cases above, the velocity of the aircraft is 500kts (and therefore, $\mathcal{A} = \mathcal{B} = [-1.3, 1.3]$ °sec^{-1}).**

and are flying opposite directions). This enables one to reduce the computational time required by the method.

2. The velocities are directly read from the ETMS data. $\mathcal{A}$ and $\mathcal{B}$ are computed using the procedure of the previous section.

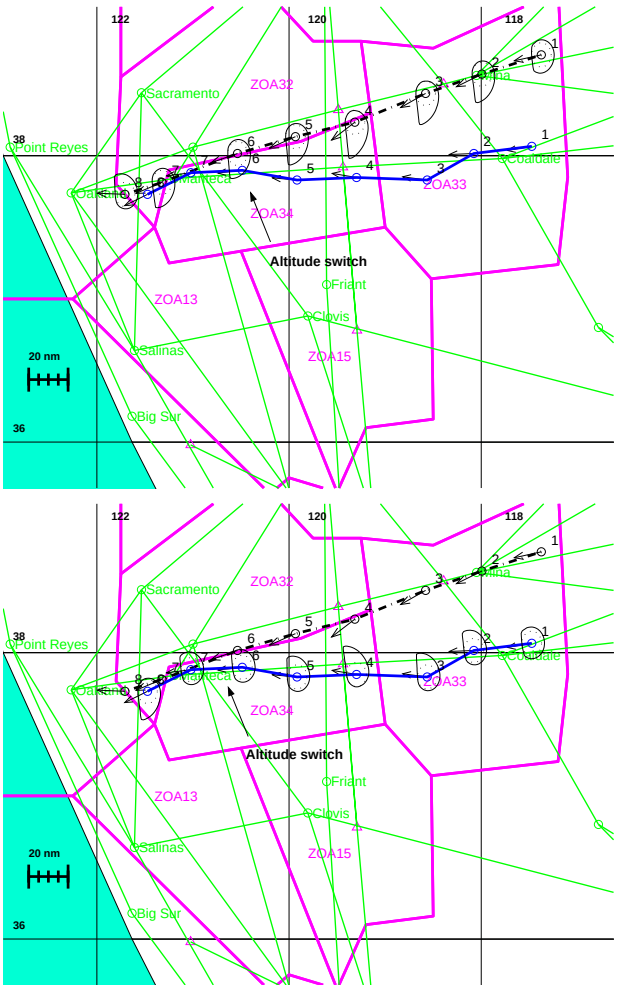

Fig. 7 **Aircraft 1 (solid), arriving from Dallas/Fort Worth Airport to Oakland (OAK); aircraft 2 (dash-dotted), arriving from Indianapolis Airport to OAK. Both aircraft are initially supposed to follow the Madwin 3 arrival (from Coaldale and Mina respectively), entering the Oakland ARTCC (positions labeled 1) at 35,000ft. At the position labeled 6, both aircraft are in $\mathcal{G}(\tau)$ w.r.t. the other aircraft, for $\tau = 3$min. An altitude change - while descending into OAK - is observed shortly after (aircraft 1 descends to 14,000ft, while aircraft 2 descends to 24,000ft). This action of ATC keeps the aircraft separated.**

3. The computation of $\mathcal{G}(\tau)$ is performed by solving the HJE PDI, as described in the previous section. It is realized for an arbitrary choice of a and b, and accounts for a possible blunder of aircraft b. At step 3', the labels a and b are inverted, so that a possible blunder of aircraft a is also taken into account.

4. The relative position of the two aircraft can be computed directly from the latitude and longitude of each aircraft. We ignore earth curvature for conflicts, since the size of the reachable sets for our computations rarely exceeds 50nm. The heading difference ψ_r can be directly computed from the ETMS data as well (Figure 4). Using ψ_r, one can slice $\mathcal{G}(\tau)$ at ψ_r (Figure 2). This provides a

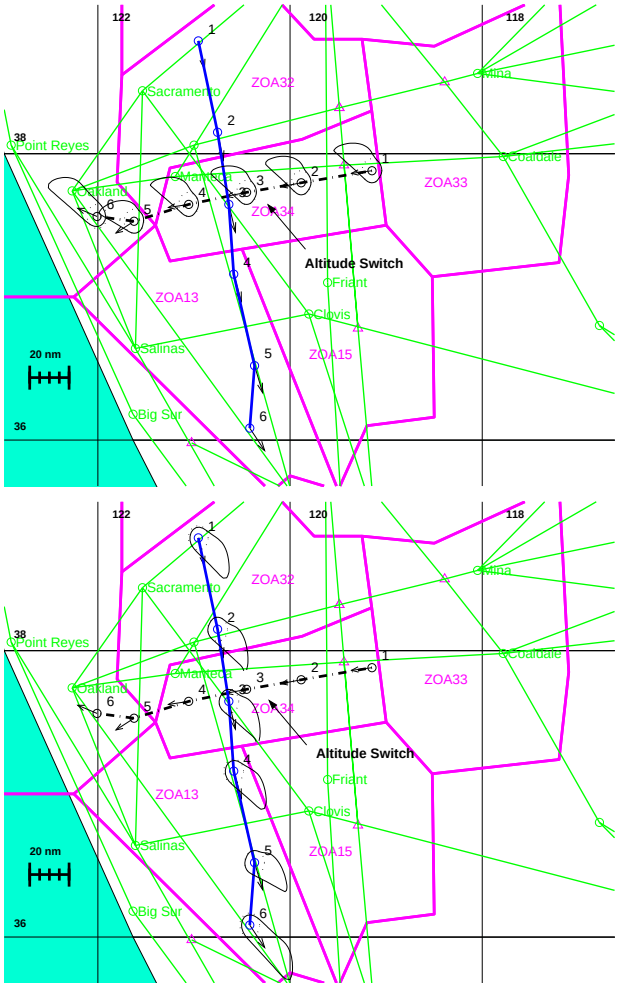

Fig. 8 Aircraft 1 (dash dotted), arriving from Philadelphia Airport to San Francisco Airport (SFO); aircraft 2 (solid), en route from Ted Stevens Anchorage Airport to Los Angeles Airport. Aircraft 1 is using the Modesto 2 arrival to SFO. Interpolation of the positions of both aircraft between the labels 2 and 3 shows that aircraft 2 is in the $\mathcal{G}(\tau)$ for aircraft 1. ATC avoids the conflict by commanding aircraft 1 to descend to 24,000ft (which initiates the descent into SFO).

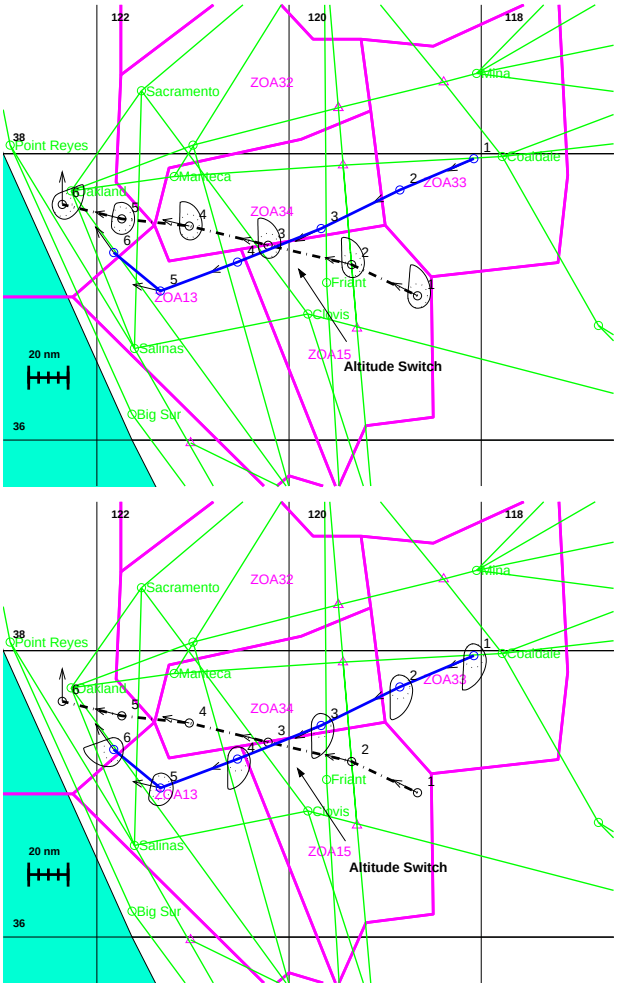

Fig. 9 Aircraft 1 (solid), arriving from Chicago O'Hare Airport to San Jose Airport (SJC); aircraft 2 (dash-dotted), arriving from Phoenix Sky Harbor Airport to San Francisco Airport (SFO). Aircraft 1 is on El Nido arrival at 35,000ft entering through Coaldale; aircraft 2 is on Modesto 2 arrival, entering through Clovis. Interpolation between positions 2 and 3 shows that aircraft 2 is in the $\mathcal{G}(\tau)$ of aircraft 1. The action of ATC coincides with their respective descents: aircraft 1 is commanded to descend to 24,000ft, while aircraft 2 is commanded to descend to 11,000ft (starting from position 2 and 3 respectively).

2D set. Plotting (x_r, y_r) on top of the slice of $\mathcal{G}(\tau)$ shows if the aircraft b is inside $\mathcal{G}(\tau)$ or not, and is a potential threat or not. All examples which follow will be displayed in this format for readability. In practice, the test $(x_r, y_r, \psi_r) \in \mathcal{G}(\tau)$ can be automated using level set methods.[20] It is instantaneous and consists of evaluating ϕ numerically from a grid using an interpolation subroutine.

5. If $(x_r, y_r, \psi_r) \in \mathcal{G}(\tau)$, there is a potential LOS within τ time units. Our method also allows one to check if $(x_r, y_r, \psi_r) \in \mathcal{G}(\tau) \backslash \mathcal{G}(\tau')$ where $\tau' \leq \tau$. In this case, if τ' is large enough to climb or descend to the next floor, a LOS can be avoided by altitude change.

6. All pairs of aircraft have to be tested for potential threats.

7. The update rate of the ETMS data is on the order of 3 min., as appears in the examples presented in this article.

Applications and validation of the results

Several *metrics* have been defined in the past to help the decision making process in ATC.[16] The specific metric used here is *time to minimum separation*. It is easy to adapt our mathematical formulation to other metrics such as *predicted minimum separation* or *estimated time to closest point of approach*,[16] using an equivalence theorem.[20] The goal of the rest of the paper is to show that this metric is appropriate for short term LOS detection.

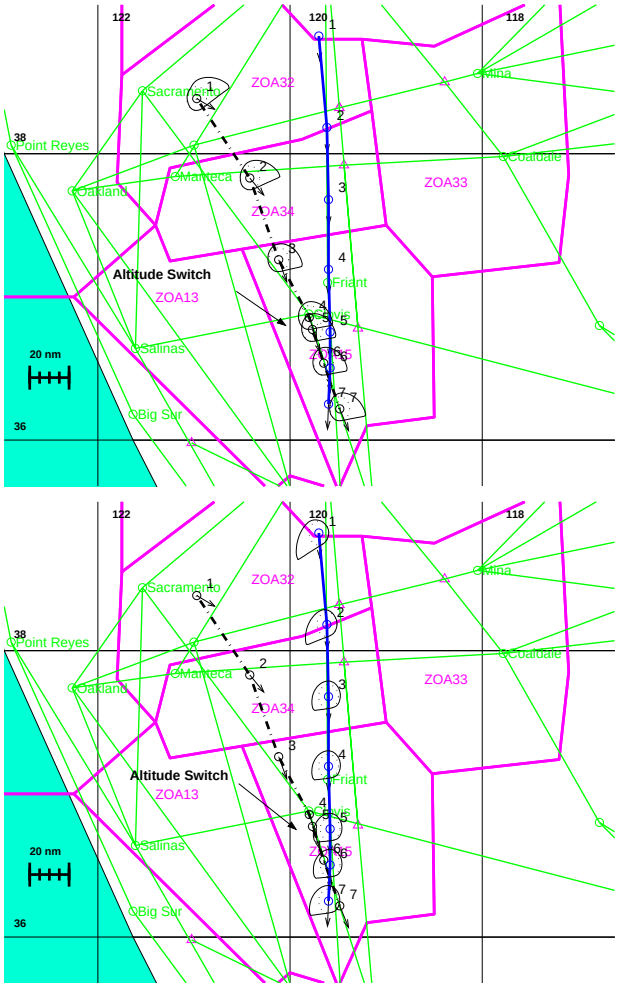

Fig. 10 Aircraft 1 (solid), en route from Reno/Tahoe Airport to Los Angeles Airport; aircraft 2 (dash-dotted), en route from Sacramento Airport to San Diego Airport; both aircraft are cruising at 33,000ft when aircraft 1 is commanded to climb to 37,000ft by the ATC, at the position labeled by 4. At position 5, both aircraft are in the $\mathcal{G}(\tau)$ of the other aircraft.

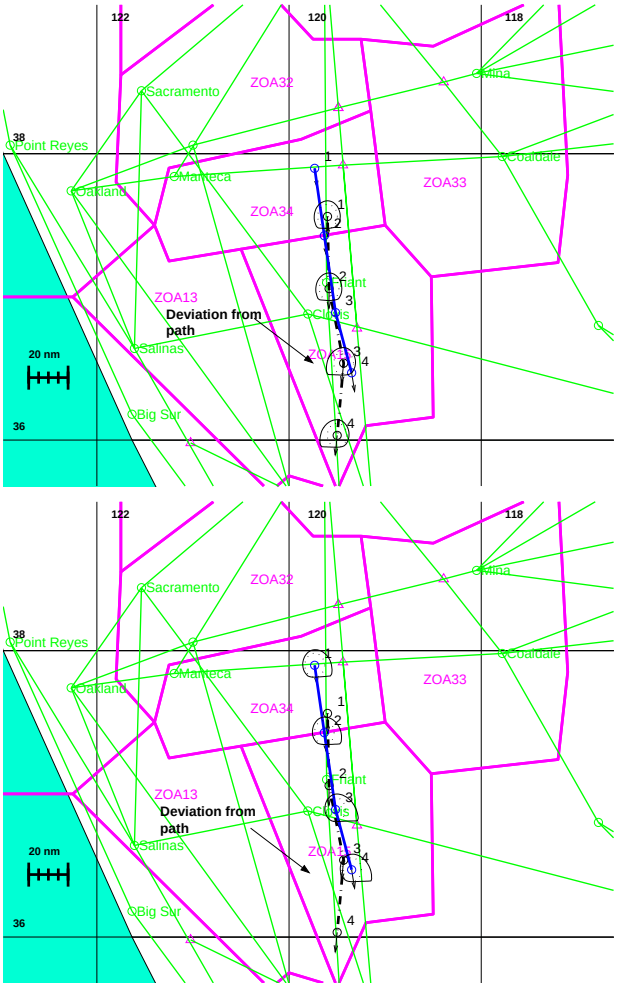

Fig. 11 Aircraft 1 (solid) and aircraft 2 (dash-dotted) are both en route from Seattle-Tacoma Airport to Los Angeles Airport, cruising at 33,000ft. Keeping aircraft on their path (Jetway 7) would lead to LOS, predicted by our algorithm, using interpolation between 3 and 4. The maneuver chosen by ATC (aircraft 2 deviates from its path, and joins again) is not obvious from the data, but avoids LOS.

Our method provides LOS prediction for aircraft pairs, and the corresponding horizontal maneuvers to apply to one of the aircraft in order to avoid the LOS. It is not always realistic however to apply the optimal heading changes provided by ω_a^* directly to a real aircraft: even though it has already been implemented in practice for UAVs,[26] the technical difficulties for communicating ω_a^* from the ground ATC to an aircraft are still significant, and enhanced by the ill-behaved nature of ω_a^*.[20] However, our goal is not an implementation of the controller provided by our method, but the use of the reachable set as a safety set for LOS alert; the maneuver assignment is left to ATC.

In order to assess the value of our metric, we run it on an ETMS data sample, and rate its success in the following way. We classify the scenarios encountered by the method in four categories. Each pair of aircraft treated can be in one of the following cases:

- *Detected conflicts.* A conflict is detected when the method presented in the previous section assesses a threat, and the ETMS data unambiguously shows an actual conflict resolution by the ATC. We consider the following conflict resolution protocols unambiguous: (1) altitude change prior to LOS, (2) significant heading change before LOS. Case (1) is easily detected from the altitude tag in the ETMS data. Case (2) is harder and requires comparison with filed flight plan or current heading.

- *Conflict detected with interpolation.* Because of the sampling rate of ETMS data (one update every 3 min.), our method might miss occurrences of $(x_r, y_r, \psi_r) \in \mathcal{G}(\tau)$. Interpolation of the flight path between the sample points might however provide a *detected conflict* from the previous category. Since this was not observed directly from the data, but from an interpolation, we call this

Floor	D.C.	D.C.w.I	F.A.	N.A	T.C.P.
250	3	2	0	205	210
270	1	1	0	256	258
290	2	2	0	288	292
310	0	1	0	186	187
330	7	2	1	198	208
350	6	6	1	361	374
370	0	0	0	61	61
tot.	19	14	2	1555	1590

Table 3 Sample study for one hour of ETMS data for flight levels ranging from 25,000ft to 37,000ft. All aircraft pairs are classified in one of the four categories: detected conflicts (D.C), detected conflicts with interpolation (D.C.w.I), false alarms (F.A.), not applicable (N.A). The sum is shown in the last column as Total Counted Pairs (T.C.P.).

"conflict detected with interpolation". An implementation in ATC systems would not suffer from such problems because the update rate would be much faster.

- *False alarms.* When our method detects a threat, but no reaction from the ATC was observed in the ETMS data, we call the situation a false alarm. It is due to the fact that our method is conservative: one of the aircraft did not do the worst possible action ω_b^* which is accounted for in our algorithm.
- *Not applicable.* Any pair of aircraft for which no threat is detected. Most of the aircraft pairs will fall into this category, because they are very far from each other.

We realized a sample study for one hour of traffic for all altitudes between floors 250 and 370 (25,000 ft and 37,000 ft respectively) for the complete Oakland ARTCC, and classified all aircraft pairs using the categories above. The result is presented in Table 3. The number of "not applicable" pairs is obviously greater than all other numbers; it illustrates the fact that the number of conflicting pairs of aircraft is small in comparison with the total number of pairs of airborne aircraft. We see from Table 3 that our method only gives two false alarms out of 33 cases, which is satisfying. Figures 7 to 11 illustrate some of the detected conflicts and conflicts detected with interpolation accounted in Table 3. We also observe that most of the cases we found in this study were conflict avoidance scenarios resolved by altitude changes. Very often, these altitude changes coincide with descent of aircraft into destination airports, as in Figures 7, 8 and 9.

To further justify the usefulness of the method proposed, we also need to show that our results do not saturate the airspace display with unsafe sets, i.e. that the cumulative size of the unsafe areas does not cover too large a portion of the airspace. We determine the total airspace area which our method computes as unsafe, at a given time (i.e. cumulating all aircraft pairs) and plot its value divided by the area of just the protected zones $\mathcal{G}_0$, over a 24 hour period. The result is shown in Figure 12 for two flight altitudes. We see that the results never exceed a factor of 4: the total unsafe area marked by our algorithm is never bigger than 4 times the area enclosed by the safety disks of all aircraft. This confirms our belief that this method is applicable to a real ATC system, in that it does not produce too many false alarms or saturate the display with large unsafe sets.

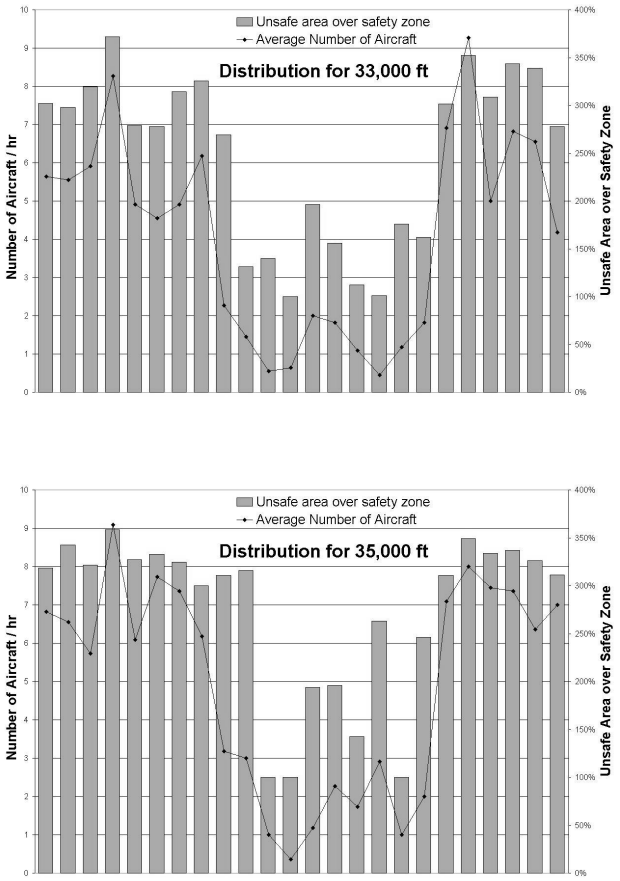

Fig. 12 Average number of aircraft per hour at a given altitude, over a 24 hours period. Ratio of the area labeled potentially unsafe $\mathcal{G}(\tau)$ over the sum of the protected areas ($\mathcal{G}_0$). This ratio does not exceed 4, which confirms that our method provides results which are compatible with current ATC displays (there will be no saturation of the airspace display with unsafe sets).

Conclusion

We used the differential game framework to provide a metric for LOS threat detection. Our formulation posed a LOS threat problem as a two player game, and computes safe sets given different assumed allowed actions of the two players. We applied recently developed numerical techniques to traditionally analytically solved framework of differential games. True ATC cases were cast in this framework and investigated. We used ETMS data available for the Oakland ARTCC as a testbed for our work, and showed with this data that our results concord with human observed behavior at

the ARTCC. Therefore, we are confident that the LOS threat metric defined by our method is appropriate for high altitude traffic and will be a useful tool for aided decision making.

A key advantage of this technique over previously developed collision detection methods is that it does not assume anything about the type of blunder that may occur — rather it computes worst case blunders from the aircraft kinematic configuration. As can be seen by the results in this paper, these blunders are often not intuitive, yet entirely possible.

We also think that this method could be applied to maneuver conformance monitoring for conflict avoidance[25] for example: the main output of this method is a set of parameters inside of which we can guarantee that some goal has been achieved; for the present study, it was separation, but it could as well be maneuvers. So far, the limitation of computational power allows us only to consider kinematics, but as computational power increases, it might be possible to also include accelerations in the present computations. Notice also that this method is relevant for short term conflicts (τ on the order of 3min). For converging traffic[1, 2, 9, 11, 13] in which aircraft are lined up in the last 100nm or 200nm of their flight (next to the destination airports), where the conflicts are often resolved by speed changes or vector for spacing, other methods should be used.[2, 3, 9] Even though loss of separation might happen in merging traffic[2, 3, 9] as well, we feel that appropriate techniques to employ in order to prevent them from happening are optimization techniques rather than differential game theory. These problems are also the focus of ongoing research,[3] and investigates efficient methods to maintain conflict free structures in the flow while regulating it, and it is closely linked with airspace capacity constraints issues.[2–4, 9, 17]

Acknowledgments

We are grateful to Shon Grabbe (NASA Ames) for his help on FACET and ETMS data, and to George Meyer (NASA Ames) for fruitful discussions and suggestions about the reachability problem.

References

[1] A. M. Bayen, P. Grieder, and C. J. Tomlin. A control theoretic predictive model for sector-based air traffic flow. In *Proceedings of the AIAA Conference on Guidance, Navigation and Control*, Monterey, CA, August 2002.

[2] A. M. Bayen, P. Grieder, C. J. Tomlin, and G. Meyer. Lagrangian delay predictive models for sector-based air traffic flow. Submitted Dec. 2002 to the *AIAA Journal on Guidance, Dynamics and Control*.

[3] A. M. Bayen and C. J. Tomlin. Real-time discrete control law synthesis for hybrid systems using MILP: applications to congested airspaces. To appear in the *Proceedings of the American Control Conference*, Denver, CO, June 2003.

[4] D. Bertsimas and S. Stock Patterson. The air traffic flow management problem with enroute capacities. *Operations Research*, 46:406–422, 1998.

[5] K. Bilimoria, B. Sridhar, G. Chatterji, K. Seth, and S.Graabe. FACET: Future ATM concepts evaluation tool. In *Proceedings of the 3rd USA/Europe Air Traffic Management R&D Seminar*, Naples, Italy, June 2001.

[6] P. Cardaliaguet, M. Quincampoix, and P. Saint-Pierre. Set-valued numerical analysis for optimal control and differential games. In M. Bardi, T.E.S. Raghavan, and T. Parthasarathy, editors, *Stochastic and Differential Games: Theory and Numerical Methods*, Annals of the International Society of Dynamic Games. Birkhuser, 1999.

[7] M. G. Crandall, L. C. Evans, and P.-L. Lions. Some properties of viscosity solutions of Hamilton-Jacobi equations. *Transactions of the American Mathematical Society*, 282(2):487–502, 1984.

[8] M. G. Crandall and P.-L. Lions. Viscosity solutions of Hamilton-Jacobi equations. *Transactions of the American Mathematical Society*, 277(1):1–42, 1983.

[9] D. Dugail, E. Feron, and K. Bilimoria. Conflict-free conformance to en-route flow-rate constraints. In *Proceedings of the AIAA Conference on Guidance, Navigation and Control*, Monterey, CA, August 2002.

[10] R. Y. Gazit. *Aircraft Surveillance and Collision Avoidance using GPS*. PhD thesis, Department of Aeronautics and Astronautics, Stanford University, 1996.

[11] J. Histon and R. J. Hansman. The impact of structure on cognitive complexity in air traffic control. Technical Report ICAT-2002-4, MIT, June 2002.

[12] R. Isaacs. *Differential Games*. Dover (John Wiley), 1999 (1965).

[13] D. Jackson and J. Chapin. Redesigning air traffic control: An exercise in software design. *IEEE Software*, 17(3):63–70, 2000.

[14] J.W. Jackson and S.M. Green. Control applications and challenges in air traffic management. In *Proceedings of the 1998 American Control Conference*, Philadelphia, PA, 1998.

[15] S. Kahne and I. Frolow. Air traffic management: Evolution with technology. *IEEE Control Systems Magazine*, 16(4):12–21, 1996.

[16] J. Kuchar and L. Yang. A review of conflict detection and resolution modeling methods. *IEEE Transactions on Intelligent Transportation Systems*, 1(4):179–189, 2000.

[17] P. K. Menon, G. Sweriduk, and K. Bilimoria. A new approach for modeling, analysis and control of air traffic flow. In *Proceedings of the AIAA Conference on Guidance, Navigation and Control*, Monterey, CA, August 2002.

[18] A. W. Merz. The game of two identical cars. *Journal of Optimization Theory and Applications*, 9(5):324–343, 1972.

[19] I. Mitchell. *Application of Level Set Methods to Control and Reachability Problems in Continuous and Hybrid Systems*. PhD thesis, Scientific Computing and Computational Mathematics, Stanford University, 2002.

[20] I. Mitchell, A. M. Bayen, and C. J. Tomlin. Computing reachable sets for continuous dynamic games using level set methods. Submitted January 2002 to *IEEE Transactions on Automatic Control*.

[21] M. S. Nolan. *Fundamentals of Air Traffic Control, 3rd Edition*. Brooks/Cole Publishing, Pacific Grove, CA, 1999.

[22] M. Pachter and T Miloh. The geometric approach to the construction of the barrier surface in differential games. *Computers and Mathematics with Applications*, 13(1-3):47–67, 1987.

[23] T. S. Perry. In search of the future of air traffic control. *IEEE Spectrum*, 34(8):18–35, 1997.

[24] N. Pujet and E. Feron. Flight plan optimization in flexible air traffic environments. In *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, San Diego, CA, August 1996.

[25] T.G. Reynolds and R.J. Hansman. Conformance monitoring approaches in current and future air traffic environments. In *Proceedings of the 21st Digital Avionics Systems Conference*, Irvine, CA, Oct. 2002.

[26] R. S. Teo, J. S. Jang, and C. J. Tomlin. Tests performed on Dragonfly Aircraft, NASA Ames, September 2002.

[27] C. J. Tomlin, J. Lygeros, and S. Sastry. A game theoretic approach to controller design for hybrid systems. *Proceedings of the IEEE*, 88(7):949–970, 2000.