A Probabilistic Approach to Aircraft Conflict Detection

M. Prandini, J. Hu, J. Lygeros, and S. Sastry

Abstract

In this paper, we concentrate primarily on the detection component of conflict detection and resolution schemes operating at the mid-range and short range level of the air traffic management process.
Probabilistic models for mid-term and short term prediction are introduced. Based on the mid-term prediction model, the maximum instantaneous probability of conflict is proposed as criticality measure of a two aircraft encounter situation. The computational issues involved in its estimate are solved by resorting to randomized methods which provide also quantitative bounds on the level of approximation introduced. As for short range detection, closed-form analytical expressions - though approximations - for the probability of conflict are obtained by using the short term prediction model. Based on these expressions, an algorithm for decentralized conflict detection and resolution that generalizes potential fields ideas to path planning in a probabilistic dynamic environment is suggested.
The proposed approaches are shown to be promising using Monte Carlo simulations.

Keywords

Air traffic management system automation, conflict detection, probabilistic modeling.

I. Introduction

Despite technological advances such as the availability of powerful on-board computers and advanced positioning and communication systems as, for example, Global Positioning System (GPS) and Automatic Dependence Surveillance-Broadcast (ADS-B), the current Air Traffic Management System (ATMS) is still based on

- *A rigidly structured airspace*, the aircraft being forced to fly along predefined jetways without the possibility of selecting optimal routes and utilizing favorable winds (see Figure 1), with a few exceptions at those altitudes where the air traffic density is low;
- *A centralized, mostly human-operated system architecture*, the Air Traffic Controller (ATC) on the ground being responsible of aircraft separation by issuing adequate trajectory specifications to the pilots.

For a detailed account of the current ATMS practice, see *e.g.* [1] and [2].
The increasing demand for air travel is stressing the current ATMS, and it is likely to cause both *safety* and *performance* degradation. It is believed, however, that by increasing the level of automation, efficiency of ATMS can be improved and tasks of human operators can be simplified, so as to handle the increased demand in air traffic in a more reliable way, even enhancing the level of safety of the current system.
The primary concern of all advanced ATMS is to guarantee safety of air travel. Safety is typically quantified in terms of numbers of conflicts, *i.e.*, situations where two aircraft come closer than a certain distance from one another. The safety distance is encoded by means of a minimum allowed *horizontal separation* and a minimum *vertical separation*. Currently, for en-route airspace the minimum horizontal separation is 5 nautical miles (nmi), while inside the Terminal Radar Approach Control (TRACON) area it is reduced to 3

Research supported by DARPA under grant F33615-98-C-3614, by NASA under grant NAG 2-1039, and by ARO under grant MURI DAAH 04-96-1-0341.

M. Prandini is with the Department of Electronics for Automation, University of Brescia - 25123 Brescia, Italy. E-mail: prandini@ing.unibs.it

J. Hu and S. Sastry are with the Department of Electrical Engineering and Computer Sciences, University of California at Berkeley - Berkeley CA 94720, U.S.A. E-mail: {jianghai, sastry}@robotics.eecs.berkeley.edu

J. Lygeros is with the Engineering Department, University of Cambridge - Cambridge, CB2 1PZ, U.K. E-mail: jl290@eng.cam.ac.uk

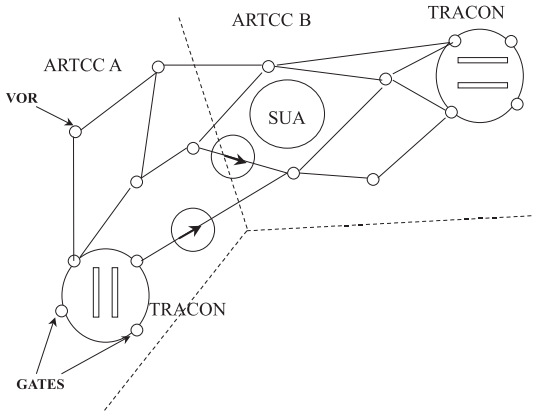


Fig. 1. A sketch of the current ATMS structure organized in Air Route Traffic Control Centers (ARTCCs), Special Use Airspace (SUA) areas, TRACONs facilities, with the aircraft flying along VHF Omni-Directional Range (VOR) jetways and entering the TRACON through gates.

nmi. The minimum vertical separation is 2000 feet (ft) above the altitude of $29,000$ ft (FL290), and 1000 ft below FL290.

To prevent conflicts, ATMS resorts to a two stage process. In the first stage *conflict detection* is performed: the positions of the aircraft in the future are predicted based on their current positions and flight plans, and they are compared so as to detect potential situations of conflict. Once a potential conflict has been detected, the trajectories of the aircraft involved in the conflict are re-planned in the *conflict resolution* stage.
Conflict detection and resolution is actually given consideration at three different levels of the ATM process, which differ in the time horizon over which conflict detection and resolution is performed:

Long range: Some form of conflict prediction and resolution is carried out at the level of the entire National Airspace System (NAS), over a horizon of several hours. It involves composing flight plans and airline schedules (on a daily basis, for example) to ensure that airport and sector capacities are not exceeded ([3], [4], [5]). This is typically accomplished using large scale integer and linear programming techniques.
Mid-range: Conflict prediction and resolution is carried out by ATCs, over horizons of the order of tens of minutes. It involves modifying the pre-planned flight plan on-line to ensure adequate aircraft separation. Semi-automated tools have been developed to assist ATCs with these decisions, for example the Center TRACON Automation System (CTAS, [6]) integrating the algorithm developed in [7], [8], operating within the terminal area and the en-route airspace close to the terminal area, and the conflict probe based on the User Request Evaluation Tool (URET) ([9]), operating within the en-route airspace. The algorithms and techniques of [10], [11] can also be used at this level.
Short range: Conflict prediction and resolution is also carried out on board the aircraft at the Flight Management System (FMS) level, over horizons of seconds to minutes.The Traffic Alert and Collision Avoidance System (TCAS, [12], [13]) currently operating on all commercial aircraft carrying more than 30 passengers is such a prediction/resolution algorithm. The algorithms of [10], [11] also belong to this category.

The reader is referred to [14] for a survey of the different conflict detection and resolution schemes presented in the literature, the main classification criterion being the modeling method used for projecting the aircraft position in the future.
One of the difficulties in predicting the aircraft future position consists in modeling the perturbations influencing its motion. The actual motion of the aircraft is in fact affected by uncertainty, due mainly to wind, but also to errors in tracking, navigation, and control. However, the resultant error with respect to the nominal trajectory can reasonably be modeled as the sum of a large number of independent random perturbations acting in disjoint time intervals, and thus it is expected to be Gaussian. This hypothesis was indeed verified by air traffic data in [15], [16].

In general terms, in the probabilistic approach to conflict prediction, the uncertainty affecting the aircraft motion is taken into account by considering the ensemble of sample paths and computing the probability of projected conflicts, thus avoiding the conservativeness of the worst-case approach (this is the technique used in [11], [17]), but still being robust with respect to uncertainty. The main issues are then building a simple but still realistic probabilistic prediction model, and computing the probability of conflict based on such a model.
In [7], [8], [10], a description of the global effect of the perturbations affecting the aircraft motion over a 20 minutes time horizon is given in probabilistic terms, by distinguishing the resultant uncertainty in the *along-track* and in the *cross-track* directions. Specifically, the tracking errors are typically described as zero mean Gaussian random variables with the variance of the along-track component, $\sigma_a^2(t)$, growing quadratically with time:

$$\sigma_a^2(t) \sim r_a^2 t^2, \tag{1}$$

the variance of the horizontal cross-track component, $\sigma_c^2(t)$, growing quadratically with the traveled distance, $s(t)$, and then saturating at a fixed value:

$$\sigma_c^2(t) \sim r_c^2 s^2(t), \quad sat\{\sigma_c^2(t)\} = \bar{\sigma}_c^2, \tag{2}$$

and, finally, the variance of the vertical cross-track component remaining constant.
In [7], it is argued that this model is fairly accurate for predicting the position of an aircraft over a mid-term horizon, as it reflects the fact that pilots tend to correct cross-track errors in the short term, while dealing with along-track errors in the long term, using small changes in speed.

Based on this probabilistic description of the aircraft motion, two main methods have been introduced for computing the probability of conflict for a two aircraft encounter situation. In [7], a closed-form analytic expression for the probability of conflict for level flight is derived under specific assumptions, *i.e.*, the two aircraft flying along straight lines with constant tracking errors over the whole time horizon. In spite of its simplicity, which makes it very attractive for on-line computation, the exact interpretation of the proposed criticality measure remains to be clarified when such simplifying assumptions are not verified. In [10], [18], Monte Carlo simulation is used to compute the probability of conflict. This approach does not require particular assumptions and allows for many different scenarios, but it is not suitable for on-line computation.

In this paper, we introduce probabilistic prediction models for mid-range and short range conflict detection which are both inspired by the empirically motivated description of the perturbations acting on the aircraft motion described above. For simplicity, we deal with the level flight case. The extension of the proposed approaches to the non-level flight case is straightforward as remarked in Section IV.

As for mid-range conflict detection (Section II), the introduced prediction model is simple and realistic, but it does not allow the derivation of closed-form expression for the probability of conflict. Based on this model, we actually suggest a measure different from the probability of conflict for evaluating the criticality of a two aircraft encounter situation, and give a procedure for computing its estimate and also characterizing the level of accuracy achieved. An algorithm for mid-term conflict detection is then proposed and its performances are compared by Monte Carlo simulations to the ones achieved by the algorithm in [7], which uses an approximation of the probability of conflict as criticality measure. The simulations are based on a more detailed, stochastic ordinary differential equation model for the motion of the aircraft, and they show that the proposed mid-term conflict detection algorithm has in general superior performances to the ones of [7]. On the other hand, the criticality measure in [7] for level flight is easier to be computed since it is in fact given by a closed-form expression. However, no closed-form expression is available for the non-level flight case. On the contrary, we shall show that, if the level of approximation introduced is kept fix, the computational load in computing our criticality measure does not significantly increase in the 3-D case with respect to the 2-D case. Still, further work has to be done to evaluate the performance of our approach in terms of computational load as a function of the level of approximation introduced.
In Section III, we propose a model for short term prediction which is basically an approximation of the mid-term prediction model. The aircraft motion is in fact modeled as a deterministic motion plus a (scaled) Brownian motion perturbation whose variance grows linearly with time, instead of quadratically. According

to this model, the probability of conflict is shown to be the probability that a Brownian motion escapes from a time-varying safe region. Approximate expressions for the probability of conflict are obtained in closed-form for both the finite and infinite horizon cases, thus allowing for fast computations. Based on these expressions, an algorithm is proposed for decentralized conflict resolution. This algorithm is used for evaluating the efficacy of the proposed criticality measure by Monte Carlo simulations.

Developing tools for automated conflict detection and resolution is currently a research topic of great interest to the air traffic control community. This interest is motivated by the Federal Aviation Administration (FAA) initiatives to increase the ATMS flexibility by allowing aircraft to fly more user preferred routes from origin to destination, in the perspective of the realization of the "Free Flight" paradigm ([19]). Possible implications of our research on the current FAA program are discussed in the conclusions.

II. Mid-range conflict detection

A conflict prediction and resolution scheme at the ATC level can be viewed as a feedback control scheme, where the ATC, aircraft, and radar correspond to the *plant* (ATC and radar playing the role of actuators and sensors respectively), and the detection and resolution components correspond to the *controller*. Our goal is to design the two "controller" modules and verify that the closed loop system possesses certain desirable properties. In the present paper, we restrict our attention to specific suggestions for detection and validation, describing only briefly in Section IV our current work on mid-range conflict resolution.

We consider two aircraft flying in the same region of the airspace, each following its individual *flight plan*. The flight plan is assumed to consist of a sequence of *way points* and a sequence of *speeds* for moving between them. Based on this information and on the measurements of the current positions of the aircraft, the models developed in [7], [8] can be modified to provide empirically motivated estimates of the probability distribution for the projected positions of the two aircraft in the near future. One can then define the *instantaneous probability of conflict* PC at a future time as the probability that an aircraft will enter the protected zone around the other one. Conflict detection consists of estimating such a probability and warranting some action when it is 'high'. The design choices that enter into conflict detection are the different ways of weighting the value of PC at various time instants, and the different ways of computing estimates of PC. Here, we declare conflicts based on the maximum value of PC over the prediction horizon. Randomized techniques are used to both estimate PC and compute its maximum. The advantage of randomized techniques is that they tend to be computationally efficient. They also provide analytical bounds on the accuracy of the approximation involved, provided one makes appropriate design choices. The parameters that one needs to set are the prediction horizon (taken here to be 20 minutes as a reasonable look ahead time) and the threshold for declaring a conflict. Systematic guidelines for setting threshold values can be found in [20].

Performance of a conflict prediction/resolution scheme is measured in terms of safety, impact on ATC workload, and efficiency (impact on fuel consumption, deviations from schedule, etc.). Here, we restrict our attention only to the issue of safety. In fact safety is closely coupled to ATC workload, as the controllers may choose to disregard disruptive advisories.
We model the aircraft motion by means of a stochastic differential equation and evaluate the performance of the detection scheme in terms of probability of false alarms and successful alerts by Monte Carlo simulations. The simulation results are also used to tune the threshold of the algorithm, in an attempt to optimize the tradeoff between the probability of successful alerts and the probability of false alarms. At a later stage we hope to be able to provide theoretical bounds on the safety of the proposed design - including the resolution component - and estimate its impact on ATC workload by means of human-in-the-loop simulations.

This section is organized as follows. In Section II-A, we introduce the probabilistic framework in which we develop our detection scheme and perform the validation. In Section II-B, we describe the detection scheme and explain how we deal with the involved computational issues in order to formulate an algorithm for its implementation. Finally, simulation results are reported in Section II-C.

A. Probabilistic Models

In this section, we introduce stochastic models for prediction and validation which are based on the knowledge of the aircraft flight plan and are inspired by the probabilistic description given in [7] and reviewed in the introduction.

We assume that the trajectory specifications the aircraft receives from the ATC are in terms of a sequence of way points $\{P_j\}_{j=0,\dots,n}$, $P_j \in \mathbb{R}^3$ (or $\mathbb{R}^2$), to follow and a sequence of speeds, $\{v_j\}_{j=1,\dots,n}$, $v_j \in \mathbb{R}_+$, between way points. These two sequences constitute the so-called *flight plan*.
At a given time, way points in the past are discarded from the flight plan, therefore at all times the first way point, P_0, encodes the current position of the aircraft. We assume that the flight plan of the aircraft over the time horizon of interest is known a-priori in its entirety, except of course P_0, which is measured by radar, and depends on how well the aircraft is tracking the flight plan. In this regard, in this paper we suppose that the initial position of the aircraft is known precisely, *i.e.*, we ignore errors in the radar measurements. This assumption can in fact be justified by noticing that uncertainty in the initial position will become rapidly dominated by the perturbation to the aircraft motion as time goes on ([10], [7]). A model including radar measurement errors is proposed in [21].

In the following, we shall deal with the level flight case. Hence, $P_j \in \mathbb{R}^2$, $j = 0, \dots, n$. This is for ease of explanation, since the generalization to the three dimensional case is straightforward with the only added difficulty consisting of more complex notation ([22]).

Prediction model. For the purpose of conflict prediction, we assume that each aircraft tries to follow its flight plan moving along the straight line joining successive way points P_{j-1} and P_j with the prescribed speed v_j. Then, the *nominal arrival times* $\{T_j\}_{j=0,\dots,n}$ of the aircraft at the ordered way points sequence $\{P_j\}_{j=0,\dots,n}$ can be recursively computed by:

$$T_j = \frac{\|P_j - P_{j-1}\|}{v_j} + T_{j-1}, \tag{3}$$

for $j = 1, \dots, n$, initialized with $T_0 = 0$ (recall that P_0 encodes the current position of the aircraft).
Based on the nominal arrival times, the *nominal distance traveled* by the aircraft, $s(t) \in \mathbb{R}_+$, and its *nominal position*, $p(t) \in \mathbb{R}^2$, at time $t \in (T_{j-1}, T_j]$, are respectively given by $s(t) = v_j(t - T_{j-1}) + s(T_{j-1})$ and $p(t) = p(T_{j-1}) + v_j(t - T_{j-1})\frac{P_j - p(T_{j-1})}{\|P_j - p(T_{j-1})\|}$, $j = 1, \dots, n$, initialized with $s(T_0) = 0$ and $p(T_0) = P_0$.

The projection of the aircraft position in the future is affected by uncertainty. Following [8], [10], we assume that the aircraft predicted position can be modeled as a multivariate Gaussian random variable.
Precisely, let $x(t) := (x_1(t), x_2(t))^T \in \mathbb{R}^2$ denote the aircraft *predicted position* at time t. We then assume that: $x(t) \sim \mathcal{N}(m(t), V(t))$, where we set the mean:

$$m(t) = p(t). \tag{4}$$

As for the covariance matrix, $V(t)$, we assume that the variance of the predicted position increases with time in every direction, reflecting the fact that the uncertainty on the aircraft position tends to increase the further we try to make a prediction into the future. We actually use the description in [7], [8], where the variances of the predicted position in the along-track and cross-track directions, $\sigma_a^2(t)$ and $\sigma_c^2(t)$, grow according to equation (1) and (2), respectively, and the along-track and cross-track error components are assumed to be independent.
Denote with ϑ_j the heading of the aircraft at time $t \in [T_{j-1}, T_j)$, *i.e.*, the angle that vector $P_j - P_{j-1}$ makes with the x_1 axis of the global framework in which the P_j's are given. Then, the covariance matrix $V(t)$ for $t \in [T_{j-1}, T_j)$ has the following expression:

$$V(t) = R(\vartheta_j)\bar{V}(t)R(\vartheta_j)^T \tag{5}$$

where $\bar{V}(t) := \mathrm{diag}(\sigma_a^2(t), \sigma_c^2(t))$ is the covariance matrix in the body coordinate frame and $R(\vartheta)$ is the

rotation matrix associated with the angle ϑ, *i.e.*,

$$R(\vartheta) = \begin{pmatrix} \cos\vartheta & -\sin\vartheta \\ \sin\vartheta & \cos\vartheta \end{pmatrix}. \tag{6}$$

In Figure 2 the prediction model for the aircraft motion is drawn with the ellipses representing equi-probability curves of the probability distribution $\mathcal{N}(m(t), V(t))$ along the aircraft nominal trajectory. Note that $V(t)$ "jumps" instantaneously at the way points; more specifically, it is discontinuous as a function of time, with the discontinuity points being at the time $\{T_j\}_{j=1,\ldots,n}$ of heading changes. Considering instantaneous turns introduces, in general, inaccuracy in the evaluation of the probability of conflict in the region near the way points. On the other hand, these discontinuities can be avoided by using a more realistic nominal trajectory with smooth turns, such as the one computed by the trajectory synthesizer implemented in CTAS.

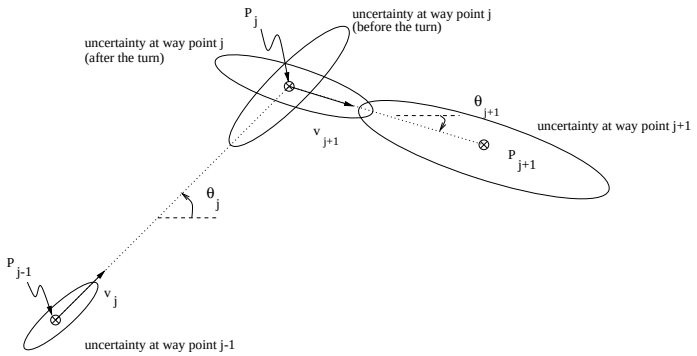


Fig. 2. Mid-range prediction model for the aircraft motion.

Validation model. To validate a mid-range conflict detection algorithm using statistics collected through Monte-Carlo simulation, we need stochastic models to generate trajectories for the aircraft motion.
The model proposed for predicting the position of an aircraft in the future consists of the probability distributions of the random variables representing the aircraft projected positions over a 20 minutes horizon. On the one hand, this model is simple and allows fast computations, which makes it ideal for on-line conflict detection. On the other hand, it does have certain built-in inaccuracies, which may make it inappropriate for generating trajectories for simulation or validation. In particular, the prediction model

1. does not describe the possible correlation among the random variables describing the projected positions of the aircraft at different time instants,
2. ignores the possible correlation between the cross-track and along-track errors,
3. uses simple nominal position assumptions and supposes instantaneous turns.

For validation, we would like to have a model to generate trajectories that alleviates as many of the drawbacks of the prediction model as possible. This validation model will invariably be more complicated than the prediction model, and therefore more difficult to compute with; this is not a major concern however, as it will only be used off-line.
In this paper, we introduce a stochastic ordinary differential equation model for the aircraft motion that succeeds in eliminating one of the sources of inaccuracy: it provides a formal way of correlating the positions of a single aircraft at different points in time, while maintaining at each time instant the statistics of the prediction model.

Consider an aircraft moving in $\mathbb{R}^2$, and let $x = (x_1, x_2)^T$ denote its position with respect to a global inertial coordinate frame. Assume its velocity has magnitude v and makes an angle ϑ with respect to the x_1 axis. Consider now a body coordinate frame $\chi = (\chi_1, \chi_2)$ with χ_1 aligned with the aircraft velocity, χ_2 perpendicular to it and lying on the plane on which the aircraft flies. The two frames are related through a coordinate transformation:

$$x = R(\vartheta)\chi + p, \tag{7}$$

where $R(\vartheta)$ is the rotation matrix of equation (6) and p denotes the position of the origin of the body frame with respect to the inertial frame.
In light of what we discussed so far, equation(7) is to be reinterpreted as follows:

- the origin p of the body coordinate frame represents the nominal position of the aircraft,
- χ_1 and χ_2 represent the variation of the aircraft position with respect to the nominal, in the along-track and cross-track directions respectively,
- x represents the actual position of the aircraft once one takes into account the uncertainty along the along-track and cross-track directions.

We can now derive a kinematic model for the motion of the aircraft in these coordinate frames. We assume that the perturbation affecting the aircraft motion is generated through the stochastic differential equation:

$$\dot{\chi} = A\chi + \eta \tag{8}$$

where $\chi(0) \sim \mathcal{N}(0, V_\chi(0))$ and $\eta := (\eta_1, \eta_2)^T$ is a white Gaussian noise process independent of $\chi(0)$, with zero mean and covariance $V_\eta(t)$. We shall, in fact, show in the paragraph below on the tuning of the validation model parameters that by appropriately selecting matrix A and the covariance matrices $V_\chi(0)$ and $V_\eta(t)$, we are able to obtain uncorrelated Gaussian along-track and cross-track errors whose variance respectively keeps growing with time and saturates according to equations (1) and (2).

By differentiating equation (7), and using equation (8), we get a nonlinear kinematic model for the aircraft motion which, by modeling turns as discrete events occurring at the way points, reduces to the piecewise linear stochastic differential equation:

$$\begin{cases} \dot{x} = A(\theta_j)(x-p) + B(\theta_j)v_j + C(\theta_j)\eta \\ \dot{p} = D(\theta_j)v_j \end{cases}, \quad t \in [T_{j-1}, T_j),\ j \geq 1 \tag{9}$$

with initial conditions $x(0) = p(0) = P_0$, and switching times $\{T_j\}$ given by equation (3), where we set $A(\vartheta) := R(\vartheta)AR(\vartheta)^T$, $B(\vartheta) = D(\vartheta) := R(\vartheta)(1,0)^T$, $C(\vartheta) := R(\vartheta)$.

The validation model can then be thought of as a *hybrid stochastic dynamical system*, with discrete events *Turn* occurring at the way points, the effect of *Turn* being the update of ϑ and v. Based on this interpretation, we are currently studying a method for formally evaluating the safety properties of detection algorithms formulated in this probabilistic framework.

Tuning of the validation model parameters. For our stochastic validation model to resemble the statistics derived from air traffic data ([15], [16]), we need to appropriately tune its parameters, that is the covariance matrices $V_\chi(0)$ and $V_\eta(t)$, and the matrix A of the tracking error model (8). The tuning is based on the following proposition, whose proof is omitted since it easily follows from standard results for the solution of stochastic differential equations (see *e.g.* [23]).

Proposition 1 *Consider the stochastic differential equation* $\dot{\chi}(t) = A\chi(t) + \eta(t)$, $t \in [0,T]$, *where* $\eta(\cdot)$ *is a white Gaussian noise process with* $\eta(t) \sim \mathcal{N}(0, diag(V_{\eta_1}(t), V_{\eta_2}))$, $t \in [0,T]$, *independent of* $\chi(0) \sim \mathcal{N}(0, diag(V_{\chi_1}(0), V_{\chi_2}(0)))$. *Set*

$$V_{\chi_1}(0) = V_{\chi_2}(0) = 0, \quad V_{\eta_1}(t) = 2r_a^2 t, \quad \frac{V_{\eta_2}}{2a} = \bar{\sigma}_c^2, \ \text{and } A = \begin{pmatrix} 0 & 0 \\ 0 & -a \end{pmatrix}, \ \text{where } a = \frac{r_c}{\bar{\sigma}_c} v_1. \tag{10}$$

Then, $\chi_1(\cdot)$ *and* $\chi_2(\cdot)$ *are independent Gaussian processes with zero mean and variance*

$$var[\chi_1(t)] = r_a^2 t^2; \tag{11}$$

$$var[\chi_2(t)] = \bar{\sigma}_c^2 (1 - \exp(-2\frac{r_c}{\bar{\sigma}_c} v_1 t)). \tag{12}$$

By tuning the parameters in equation (8) according to equation (10), we then have that $\chi(\cdot)$ resembles the empirically observed characteristics of the along-track and cross-track errors. As a matter of fact, $var[\chi_1(t)]$

in (11) has the same growth rate as the along-track variance $\sigma_a^2(t)$ in (1), $var[\chi_2(t)]$ in (12) is monotonically increasing to the saturation value $\bar{\sigma}_c^2$ as the cross-track variance $\sigma_c^2(t)$ in (2). Moreover, the time constant of the decaying exponential in equation (12) is equal to 1/2 of the time $\bar{t} = \frac{\bar{\sigma}_c}{r_c v_1}$ required for $\sigma_c^2(t)$ to reach the saturation value $\bar{\sigma}_c^2$ in the case when the aircraft trajectory is a straight line traveled with speed v_1. For $r_c = 1/57$ and $\bar{\sigma}_c = 1$ nmi (values estimated on real traffic data, [15], [16]) and for $v_1 = 500$ nmi/h (typical cruising speed) such a time is about 6 minutes. Therefore, by setting $a = \frac{r_c}{\bar{\sigma}_c} v_1$, we get that $var[\chi_2(t)]$ reaches 86% of the saturation value in about 6 minutes.

B. Conflict detection algorithm

Consider N_a aircraft sharing a region of the airspace and flying at the same altitude. A situation of conflict is encountered when an aircraft enters the *protected zone* around another aircraft, which for level flight in the en-route space is a circle of radius $\rho = 5$ nmi:

$$\mathcal{C} = \left\{ d \in \mathbb{R}^2 : \|d\| \leq \rho \right\}. \tag{13}$$

We assume that each aircraft, say i, has assigned its own flight plan $\{P_j^i\}_{j=0,\ldots,n_i}$, $P_j^i \in \mathbb{R}^2$ and $\{v_j^i\}_{j=1,\ldots,n_i}$, $v_j^i \in \mathbb{R}_+$. Then, the overall encounter situation can be described through the *configuration*, γ, of the N_a aircraft system, which consists of the flight plans of all aircraft, *i.e.*, $\gamma = \left\{ \{P_j^i\}_{j=0,\ldots,n_i}, \{v_j^i\}_{j=1,\ldots,n_i} \right\}_{i=1,\ldots,N_a}$.

A conflict detection scheme consists of extracting some measure, $C(\gamma)$, of how critical the current configuration, γ, is, comparing this measure to a threshold, $\bar{C}$, and declaring a conflict if the threshold is exceeded. The process should be repeated every time the configuration, γ, changes, that is every time a new measurement comes in from the radar, every time the ATC changes a flight plan, etc. In pseudo-code, this scheme can be encoded by:

Algorithm 1 (General Conflict Detection)
when γ changes **do**
begin
Compute $C(\gamma)$
if $C(\gamma) > \bar{C}$ declare a conflict
end

The rest of this section concerns primarily with the implementation of this piece of pseudo-code. This involves deciding on a measure of criticality, $C(\cdot)$, developing an efficient algorithm for computing it for a given configuration, γ, and setting the threshold $\bar{C}$.
We now introduce a probabilistic measure of criticality C for a two aircraft encounter situation, based on the prediction model presented in Section II-A. Then, we present an algorithm for computing it making use of stochastic estimation algorithms. As for the threshold $\bar{C}$, it can be set either by heuristic arguments such that used in [8] or by referring to the procedure suggested in [20]. We postponed the discussion on the computation of the threshold to Section II-C, where the performance of our detection algorithm is validated by Monte Carlo simulations using the ODE model of Section II-A.

Measure of criticality. Consider two aircraft, A and B, flying at the same altitude, and let x^A and x^B denote their positions in the inertial reference frame. Given the probability density function $p_{d_t}(y)$ for the separation $d(t) := x^A(t) - x^B(t)$ of the two aircraft at time t, the *instantaneous probability of conflict* at time t is given by:

$$PC(t) = \int_{y \in \mathcal{C}} p_{d_t}(y) dy. \tag{14}$$

In the algorithm proposed here, conflict detection is based on the maximum value of the probability of conflict $PC(t)$ over a finite horizon of length T. The time horizon T is chosen to be equal to 20 minutes. This is because the computation of $PC(\cdot)$ is based on the prediction model introduced in Section II-A, which is in fact valid over a 20 minutes horizon. Thus, we consider Algorithm 1 with

$$C(\gamma) := \max_{t \in [0,T]} PC(t).$$

In principle, this algorithm can be used irrespectively of which is the probability distribution of the aircraft positions, the main problem being integrating the resultant distribution for their separation $d(t)$ over the protected zone $\mathcal{C}$. Based on the prediction model in Section II-A, however, we consider the case when the two aircraft positions $x^A(t)$ and $x^B(t)$ are Gaussian random variables, $x^i(t) \sim \mathcal{N}(m^i(t), V^i(t))$, $i = A, B$, with $m^i(t)$ and $V^i(t)$ given by (4) and (5). Moreover, we make the assumption that they are uncorrelated. Then, the separation $d(t)$ between the predicted positions of two aircraft at time t is a Gaussian random variable, $d(t) \sim \mathcal{N}(\mu(t), Q(t))$, with mean $\mu(t) := m^A(t) - m^B(t)$ and covariance matrix $Q(t) := V^A(t) + V^B(t)$.

It is important to note that the accuracy of conflict prediction is limited by the uncorrelation assumption. The tracking error is in fact primarily due to wind, which may correlate the positions of the two aircraft, especially near the conflict point where the aircraft are close to each other. Still, the independence assumption between the tracking noises of the two aircraft, though unrealistic, is commonly used ([8], [10]). Modeling the tracking error correlation properties is not dealt with in this paper, though it obviously needs to be addressed in future work.

A randomized conflict detection algorithm. The major obstacle in the implementation of the proposed conflict detection scheme is the computation of $C(\gamma)$. The problem is that one cannot derive an analytical expression for $PC(\cdot)$, and even the evaluation of $PC(t)$ for a given t is time consuming. This obstacle turns out to be overwhelming in the on-line application of the detection scheme, where computation is subject to time constraints. One then has to give up the ambitious objective of finding an exact solution and look for an approximate one.
In this paper, we introduce an algorithm for computing an approximate solution to the optimization problem and we also set up a methodology to provide a quantitative estimate of the level of approximation introduced. The obtained results rests on the theory of empirical processes, whose main aim is to study how to estimate unknown quantities through experimentation [24].

Suppose for the time being that given a $t \in [0, T]$, we are able to compute $PC(t)$ with no error. Then, we can resort to the following algorithm in order to compute $C(\gamma)$.

Algorithm 2 (Randomized Estimate of $C(\gamma)$)

Initialization: Choose an integer N and set $C'(\gamma) = 0$
for $i = 1, \ldots, N$
begin
  Extract at random $t_i \in [0, T]$ according to the uniform probability distribution
  if $C'(\gamma) < PC(t_i)$ **then** $C'(\gamma) = PC(t_i)$
end

Clearly, since we are testing just N values of $PC(t)$ for determining the maximum over the time interval $[0, T]$, C' will be an approximation of C. In addition, the quality of the approximation is random due to the stochastic selection of the t_i's. Nevertheless, a quantitative statement can be proved showing that C' is a good approximation in a probabilistic sense based on the following theorem.

Theorem 1 (Randomized optimization) *Let $(Z, \mathcal{F}, \mathcal{P})$ be a probability space, and let $f : Z \to \mathbb{R}$ be a measurable function. Extract N independent samples $z_1, z_2, \ldots, z_N$ from Z according to the probability distribution $\mathcal{P}$ and define $\bar{f} := \max_{z \in \{z_i\}_{i=1}^N} f(z)$. Fix an arbitrary real number $\beta \in (0, 1)$. Then,*

$$\mathcal{P}\{z \in Z : f(z) > \bar{f}\} \leq \beta, \tag{15}$$

with probability greater than $1 - (1 - \beta)^N$.

Proof *See Lemma 11.1 in [24].* □

In the previous statement, $\bar{f}$ is a random variable defined on the product space Z^N with the product probability measure $\mathcal{P}^N := \mathcal{P} \times \ldots \times \mathcal{P}$ hosting the random extraction $(z_1, z_2, \ldots, z_N)$. Thus, $\mathcal{P}\{z \in Z : f(z) > \bar{f}\} \leq \beta$ may or may not be satisfied according to such a random extraction and, therefore, $\mathcal{P}\{z \in Z : f(z) > \bar{f}\} \leq \beta$ defines a probabilistic event in the space Z^N. Theorem 1 says that probability

$\mathcal{P}^N$ of this event is greater than $1-(1-\beta)^N$.
A possible interpretation of equation (15) is the following. Suppose that an opponent would like to determine a $\bar{z}$ such that $f(\bar{z}) > \bar{f}$ and she uses the same probabilistic strategy as in the randomized algorithm: she randomly selects a $\bar{z}$ in Z according to $\mathcal{P}$. Then, her probability of "beating" $\bar{f}$ is not greater than β. In Theorem 1, parameter β is arbitrary and, therefore, the probability of success left to the opponent can be reduced at will. On the other hand, (15) is not a deterministic statement and it only holds true with probability $\mathcal{P}^N$ at least equal to $1-(1-\beta)^N$. As β approaches 0, $1-(1-\beta)^N$ tends to 0 and so the statement becomes evanescent. In addition, we observe that statement (15) only quantifies the probability that one can improve result $\bar{f}$ by randomly selecting a new parameter $\bar{z}$. On the other hand, Theorem 1 says nothing on how large such an improvement can be.

In our case, we are interested in maximizing the probability of conflict $PC(t)$ over the time interval $[0,T]$. Note that the function $PC : [0,T] \to [0,1]$ is not smooth since the Gaussian probability density function p_{d_t} appearing in the expression (14) of $PC(t)$ loses smoothness in the set of instants $\{t \in \{T_j^A\}_{j=0,\ldots,n_1} \cup \{T_j^B\}_{j=0,\ldots,n_2} \cup \{T_*^A, T_*^B\}$ with $t \leq T\}$, where $\{T_j^i\}_{j=0,\ldots,n_i}$, $i = A, B$, are the nominal times of arrival of aircraft i at its way points sequence, and T_*^i is the time instant when the cross-track error variance of aircraft i saturates. However, this is a finite set. Hence, if Z is taken to be the interval $[0,T]$, $\mathcal{F}$ the Borel σ-algebra on $[0,T]$ and $\mathcal{P}$ the uniform probability distribution, Theorem 1 can be applied with function $f(z) = PC(z)$. Denote with $\lceil c \rceil$ the smallest integer greater than c. Then, we have

Assertion 1 *Fix $\delta \in (0,1)$, $\beta \in (0,1)$, and set $N := \lceil \frac{\ln(\delta)}{\ln(1-\beta)} \rceil$. Then, if the random extractions in Algorithm 2 are independent, C' given by Algorithm 2 is an approximation of C in the sense that there exists an exceptional set $S \subset [0,T]$ of Lebesgue measure at most βT such that $\sup_{[0,T]\setminus S} PC(t) \leq C' \leq \sup_{[0,T]} PC(t)$ with probability at least $1-\delta$.*

Next, we introduce a method to compute a uniformly good approximation of $PC(t)$ over every finite set of time instants $\{t_1, t_2, \ldots, t_N\}$. This method will allow us not only to circumvent the difficulty of integrating the probability density function $p_{d_t}(\cdot)$ over the protected zone $\mathcal{C}$, but also to estimate the maximum of $PC(t)$ over $[0,T]$ when combined with the randomized optimization.
Recall that $PC(t)$ is the probability that a random variable with time-dependent probability distribution $\mathcal{N}(\mu(t), Q(t))$ takes values in the protected zone $\mathcal{C}$ given in (13). By an appropriate change of coordinates, however, we can use a different viewpoint seeing $PC(t)$ as the probability that a random variable with standard normal distribution $\mathcal{N}(0,I)$ takes values in a time-dependent set $\mathcal{C}_t$. The required change of coordinates is given in the following proposition, whose proof is omitted since it is easy to be verified. Note that a similar procedure is followed in [7].

Proposition 2 *Set $w := L(t)^{-1}[d(t) - \mu(t)]$, where $L(t)$ is the Cholesky factorization of the covariance matrix $Q(t)$. Then, w is a standard 2-D Gaussian random variable and $PC(t)$ can be computed as the probability of w to assume values in the set $\mathcal{C}_t := \{w \in \mathbb{R}^2 : L(t)w + \mu(t) \in \mathcal{C}\}$.*

This suggests the following algorithm for probabilistically estimating $PC(t)$ for a given time instant t.

Algorithm 3 (Randomized Estimate of PC)

Initialization Choose an integer M and set $PC'(t) = 0$
for $j = 1, \ldots, M$ **do**
begin
  Extract at random $w_j \in \mathbb{R}^2$ according to the Gaussian probability density function $\mathcal{N}(0,I)$
  if $w_j \in \mathcal{C}_t$ **then** $PC'(t) = PC'(t) + 1$
end
$PC'(t) = \frac{PC'(t)}{M}$

Again, $PC'(t)$ is a random approximation of $PC(t)$, due to the stochastic selection of the w_j's. Nevertheless, a quantitative statement can be proven showing that it is a good approximation in a probabilistic sense.

Theorem 2 (Estimation of Probability Measures) *Let $(Z, \mathcal{F}, \mathcal{P})$ be a probability space. Consider a finite collection of sets $\mathcal{A} = \{A_1, \ldots, A_N\} \subset F$. Extract M independent samples $z_1, \ldots, z_M$ from Z in accordance with $\mathcal{P}$ and define $\hat{\mathcal{P}}(A_i) := \frac{1}{M} \sum_{j=1}^{M} 1_{z_j \in A_i}$, $i = 1, \ldots, N$, where $1_{z_j \in A_i} = 1$ if $z_j \in A_i$, 0 otherwise. Fix $\epsilon \in (0,1)$. Then,*

$$\mathcal{P}^M\{(z_1, \ldots, z_M) \in Z^M : \sup_{A \in \{A_i\}_{i=1}^N} |\hat{\mathcal{P}}(A) - \mathcal{P}(A)| > \epsilon\} \leq 2N \exp(-2M\epsilon^2). \tag{16}$$

Proof *By Chernoff bound ([24]).* □

Equation (16) means that each finite collection of sets $\mathcal{A}$ has the property of uniform convergence of empirical probabilities (UCEP) since, for each fixed accuracy level ϵ, the probability that the approximation error in estimating the probability measure of a set in $\mathcal{A}$ exceeds ϵ tends to zero uniformly over $\mathcal{A}$, as the number M of samples goes to infinity. It is quite intuitive that the UCEP property is a necessary condition for successfully applying Algorithm 2 using an estimate of the function $PC(\cdot)$ to be optimized instead of the true function. Since by Proposition 2, $PC(t)$ is the measure of the set $\mathcal{C}_t = \{w \in \mathbb{R}^2 : L(t)w + \mu(t) \in \mathcal{C}\}$ according to $\mathcal{N}(0, I)$, then, if Z is taken to be $\mathbb{R}^2$, $\mathcal{F}$ the Borel σ-algebra on $\mathbb{R}^2$ and $\mathcal{P} = \mathcal{N}(0, I)$, Theorem 2 can be applied to get the following result for the estimate PC' in Algorithm 3.

Assertion 2 *Fix $\delta \in (0,1)$, $\epsilon \in (0,1)$, and set $M = \lceil \frac{1}{2\epsilon^2} \ln \frac{2N}{\delta} \rceil$. Then, if the random extractions in Algorithm 3 are independent, $PC'(t)$ is an uniform approximation of $PC(t)$ to accuracy ϵ with confidence $1-\delta$ over every finite set of time instants $\{t_1, \ldots, t_N\}$, i.e., $\sup_{t \in \{t_i\}_{i=1}^N} |PC'(t) - PC(t)| \leq \epsilon$, with probability at least $1 - \delta$.*

One has then to use some optimization technique to compute the maximum of $PC'(t)$ over $[0, T]$. Observe that maximizing even the modified function $PC'(t)$ is not easy, since there is not a closed-form expression for it. This is the reason why we finally resort to the randomized optimization method described before, thus obtaining a fully randomized procedure for the approximate evaluation of $C(\gamma)$.

Algorithm 4 (Randomized Conflict Detection)

Initialization: Fix $\epsilon \in (0,1)$, $\beta \in (0,1)$, $\delta \in (0,1)$. Set $N = \lceil \frac{\ln(\frac{\delta}{2})}{\ln(1-\beta)} \rceil$, $M = \lceil \frac{1}{2\epsilon^2} \ln \frac{4N}{\delta} \rceil$
when γ changes **do**
begin
  Extract at random $w_j \in \mathbb{R}^2$, $j = 1, \ldots, M$, according to the Gaussian density function $\mathcal{N}(0, I)$
  Extract at random $t_i \in [0, T]$, $i = 1, \ldots, N$, according to the uniform density function
  for $i = 1, \ldots, N$ **do**
  begin
    Compute $\mu(t_i)$ and $Q(t_i)$
    Compute the Cholesky decomposition $Q(t_i) = L(t_i)L(t_i)^T$
    Set $PC'(t_i) = 0$
    for $j = 1, \ldots, M$ **do**
    begin
      if $L(t_i)w_j + \mu(t_i) \in \mathcal{C}$ **then** $PC'(t_i) = PC'(t_i) + 1$
    end
    $PC'(t_i) = \frac{PC'(t_i)}{M}$
    if $PC'(t_i) > C$ declare a conflict
  end
end

The algorithm therefore declares a conflict if and only if the estimate of the measure of criticality

$$C'(\gamma) := \max_{t \in \{t_i\}_{i=1}^N} PC'(t),$$

where $PC'(t)$ is computed according to Algorithm 3, exceeds the threshold $\bar{C}$. Under the assumption that all the random extractions are made independently of one another, the following theorem provides an estimate of the accuracy of our approximation. Similar accuracy estimates are given in [25], [26], [27], where randomized methods are applied to robust and adaptive control.

Theorem 3 (Approximate estimation of C) *Given $\epsilon \in (0,1)$, $\beta \in (0,1)$ and $\delta \in (0,1)$, $C'(\gamma)$ is an approximate estimate of $C(\gamma)$ to accuracy 2ϵ and level β with confidence $1-\delta$ in the sense that*

$$\mathcal{P}\{t \in [0\ T] : PC(t) > C'(\gamma) + 2\epsilon\} \leq \beta,$$

with probability greater than $1-\delta$, where $\mathcal{P}$ denotes the uniform probability distribution on $[0,T]$.

Proof *C' is a random variable on the probability space $[0,T]^N \times \mathbb{R}^{2M}$. As t_i's are independent of one another and the same holds for w_j's, and also considering that the extraction of $\{t_i\}_{i=1}^N$ is independent of the extraction of $\{w_j\}_{j=1}^M$, we can conclude that the probability measure in $[0,T]^N \times \mathbb{R}^{2M}$ is just the product probability measure $\mathcal{P}^N \times \mathcal{Q}^M$, where $\mathcal{P}$ and $\mathcal{Q}$ respectively denote the uniform probability distribution on $[0,T]$ and the standard Gaussian probability distribution on $\mathbb{R}^2$. Now, according to Theorem 1 relation*

$$\mathcal{P}\{t \in [0\ T] : PC(t) > \max_{t \in \{t_i\}_{i=1}^N} PC(t)\} \leq \beta, \tag{17}$$

holds true with probability $\mathcal{P}^N$ greater than $1-(1-\beta)^N$. Set

$$q := \mathcal{Q}^M\{(w_1, \ldots, w_M) \in \mathbb{R}^{2M} : \sup_{t \in \{t_i\}_{i=1}^N} |PC'(t) - PC(t)| > \epsilon\}. \tag{18}$$

Putting together these two results we conclude that the following holds with a product probability $\mathcal{P}^N \times \mathcal{Q}^M$ not less than $1-[(1-\beta)^N + q]$

$$\mathcal{P}\{t \in [0,T] : PC(t) > PC(\arg\max_{t \in \{t_i\}_{i=1}^N} PC'(t)) + 2\epsilon\} \leq \mathcal{P}\{t \in [0,T] : PC(t) > PC(\arg\max_{t \in \{t_i\}_{i=1}^N} PC(t))\} \leq \beta,$$

where the first inequality follows from (18) and the second from (17). Thus, maximizing $PC'(t)$ over $\{t_i\}_{i=1}^N$ leads to an approximate maximum of $PC(t)$ to accuracy 2ϵ and level β with confidence $1-\delta$ where $\delta = (1-\beta)^N + q$. If we now use the estimate $q \leq 2Ne^{-2M\epsilon^2}$ given by equation (16), we can easily conclude that using $N = \left\lceil \frac{\ln(\frac{\delta}{2})}{\ln(1-\beta)} \right\rceil$ time instants t's and $M = \left\lceil \frac{1}{2\epsilon^2} \ln \frac{4N}{\delta} \right\rceil$ vectors w's suffices to approximately maximize $PC(t)$ over $[0,T]$ to accuracy 2ϵ and level β with confidence $1-\delta$. □

Note that the number of samples needed to achieve a certain approximation level in terms of the accuracy, level and confidence, is independent of the nature of the sample space and of the probability distribution. In particular, this means that it does not depend on the dimension of the Euclidean space from which the w's are extracted. Thus, the computational load does not significantly increase in the 3-D case with respect to the 2-D case ([22]). This is not be the case if one would resort to numerical methods based on gridding or to approximate analytic methods such the one proposed in [8]. However, further work is required to evaluate the computational efficiency of our approach to mid-term conflict detection, as a function of the accuracy, level and confidence parameters.

C. Validation

In order to validate the conflict detection algorithm introduced in Section II-B, we use Monte Carlo simulations based on the ODE model of the aircraft motion described in Section II-A. We compare the performance achieved by Algorithm 4 in Section II-B, with the one achieved by the algorithm in [8]. The algorithm in [8] is in fact based on the same description of the uncertainty affecting the aircraft motion on which our prediction model is based. Differently from our approach, the criticality measure of a two

aircraft encounter adopted in [8] is an approximation of the probability of conflict. This approximation consists of computing the probability of conflict as if each aircraft were flying at a constant velocity, with a constant tracking error. The values for the velocities, and the mean and variance of the Gaussian random variables describing the tracking errors are set equal to the velocities and tracking error characteristics at the point of minimum deterministic separation (see [7] for more details). An analytical expression for such an approximation is given for the level flight case, while, as for the 3-D case, one has to resort to a numerical procedure ([28]).

The protocol for evaluating the performance of each detection algorithm consists of the following steps:

1. Given the flight plans of the two aircraft, generate pairs of aircraft trajectories over a 20 minutes time horizon according to the discretized version of the stochastic differential equation (9) (sample time set equal to 1 second), with $r_a = 0.25\ nmi/min$, $r_c = 1/57$, and $\bar{\sigma}_c = 1\ nmi$;
2. For each pair of simulated trajectories, execute the conflict detection algorithm at every radar measurement time (radar measurement time set equal to 12 seconds), each time using the updated flight plans and time horizon. These are respectively obtained by removing the way points which have been surpassed and setting the first way point equal to the current radar measurement, and by subtracting the elapsed time from the 20 minutes initial horizon. A conflict is declared as soon as the estimated value of the criticality measure exceeds the prescribed threshold;
3. Compute the *probability of false alarm* P(FA) and the *probability of successful alert* P(SA), *i.e.*, the ratio between the number of alerts issued when there was no conflict and the total number of cases when there was no conflict (P(FA)), and the ratio between the number of alerts issued before a situation of conflict effectively happens and the total number of conflicts (P(SA)).
4. Plot the *System Operating Characteristic* (SOC) curve, *i.e.*, the probability of successful alert versus the probability of false alarm parameterized by the threshold, and choose the optimal threshold.

Note that when using the detection algorithm for avoiding conflict situations, response times of ATC and pilots jointly with the aircraft maneuverability could make some of the detected conflicts not preventable, thus reducing the number of successful alerts. Modeling these aspects is not dealt with in the present paper.

It is evident that the computed P(FA) and P(SA) estimates depend on the value of the threshold used to decide whether an alert should be issued, both being decreasing as functions of the threshold. An ideal conflict detection scheme should operate at point $(0, 1)$, where there are no false alarms and all the situation of conflicts are detected. On the other hand, a real conflict detection scheme cannot operate at this point - irrespective of the threshold chosen - due to the uncertainty affecting the aircraft positions. However, the more the SOC curve approaches the point $(0, 1)$, the better is the performance of the system. The threshold can, therefore, be defined on the basis of this consideration as the one that corresponds to the minimum distance of the SOC curve from the ideal operating point $(0, 1)$ thus striking an "optimal" compromise between the number of false alarms and successful alerts as suggested in [20], [29].

We now describe the results obtained by Monte Carlo simulations in two different encounter situations. In the first example, the aircraft nominal trajectories are straight lines traveled with constant speeds, whereas in the second example the lines joining the way points describe a more complex zig-zag pattern. For these two cases, we draw the SOC curve and compare the performance obtained by Algorithm 4 with those obtained by implementing the detection algorithm described by Erzberger et al. in [8]. The parameters r_a, r_c and $\bar{\sigma}_c$ are set equal to the empirical values $r_a = 0.25\ nmi/min$, $r_c = 1/57$ and $\bar{\sigma}_c = 1\ nmi$ in both the prediction and simulation models.

The simulations show that Algorithm 4 can deal with more general encounter situations than the algorithm in [8]. Specific comments on the obtained results are given at the end of the section.

*Example 1 (*30° *path crossing angle configuration*)
Consider the case when the two aircraft are flying straight at the same altitude along paths whose crossing angle is 30° at speeds of 480 nmi/h and 500 nmi/h. For sake of clarity, in the left side of Figure 3 we have drawn a realization of the trajectories obtained by performing step 1 of the described simulation protocol. For each pair of trajectories, the initial nominal minimum separation and time to minimum separation are respectively 5 nmi and 4 min. As we iterate the algorithms along the trajectories according to step 2, these

quantities keep changing and in fact after about 4 minutes the time to the closest point of approach becomes zero. By running Algorithm 4 and the one proposed in [8] on 1000 pairs of trajectories generated by the

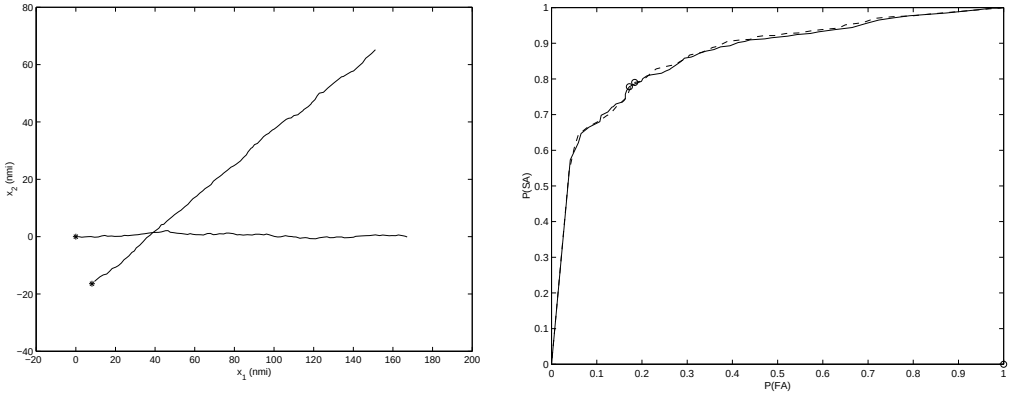


Fig. 3. Left: Sample pair of simulated trajectories. The $\star$ stand for the starting points. Right: SOC curves for Algorithm 4 (solid) & Erzberger algorithm (dashed). The $\circ$ stand for the "optimal" threshold points (Example 1: $\epsilon = 0.05$; $\delta = 0.1$; $\beta = 0.05$; 1000 simulations).

stochastic ODE modeling the aircraft motion, we estimate the probability of successful alert and false alarm and draw the SOC curve for both algorithms. The result is plotted in the right side of Figure 3, where the solid line curve corresponding to the detection algorithm proposed in this paper is almost overlapped to the dashed line curve corresponding to the algorithm in [8]. In this example, the "optimal" threshold value $\bar{C}$, *i.e.*, the one at the point of the SOC curve nearest to ideal operating point $(0, 1)$, and the "optimal" values for P(FA) and P(SA) are similar for the two algorithms ($\bar{C} \simeq 0.85$, P(FA)$\simeq 0.18$, P(SA)$\simeq 0.78$). □

Example 2 (zig-zag flight paths configuration)
In this example, we consider the case when the sequence of way points in the flight plans describes a zig-zag configuration at a fixed altitude with speeds $v_1^A = 505$ nmi/h and $v_2^A = 455$ nmi/h, $v_1^B = 460$ nmi/h and $v_2^B = 470$ nmi/h. In Figure 4 a realization of the trajectories is represented. The estimates of the probability

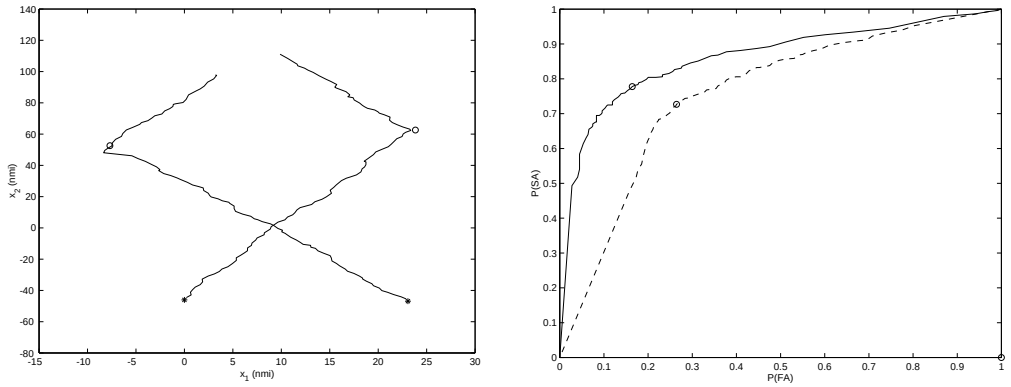


Fig. 4. Left: Sample pair of simulated trajectories. The $\star$ stand for the starting points. Right: SOC curves for Algorithm 4 (solid) & Erzberger algorithm (dashed). The $\circ$ stand for the "optimal" threshold points (Example 2: $\epsilon = 0.05$; $\delta = 0.1$; $\beta = 0.05$; 1000 simulations).

of successful alert and false alarm obtained by 1000 Monte Carlo samples are used to plot the SOC curves in the right side of Figure 4, where the solid line SOC curve corresponds to the detection algorithm proposed in this paper. In this case, the "optimal" threshold values for the introduced detection procedure and the one proposed in [8] are respectively $\bar{C} = 0.62$, corresponding to P(FA)=0.164 and P(SA)=0.778, and $\bar{C} = 0.79$, corresponding to P(FA)=0.264 and P(SA)=0.727. *Thus, Algorithm 4 has a slightly higher probability of*

successful detection and a false alarm probability of 16.4%*, which is* 3/5 *that of [8].*
Note that for each value of P(SA), the algorithm in [8] corresponds to a higher value of P(FA) with respect to our approach. □

The two algorithms give similar results when the aircraft are flying along straight lines, whereas when the cross pattern is more complex our algorithm generally performs better than the algorithm in [8]. This actually is also the case when the trajectories are straight lines traveled at constant speed, but the prediction models for the two algorithms take into account the uncertainty in the current position ([21]). Figure 5 represents the plots of P(FA) and P(SA) as functions of the threshold obtained by 1000 Monte Carlo simulations for the same encounter as in Example 1, but considering a cross-track error with constant variance $\sigma_c^2 = \bar{\sigma}_c^2$ in the prediction models. Note that in this case, irrespectively of the threshold value, P(FA) given by [8] is higher than P(FA) given by our approach, and this increase of P(FA) is not compensated by an adequate increase of the probability of successful alert. The same behavior actually occurs in Example 2.

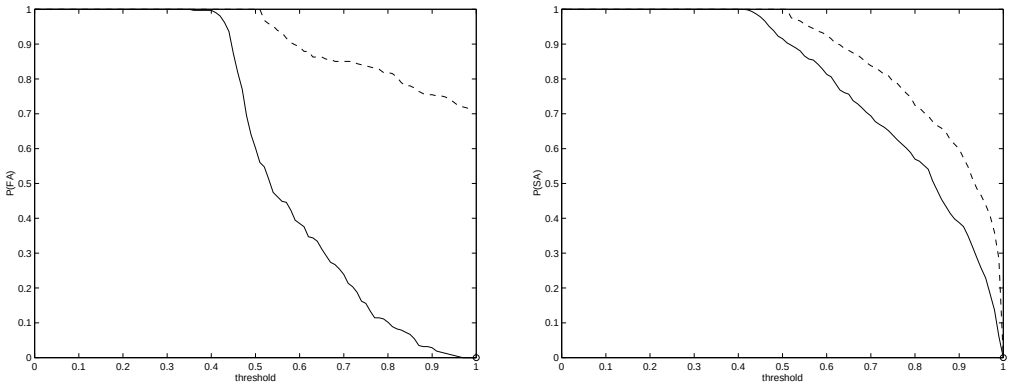


Fig. 5. Plots of P(FA) and P(SA) vs threshold for our algorithm (solid) & Erzberger algorithm (dashed) (Example 1 with constant cross-track variance: $\epsilon = 0.05$; $\delta = 0.1$; $\beta = 0.05$; 1000 simulations).

In our opinion, an explanation for the obtained results is that the measure of criticality used for detecting a situation of conflict in [8] is computed by integrating the probability density function of the distance at the minimum separation point over a strip that is an extension of the circular domain of integration for computing the probability of conflict at that time instant. Thus, it can be viewed as an over approximation of the instantaneous probability of conflict. On the other hand, the criticality measure in [8] is the probability of conflict for a *fictitious encounter* which extends in both the time directions to an infinite horizon, with the aircraft keeping flying at constant velocities with a constant tracking error. So, if either the trajectories are not straight lines or they are diverging and the current position is affected by uncertainty, the probability of conflict computed in [8] may highly overestimate the probability of conflict of the real encounter. This happens for example in the case when a collision occurs along the nominal trajectories of the fictitious encounter. Hence, a higher value for the probability of false alarms is found in the simulations.

It is important to observe that different configurations lead to different SOC curves and therefore to different optimal thresholds. A sensitivity analysis of the dependence of the threshold on the flight plans should be performed by parameterizing them as a function of the crossing angles, minimum deterministic distance, and time of minimum distance, in order to set a value for the threshold which is appropriate for the typically encountered path configurations. Other aspects that should be taken into account for the tuning of the threshold are the detection of conflict a certain amount of time before it occurs and the prediction of its occurrence time (so as to avoid "last minute" maneuvers). These aspects, in fact, highly influence the efficiency of every prediction/resolution scheme involving the human-in-the-loop component. These issues are currently under study.

III. Short range conflict detection

In this section, we propose to adopt a sample path viewpoint in order to measure the criticality of an encounter situation. The approach is motivated by the fact that the motion of each aircraft involved in the encounter is subject to perturbations, whose overall effect in short time horizons can be modeled as a white noise affecting the aircraft velocity. In this way a different likelihood is attributed to the different admissible paths of each aircraft, and the probability of conflict of the encounter can be computed as the probability that a path of some aircraft enters the protected zone of another aircraft. The proposed approach is briefly explained next, whereas the mathematical derivations are postponed to the following subsections.

Consider the following first order stochastic differential equation in $\mathbb{R}$:

$$\frac{ds}{dt}(t) = f(t) + w(t), \tag{19}$$

where $f(\cdot)$ is a piecewise Lipschitz continuous function defined on $[0, \infty)$ and $w(\cdot)$ is a white noise with power spectral density ν^2, *i.e.*, $E[w(t)w(t+z)] = \nu^2\,\delta(z)$ for all $z, t \geq 0$.
Equation (19) can be used to model the motion of an aircraft. In such a case, ds/dt represents the aircraft ground speed, f is the air speed which can be directly controlled by the aircraft, and w models mainly local wind effects such as air turbulence, as well as deviations due to mechanical and human factors, [30]. Integrating equation (19), we have

$$s(t) = \int_0^t f(z)\,dz + b(t), \tag{20}$$

where $b(t) = \int_0^t w(z)\,dz$ is a Gaussian process with stationary, independent increment, whose mean value and variance are respectively $E[b(t)] = \int_0^t E[w(z)]\,dz = 0$ and $\mathrm{Var}[b(t)] = \int_0^t\!\int_0^t E[w(z_1)w(z_2)]\,dz_1\,dz_2 = \nu^2 t$. This last equation and the Kolmogorov continuity theorem imply that $b(t)$ has a continuous version $B(t)$, *i.e.*, $P(\{\omega : b(t,\omega) = B(t,\omega)\}) = 1$ for all $t \geq 0$, where $\{B(t,\omega),\, t \geq 0\}$ is continuous in t with probability one (see [31]). The continuity of $b(t)$ is actually required when equation (19) is used to model the aircraft motion. Note that, after scaling by $1/\nu$, $b(t)$ is a standard Brownian Motion (BM). A BM possesses many unusual local properties. For example, at any fixed time its sample path is not differentiable with probability one. However, this is not of a major concern since we are interested in its collective properties, *i.e.*, the probability that the perturbed trajectory experiences a large deviation from the nominal one or, more precisely, that $s(\cdot)$ evolves outside some *safe set* thus causing a conflict. In [32] the approach based on equation (19) is used for highway safety analysis.
By subtracting the nominal motion $\int_0^t f(z)\,dz$ and properly scaling, we can therefore reduce the perturbed motion in (20) to a standard BM. Correspondingly, the safe set is transformed into some time-varying set. The problem of computing the probability of conflict then reduces to calculating the escaping probability of a standard BM with respect to a time-varying region. In general, however, it is very hard - if not impossible - to get an analytical expression for such a probability. A case when this is possible is the following.

Lemma 1 (Bachelier-Levy, [33]) *Let $B(t)$ be a standard one dimensional BM starting at the origin. Fix $\mu \in \mathbb{R}$ and define $\tau := \inf\{t \geq 0 : B(t) = a - \mu t\}$ to be the first time $B(t)$ reaches a point which is moving with speed μ towards the origin starting at position $a > 0$. Then, τ has probability density function:*

$$p_\tau(t) = \frac{a}{\sqrt{2\pi t^3}} \exp[-\frac{(a-\mu t)^2}{2t}], \quad t \geq 0. \tag{21}$$

This lemma will prove to be of utmost importance for estimating the probability of conflict in the - more complex - situations of real interest, which will in fact be reduced to the case described in the lemma by appropriate approximations.

The rest of the present section is organized as follows. We start by detailing the prediction model we propose for short range conflict detection (Section III-A). In Section III-B, we derive closed-form approximations of the probability of conflict. Finally, in Section III-C, we describe a decentralized conflict resolution

algorithm based on the introduced closed-form approximations. This algorithm is used to analyze the effectiveness of the proposed criticality measure by Monte Carlo simulations.

A. Prediction model

Consider an aircraft flying with constant speed along a straight line on a plane. Denote with $u = (u_1, u_2)^T$ the aircraft velocity and with ϑ the angle that vector u makes with the x_1 axis of a global coordinate frame. The aircraft motion can then be represented by the following kinematic model:

$$x(t) = u\,t + R(\vartheta)\Sigma B(t) \tag{22}$$

where $R(\vartheta)$ is the rotation matrix given by equation (6), $\Sigma = \mathrm{diag}(\nu_a, \nu_c)$, ν_a^2, ν_c^2 being the power spectral densities of the perturbations affecting the motion in the along-track and cross-track directions, and $B(t)$ is a standard 2-D BM.

Note that under the constant velocity assumption, the variances of the along-track and cross-track perturbations given in equations (1) and (2) can be approximated as linear in time over short time horizons. Therefore, the proposed model can be used for short term prediction. As for the values of ν_a, ν_c, setting $r_c = 1/57$ and $r_a = 0.25$ nmi/min ([15], [10]), and considering standard cruising speeds of commercial aircraft, *e.g.*, $v = 480$ nmi/h, the growth rate of $\sigma_c^i(t)$ with time is given by $r_c v \simeq 0.14$ nmi/min which is lower than the growth rate r_a of $\sigma_a^i(t)$. Hence, typically $\nu_a > \nu_c$.

Model (22) seems also suitable for the free flight scenario prospected for the future ATMS, [19]. In this case in fact the saturation phenomenon attributed in [7] to pilot feedback to maintain the aircraft on a *specific* trajectory is not so significant, since, according to the free flight paradigm, each aircraft receives advisories rather than mandatory trajectory specifications from the ATCs. Moreover, early conflict detection and resolution is responsibility of individual aircraft which have only partial information about the intentions of the surrounding aircraft - mainly their current positions and headings -, and which cannot rely on the fact that the intruder will make an effort to maintain its current heading precisely.

B. Closed-form expressions for the probability of conflict

Consider two aircraft, labeled A and B, flying at the same altitude. Assume without loss of generality that at time $t = 0$ aircraft A is at the origin of a global coordinate frame, flying along the x_1 axis from left to right with a velocity $u^A \in \mathbb{R}^2$, while aircraft B is at position $\Delta x_0 \in \mathbb{R}^2$, flying with a velocity $u^B \in \mathbb{R}^2$ which makes an angle ϑ with the x_1 axis. A conflict occurs if aircraft B enters the protected zone of aircraft A or vice versa (see Figure 6).

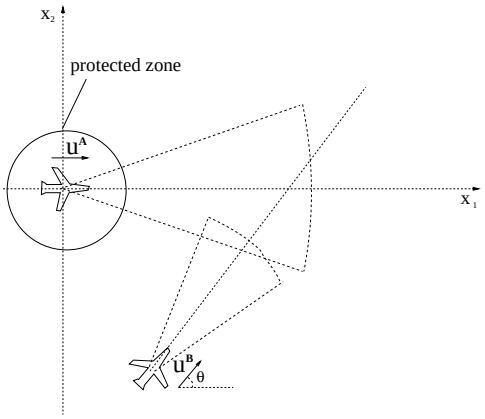


Fig. 6. Encounter situation for two aircraft flying at the same altitude.

Denote with $x^A(t)$ and $x^B(t)$ the position of aircraft A and B, respectively. By the kinematic model (22),

we get the following description for the two aircraft system:

$$\begin{cases} x^A(t) = u^A t + \Sigma B^A(t) \\ x^B(t) = \Delta x_0 + u^B t + R(\vartheta)\Sigma B^B(t) \end{cases} \tag{23}$$

where $B^A(t)$ and $B^B(t)$ are independent standard 2-D BMs. We assume that the BMs $B^A(t)$ and $B^B(t)$ start at the origin, thus ignoring errors in the radar and GPS measurements, as in the mid-term prediction model. Subtracting the first equation from the second in (23) leads to the following description of the aircraft relative position:

$$\Delta x(t) = \Delta x_0 + \Delta u\, t - n(t), \tag{24}$$

where we set $\Delta x(t) := x^B(t) - x^A(t)$, $\Delta u := u^B - u^A$, and $n(t) := \Sigma B^A(t) - R(\vartheta)\Sigma B^B(t)$.
Equation (24) suggests that one can think of the motion of aircraft A as consisting only of the perturbation $n(t)$, and the motion of aircraft B as deterministic with constant velocity Δu starting at Δx_0. Furthermore, it is shown in Proposition 3 that the Gaussian process $n(t)$ can be reduced to a standard 2-D BM by a coordinate transformation, which is similar to the one used in Proposition 2.

Proposition 3 *Set $P := \sqrt{2}\, R(\frac{\vartheta}{2})\Lambda$, with $\Lambda = diag(\lambda_1, \lambda_2)$, where*

$$\begin{cases} \lambda_1 = \sqrt{\nu_a^2 \cos^2(\frac{\vartheta}{2}) + \nu_c^2 \sin^2(\frac{\vartheta}{2})} \\ \lambda_2 = \sqrt{\nu_a^2 \sin^2(\frac{\vartheta}{2}) + \nu_c^2 \cos^2(\frac{\vartheta}{2})} \end{cases} . \tag{25}$$

Then, the stochastic process $\bar{n}(t) := P^{-1} n(t)$ is a standard 2-D BM starting at zero.

Proof *The thesis easily follows once one observes that $[\Sigma^2 + R(\vartheta)\Sigma^2 R(\vartheta)^T]t$ is the covariance matrix of $n(t)$ and it is equal to $PP^T t$.* □

By using the coordinate transformation with matrix P^{-1} given in Proposition 3, equation (24) can be rewritten as follows

$$\Delta s(t) = \Delta s_0 + u\, t - \bar{n}(t), \tag{26}$$

where $\Delta s(t) := P^{-1}\Delta x(t)$ is the relative position of the two aircraft in the new coordinate system, $\Delta s_0 := P^{-1}\Delta x_0$, $u := P^{-1}\Delta u$, and $\bar{n}(t) = P^{-1} n(t)$ is a 2-D standard BM starting at zero. Therefore, we can view the motion of aircraft A as a standard 2-D BM starting at the origin and the motion of aircraft B as a motion at constant velocity u starting at Δs_0.
In the new coordinate system, the circular protection zone of radius $\rho = 5$ nmi around aircraft B is transformed into an ellipse initially centered at $\Delta s(0) = \Delta s_0$ and then moving along with aircraft B at velocity u (see Figure 7), with boundary described by:

$$\lambda_1^2 (x_1 - \Delta s_1(t))^2 + \lambda_2^2 (x_2 - \Delta s_2(t))^2 = \rho^2/2, \tag{27}$$

where λ_1 and λ_2 are given in (25), and the evolution in time of $\Delta s(t)$ is governed by equation (26). A conflict then occurs if and only if the 2-D standard BM $\bar{n}(t)$ starting at the origin (*i.e.*, aircraft A in the new coordinate system) ever wanders into this moving ellipse.

Before estimating the probability of such an event, we introduce some definitions that will be used in the derivations. We define x_d to be the distance of the origin from the line h along which aircraft B is flying, and a to be the distance from the position Δs_0 of aircraft B at $t = 0$ to the projection H of the origin on h, as indicated in Figure 7 representing the new coordinate system. Then, x_d and a can be computed as follows:

$$x_d = \frac{|\Delta s_0^T R(\frac{\pi}{2}) u|}{\|u\|}, \qquad a = -\frac{\Delta s_0^T u}{\|u\|}. \tag{28}$$

Observe that a positive value for a indicates that the two aircraft are approaching each other, whereas a negative value for a indicates that they are flying away from each other. Then, if we ignore the effect of the

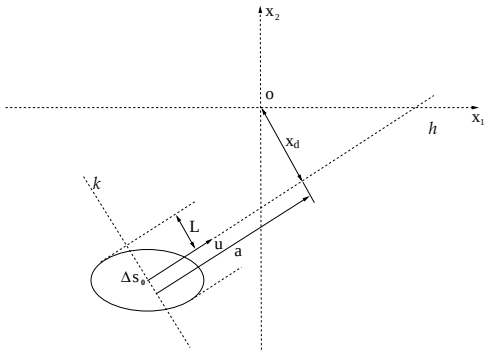


Fig. 7. Transformed protection zone.

noise, in the new coordinate system the minimal separation distance during the encounter is given by x_d if $a > 0$, and by $\|\Delta s_0\|$, *i.e.*, the distance at time $t = 0$, if $a < 0$.

Denote the conflict event, *i.e.*, the event that $\bar{n}(t)$ ever wanders into the moving ellipse, by F. The probability of conflict $P(F)$ does not admit a closed-form formula. However, we can approximate it by a "decoupled" event. Let k be the line passing through the center of the ellipse and orthogonal to u which moves along with the ellipse with velocity u (see Figure 7). The projected width $2L$ of the ellipse along line k can be computed as follows:

$$L = \frac{\rho}{\lambda_1 \lambda_2} \sqrt{\frac{u_1^2 \lambda_1^2 + u_2^2 \lambda_2^2}{2(u_1^2 + u_2^2)}}, \tag{29}$$

where λ_1 and λ_2 are given in equation (25). Denote by τ the first time $\bar{n}(t)$ hits k and define F' to be the event that the first time $\bar{n}(t)$ hits line k, it is within a distance of L from the center of the ellipse. In [34], it is shown that $P(F')$ is a good estimate of the probability of conflict $P(F)$ when the difference of the two aircraft velocities is much larger than the variance growth rate of the BM, which is in fact the case of interest. The intuition for this is that when the velocity of the moving ellipse is high, the only dimension of the ellipse that is relevant for the event F is that perpendicular to u. The interested reader is referred to [34] for a formal discussion on the quality of such an approximation.

With no loss of generality, in order to compute $P(F')$ we assume that u is aligned with the positive x_1 axis. As a matter of fact, the axes rotation eventually necessary to make u aligned with the positive x_1 axis can be incorporated into the transformation matrix P, and still the motion of aircraft A remains a standard BM, since BM is invariant with respect to rotations.

Assume that the two aircraft are approaching each other. Then, a given by equation (28) is positive. Moreover, time τ for aircraft A to reach line k has evidently the distribution $p_\tau(\cdot)$ given by equation (21) in Lemma 1 with $\mu = \|u\|$. The approximate probability of conflict $P(F')$ can then be written as:

$$P(F') = \int_0^\infty p_\tau(t) \int_{|y - x_d| < L} \frac{1}{2\pi t} \exp(-\frac{y^2}{2t})\, dy\, dt = \int_0^\infty p_\tau(t) \big[Q(\frac{x_d - L}{\sqrt{t}}) - Q(\frac{x_d + L}{\sqrt{t}}) \big]\, dt, \tag{30}$$

where we set $Q(y) := \int_y^\infty \frac{1}{\sqrt{2\pi}} \exp(-z^2/2)\, dz$.

Note that by Formula 1 in Appendix A, $E[\tau] = a/\mu$. When μ is greater than 1, $p_\tau(t)$ concentrates near $t_0 := a/\mu$. If we then use the 0-th order expansions of $Q(\frac{x_d - L}{\sqrt{t}})$ and $Q(\frac{x_d + L}{\sqrt{t}})$ around t_0 in (30), we get:

Assertion 3 *The probability of conflict can be approximated by:*

$$P^0 := Q(\frac{x_d - L}{\sqrt{t_0}}) - Q(\frac{x_d + L}{\sqrt{t_0}}). \tag{31}$$

where L is given by equation (29), a and x_d are given by equation (28), and we set $t_0 := a/\mu$.

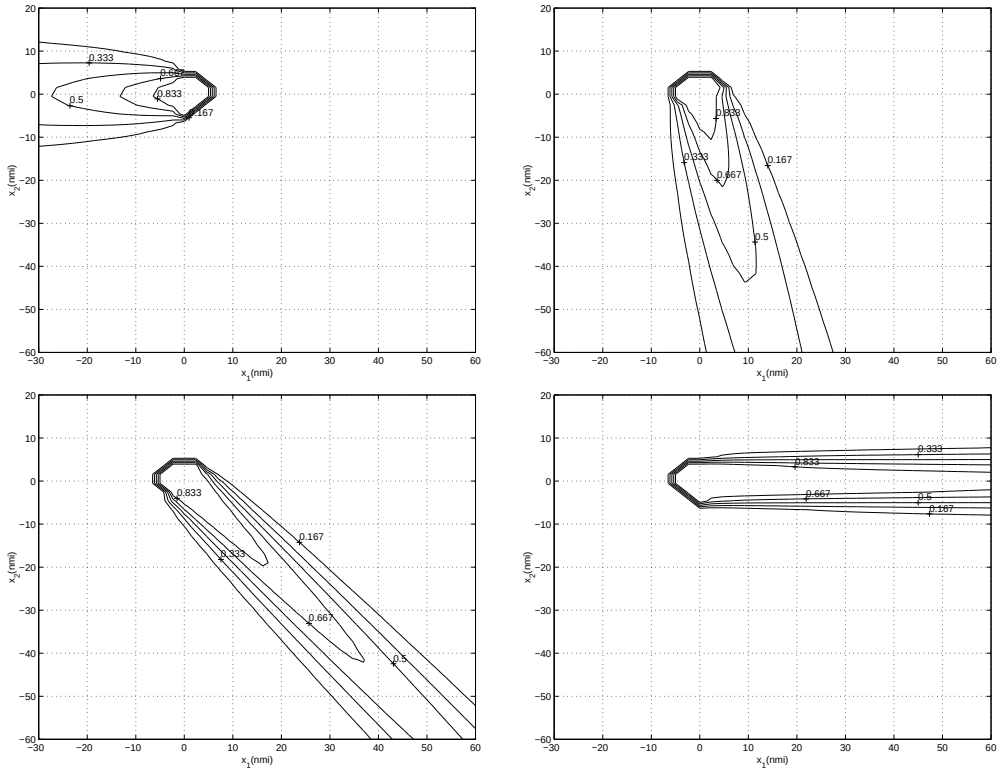


Fig. 8. P^0 for path angles $\vartheta = 0^\circ$ (upper left); $\vartheta = 45^\circ$ (upper right); $\vartheta = 90^\circ$ (lower left); $\vartheta = 180^\circ$ (lower right) with $\|u^A\| = 7$ nmi/min, $\|u^B\| = 8$ nmi/min, $\nu_a = 2$, ν_c=1.

In [34], it is shown that P^0 is a remarkably sharp estimate of $P(F')$ computed by numerical integration even in the case when $\mu = \|u\|$ is much smaller than 1 (the approximation error when $\mu > 1$ is barely visible). Higher order approximations of the probability of conflict can also be obtained.

Figure 8 shows some level curves of P^0 as a function of the initial position Δx_0 of aircraft B for the path angles $\vartheta = 0^\circ$, 45°, 90°, 180°. In each plot, aircraft A is moving from left to right with velocity $\|u^A\| = 7$ nmi/min starting at the origin and aircraft B is moving with velocity $\|u^B\| = 8$ nmi/min. To see the results more compactly, we used relatively large ν_a and ν_c ($\nu_a = 2$ and $\nu_c = 1$). Note that in all the examples ν_a and ν_c are treated as adimensional quantities with the understanding that their squares are equal to the values in nmi^2/min of the along-track and cross-track power spectral densities of the BM appearing in the aircraft model (22). Observe that as the cross path angle ϑ increases, the region delimited by the same P^0 equi-probability line gets more and more extended. This means that for a fixed distance of the two aircraft, the situation of headon conflict ($\vartheta = 180^\circ$) is the most dangerous, whereas the situation of overtake conflict ($\vartheta = 0^\circ$) is the least dangerous.

We now introduce an expression for the probability of conflict within a time horizon t_f. Limiting the prediction time horizon to be finite makes in fact much sense, since for the same value of the probability of conflict, the more dangerous situations are the ones with smaller projected collision time. Moreover, by considering a finite prediction horizon, we avoid problems similar to those pointed out in Section II-C for the algorithm in [8].
The probability of conflict within a fixed horizon t_f is computed as follows:

$$P(F' \cap \{\tau \le t_f\}) = \int_0^{t_f} p_\tau(t) \int_{|y-x_d|<L} \frac{1}{2\pi t} \exp(-\frac{y^2}{2t})\, dy\, dt = \int_0^{t_f} p_\tau(t) \big[Q(\frac{x_d - L}{\sqrt{t}}) - Q(\frac{x_d + L}{\sqrt{t}})\big]\, dt,$$

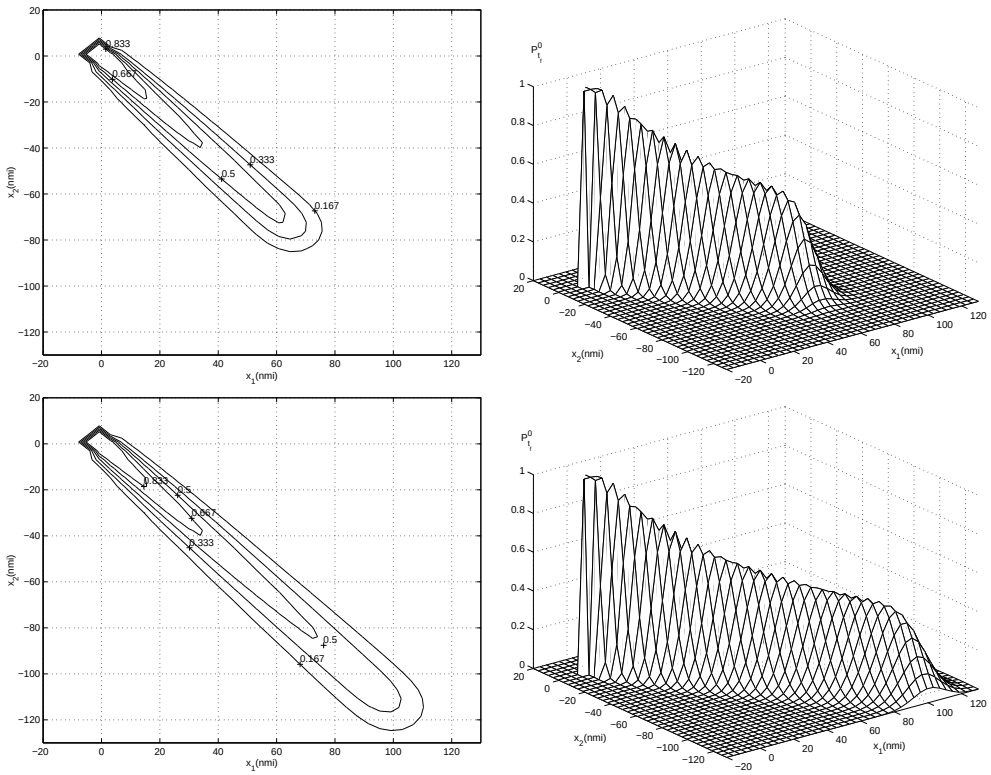


Fig. 9. $P^0_{t_f}$ for $t_f = 10$ min (top); $t_f = 15$ min (bottom) with $\vartheta = 90°$, $\|u^A\| = 7$ nmi/min, $\|u^B\| = 8$ nmi/min, $\nu_a = 2$, $\nu_c = 1$.

which, by using 0-th order series expansions of $Q(\frac{x_d - L}{\sqrt{t}})$ and $Q(\frac{x_d + L}{\sqrt{t}})$ around t_0, and Formula 2 in Appendix A, leads to

Assertion 4 *The probability of conflict within time t_f can be approximated by:*

$$P^0_{t_f} := \left[Q(\frac{a - \mu t_f}{\sqrt{t_f}}) + \exp(2a\mu)Q(\frac{a + \mu\, t_f}{\sqrt{t_f}})\right]\left[Q(\frac{x_d - L}{\sqrt{t_0}}) - Q(\frac{x_d + L}{\sqrt{t_0}})\right] \tag{32}$$

where L is given by equation (29), a and x_d are given by equation (28), and we set $t_0 := a/\mu$.

Figure 9 shows some level curves of $P^0_{t_f}$ for $\vartheta = 90°$, in the two cases when $t_f = 10$ min and $t_f = 15$ min. Note that $P^0_{t_f}$ looks like a truncation of the corresponding infinite horizon version. Moreover, it is not surprising that as ν_a and ν_c increase, the truncation becomes smoother.
We also computed higher order approximations of the probability of conflict for both the infinite and finite horizon cases, but they were not significantly different from the 0-th order ones.

Remark 1 *In [35], the expression (31) for the probability of conflict is obtained by a different approach, which mainly consists of computing the probability that the aircraft distance at the nominal time of closest approach is lower than the minimum allowed distance, given that their initial positions and velocities are uncertain. In [35], it is then proposed to use a level curve of (31) to delimit the "alert zone" which is the protection zone around the aircraft where ATC assistance for conflict avoidance is required in a free flight setting. By approaching the problem from the process point of view, we get more general results which can for example be easily extended to the finite horizon case. This allows the alert zone to be appropriately chosen based also on the look-ahead time horizon.*

C. A decentralized conflict resolution algorithm for validation

Many contributions have been presented in the literature to deal with the issue of providing coordination maneuvers which are guaranteed to be safe for the two aircraft case, both in a deterministic (see *e.g.* [11], [12], [36], [37]) and in a probabilistic setting (see *e.g.* [8], [35]). In particular, TCAS ([12]), currently operating on all passenger aircraft carrying more than 30 seats, deals with short range conflict resolution by choosing among a predefined set of maneuvers the least aggressive one that ensures a safe distance between two aircraft which are predicted to incur in a conflict situation. In TCAS, however, uncertainty in the aircraft position is not taken into account when predicting a conflict.
Conflicts involving multiple aircraft, besides being more general, may actually occur in high-traffic areas. Moreover, it might also happen that by solving a two aircraft conflict without taking into consideration the other surrounding aircraft, one generates a conflict with a third aircraft, or that by solving the conflicts pairwise, no feasible resolution maneuver can be found. Despite of that, there are only a few contributions treating directly the multiple aircraft case, and they typically consider a deterministic setting. Among them, we mention [38], [39], [40], [41], where the approach to multiple aircraft conflict resolution is based on the optimization of a cost function, which is suitably selected so as to take into account practical factors such as fuel consumption and passenger comfort. The computational issues involved in such a constrained optimization problem are solved by genetic algorithms in [38], semidefinite programming combined with a branch-and-bound search in [39], and an iterative method based on approximating the performance index as quadratic and the constraints as linear in [40]. The main drawback of these approaches is that there is no guarantee that the obtained solution is actually optimal. In [41], resolution maneuvers are classified into different homotopy types, and the problem is reduced to a convex optimization problem within each type. Therefore, one can, in principle, compute the global optimum by comparing the optimal solutions associated with all the different types of maneuvers. The number of types, however, highly grows with the number of aircraft involved in the encounter.

We introduce in this section an algorithm for multiple aircraft conflict resolution which uses the probability of conflict calculated in Section III-B to guide each aircraft to its desired destination, while avoiding possible situations of conflict with other aircraft flying in the same region of the airspace. Differently from the algorithms for multiple aircraft conflict resolution mentioned above, in our algorithm the uncertainty affecting the aircraft motion is taken into account. Moreover, we consider decentralized conflict resolution, since each aircraft solves its own conflicts with the other aircraft based on information regarding their positions/intentions, but without coordinating with them. However, in contrast with the worst case approach to decentralized conflict resolution in [11], [17] where the aircraft do not trust each other, the idea here is that each aircraft assumes that the other aircraft will try to behave rationally but this effort is undermined by the various uncertainties inherent in the environment.

Our algorithm is inspired by [42] where the potential and vortex field method is applied to generate coordination maneuvers for multiple aircraft encounters. The main difference from this contribution is that we use the information not only on the current positions but also on the current headings and speeds of the surrounding aircraft in order to generate the repulsive force acting on the aircraft. Thus, the two cases when two aircraft are at the same distance, but approaching and flying away from each other, do not lead to the same repulsive force, and hence abrupt avoidance maneuvers are avoided.

In this paper, the probabilistic algorithm for conflict resolution is used for assessing the significance of the proposed criticality measure by Monte Carlo simulations. In [41], this algorithm is used as 'random type chooser', thus providing a randomized solution to the combinatorial optimization problem.

Consider the case when two aircraft (labeled A and B) are currently at positions a^A and a^B and have destinations b^A and b^B respectively. Assume their initial headings are toward their own destinations and that they fly at a constant speed, say v^A and v^B respectively.
At each time instant t, the probability of conflict $P_c(t)$ can be calculated by equation (31) (or equation (32) with some fixed horizon t_f) using the current positions $x^A(t), x^B(t)$ and velocities u^A, u^B of the two aircraft.

We define for each aircraft three particular headings of interest:

- *Current heading ϑ_c:* Direction along which the aircraft is currently flying.
- *Destination heading ϑ_d:* Direction defined by the current aircraft position and its desired destination.
- *Gradient heading ϑ_g:* Direction corresponding to the highest decrease of the probability of conflict. Since both aircraft maintain their velocities within a short time, ϑ_g can be chosen as the direction of the negative gradient of P_c as a function only of the current position of the aircraft.

Our resolution strategy aims at making each aircraft reach the desired destination while avoiding situations of conflict by appropriately changing the heading. This is done by updating the aircraft headings at every Δt time instants by means of the following algorithm, which in fact represents the heart of our detection and resolution scheme.

Algorithm 5 (Decentralized conflict detection and resolution)

Compute $P_c(k)$ based on $x^A(k), x^B(k), u^A(k), u^B(k)$
for $i = A, B$ **do**
begin
Compute $\vartheta_c^i(k)$, $\vartheta_d^i(k)$, and $\vartheta_g^i(k)$ given $x^A(k), x^B(k), u^A(k), u^B(k)$.
Set $\bar{\vartheta}^i(k) := P_c(k)\vartheta_g^i(k) + (1 - P_c(k))\vartheta_d^i(k)$.
Given $\vartheta_c^i(k-1)$ choose the new headings at step k as follows:

$$\vartheta_c^i(k) = \begin{cases} \bar{\vartheta}^i(k), & \text{if } |\bar{\vartheta}^i(k) - \vartheta_c^i(k)| < \beta \\ \vartheta_c^i(k-1) + \beta \cdot \mathrm{sgn}(\bar{\vartheta}^i(k) - \vartheta_c^i(k-1)) & \text{otherwise.} \end{cases}$$

end

where β is the maximal turn angle allowed per time step and we used $z(k)$ to denote $z(k\Delta t)$.

The ideal new heading $\bar{\vartheta}^i(k)$ is the weighted sum of destination heading and gradient heading, with weights depending on $P_c(k)$. Intuitively, if $P_c(k)$ is high, then decreasing the probability of conflict becomes a priority and therefore the aircraft should pursue the gradient direction more. If, instead, $P_c(k)$ is negligible, then the aircraft should pursue the destination direction. In any case, due to the nonholonomic nature in aircraft, the deviation from current heading is restricted by β. This is the reason why the new heading is chosen to be the one nearest to $\bar{\vartheta}^i(k)$ within the allowed range.

The validation protocol adopted in performing the Monte Carlo simulations is as follows:

1. At every iteration, execute
   (a) *An initialization step*, where given the flight plans of the two aircraft, one computes $\vartheta_d^i(0)$ and set $x^i(0) = a^i$, $\vartheta_c^i(-1) = \vartheta_d^i(0)$, $u^i(0) = (v^i \cos(\vartheta_c^i(-1)), v^i \sin(\vartheta_c^i(-1)))^T$, $i = A, B$, and $k := 0$;
   (b) *A resolution step*, where one chooses the current heading at time k by means of Algorithm 5;
   (c) *An updating step*, where one updates the aircraft velocities and positions at time step $k+1$ as follows:

$$u^i(k+1) = (v^i \cos(\vartheta_c^i(k)), v^i \sin(\vartheta_c^i(k)))^T$$
$$x^i(k+1) = x^i(k) + \Delta t\, u^i(k+1) + R(\vartheta_c^i(k))\, \mathrm{diag}(\nu_a, \nu_c)\, \Delta t\, n^i(k+1),$$

   $i = A, B$, where $R(\vartheta)$ is the rotation matrix of angle ϑ, and n^A and n^B are independent 2-D white Gaussian noises with unit covariance matrices.
   (d) *A verification step*, where one check if both the aircraft have reached their destinations. If this is not the case, one sets $k := k + 1$ and go back to step b.
2. After a prescribed number of iterations, build the histogram of the minimum separation distance between the aircraft.

This approach can be viewed as a receding horizon control method, with Δt and the horizon over which P_c is computed respectively representing the control horizon and the prediction horizon. Hence, in each simulation of the encounter we actually test our short range conflict detection and resolution algorithm repeatedly.

Simulation results are shown in Figure 10 for two typical encounters: headon encounter and orthogonal encounter. The speeds of the two aircraft are chosen such that it takes 30 minutes for them to fly from their starting positions (marked with stars) to their destination positions (marked with diamonds) along the unresolved straight line motion. We used equation (31) for computing $P_c(k)$. Moreover, we set $\nu_a =$

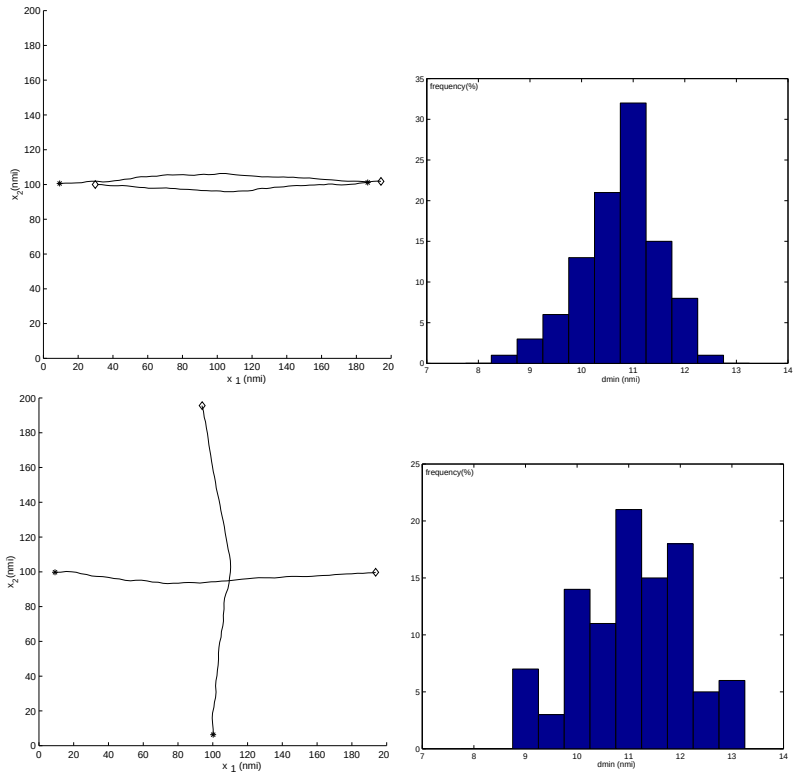


Fig. 10. Resolution maneuver for two aircraft encounter situations ($\nu_a = 0.35, \nu_c = 0.2$, $\beta = \pi/25$).

$0.35, \nu_c = 0.2$, and $\beta = \pi/25$. The values of ν_a and ν_c are chosen so as to get a first order approximation of the growing rate of the along-track and cross-track variances in (1) and (2).
The histograms obtained by running 100 simulations show that in both cases the minimum separation distance is respected and, moreover, most of the minimum separation distances are greater than 2ρ. The factor 2 arises since each aircraft tries to maintain a distance of ρ from the other one, and there is no coordination between them.

In the simulations, the two aircraft update their heading every $\Delta t = 1$ min. Hence, a zoom into the resolution trajectories would reveal lots of chattering between positions corresponding to higher P_c and lower P_c. This behavior is expected, since the function P_c is very sensitive to the heading of the two aircraft. In order to get flyable paths, we can either update the heading at a larger time step or predict a certain period of time further into the future and take the average of the resolution headings during that period (which acts as a low pass filter). This is not dealt with in the present paper.

The algorithm can be extended to the multiple aircraft case. Suppose we have n aircraft sharing the same region of the airspace. Then, all aircraft pairs have to be considered and the new heading of each aircraft has to be computed taking into account the overall probability of conflict.
For each aircraft, say i, $i \in \{A_1, A_2, \cdots, A_n\}$, we compute the probability of conflict between it and any of the other aircraft. Denote with $P_c^{ij}(k)$ the probability of conflict between aircraft i and j, and with $\vartheta_g^{ij}(k)$ the corresponding gradient heading. Then, the new heading $\vartheta_c^i(k)$ for aircraft i is chosen based on the expression given in Algorithm 5 for the two aircraft case, but with the ideal heading computed as follows:

$$\bar{\vartheta}^i(k) = \bar{P}_c(k) \frac{\sum_{j \neq i} P_c^{ij}(k) \vartheta_g^{ij}(k)}{\sum_{j \neq i} P_c^{ij}(k)} + (1 - \bar{P}_c(k)) \vartheta_d^i(k),$$

where $\bar{P}_c(k) = \max_{j \neq i} P_c^{ij}(k)$ is the maximum probability of conflict among all aircraft pairs involving

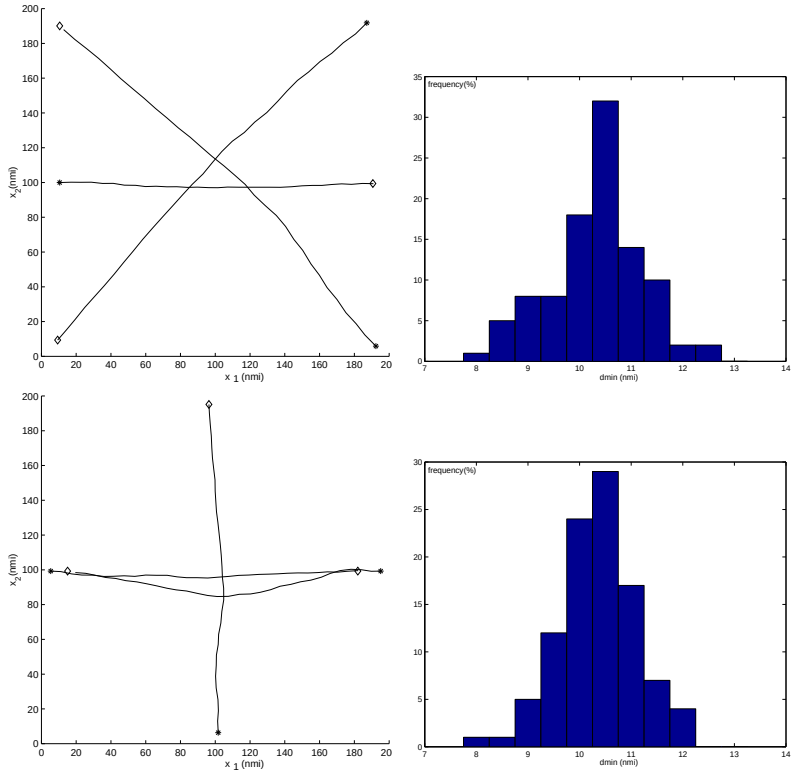


Fig. 11. Resolution for three aircraft encounter situations ($\nu_a = 0.35, \nu_c = 0.2$, $\beta = \pi/25$).

aircraft i and is used as an indicator of the degree of danger for aircraft i in the multiple aircraft setting.

Simulation results for two different three aircraft encounter situations are shown in Figure 11. All parameters remain the same as in the two aircraft case. Since the airspace is not considerably more congested than as in the two aircraft case, still most of the minimum separation distances are centered around 2ρ.
The situation becomes more complicated when the number of aircraft increases to, say, eight as it is shown in Figure 12. The first encounter situation is a symmetric encounter where the eight aircraft pass through a common point at angles evenly distributed in $[0, 2\pi]$. In the second encounter situation, the eight aircraft are divided into two groups. The aircraft in each group follow each other in a streamline and the trajectories of the two groups are orthogonal to each other. In this simulation, we set $\rho = 10$ instead of 5 nmi just for the maneuvers to be more evident in the pictures. Note that in this case, the minimal separation of ρ nmi is satisfied for 92 out of 100 runs for the first example and 94 out of 100 runs for the second one. This shows that the algorithm cannot guarantee absolute safety. We are currently analyzing its safety properties as a function of the number of aircraft involved in the encounter and the uncertainty level.

IV. Conclusions

In this paper, we dealt with aircraft conflict detection at the mid-range and short range levels of the ATMS. Starting from an empirically motivated probabilistic description of the aircraft motion, stochastic models are proposed for mid-term and short term prediction of the aircraft positions, thus allowing the corresponding criticality measures to take into account the various sources of uncertainty inherent in the environment.

Although we focused on the planar case, the extension to the 3-D case is straightforward. As for mid-term conflict detection, the computation of the proposed criticality measure can still be performed in an efficient way by the randomized methods as remarked at the end of Section II-B. As for the short range conflict detection, in a similar way to the 2-D case, analytical approximations for the probability of conflict can be

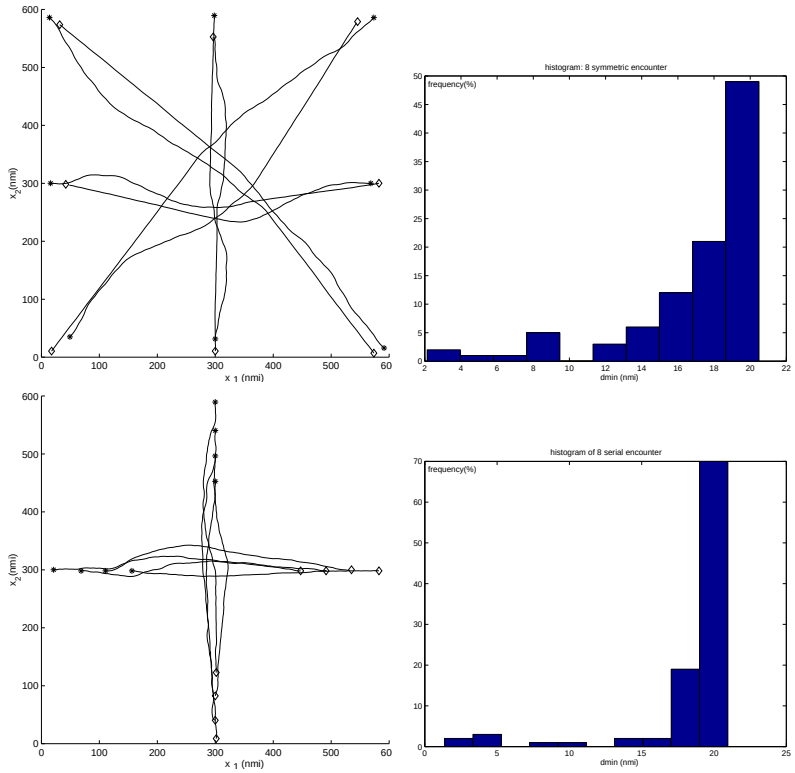


Fig. 12. Resolution for eight aircraft encounter situations ($\nu_a = 0.35, \nu_c = 0.2$, $\beta = \pi/25$, $\rho = 10$ nmi).

computed for the 3-D case, and then be used to derived a decentralized 3-D resolution algorithm. However, it is considerably harder to get meaningful bounds for the error of such approximations.

We are currently working on formulating algorithms for mid-range conflict resolution and investigating different design alternatives. In particular, we are studying a model predictive control approach to achieve this goal. The idea is to use the current radar measurements, the introduced detection module and an optimization algorithm to minimize an appropriate *resolution cost function* over all possible flight plans. ATC is notified if changes in the upcoming way points are imminent. The process is repeated every time a new radar measurement becomes available.
The design choices that enter into conflict resolution are the different resolution cost functions and the different optimization techniques. The optimization could be carried out by randomized algorithms, but one has to carefully investigate the smoothness properties of the cost to be optimized as a function of the controlled variables, *i.e.* the flight plans.
The parameters one needs to set are: How far in advance should ATC be notified of flight plan changes (partly determined by human capabilities) and the bounds on the allowable flight plans (partly determined by aircraft capability and ATC protocol).
Once the prediction/resolution algorithm has stabilized, we hope to be able to test it in human-in-the-loop simulations. This will allow us to tune the various parameters, and assess its impact on ATC workload.

In parallel we are working towards a methodology for formally evaluating the safety properties of the proposed algorithms. This will hopefully lead to a more general probabilistic verification methodology for hybrid systems.

Appendix

I. Formulae

The formulae reported in this section are derived in [34]. The proofs are omitted for brevity.

Formula 1 *If* $x > 0, v > 0$, *then* $\int_0^\infty \frac{x}{\sqrt{2\pi t}} \exp[-\frac{(x-vt)^2}{2t}]\, dt = \frac{x}{v}$.

Formula 2 *Define* $Q(x) := \int_x^\infty \frac{1}{\sqrt{2\pi}} \exp(-z^2/2)\, dz$. *Then, for* $a > 0, b \neq 0, s > 0$

$$\int_0^s \frac{a}{\sqrt{2\pi t^3}} \exp[-\frac{(a-bt)^2}{2t}]\, dt = Q(\frac{a-bs}{\sqrt{s}}) + \exp(2ab) Q(\frac{a+bs}{\sqrt{s}}).$$

Acknowledgments

The authors would like to thank A. Nilim for his help with the simulations, and to acknowledge C. Tomlin, G. Pappas and J. Kosecka for the stimulating discussions. Finally, the authors thank the anonymous reviewers for their useful comments and suggestions.

References

[1] C. Tomlin, *Hybrid control of air traffic management system*, Ph.D. thesis, University of California, Berkeley, 1998.

[2] M.S. Nolan, *Fundamentals of Air Traffic Control*, Brooks/Cole, Wadsworth, 3rd edition, 1998.

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

[4] P.B.M Vranas, D. Bertsimas, and A.R. Odoni, "Dynamic gound-holding policies for a network of airports," *Transportation Science*, vol. 28, pp. 275–291, 1994.

[5] Christos G. Panayiotou and Christos G. Cassandras, "A sample path approach for solving the ground-holding policy problem in air traffic control," in *Proc. 38th IEEE Conference on Decision and Control*, Phoenix, AZ, 1999.

[6] Heinz Erzberger, Thomas J. Davis, and Steven Green, "Design of center-tracon automation system," in *Proceedings of the AGARD Guidance and Control Syposium on Machine Intelligence in Air Traffic Management*, 1993, pp. 11.1–11.12.

[7] R.A. Paielli and H.Erzberger, "Conflict probability estimation for free flight," *Journal of Guidance, Control and Dynamics*, vol. 20, no. 3, pp. 588–596, 1997.

[8] H. Erzberger, R.A. Paielli, D.R. Isaacson, and M.M. Eshow, "Conflict detection and resolution in the presence of prediction error," in *1st USA/Europe Air Traffic Manegement R & D Seminar*, 1997.

[9] D. Brudnicki and A. McFarland, "User Request Evaluation Tool (URET) conflict probe performance and benefits assessment," in *1st USA/Europe Air Traffic Manegement R & D Seminar*, 1997.

[10] L. Yang and J. Kuchar, "Prototype conflict alerting logic for free flight," in *Proc. 35th AIAA Airspace Science Meeting & Exhibit, AIAA 97-0220*, January 1997.

[11] C. Tomlin, G.J. Pappas, and S. Sastry, "Conflict resolution for air traffic management: a study in multi-agent hybrid systems," *IEEE Trans. on Automatic Control*, vol. 43, pp. 509–521, 1998.

[12] Radio Technical Commission for Aeronautics, "Minimum operational performance standards for traffic alert and collision avoidance system (TCAS) airborn equipment," Tech. Rep. RTCA/DO-185, RTCA, September 1990, Consolidated Edition.

[13] J. Lygeros and N. Lynch, "On the formal verification of the TCAS conflict resolution algorithms," in *Proc. 36th IEEE Conf. on Decision and Control, San Diego*, 1997, pp. 1829–1834.

[14] J. Kuchar and L.C. Yang, "Survey of conflict detection and resolution modeling methods," in *AIAA Guidance, Navigation, and Control Conf.*, New Orleans, LA, 1997.

[15] M.G. Ballin and H. Erzberger, "An analysis of landing rates and separations at Dallas/Ft. Worth Airport," Tech. Rep., NASA Technical Memorandum, TM 110397, July 1996.

[16] R.A. Paielli, "Empirical test of conflict probability estimation," in *2st USA/Europe Air Traffic Manegement R & D Seminar*, 1998.

[17] C. Tomlin, J. Lygeros, and S. Sastry, "Controller design for hybrid systems," *Proceedings of the IEEE*, 2000, To Appear.

[18] L.C. Yang and J. Kuchar, "Using intent information in probabilistic conflict analysis," in *AIAA Guidance, Navigation, and Control Conf.*, Boston, MA, 1998.

[19] Radio Technical Commission for Aeronautics, "Minimum aviation system performance standards for automatic dependent surveillance-broadcast (ADS-B)," *Technical report*, vol. RTCA-186, 1997, DRAFT 4.0.

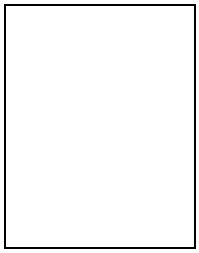
[20] J.K. Kuchar, *A Unified Methodology for the Evaluation of Hazard Alerting Systems*, Ph.D. thesis, Massacusetts Institute of Technology, 1995.

[21] M. Prandini, J. Lygeros, A. Nilim, and S. Sastry, "A probabilistic framework for aircraft conflict detection," in *AIAA Guidance, Navigation and Control Conf.*, Portland, OR, 1999.

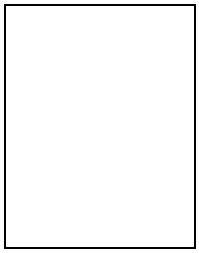
[22] M. Prandini, J. Lygeros, A. Nilim, and S. Sastry, "Randomized algorithms for probabilistic aircraft conflict detection," in *IEEE Conference on Decision and Control*, Pheonix, AZ, 1999.

[23] G. Chen, G. Chen, and S. Hsu, *Linear stochastic control systems*, CRC Press, Inc., 1995.

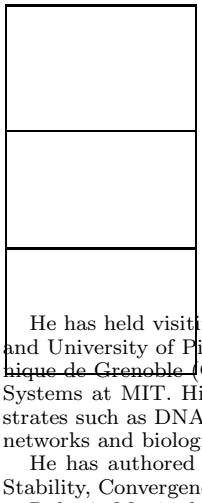
[24] M. Vidyasagar, *A theory of learning and generalization: with applications to neural networks and control systems*, Springer-Verlag, London, 1997.
[25] M. Vidyasagar, "Statistical learning theory and its applications to randomized algorithms for robust controller synthesis," in *Plenary Lectures and Mini-Courses at the Europ. Control Conf.*, July 1997, pp. 161–189.
[26] M. Prandini, *Adaptive LQG control: optimality analysis and robust controller design*, Ph.D. thesis, University of Brescia, 1998.
[27] M.C. Campi and M. Prandini, "Randomized algorithms for the synthesis of adaptive controllers," in *MTNS 98 Conf.*, Padova, Italy, 1998.
[28] R.A. Paielli and H. Erzberger, "Conflict probability estimation generalized to non-level flight," *Air Traffic Control Quarterly*, vol. 7, no. 3, pp. 195–222, 1999.
[29] James K. Kuchar, "Methodology for alerting-system performance evaluation," *Journal of Guidance, Control and Dynamics*, vol. 19, no. 2, pp. 438–444, 1996.
[30] J. Hu, J. Lygeros, M. Prandini, and S. Sastry, "Aircraft conflict prediction and resolution using Brownian Motion," in *IEEE Conference on Decision and Control*, Pheonix, AZ, 1999.
[31] B. Oksendal, *Stochastic Differential Equations, an introduction with application. Fifth edition*, Springer-Verlag, 1998.
[32] J. Hu, J. Lygeros, M. Prandini, and S. Sastry, "A probabilistic framework for highway safety analysis," in *IEEE Conference on Decision and Control*, Pheonix, AZ, 1999.
[33] R. Durrett, *Probability: theory and examples, Second edition*, Duxbury Press, 1996.
[34] J. Hu, "A study of conflict detection and resolution in free flight," M.S. thesis, University of California, Berkeley, 1999.
[35] J. Krozel and M. Peters, "Strategic conflict detection and resolution for free flight," in *IEEE Conference on Decision and Control*, San Diego, 1996.
[36] Y. Zhao and R. Schultz, "Deterministic resolution of a two aircraft conflict in free flight," in *AIAA Guidance, Navigation, and Control Conf.*, New Orleans, LA, 1997.
[37] J. Krozel, T. Mueller, and G. Hunter, "Free flight conflict detection and resolution analysis," in *AIAA Guidance, Navigation and Control Conf.*, San Diego, CA, 1996.
[38] N. Durand, J.M. Alliot, and J. Noailles, "Automatic aircraft conflict resolution using genetic algorithms," in *11th Annual ACM conference on applied computing, ACM/SAC*, Philadelphia, 1996.
[39] E. Frazzoli, Z.H. Mao, J.H. Oh, and E. Feron, "Resolution of conflicts involving many aircraft via semidefinite programming," AIAA Journal of Guidance, Control and Dynamics, to appear.
[40] P.K. Menon, G.D. Sweriduk, and B. Sridhar, "Optimal strategies for free-flight air traffic conflict resolution," *AIAA Journal of Guidance, Control and Dynamic*, vol. 22, no. 2, pp. 202–211, 1999.
[41] J. Hu, M. Prandini, and S. Sastry, "Optimal maneuver for multiple aircraft conflict resolution: a braid point of view," in *IEEE Conference on Decision and Control*, Sidney, Australia, 2000, To appear.
[42] J. Kosecka, C. Tomlin, G. Pappas, and S. Sastry, "Generation of Conflict Resolution Maneuvers For Air Traffic Management," in *IEEE Conference on Intelligent Robotics and System '97*, 1997.



Maria Prandini received the Laurea degree in electrical engineering from the Politecnico of Milano, Italy, in 1994, and the PhD degree in electrical engineering from the University of Brescia, Italy, in 1998. From March to July 1998, she was a visiting scholar at the Delft University of Technology, the Netherlands. Since August 1998, she has been a visiting post-doctoral researcher at the Electrical Engineering and Computer Sciences Department, University of California, Berkeley. She is currently a research fellow at the University of Brescia, Italy. Her research interests include identification and adaptive control of stochastic systems, air traffic management, probabilistic pursuit-evasion games, and verification of hybrid systems.



Jianghai Hu received the B.E. degree in automatic control from Xi'an Jiaotong University, P.R. China, in 1994. In 1999, he received the M.S. degree in electrical engineering from the University of California at Berkeley, where he is currently pursuing the Ph.D. degree in electrical engineering. His research interests include stochastic and optimal control of hybrid systems, air traffic management, and geometric control.



John Lygeros

Shankar Sastry received his Ph.D. degree in 1981 from the University of California, Berkeley. He was on the faculty of MIT as Asst. Professor from 1980-82 and Harvard University as a chaired Gordon Mc Kay professor in 1994. He is currently a Professor of Electrical Engineering and Computer Sciences and a Professor of Bioengineering. He was the Director of the Electronics Research Laboratory at Berkeley an organized research unit on the Berkeley campus conducting research in computer science and all aspects of electrical engineering. During his Directorship from 1996-99 the laboratory grew from $29Mto$ 50 M in volume of extra-mural funding. He is currently on leave from Berkeley as Director of the Information Technology Office at DARPA, Washington DC.

He has held visiting appointments at the Australian National University, Canberra the University of Rome, Scuola Normale and University of Pisa, the CNRS laboratory LAAS in Toulouse (poste rouge), Professor Invite at Institut National Polytechnique de Grenoble (CNRS laboratory VERIMAG), and as a Vinton Hayes Visiting fellow at the Center for Intelligent Control Systems at MIT. His areas of research are embedded and autonomous software, computer vision, computation in novel substrates such as DNA, nonlinear and adaptive control, robotic telesurgery, control of hybrid systems, embedded systems, sensor networks and biological motor control.

He has authored over 250 research papers. In terms of books, he is a coauthor (with M. Bodson) of "Adaptive Control: Stability, Convergence and Robustness, Prentice Hall, 1989." and (with R. Murray and Z. Li) of "A Mathematical Introduction to Robotic Manipulation, CRC Press, 1994". His solo book "Nonlinear Systems: Analysis, Stability and Control" has just been published by Springer-Verlag in 1999. He has co-edited " Hybrid Control II" and "Hybrid Control IV" and "Hybrid Control V", with P. Antsaklis, A. Nerode, and W. Kohn, Springer Lecture Notes in Computer Science, 1995, 1997, and 1999, respectively and co edited "Hybrid Systems Computation and Control" Springer-Verlag Lecture Notes in Computer Science with Henzinger in 1998 and "Essays in Mathematical Robotics", Springer-Verlag IMA Series with Baillieul and Sussmann. Books on Embedded Software and Structure from Motion in Computer Vision are in progress.

Prof. Sastry was an Associate Editor of the IEEE Transactions on Automatic Control, IEEE Control Magazine, IEEE Transactions on Circuits and Systems, and the Journal of Mathematical Systems, Estimation and Control and is an Associate Editor of the IMA Journal of Control and Information, the International Journal of Adaptive Control and Signal Processing and the Journal of Biomimetic Systems and Materials. He received the President of India Gold Medal in 1977, the IBM Faculty Development award for 1983-1985, the NSF Presidential Young Investigator Award in 1985 and the Eckman Award of the of the American Automatic Control Council in 1990, an M. A. (honoris causa) from Harvard in 1994, Fellow of the IEEE in 1994, the distinguished Alumnus Award of the Indian Institute of Technology in 1999, and the David Marr prize for the best paper at the International Conference in Computer Vision in 1999.

He has supervised 34 doctoral students to completion and over 50 MS students. His students now occupy leadership roles in several locations such as Dean of Engineering at Caltech, Director of Information Systems Laboratory, Stanford, Army Research Office, and on the faculties of every major university in the United States and abroad.

Caption of figure 1:
A sketch of the current ATMS structure organized in Air Route Traffic Control Centers (ARTCCs), Special Use Airspace (SUA) areas, TRACONs facilities, with the aircraft flying along VHF Omni-Directional Range (VOR) jetways and entering the TRACON through gates.
Caption of figure 2:
Mid-range prediction model for the aircraft motion.
Caption of figure 3:
Left: Sample pair of simulated trajectories. The $\star$ stand for the starting points. Right: SOC curves for Algorithm 4 (solid) & Erzberger algorithm (dashed). The $\circ$ stand for the "optimal" threshold points (Example 1: $\epsilon = 0.05$; $\delta = 0.1$; $\beta = 0.05$; 1000 simulations).
Caption of figure 4:
The $\star$ stand for the starting points. Right: SOC curves for Algorithm 4 (solid) & Erzberger algorithm (dashed). The $\circ$ stand for the "optimal" threshold points (Example 2: $\epsilon = 0.05$; $\delta = 0.1$; $\beta = 0.05$; 1000 simulations).
Caption of figure 5:
Plots of P(FA) and P(SA) vs threshold for our algorithm (solid) & Erzberger algorithm (dashed) (Example 1 with constant cross-track variance: $\epsilon = 0.05$; $\delta = 0.1$; $\beta = 0.05$; 1000 simulations).
Caption of figure 6:
Encounter situation for two aircraft flying at the same altitude.
Caption of figure 7:
Transformed protection zone.
Caption of figure 8:
P^0 for path angles $\vartheta = 0^\circ$ (upper left); $\vartheta = 45^\circ$ (upper right); $\vartheta = 90^\circ$ (lower left); $\vartheta = 180^\circ$ (lower right) with $\|u^A\| = 7$ nmi/min, $\|u^B\| = 8$ nmi/min, $\nu_a = 2$, ν_c=1.
Caption of figure 9:
$P^0_{t_f}$ for $t_f = 10$ min (top); $t_f = 15$ min (bottom) with $\vartheta = 90^\circ$, $\|u^A\| = 7$ nmi/min, $\|u^B\| = 8$ nmi/min,$\nu_a = 2$, $\nu_c = 1$.
Caption of figure 10:
Resolution maneuver for two aircraft encounter situations ($\nu_a = 0.35, \nu_c = 0.2$, $\beta = \pi/25$).
Caption of figure 11:
Resolution for three aircraft encounter situations ($\nu_a = 0.35, \nu_c = 0.2$, $\beta = \pi/25$).
Caption of figure 12:
Resolution for eight aircraft encounter situations ($\nu_a = 0.35, \nu_c = 0.2$, $\beta = \pi/25$, $\rho = 10$ nmi).