A Probabilistic Approach to Aircraft Conflict Detection

Maria Prandini, Jianghai Hu, John Lygeros, and Shankar Sastry, *Fellow, IEEE*

Abstract—Conflict detection and resolution schemes operating at the mid-range and short-range level of the air traffic management process are discussed. Probabilistic models for predicting the aircraft position in the near-term and mid-term future are developed. Based on the mid-term prediction model, the maximum instantaneous probability of conflict is proposed as a criticality measure for two aircraft encounters. Randomized algorithms are introduced to efficiently estimate this measure of criticality and provide quantitative bounds on the level of approximation introduced. For short-term detection, approximate closed-form analytical expressions for the probability of conflict are obtained, using the short-term prediction model. Based on these expressions, an algorithm for decentralized conflict detection and resolution that generalizes potential fields methods for path planning to a probabilistic dynamic environment is proposed. The algorithms are validated using Monte Carlo simulations.

Index Terms—Air traffic management automation, conflict detection, probabilistic modeling, randomized algorithms.

I. INTRODUCTION

DESPITE technological advances such as powerful on-board computers, advanced flight management systems (FMS) and positioning and communication systems such as the global positioning system (GPS) and automatic dependence surveillance-broadcast (ADS-B), the current air traffic management system (ATMS) is still, to a large extent, based on:

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

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 USA (e-mail: jianghai@robotics.eecs.berkeley.edu; sastry@robotics.eecs.berkeley.edu).

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

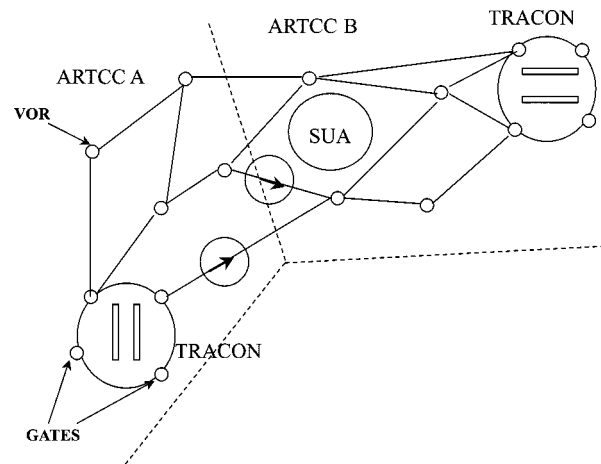Publisher Item Identifier S 1524-9050(00)11406-1.

Fig. 1. A sketch of the current ATMS structure organized in air route traffic control centers (ARTCCs), special use airspace (SUA) areas, TRACON facilities, with the aircraft flying along very high frequency (VHF) omni-directional range (VOR) jetways and entering the TRACON through gates.

For a more detailed account of the current ATMS practice (see, for example, [1] and [2]).

The increasing demand for air travel is stressing the current ATMS. This is likely to cause both safety and performance degradation in the near future. It is believed that by increasing the level of automation, the efficiency of the ATMS can be improved and the tasks of human operators can be simplified. This will allow them to handle the increased demand in air traffic in a more reliable way, possibly enhancing the level of safety over the current system.

The primary concern of all advanced ATMS is to guarantee safety. Safety is typically quantified in terms of numbers of conflicts, i.e., situations where two aircraft come closer than a certain distance to 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 nmi, while inside the terminal radar approach control (TRACON) area it is reduced to 3 nmi. The minimum vertical separation is 2000 ft above the altitude of 29 000 ft (FL290), and 1000 ft below FL290.

The prevention of conflicts involves 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. A potential conflict is declared if the predicted positions of any two aircraft violate the safe separation requirements at some time in the future. In this case, the trajectories of the aircraft involved in the conflict are replanned in the *conflict resolution* stage.

Conflict detection and resolution is performed at three different levels of the ATM process.

1) Long Range: Some form of conflict prediction and resolution is carried out at the level of the entire national airspace system (NAS), over a time horizon of several hours [3]–[5]. It involves composing flight plans and airline schedules (on a daily basis, for example) to ensure that airport and sector capacities are not exceeded. This is typically accomplished using large-scale integer and linear programming techniques.

2) Mid-Range: Conflict prediction and resolution is carried out by ATCs, over horizons of the order of tens of minutes. It involves modifying the 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]), which is based on the algorithms of [7], [8], operates within the terminal area and the en-route airspace around it. On the other hand, the conflict probe based on the user request evaluation tool (URET, [9]), operates within the en-route airspace. The algorithms and techniques of [10], [11] can also be used at this level.

3) Short Range: Conflict prediction and resolution is also carried out on board the aircraft by the FMS, 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] and [11] also belong to this category.

An excellent survey of the different conflict detection and resolution schemes found in the literature is presented in [14], where the schemes are classified according to the modeling method used for projecting the aircraft position in the future.

One of the difficulties in predicting aircraft positions is modeling the perturbations influencing their motion. The future motion of the aircraft is affected by uncertainty, due primarily to wind, but also to errors in tracking, navigation, and control. However, the resultant deviation from 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 a 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. This avoids the conservativeness of the worst-case approach (used in [11], [17], for example), but is still robust with respect to uncertainty. The main issues are then building a simple but realistic probabilistic prediction model and computing the probability of conflict based on that model.

In [7], [8], and [10], a probabilistic description of the global effect of the perturbations entering the aircraft motion is proposed. The tracking errors are described by 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 $s(t)$ until it saturates at a fixed value $\overline{\sigma}_c^2$

$$\sigma_c^2(t) \sim \min\{r_c^2 s^2(t), \overline{\sigma}_c^2\} \tag{2}$$

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 of the order of 20 min, 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 the above probabilistic description of the aircraft motion, two main methods have been proposed for computing the probability of conflict for two aircraft encounters. In [7], a closed-form analytic expression for the probability of conflict for level flight is derived, assuming that the two aircraft fly 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 is unclear when these simplifying assumptions are not satisfied. In [10] and [18], Monte Carlo simulation is used to compute the probability of conflict. This approach does not require particular assumptions and can be applied to many different scenarios, but is computationally intensive and therefore may not be suitable for on-line implementation (though promising attempts have been reported in the literature [18]).

In this paper, we introduce two probabilistic prediction models, one for mid-range and one for short-range conflict detection. Both are inspired by the empirically motivated probabilistic model described above. The two models are intended to work together at different levels of the ATMS. The mid-range model is meant to provide centralized conflict information to the ATC, in a role similar to that of CTAS in the current ATMS. The short-range model on the other hand operates at a "lower level" to provide decentralized advisories to the pilots, in a role similar to the one currently played by TCAS. For simplicity, we deal with the level flight case. The extension of the proposed approaches to the nonlevel flight case is straightforward as remarked in Section IV.

The prediction model developed for mid-range conflict detection Section II, is simple and realistic, but does not allow the derivation of a closed-form expression for the probability of conflict. Based on this model, we propose a different measure of criticality for two aircraft encounters, give a procedure for estimating it, and quantify the level of accuracy achieved. An algorithm for mid-term conflict detection is then proposed and its performance is compared to that of the algorithm in [7] by Monte Carlo simulation. The simulations are based on a more detailed, stochastic, ordinary differential equation model for the motion of the aircraft. They suggest that the proposed mid-term conflict detection algorithm has superior performance to the one of [7], at the cost of increased computational load. However, we show that, for the same level of approximation, the load in computing our criticality measure does not significantly increase when we go from the two-dimensional (2-D) to the three-dimensional (3-D) case; in contrast, the computational load for

the algorithm of [7] is likely to increase substantially, since a closed-form expression like the one used in [7] for level flight is not available for nonlevel flight. Further work to evaluate the performance of our approach in terms of computational load as a function of the level of approximation introduced is currently underway.

In Section III, we propose a model for short term prediction based on an approximation of the mid-term prediction model. The aircraft motion is modeled as a deterministic motion plus a (scaled) Brownian motion perturbation whose variance grows linearly with time, instead of quadratically. 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, thus allowing for very 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 short-term 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) "free flight" initiative [19], an attempt to increase the ATMS flexibility by allowing aircraft to fly more user preferred routes from origin to destination. 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 section, 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] and [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 one aircraft will enter the protected zone around the other one. Conflict detection consists of estimating this probability and initiating 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 a 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 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. It should be noted, however, that 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 models on which we base our detection scheme and the validation. In Section II-B, we describe the detection scheme, explain how we deal with the computational issues and formulate a randomized algorithm for its implementation. Simulation results are reported in Section II-C.

A. Probabilistic Models

We restrict our attention to the level flight case. This is primarily for ease of exposition, since the generalization to the 3-D case is straightforward, the only added difficulty being the more complex notation [22]. We assume that the trajectory specifications the aircraft receives from the ATC are in terms of a sequence of way points $\{P_j\}_{j=0,...,n}, P_j \in \mathbb{R}^2$, and a sequence of speeds, $\{v_j\}_{j=1,...,n}, v_j \in \mathbb{R}_+$, for moving between consecutive way points. These two sequences constitute the *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 paper, we ignore errors in the radar measurements. This assumption can be justified by noticing that uncertainty in the initial position quickly becomes dominated by the perturbation to the aircraft motion as time goes on [7], [10]. A model including radar measurement errors is proposed in [21].

1) 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. The *nominal arrival times*
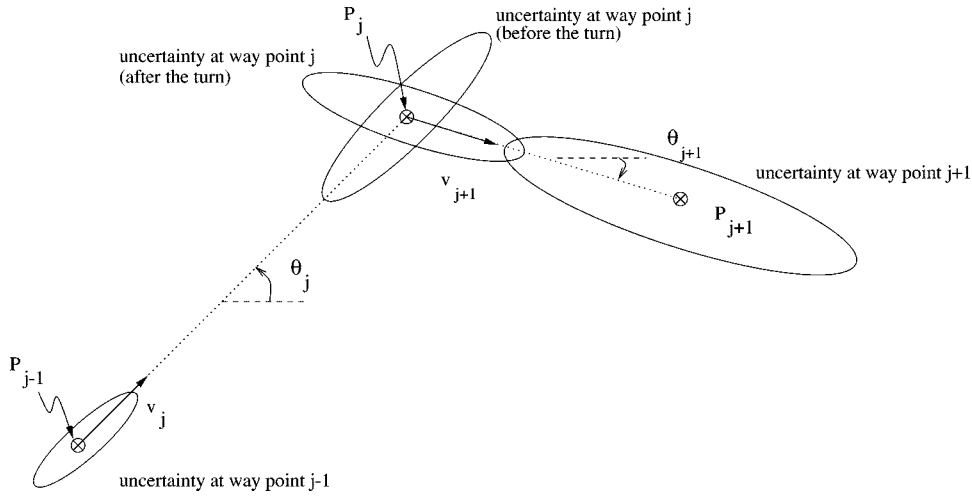
Fig. 2.   Mid-range prediction model for the aircraft motion.

$\{T_j\}_{j=0, \ldots, n}$ of the aircraft at the way points can be recursively computed by

$$T_j = \frac{\|P_j - P_{j-1}\|}{v_j} + T_{j-1} \qquad (3)$$

for $j = 1, \ldots, n$, 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 $p(t) = p(T_{j-1}) + v_j(t - T_{j-1})(P_j - p(T_{j-1}))/\|P_j - p(T_{j-1})\|$, $j = 1, \ldots, n$, initialized with $s(T_0) = 0$ and $p(T_0) = P_0$.

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

$$m(t) = p(t). \qquad (4)$$

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 project into the future. Following [7] and [8], we assume that 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 (1) and (2), respectively, and that the along-track and cross-track error components are independent.

Denote by 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_js are given. Then, the covariance matrix $V(t)$ for $t \in [T_{j-1}, T_j)$ is given by

$$V(t) = R(\vartheta_j)\overline{V}(t)R(\vartheta_j)^T \qquad (5)$$

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

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

In Fig. 2, the prediction model for the aircraft motion is drawn with the ellipses representing equiprobability 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 precisely, 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. The assumption of instantaneous turns introduces, in general, inaccuracy in the evaluation of the probability of conflict near the way points. 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.

2) 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 above for predicting the position of an aircraft in the future is simple and allows fast computations, which makes it ideal for on-line conflict detection. However, it has 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 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.

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 $x = (x_1, x_2)$ denote its position with respect to a global coordinate frame. Assume its velocity has magnitude v and makes an angle ϑ with respect to the x_1 axis. Consider a body coordinate frame $\chi = (\chi_1, \chi_2)$ with χ_1 aligned with the aircraft velocity and χ_2 perpendicular to it. The two frames are related through a coordinate transformation

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

where $R(\vartheta)$ is the rotation matrix of (6) and p denotes the position of the origin of the body frame with respect to the inertial frame.

In light of the above discussion, (7) can 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 in the along-track and cross-track directions.

We assume that the perturbation affecting the aircraft motion is generated through the stochastic ordinary differential equation (ODE)

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

where $\chi(0) \sim \mathcal{N}(0, V_\chi(0))$ and $\eta := (\eta_1, \eta_2)$ is a white Gaussian noise process independent of $\chi(0)$, with zero mean and covariance $V_\eta(t)$. We shall show below 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 mimics the behavior of (1) and (2).

By differentiating (7) and using (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

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

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

The validation model can be thought of as a *stochastic hybrid dynamical system*, with discrete events *turn* occurring at the way points. The effect of *turn* is to update 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.

3) 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. The tuning is based on the following proposition, whose proof is omitted since it easily follows from standard results for stochastic ODEs [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 $\eta(t) \sim \mathcal{N}(0, \text{diag}(V_{\eta_1}(t), V_{\eta_2}(t))), t \in [0, T]$, independent of $\chi(0) \sim \mathcal{N}(0, \text{diag}(V_{\chi_1}(0), V_{\chi_2}(0)))$. Set

$$V_{\chi_1}(0) = V_{\chi_2}(0) = 0, \quad V_{\eta_1}(t) = 2r_a^2 t, \quad V_{\eta_2}(t) = 2a\overline{\sigma}_c^2$$

and

$$A = \begin{pmatrix} 0 & 0 \\ 0 & -a \end{pmatrix}, \quad \text{where} \quad a = \frac{r_c}{\overline{\sigma}_c}v_1. \tag{10}$$

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

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

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

By tuning the parameters in (8) according to (10), we can make $\chi(\cdot)$ resemble the empirically observed characteristics of the along-track and cross-track errors. In fact, $\text{var}[\chi_1(t)]$ in (11) has the same growth rate as the along-track variance $\sigma_a^2(t)$ in (1) and $\text{var}[\chi_2(t)]$ in (12) monotonically increases to the saturation value $\overline{\sigma}_c^2$ as the cross-track variance $\sigma_c^2(t)$ in (2). The time constant of the decaying exponential in (12) is equal to 1/2 of the time $\overline{t} = \overline{\sigma}_c/r_c v_1$ required for $\sigma_c^2(t)$ to reach the saturation value $\overline{\sigma}_c^2$ in the case when the aircraft trajectory is a straight line traveled with speed v_1. For $r_c = 1/57$ and $\overline{\sigma}_c = 1$ nmi (values estimated on real traffic data [15], [16]) and for $v_1 = 500$ nmi/h (typical cruising speed) this time is about 6 min. Therefore, by setting $a = (r_c/\overline{\sigma}_c)v_1$, we get that $\text{var}[\chi_2(t)]$ reaches 86% of the saturation value in about 6 min.

B. Conflict Detection Algorithm

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

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

We assume that each aircraft, say i, has been assigned a 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 can be described through the *configuration* γ of the N_a aircraft system, which consists of the flight plans of all aircraft, i.e., $\gamma = \{\{P_j^i\}_{j=0, \ldots, n_i}, \{v_j^i\}_{j=1, \ldots, n_i}\}_{i=1, \ldots, N_a}$.

Conflict detection involves extracting some measure $C(\gamma)$ of how critical the current configuration γ is, comparing this measure to a threshold $\overline{C}$ 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, or every time the ATC changes a flight plan. In pseudocode, this can be encoded by the following.

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

The rest of this section deals with the implementation of this piece of pseudocode. 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 $\overline{C}$. We first introduce a probabilistic measure of criticality for a two aircraft encounter based on the prediction model presented in Section II-A. We then present an algorithm for computing it, making use of stochastic estimation algorithms. We postpone 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 stochastic ODE model of Section II-A.

1) 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

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

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, i.e.,

$$C(\gamma) := \max_{t \in [0, T]} PC(t). \qquad (15)$$

The time horizon T is chosen to be equal to 20 min. This is because the computation of $PC(\cdot)$ is based on the prediction model introduced in Section II-A, which is valid over a 20 min horizon [7]. As part of this work, other measures of criticality were also considered, most notably, different weighted averages of $PC(t)$ over the horizon $[0, T]$, including the average value. These measures were found to be less effective, since for most conflicts PC tends to exhibit a sharp "spike" near the point of conflict and is fairly low elsewhere.

Based on the prediction model in Section II-A, we consider the case when the two aircraft positions $x^A(t)$ and $x^B(t)$ are Gaussian random variables,[1] $x^i(t) \sim \mathcal{N}(m^i(t), V^i(t))$, $i = A, B$, with $m^i(t)$ and $V^i(t)$ given by (4) and (5). If we further assume that x^A and x^B 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 $Q(t) := V^A(t) + V^B(t)$.

---

[1]In principle, this algorithm can be used for any probability distribution of the aircraft positions, the main problem being integrating the resultant distribution for their separation $d(t)$ over the protected zone $\mathcal{C}$.

It is important to note that the assumption that the positions of the two aircraft are uncorrelated, though commonly used [8], [10], is rather unrealistic. The tracking error is 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. Modeling the tracking error correlation properties is not dealt with in this paper, though it obviously needs to be addressed in future work.

2) 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 $PC(\cdot)$; even the computation of $PC(t)$ for a given t is time consuming. This turns out to be a major obstacle for on-line implementation, where computation is subject to time constraints. To solve this problem we introduce an algorithm for computing an approximate solution to the optimization problem and set up a methodology to provide a quantitative estimate of the level of approximation introduced. Our results build 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 to compute $C(\gamma)$.

Algorithm 2 [**Randomized Estimate of** $C(\gamma)$]
Initialization: Choose an integer N and
  set $C'(\gamma) = 0$
for $i = 1, \cdots, N$
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_is. Nevertheless, a quantitative statement showing that C' is a good approximation in a probabilistic sense can be proved 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 $\overline{f} := \max_{z \in \{z_i\}_{i=1}^N} f(z)$. Fix an arbitrary real number $\beta \in (0, 1)$. Then

$$\mathcal{P}\{z \in Z: f(z) > \overline{f}\} \leq \beta \qquad (16)$$

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

    Proof: See Lemma 11.1 in [24].   □

In the previous statement, $\overline{f}$ 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, $\mathcal{P}\{z \in Z: f(z) > \overline{f}\} \leq \beta$ may or may not be satisfied depending on the random extraction and,

therefore, $\mathcal{P}\{z \in Z: f(z) > \overline{f}\} \leq \beta$ 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 (16) is the following. Suppose that an opponent would like to determine a $\overline{z}$ such that $f(\overline{z}) > \overline{f}$ and she uses the same probabilistic strategy as in the randomized algorithm: she randomly selects a $\overline{z}$ in Z according to $\mathcal{P}$. Then, her probability of "beating" $\overline{f}$ 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, (16) 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 zero, $1 - (1 - \beta)^N$ tends to zero and so the statement becomes evanescent. In addition, we observe that statement (16) quantifies the probability that one can improve result $\overline{f}$ by randomly selecting a new parameter $\overline{z}$, but says nothing about 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) $PC(t)$ of $PC(t)$ loses smoothness in the set of instants $\{t \in \{T_j^A\}_{j=0, \dots, n_1} \cup \{T_j^B\}_{j=0, \dots, n_2} \cup \{T_*^A, T_*^B\}$ with $t \leq T\}$, where T_j^i is the nominal time of arrival of aircraft i at its way point j and T_*^i is the time instant when the cross-track error variance of aircraft i saturates. Since this set is finite, Theorem 1 can be applied to $f(z) = PC(z)$, $Z = [0, T]$, $\mathcal{F}$ the Borel σ-algebra on $[0, T]$ and $\mathcal{P}$ the uniform probability distribution. Denote by $\lceil c \rceil$ the smallest integer greater than c. Then, we have the following.

Assertion 1: Fix $\delta \in (0, 1)$, $\beta \in (0, 1)$ and set $N := \lceil \ln(\delta) / (\ln(1 - \beta)) \rceil$. 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 a finite set of time instants $\{t_1, t_2, \dots, t_N\}$. 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, we can also treat $PC(t)$ 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 verify. 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 that w takes 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 $j = 1, \dots, M$ **do**
begin
  Extract at random $w_j \in \mathbb{R}^2$ according to
  the standard Gaussian probability density function $\mathcal{N}(0, I)$
  if $w_j \in \mathcal{C}_t$ **then** $PC'(t) = PC'(t) + 1$
end
$PC'(t) = PC'(t)/M$.

Again, $PC'(t)$ is a random approximation of $PC(t)$, due to the stochastic selection of the w_js. Nevertheless, a quantitative statement can be proved 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 $\mathcal{A} = \{A_1, \dots, A_N\} \subset \mathcal{F}$. Extract M independent samples $z_1, \dots, z_M$ from Z in accordance with $\mathcal{P}$ and define $\hat{\mathcal{P}}(A_i) := (1/M) \sum_{j=1}^{M} 1_{z_j \in A_i}$, $i = 1, \dots, N$, where $1_{z_j \in A_i} = 1$ if $z_j \in A_i$, 0, otherwise. Fix $\epsilon \in (0, 1)$. Then,

$$\mathcal{P}^M \left\{ (z_1, \dots, z_M) \in Z^M : \sup_{A \in \{A_i\}_{i=1}^N} |\hat{\mathcal{P}}(A) - \mathcal{P}(A)| > \epsilon \right\}$$
$$\leq 2N \exp(-2M\epsilon^2). \tag{17}$$

Proof: By the Chernoff bound (see [24]). $\qquad \square$

Equation (17) 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 $PC(\cdot)$ 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 $M = \lceil (1/2\epsilon^2) \ln(2N/\delta) \rceil$. Then, if the random extractions in Algorithm 3 are independent, $PC'(t)$ is a uniform approximation of $PC(t)$ to accuracy ϵ with confidence $1 - \delta$ over every finite set of time instants $\{t_1, \dots, t_N\}$, i.e., $\sup_{t \in \{t_i\}_{i=1}^N} |PC'(t) - PC(t)| \leq \epsilon$, with probability at least $1 - \delta$.

Combining the randomized estimation procedures for $C(\gamma)$ and $PC(t)$ leads to a fully randomized implementation of Algorithm 1.

Algorithm 4 (Randomized Conflict Detection)

Initialization: Fix $\epsilon \in (0, 1)$, $\beta \in (0, 1)$, $\delta \in (0, 1)$. Set

$$N = \left\lceil \frac{\ln\left(\frac{\delta}{2}\right)}{\ln(1 - \beta)} \right\rceil, \quad M = \left\lceil \frac{1}{2\epsilon^2} \ln \frac{4N}{\delta} \right\rceil$$

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
    $PC'(t_i) = PC'(t_i)/M$
    if $PC'(t_i) > \overline{C}$ declare a conflict
  end
end.

It is easy to see that the algorithm declares a conflict if and only if

$$C'(\gamma) := \max_{t \in \{t_i\}_{i=1}^N} \frac{1}{M} \sum_{j=1}^M 1_{\{L(t)w_j + \mu(t) \in \mathcal{C}\}} > \overline{C} \quad (18)$$

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]–[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

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

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 the t_is and w_js are independent of one another and the extraction $\{t_i\}_{i=1}^N$ is independent of the extraction $\{w_j\}_{j=1}^M$, we can conclude that the probability measure on

$[0, T]^N \times \mathbb{R}^{2M}$ 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

$$\mathcal{P}\left\{ t \in [0\ T] : PC(t) > \max_{t \in \{t_i\}_{i=1}^N} PC(t) \right\} \le \beta \quad (19)$$

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

$$q := \mathcal{Q}^M \left\{ (w_1, \ldots, w_M) \in \mathbb{R}^{2M} : \right.$$
$$\left. \sup_{t \in \{t_i\}_{i=1}^N} |PC'(t) - PC(t)| > \epsilon \right\}. \quad (20)$$

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 $1 - [(1 - \beta)^N + q]$

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

where the first inequality follows from (20) and the second from (19). 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 $q \le 2Ne^{-2M\epsilon^2}$ given by (17), we can easily conclude that using $N = \lceil (\ln(\delta/2))/(\ln(1 - \beta)) \rceil$ time instants t_i and $M = \lceil (1/2\epsilon^2) \ln(4N/\delta) \rceil$ vectors w_j suffices to approximately maximize $PC(t)$ over $[0, T]$ to accuracy 2ϵ and level β with confidence $1 - \delta$. $\square$

Note that the number of samples needed to achieve a certain approximation in terms of 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 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 the case if one resorts to numerical methods based on gridding or to approximate analytic methods such the one proposed in [8]. However, further work is required to fully evaluate the computational efficiency of our approach as a function of the accuracy, level, and confidence parameters.

C. Validation

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 with the one achieved by the algorithm in [8]. The algorithm in [8] is based on the same description of the uncertainty affecting the aircraft motion. However, the criticality measure adopted in [8] for a two aircraft encounter is an approximation of the probability of conflict. The approximation consists of computing the probability of conflict as if each aircraft was flying at a constant velocity and with constant tracking error statistics. The values for the mean and variance of the Gaussian random variables describing the tracking errors are set equal to the corresponding values at the point of minimum nominal separation (see [7] for more details). For the 2-D case the probability of conflict is further over-approximated by extending the protection zone by "sweeping" the 5 nmi disc to $\pm\infty$ in the direction of the relative velocity at the point of minimum nominal separation. This leads to an analytical approximation for the probability of conflict in the level flight case. For the 3-D case, one has to resort to a numerical approximation procedure [28].

The protocol we adopt for evaluating the performance of the detection algorithms consists of the following steps.

1) Given the flight plans of the two aircraft, generate pairs of aircraft trajectories over a 20-min time horizon according to a discretized version of the stochastic differential equation (9) (sample time set equal to 1 s), with $r_a = 0.25$ nmi/min, $r_c = 1/57$, and $\overline{\sigma}_c = 1$ 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 obtained by removing the way points which have been surpassed, setting the first way point equal to the current radar measurement and subtracting the elapsed time from the 20-min 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 of the number of alerts issued when there was no conflict over the total number of cases when there was no conflict [P(FA)] and the ratio of the number of alerts issued before a situation of conflict effectively happens over 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 the above definition of SA does not account for the fact that a conflict may be detected when it is too late to do anything about it. A more complete definition of SA should take into account the available conflict resolution strategies, the response times of ATC and pilots and the aircraft maneuverability constraints. In this paper, we simply require that a conflict should be detected at least 60 s before it occurs for a SA to be declared. A more thorough study of this issue will be undertaken once a resolution scheme has been developed.

It is evident that P(FA) and P(SA) are both decreasing functions of the threshold used to decide whether an alert should be issued. An ideal conflict detection scheme should operate at the point P(FA) = 0 and P(SA) = 1, where there are no false alarms and all the conflicts are detected. Unfortunately, a real conflict detection scheme cannot operate at this point due to the uncertainty affecting the aircraft positions. However, the more the SOC curve approaches the point $(0, 1)$, the better the performance of the system is likely to be. The threshold can be selected 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] and [29].

We now describe the results obtained by Monte Carlo simulations for 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 form a more complex zig-zag pattern. For these two cases, we draw the SOC curve and compare the performance obtained by Algorithm 4 with that obtained by implementing the detection algorithm described by Erzberger *et al.* [8]. The parameters r_a, r_c and $\overline{\sigma}_c$ are set equal to the empirical values $r_a = 0.25$ nmi/min, $r_c = 1/57$ and $\overline{\sigma}_c = 1$ nmi in both the prediction and simulation models.

Example 1 (30° path crossing angle configuration, Fig. 3): 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 Fig. 3 we have drawn a realization of the trajectories. For each pair of trajectories, the initial nominal minimum separation and time to minimum separation are, respectively, 5 nmi and 4 min. Fig. 3(a) shows the SOC curves of Algorithm 4 (solid line) and the algorithm of [8] (dashed line). The curves were obtained using 1000 pairs of trajectories generated by the stochastic ODE model. In this example, the "optimal" threshold $\overline{C}$, i.e., the one corresponding to the point of the SOC curve nearest to $(0, 1)$, and the "optimal" values for P(FA) and P(SA) are similar for the two algorithms [$\overline{C} \simeq 0.85$, P(FA) $\simeq 0.18$, P(SA) $\simeq 0.78$]. ☐

Example 2 (zig zag flight paths configuration, Fig. 4): 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, $v_2^A = 455$ nmi/h, $v_1^B = 460$ nmi/h and $v_2^B = 470$ nmi/h. The estimates of the probability of successful alert and false alarm obtained by 1000 Monte Carlo samples are used to plot the SOC curves in Fig. 4(a). Again the solid line corresponds to the detection algorithm proposed in this paper. The "optimal" threshold for Algorithm 4 is $\overline{C} = 0.62$, corresponding to P(FA) = 0.164 and P(SA) = 0.778 whereas the optimal threshold for the algorithm of [8] is $\overline{C} = 0.79$, corresponding to P(FA) = 0.264 and P(SA) = 0.727. 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 also that for each value of P(SA), the algorithm of [8] results in a higher value of P(FA). ☐

Summarizing, the two algorithms give similar results when the aircraft are flying along straight lines. When the flight plans are more complex, however, our algorithm generally performs better than the algorithm of [8]. This also turns out to be the
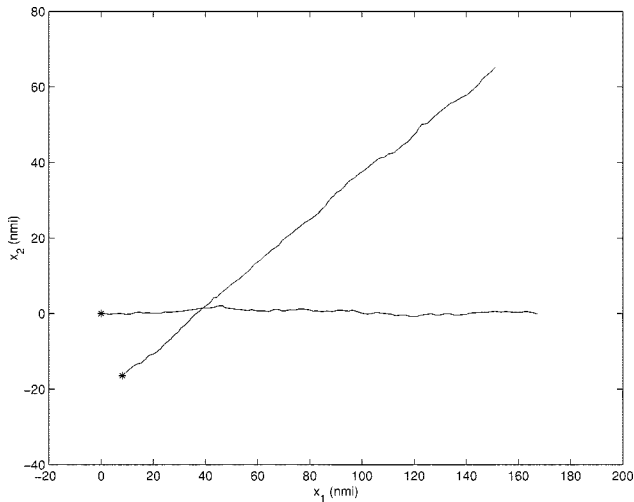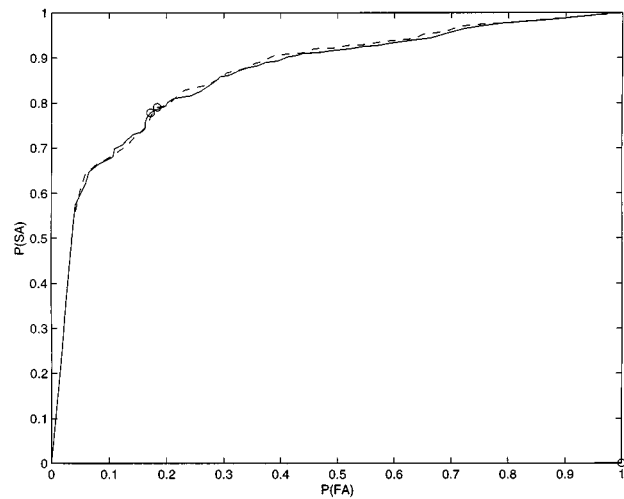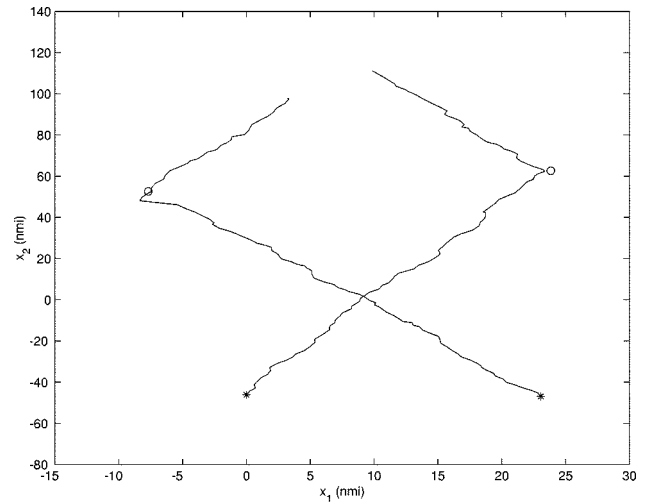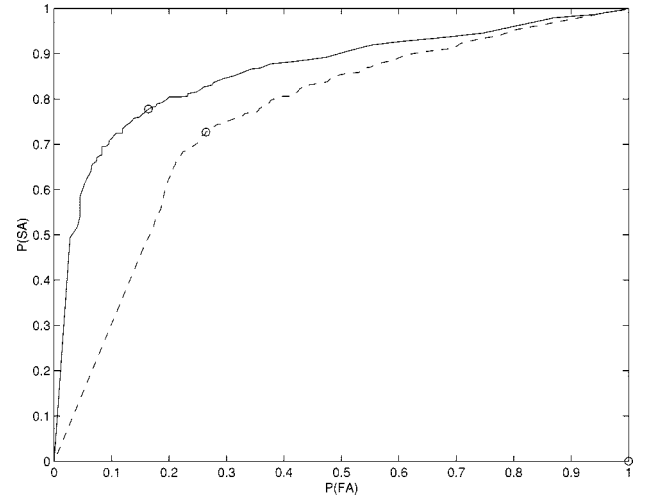
Fig. 3. (a) Sample pair of simulated trajectories. The $\star$ stand for the starting points. (b) SOC curves for Algorithm 4 (solid) and Erzberger algorithm (dashed). The $\circ$ stand for the "optimal" threshold points (Example 1: $\epsilon = 0.05$; $\delta = 0.1$; $\beta = 0.05$; 1000 simulations).

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

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]. Fig. 5 represents the plots of P(FA) and P(SA) as functions of the threshold obtained by 1000 Monte Carlo simulations for the encounter in Example 1, but considering a cross-track error with constant variance $\sigma_c^2 = \overline{\sigma}_c^2$ in the prediction models. Note that in this case, irrespective of the threshold, the P(FA) given by [8] is higher than the P(FA) given by our approach, and the increase is not compensated by an adequate increase in P(SA). Similar behavior is observed in Example 2.

In our opinion, an explanation for the results is that the measure of criticality used in [8] is an over approximation of the probability of conflict. In a sense, this criticality measure captures the probability of conflict for a fictitious *worst-case encounter*, which extends to infinity for both positive and negative times, with the aircraft flying with constant velocities and constant error statistics throughout. So, if the trajectories are not straight lines or if they are diverging and the current position is

uncertain, the probability of conflict computed in [8] may highly overestimate the probability of conflict of the real encounter. This results in a higher probability of false alarms.

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 to minimum distance.

III. SHORT-RANGE CONFLICT DETECTION

In this section, we adopt a sample path viewpoint to measure the criticality of an encounter. The approach is motivated by the fact that the motion of each aircraft involved in the encounter is subject to perturbations, whose overall effect over 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
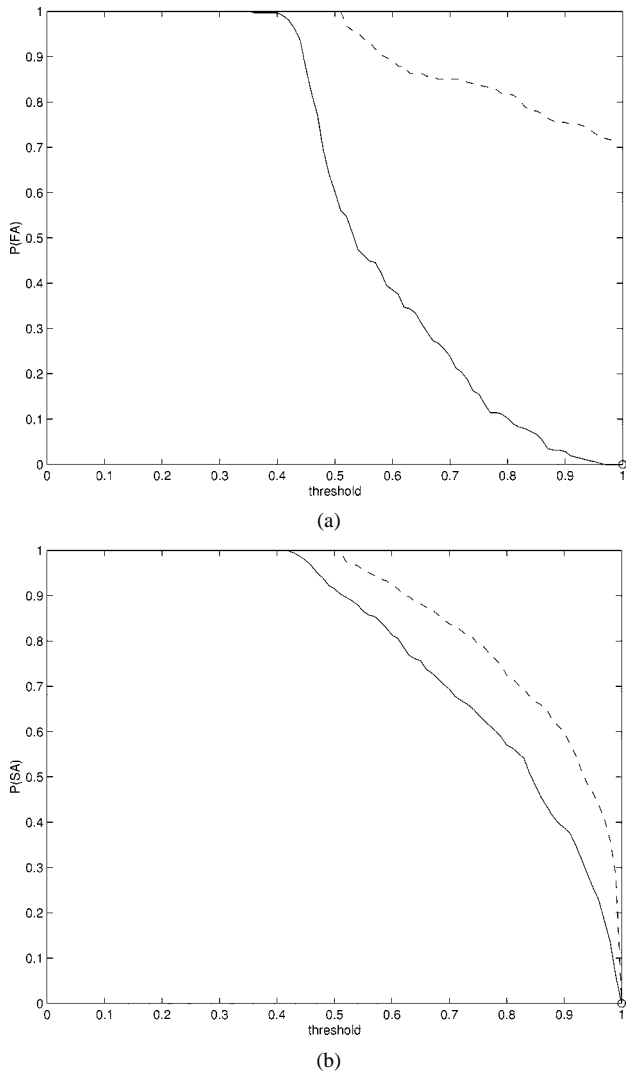
Fig. 5. (a) Plot of P(FA) and (b) plot of P(SA) versus threshold for our algorithm (solid) and Erzberger algorithm (dashed) (Example 1 with constant cross-track variance: $\epsilon = 0.05$; $\delta = 0.1$; $\beta = 0.05$; 1000 simulations). .

that a path of some aircraft enters the protected zone of another aircraft. The proposed approach is briefly explained next; the formal derivations are given in the following subsections.

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

$$\frac{ds}{dt}(t) = f(t) + w(t) \tag{21}$$

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 (21) 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 local wind effects such as air turbulence, as well as deviations due to mechanical and human factors [30]. Integrating (21), we have

$$s(t) = \int_0^t f(z)\, dz + b(t) \tag{22}$$

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 [31]. The continuity of $b(t)$ is actually required when (21) 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.

By subtracting the nominal motion $\int_0^t f(z)\, dz$ and scaling, we can reduce the perturbed motion in (22) 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 escape probability of a standard BM with respect to a time-varying region. In general, it is very difficult 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 (1-D) 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 toward the origin starting at position $a > 0$. Then, τ has probability density function

$$p_\tau(t) = \frac{a}{\sqrt{2\pi t^3}} \exp\left[-\frac{(a - \mu t)^2}{2t}\right], \qquad t \geq 0. \tag{23}$$

This lemma will prove very useful for estimating the probability of conflict in the more complex situations of real interest, which will be reduced to the case described in the lemma by appropriate approximations. In [32] a similar approach was used for highway safety analysis.

The rest of the section is organized as follows. We start by introducing 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 these approximations and validate the process by Monte Carlo simulations.

A. Prediction Model

Consider an aircraft flying with constant speed along a straight line on a plane. Let $x = (x_1, x_2)$ denote its position with respect to a global coordinate frame, $u = (u_1, u_2)$ denote the aircraft velocity, and ϑ denote the angle that vector u makes with the x_1 axis of the global frame. The aircraft motion can be represented by the stochastic kinematic model

$$x(t) = ut + R(\vartheta)\Sigma B(t) \tag{24}$$

where

$B(t)$ standard 2-D BM;

$R(\vartheta)$ rotation matrix of (6);

Σ diag(ν_a, ν_c), with 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.

Despite the nonlinear dependence on time in (1) and (2), over short-time horizons and under the constant velocity assumption, the variances of the along track and cross track perturbations can be approximated by linear functions of time. This implies that the proposed model may 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 [10], [15] and considering standard cruising speeds of commercial aircraft, e.g., $v = 480$ nmi/h, the growth rate of $\sigma_c(t)$ with time is given by $r_c v \simeq 0.14$ nmi/min, which is lower than the growth rate r_a of $\sigma_a(t)$. Hence, typically $\nu_a > \nu_c$. In all the examples ν_a and ν_c will be treated as a-dimensional 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.

The model of (24) also seems suitable for the free flight scenario envisioned for the future ATMS [19]. In this case, the saturation phenomenon, which in [7] is attributed to the pilots attempt to track a specific trajectory, may not be so significant. This is because in the free-flight paradigm, each aircraft receives advisories rather than mandatory trajectory specifications from the ATC. Moreover, under free-flight conflict, detection and resolution will be to a large extent carried out by individual aircraft, using only partial information about the intentions of the surrounding aircraft (primarily their current positions and headings). For this type of resolution, one cannot rely on the assumption 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 Fig. 6).

Denote with $x^A(t)$ and $x^B(t)$ the position of aircraft A and B, respectively. By the kinematic model (24), we get the following description for the two aircraft system:

$$\begin{cases} x^A(t) = u^A t + \Sigma B^A(t) \\ x^B(t) = \Delta x_0 + u^B t + R(\vartheta)\Sigma B^B(t) \end{cases} \qquad (25)$$

where $B^A(t)$ and $B^B(t)$ are standard 2-D BMs. As for the mid-range prediction model, we assume that $B^A(t)$ and $B^B(t)$ are independent and start at the origin (ignoring GPS and radar errors). Subtracting the first equation from the second in (25) leads to

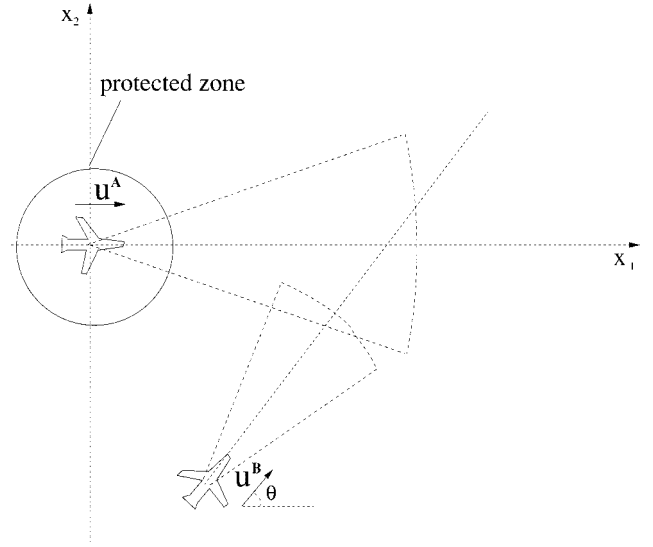$$\Delta x(t) = \Delta x_0 + \Delta u t - n(t) \qquad (26)$$

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

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 (26) 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. One can show that the Gaussian process $n(t)$ can be reduced to a standard 2-D BM by a coordinate transformation, similar to the one used in Proposition 2.

Proposition 3: Set $P := \sqrt{2} R(\vartheta/2)\Lambda$, with $\Lambda = $ diag(λ_1, λ_2), where

$$\begin{cases} \lambda_1 = \sqrt{\nu_a^2 \cos^2\left(\dfrac{\vartheta}{2}\right) + \nu_c^2 \sin^2\left(\dfrac{\vartheta}{2}\right)} \\ \\ \lambda_2 = \sqrt{\nu_a^2 \sin^2\left(\dfrac{\vartheta}{2}\right) + \nu_c^2 \cos^2\left(\dfrac{\vartheta}{2}\right)}. \end{cases} \qquad (27)$$

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

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

Equation (26) can now be rewritten as

$$\Delta s(t) = \Delta s_0 + u t - \overline{n}(t) \qquad (28)$$

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 $\overline{n}(t) = P^{-1}n(t)$ 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 Fig. 7). The boundary of the protected zone is given by

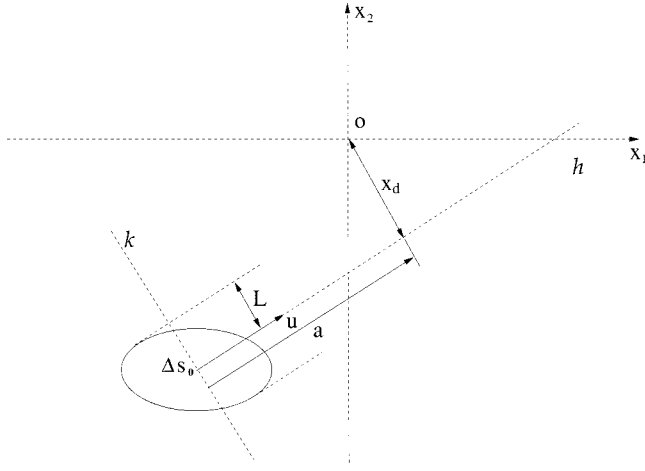$$\lambda_1^2(x_1 - \Delta s_1(t))^2 + \lambda_2^2(x_2 - \Delta s_2(t))^2 = \rho^2/2. \qquad (29)$$

Fig. 7.   Transformed protection zone.

A conflict occurs if and only if the 2-D standard BM $\overline{n}(t)$ starting at the origin (representing aircraft A in the new coordinate system) ever wanders into this moving ellipse.

Let x_d 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 of the origin on h, as indicated in Fig. 7. Then

$$x_d = \frac{|\Delta s_0^T R\left(\frac{\pi}{2}\right) u|}{\|u\|}, \qquad a = -\frac{\Delta s_0^T u}{\|u\|}. \tag{30}$$

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. If we ignore the effect of the noise, in the new coordinate system the minimal 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 by F. Since the probability of conflict $P(F)$ does not admit a closed-form formula, we will attempt to approximate it. 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 Fig. 7). The projected width $2L$ of the ellipse along line k can be computed as follows:

$$L = \frac{\rho}{\lambda_1 \lambda_2} \sqrt{\frac{u_1^2 \lambda_1^2 + u_2^2 \lambda_2^2}{2(u_1^2 + u_2^2)}} \tag{31}$$

where λ_1 and λ_2 are given in (27). Denote by τ the first time $\overline{n}(t)$ hits k and define F' to be the event that $\overline{n}(\tau)$ is within a distance of L from the center of the ellipse. One can show [34] 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 growth rate of the variance of the BM, which is, in fact, the case of interest. The intuition 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 (for a formal discussion see [34]).

Assume that u is aligned with the positive x_1 axis. This can be done by without loss of generality by an appropriate rotation, since the standard BM is invariant with respect to rotation. Assume that the two aircraft are approaching each other, i.e., $a > 0$. Then the time τ for aircraft A to reach line k has the dis-

tribution $p_\tau(\cdot)$ given by (23) with $\mu = \|u\|$. The approximate probability of conflict $P(F')$ can then be written as

$$P(F') = \int_0^\infty p_\tau(t) \int_{|y - x_d| < L} \frac{1}{2\pi t} \exp\left(-\frac{y^2}{2t}\right) dy\, dt$$

$$= \int_0^\infty p_\tau(t) \left[Q\left(\frac{x_d - L}{\sqrt{t}}\right) - Q\left(\frac{x_d + L}{\sqrt{t}}\right) \right] dt \tag{32}$$

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

Note that by Formula 1 in the Appendix, $E[\tau] = a/\|u\|$. When $\|u\|$ is greater than one, $p_\tau(t)$ concentrates near $t_0 := a/\|u\|$. Then by taking the 0th order expansion of $Q((x_d - L)/\sqrt{t})$ and $Q((x_d + L)/\sqrt{t})$ around t_0 in (32), we get the following.

Assertion 3: The probability of conflict can be approximated by

$$P^0 := Q\left(\frac{x_d - L}{\sqrt{t_0}}\right) - Q\left(\frac{x_d + L}{\sqrt{t_0}}\right) \tag{33}$$

where L is given by (31), a and x_d are given by (30), and we set $t_0 := a/\|u\|$.

In [34], it is shown that P^0 is a remarkably sharp estimate of $P(F')$, even in the case when $\|u\|$ is much smaller than one (the approximation error when $\|u\| > 1$ is barely noticeable). Higher order approximations of the probability of conflict can also be obtained [34], but the resulting improvement in the approximation is not substantial.

Fig. 8 shows level curves of P^0 as a function of the initial position Δx_0 of aircraft B for the path angles $\vartheta = 0°, 45°, 90°, 180°$. 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 values for ν_a and ν_c ($\nu_a = 2$ and $\nu_c = 1$). Observe that as the cross path angle ϑ increases, the region delimited by the same P^0 equiprobability line gets more and more extended. This indicates that, everything else being equal, the situation of head-on conflict ($\vartheta = 180°$) is the most dangerous, whereas the situation of overtake conflict ($\vartheta = 0°$) 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 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:

$$P(F' \cap \{\tau \le t_f\})$$

$$= \int_0^{t_f} p_\tau(t) \int_{|y - x_d| < L} \frac{1}{2\pi t} \exp\left(-\frac{y^2}{2t}\right) dy\, dt$$

$$= \int_0^{t_f} p_\tau(t) \left[Q\left(\frac{x_d - L}{\sqrt{t}}\right) - Q\left(\frac{x_d + L}{\sqrt{t}}\right) \right] dt$$

which, by using zero-order expansion of $Q((x_d - L)/\sqrt{t})$ and $Q((x_d + L)/\sqrt{t})$ around t_0, and Appendix, formula 2 leads to
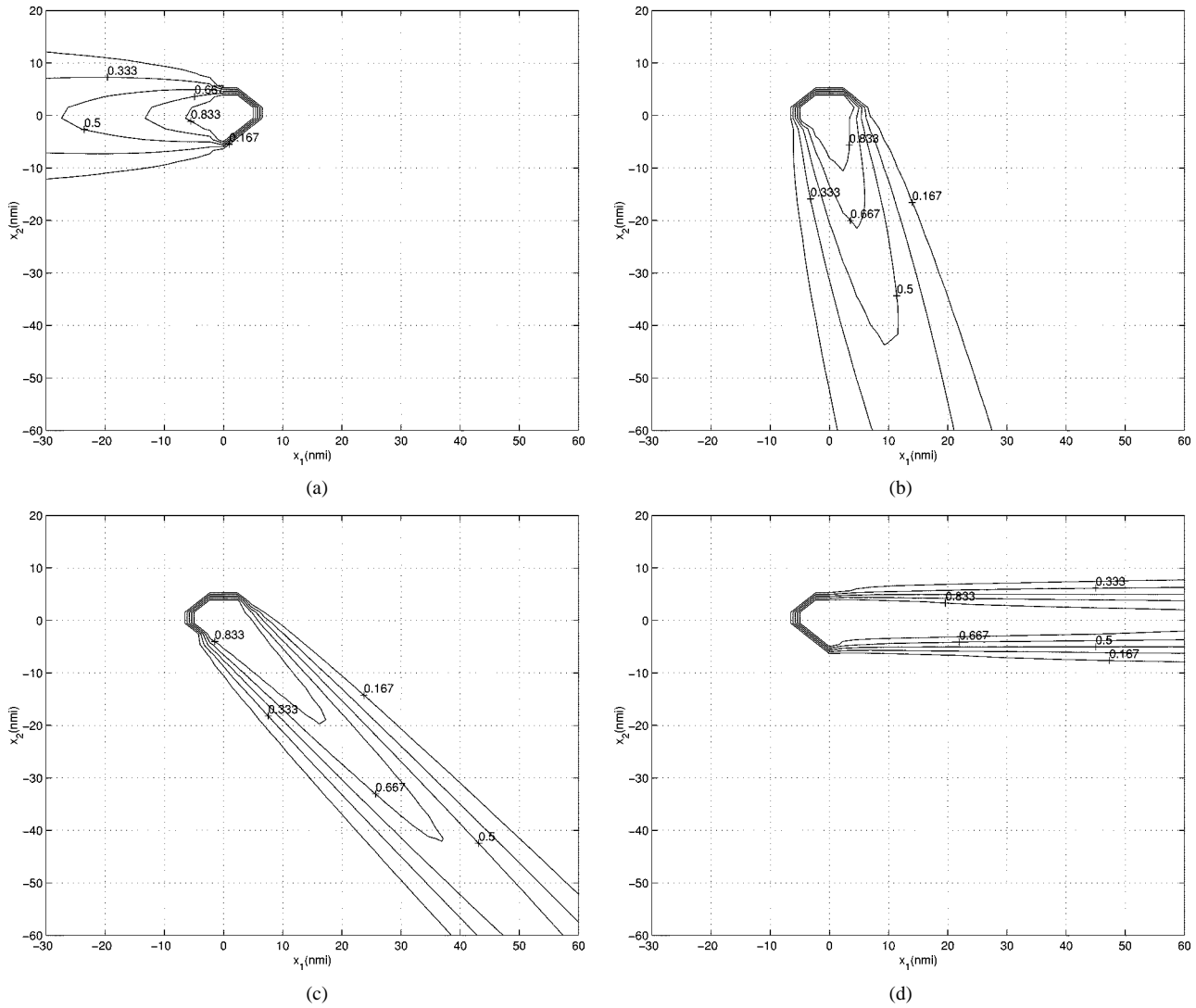
(a)

(b)

(c)

(d)

Fig. 8.   (a) P^0 for path angles $\vartheta = 0°$ and (b) $\vartheta = 45°$ with $\|u^A\| = 7$ nmi/min, $\|u^B\| = 8$ nmi/min, $\nu_a = 2$, $\nu_c = 1$. (c) $\vartheta = 90°$ and (d) $\vartheta = 180°$ with $\|u^A\| = 7$ nmi/min, $\|u^B\| = 8$ nmi/min, $\nu_a = 2$, $\nu_c = 1$.

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

$$P^0_{t_f} := \left[Q \left(\frac{a - \|u\| t_f}{\sqrt{t_f}} \right) + \exp(2a\|u\|) Q \left(\frac{a + \|u\| t_f}{\sqrt{t_f}} \right) \right]$$
$$\cdot \left[Q \left(\frac{x_d - L}{\sqrt{t_0}} \right) - Q \left(\frac{x_d + L}{\sqrt{t_0}} \right) \right] \tag{34}$$

where L is given by (31), a and x_d are given by (30), and we set $t_0 := a/\|u\|$.

Fig. 9 shows level curves of $P^0_{t_f}$ for $\vartheta = 90°$, in the two cases when $t_f = 10$ min and $t_f = 15$ min. $P^0_{t_f}$ looks like a truncation of the corresponding infinite horizon version. It is not surprising that as ν_a and ν_c increase, the truncation becomes smoother. As for the infinite horizon case, higher order approximations of the probability of conflict do not seem to be significantly better.

In [35, eq. (33)], 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 the initial positions and velocities are uncertain.

It is then proposed to use a level curve of (33) to delimit the "alert zone," which in the context of [35] is a region 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. Decentralized Conflict Resolution Algorithm

Many contributions in the literature deal with the issue of providing safe, coordinated conflict resolution maneuvers for two aircraft, both in a deterministic setting (see, for example, [11], [12], [36], [37]) and in a probabilistic setting (see, for example, [8], [35]). In comparison, there are relatively few contributions treating directly the multiple aircraft case and they mostly deal with a deterministic setting. Conflicts involving multiple aircraft that may occur in high-traffic areas are more general and more difficult to resolve. The reason is that if one tries to resolve conflicts pairwise one may generate a conflict with a third aircraft and may be unable to converge
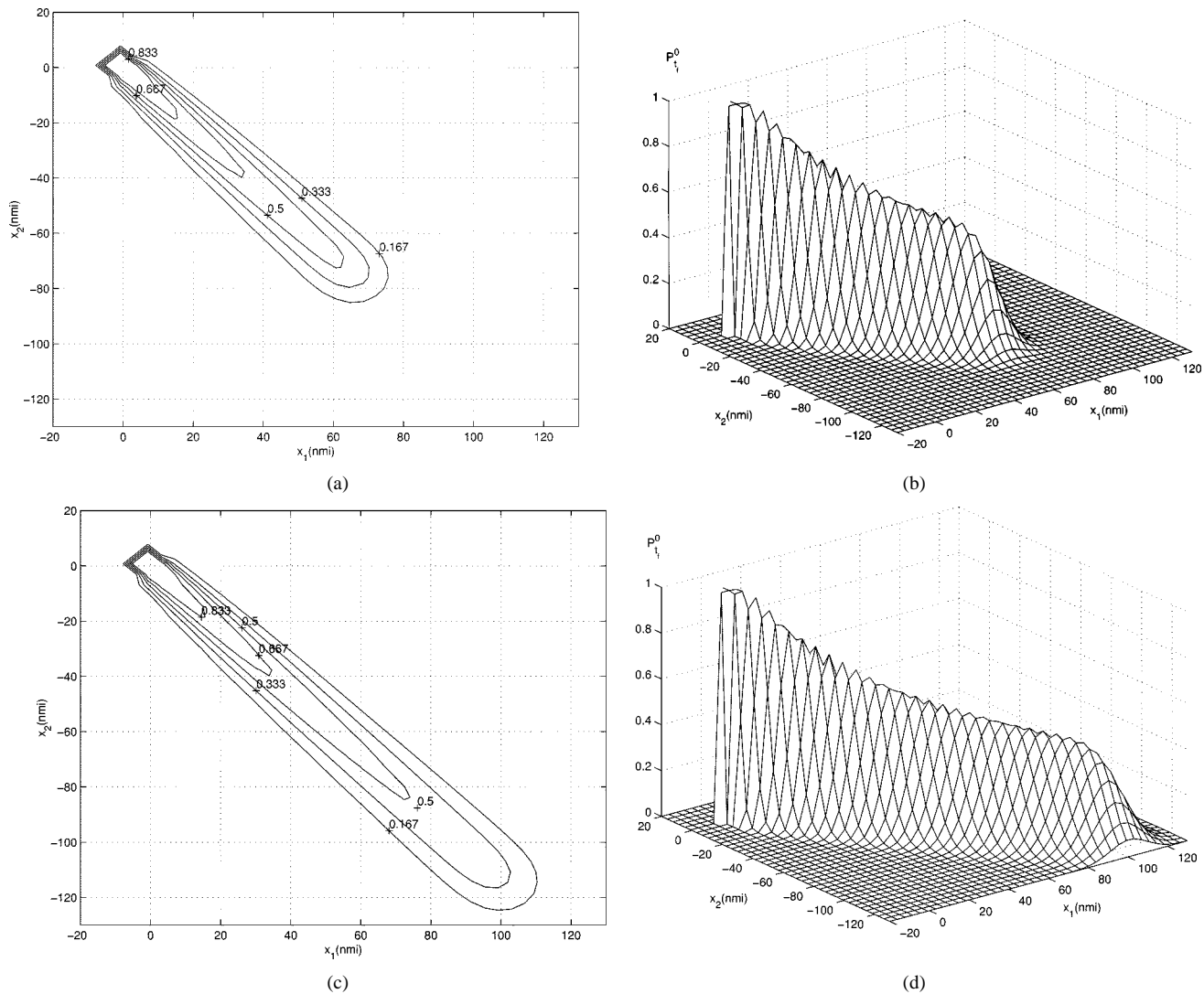
(a)

(b)

(c)

(d)

Fig. 9. (a) $P_{t_f}^0$ for $t_f = 10$ min and (b) $t_f = 15$ min with $\vartheta = 90°$, $\|u^A\| = 7$ nmi/min, $\|u^B\| = 8$ nmi/min, $\nu_a = 2$, $\nu_c = 1$. (c) $P_{t_f}^0$ for $t_f = 10$ min and (d) $t_f = 15$ min with $\vartheta = 90°$, $\|u^A\| = 7$ nmi/min, $\|u^B\| = 8$ nmi/min, $\nu_a = 2$, $\nu_c = 1$.

to a feasible solution. In [38]–[41], multiple aircraft conflict resolution is based on the optimization of a cost function, which is suitably selected to take into account practical factors such as fuel consumption and passenger comfort. The computational issues involved in the 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 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, grows rapidly with the number of aircraft involved in the encounter.

A completely different approach is taken in [42], where potential and vortex field methods are applied to generate decentralized resolution maneuvers for multiple aircraft encounters.

TCAS [12] also has some multiple aircraft resolution capabilities, but is primarily geared toward pairwise, short range encounters. TCAS predicts conflicts using a deterministic model and issues coordinated resolution advisories by selecting the least aggressive among a predefined set of maneuvers in the vertical plane. The goal is to ensure adequate vertical separation at the point of closest approach.

Here, we introduce an algorithm for multiple aircraft conflict resolution which uses the probability of conflict calculated in Section III-B to guide each aircraft to its destination, while avoiding conflict with other aircraft flying in the same region of the airspace. Unlike TCAS and the optimization based algorithms discussed above, in our algorithm the uncertainty affecting the aircraft motion is explicitly taken into account. Moreover, conflict resolution is decentralized since aircraft resolve conflicts using only information about the positions, headings, and speeds of neighboring aircraft without coordinating with them. In contrast with the worst-case approach to decentralized conflict resolution taken 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

their effort may be undermined by the uncertainties inherent in the environment.

Our approach is most closely related to that of [42]. The main difference is that we use information not only on the current positions, but also on the current headings and speeds of the surrounding aircraft 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 merits of P^0 and $P^0_{t_f}$ as criticality measures by Monte Carlo simulations. In [41], this algorithm is also used as "random type chooser," to provide 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 destinations and that they fly at a constant speed v^A and v^B, respectively. At each time instant t, the probability of conflict $P_c(t)$ can be calculated by (33) [or (34) for some finite horizon t_f] using the current positions $x^A(t)$, $x^B(t)$ and velocities $u^A(t)$, $u^B(t)$ of the two aircraft.

For each aircraft we define 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 positions of the aircraft.

Our resolution strategy aims at making each aircraft reach the desired destination while avoiding situations of conflict by appropriately changing its heading. This is done by updating the aircraft headings every Δt time instants by means of the following algorithm.

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

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

end

Here β is the maximal turn angle allowed per time step and for simplicity we use $z(k)$ to denote $z(k\Delta t)$.

The ideal new heading $\overline{\vartheta}^i(k)$ 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. Due to aircraft limitations, 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 $\overline{\vartheta}^i(k)$ within the allowed range.

The validation protocol adopted for performing the Monte Carlo simulations is the following.

1) For every Monte Carlo run perform
   a) *initialization*: given the flight plans of the two aircraft, compute $\vartheta_d^i(0)$ 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) *resolution*: choose the current heading at time k by means of Algorithm 5;
   c) *update*: update the aircraft velocities and positions at time step $k + 1$ as follows:

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

   $i = A, B$, 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) *termination*: check if both aircraft have reached their destinations; if not, set $k := k + 1$ and return to step b).

2) After a prescribed number of runs, build the histogram of the minimum separation between the aircraft. The resolution algorithm can also be interpreted as a receding horizon control algorithm, with Δt playing the role of the control horizon, and the interval over which P_c is approximated (either $[0, \infty)$ or $[0, t_f]$) playing the role of the prediction horizon. Therefore, in the simulations presented below resolution is not performed at once over the entire length of the simulation (of the order of 30 minutes). It is performed by repeatedly invoking our short-range resolution algorithm over a horizon Δt (of the order of 1 min).

Simulation results are shown in Fig. 10 for two typical encounters: head-on encounter and orthogonal encounter. The speeds of the two aircraft are chosen such that it takes 30 min for them to fly from their starting positions (marked with stars) to their destination positions (marked with diamonds) along a straight line. We used (33) for computing $P_c(k)$ and set $\Delta t = 1$, $\beta = \pi/25$, $\nu_a = 0.35$ and $\nu_c = 0.2$. The values of ν_a and ν_c are chosen 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 $\rho = 5$ nmi is respected and, moreover, most of the minimum separations
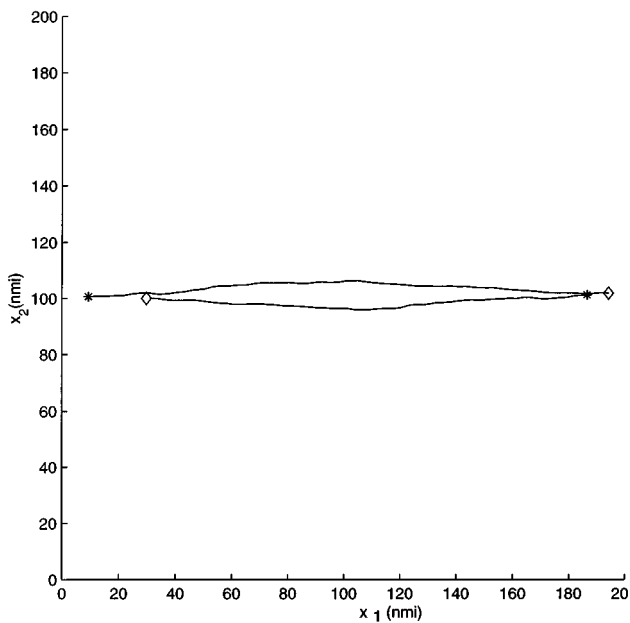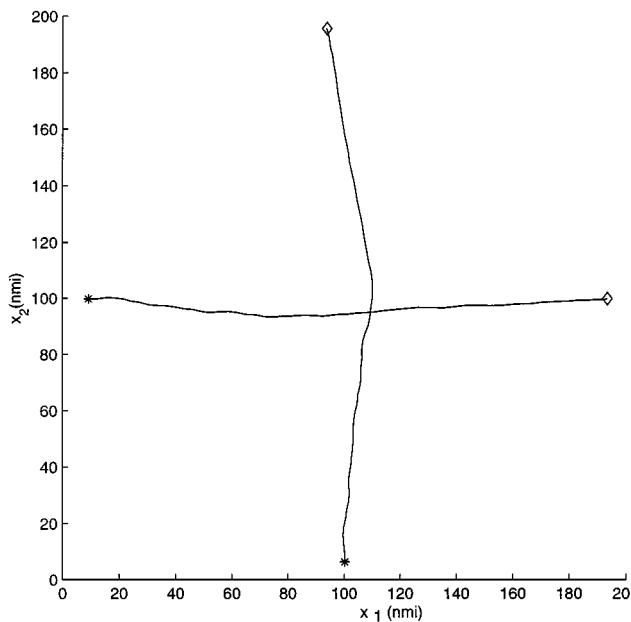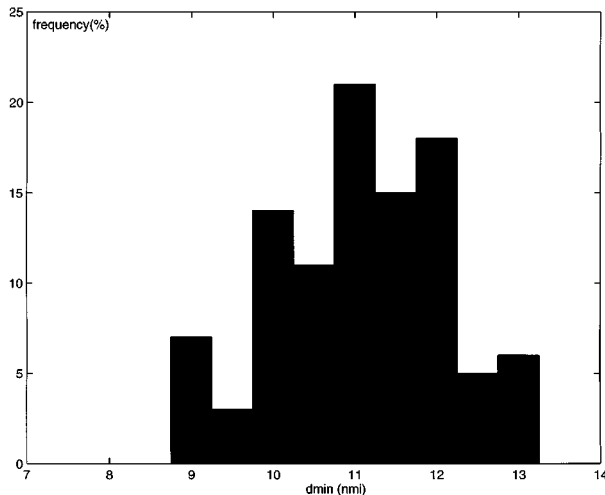
(a)
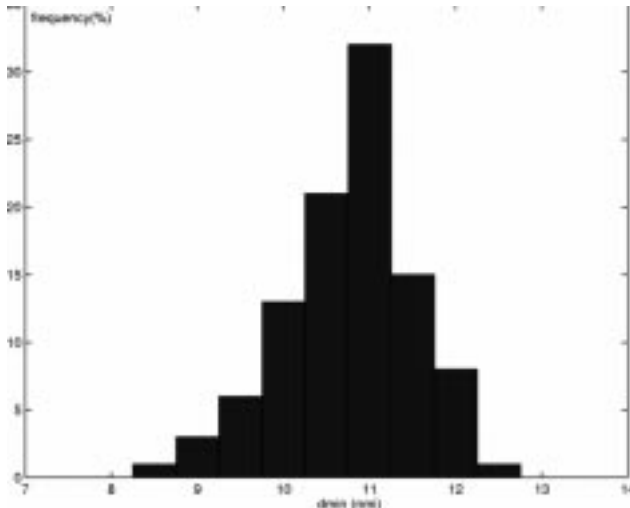
(b)

(c)

(d)

Fig. 10.   Resolution maneuver for two aircraft encounter situations ($\nu_a = 0.35$, $\nu_c = 0.2$, $\beta = \pi/25$).(a) Head-on encounter. (b) Minimum separation histogram for (a). (c) Orthogonal encounter. (d) Minimum separation histogram for (c).

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.

Zooming into the resolution trajectories reveals a lot 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. To get flyable paths, we can either increase the resolution horizon Δt or take the average of the resolution headings over a period of time (which would act as a low-pass filter).

The algorithm can be extended to the multiple aircraft case. Suppose we have n aircraft sharing the same region of the airspace. Then, the new heading of each aircraft has to be com-

puted taking into account the overall probability of conflict. For each aircraft, say i, $i \in \{A_1, A_2, \ldots, A_n\}$, 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:

$$\overline{\vartheta}^i(k) = \overline{P}_c^i(k) \frac{\displaystyle\sum_{j \neq i} P_c^{ij}(k)\vartheta_g^{ij}(k)}{\displaystyle\sum_{j \neq i} P_c^{ij}(k)} + \left(1 - \overline{P}_c^i(k)\right) \vartheta_d^i(k)$$
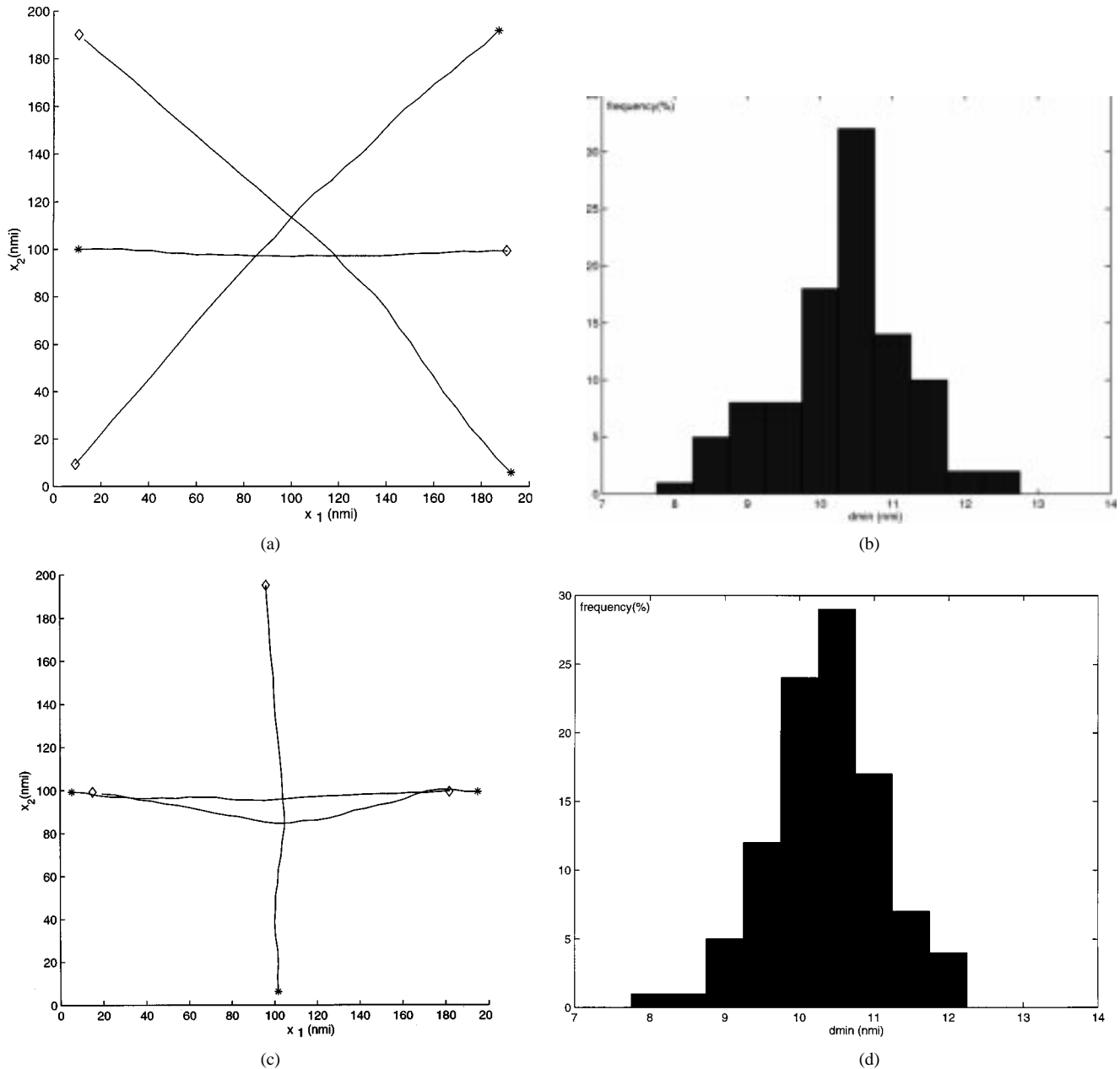
Fig. 11.   Resolution for three aircraft encounter situations ($\nu_a = 0.35$, $\nu_c = 0.2$, $\beta = \pi/25$). (a) Encounter. (b) Minimum separation histogram for (a). (c) Orthogonal encounter. (d) Minimum separation histogram for (c).

where $\overline{P}_c^i(k) = \max_{j \neq i} P_c^{ij}(k)$ is the maximum probability of conflict among all aircraft pairs involving 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 encounter situations involving three aircraft are shown in Fig. 11. All parameters remain the same as in the two aircraft case. Since the airspace is not considerably more congested all conflicts are successfully resolved and most of the minimum separations are again centered around 2ρ.

The situation becomes more complicated when the number of aircraft increases to, say, eight, as is shown in Fig. 12. The first encounter 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, 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 minimum 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 the safety properties of this algorithm as a function of the number of aircraft involved in the encounter and the uncertainty level.
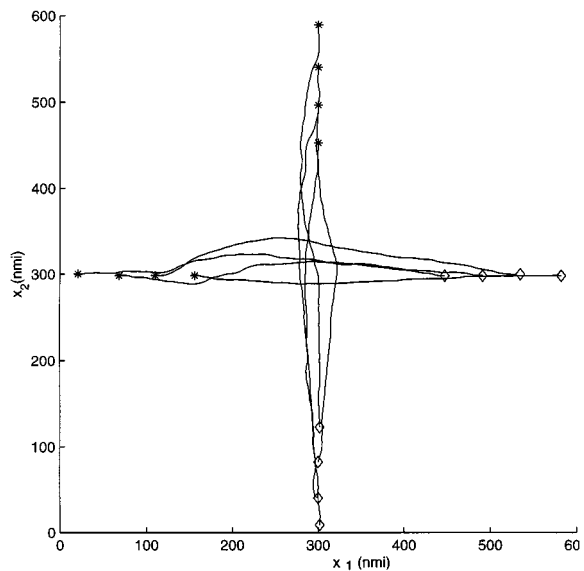
Computational load is an important issue for multiple aircraft resolution. Because all probabilities are approximated in closed form the load for pairwise resolution is very small with our algorithm. However, for multiple aircraft encounters all pairs of aircraft have to be tested, therefore, the number of operations
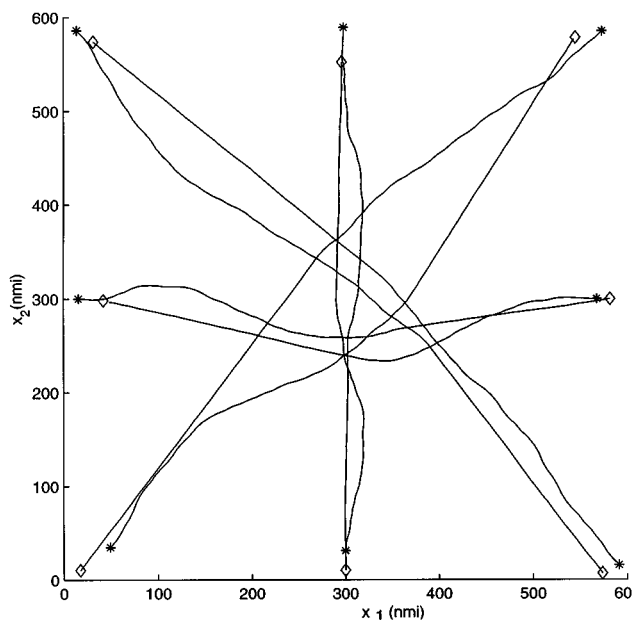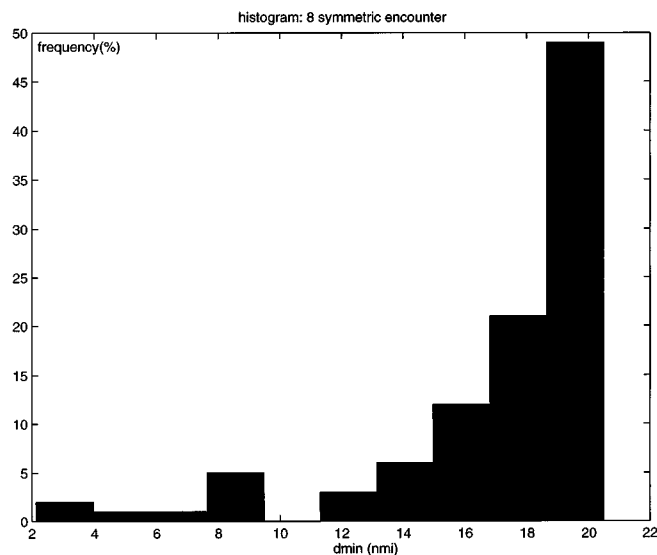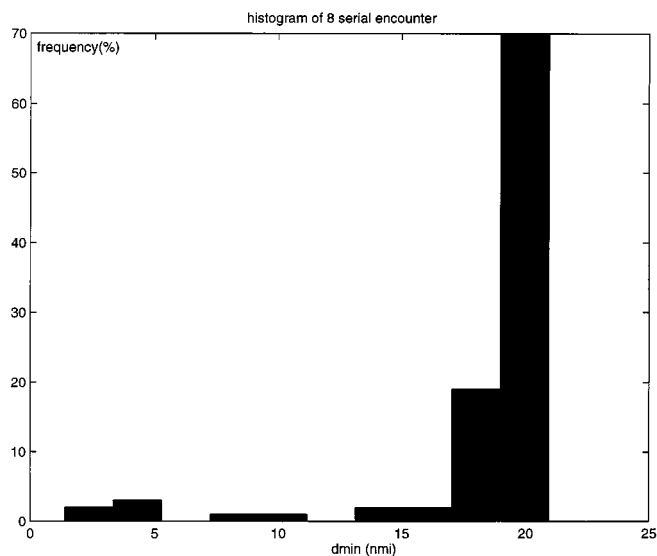
Fig. 12. Resolution for eight aircraft encounter situations ($\nu_\alpha = 0.35$, $\nu_c = 0.2$, $\beta = \pi/25$, $\rho = 10$ nmi). (a) Symmetric encounter. (b) Minimum separation histogram for (a). (c) Streamline encounter. (d) Minimum separation histogram for (c).

that need to be performed grows roughly as the square of the number of aircraft involved. This type of growth is also observed in CTAS and has been addressed using a number of heuristics to limit the search. Fortunately, the number of aircraft involved in our case is likely to be much smaller than CTAS, because of the short-range nature of our algorithm. Even though the computation becomes harder in the multiple aircraft case, our algorithm scales better than most centralized resolution algorithms, whose complexity grows exponentially in the size of the problem.

IV. SUMMARY AND CONCLUSION

In this paper, we addressed 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 were developed 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.

For mid-term prediction, randomized algorithms were used for estimating the criticality measure and formulating a conflict detection algorithm for two aircraft encounters. The performance of the randomized detection algorithm was compared to that of the detection algorithm integrated in CTAS using Monte Carlo simulations. The simulations suggest that the two algorithms have similar performance for simple encounters, whereas our algorithm tends to perform better for more complex encounters. Further study is needed to evaluate the performance of the

randomized algorithm in terms of computational load as a function of the level of approximation introduced.

For short-term prediction closed-form approximations for the probability of conflict were derived for a two aircraft encounter. Based on these approximations, we formulated a conflict resolution algorithm and used Monte Carlo simulations to demonstrate the efficacy of the use of the probability of conflict to generate safe resolution maneuvers. The extension of the approach to multiple aircraft conflict resolution revealed potential problems with this approach when the airspace becomes very congested. We are currently analyzing the effect of the uncertainty level and the number of aircraft involved in the encounter on the performance of the resolution algorithm.

Although we focused on the planar case, the extension to the 3-D case is straightforward. 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. For the short-range conflict detection, in a similar way to the 2-D case, analytical approximations for the probability of conflict can be computed for the 3-D case and then be used to derive a decentralized 3-D resolution algorithm. However, it is considerably harder to get meaningful bounds for the error of such approximations.

Both the approaches introduced here have potential application to the airborne conflict management (ACM) operational concept, which is being developed by the radio technical commission for aeronautics (RTCA) and the FAA [43]. The ACM concept includes detecting conflict situations, monitoring for conflicts that might arise if the current flight plan is changed and suggesting resolution maneuvers to prevent a violation of the separation criteria. It is envisioned that these functionalities (respectively named *conflict detection*, *conflict prevention*, and *conflict resolution*) will be available not only to the ATC but also to the pilots. The enabling technologies are GPS, ADS-B and cockpit display of traffic information (CDTI). Multiple levels of alert are proposed for conflict detection in ACM. Depending on the airspace domain and the operating flight rules, the actions in response to the different alerts may have to be coordinated with the ATC or may be undertaken autonomously by the pilot.

In this context, the mid-term detection algorithm proposed here could be used for the conflict prevention function of ACM since it can deal with longer look-ahead horizons. Besides providing warnings for potential conflicts, it may also be possible for the ATC to query the system about the safety of the changes he would like to make to the flight plans. Moreover, the level sets for the probability of conflict obtained for short-term prediction could be used to define the different alert zones, with the lower alert zone being delimited by the level curve corresponding to the lower probability of conflict. The proposed conflict resolution algorithm could be used to suggest to the pilots maneuvers to resolve short range conflicts. Notice that ADS-B and GPS together provide the position and intent (velocity) information necessary for our decentralized resolution algorithm to be implemented.

We are currently working on formulating algorithms for mid-range conflict resolution and investigating different design alternatives. We are pursuing a model predictive control approach to achieve this goal. The idea is to use the current radar measurements, the 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, and 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 flight plans. The parameters one needs to set significantly influence the acceptability of the advisories by the ATC. These parameters include how far in advance should ATC be notified of flight plan changes and the bounds on the allowable flight plans (partly determined by aircraft capability). 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 toward 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
FORMULAS

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

Formula 1: If $x > 0$, $v > 0$, then

$$\int_0^\infty \frac{x}{\sqrt{2\pi t}} \exp\left[-\frac{(x-vt)^2}{2t}\right] dt = \frac{x}{v}.$$

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

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

REFERENCES

[1] C. Tomlin, "Hybrid control of air traffic management system," Ph.D. dissertation, Univ. California, Berkeley, 1998.
[2] M. S. Nolan, *Fundamentals of Air Traffic Control*, 3rd ed. Belmont, CA: Wadsworth, 1998.
[3] D. Bertsimas and S. Stock Patterson, "The air traffic flow management problem with enroute capacities," *Operat. Res.*, vol. 46, pp. 406–422, 1998.

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

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

[6] H. Erzberger, T. J. Davis, and S. Green, "Design of center-tracon automation system," in *Proc. AGARD Guidance Contr. Symp. Mach. Intell. Air Traffic Management*, 1993, pp. 11.1–11.12.

[7] R. A. Paielli and H. Erzberger, "Conflict probability estimation for free flight," *J. Guidance, Contr., Dynam.*, 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/Eur. Air Traffic Management Res. Development Seminar*, Saclay, France, June 1997.

[9] D. Brudnicki and A. McFarland, "User request evaluation tool (URET) conflict probe performance and benefits assessment," in *1st USA/Eur. Air Traffic Management Res. Development Seminar*, Saclay, France, June 1997.

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

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

[12] Radio Tech. Commission Aeronaut., "Minimum operational performance standards for traffic alert and collision avoidance system (TCAS) airborne equipment,", Tech. Rep. RTCA/DO-185, RTCA, Sept. 1990.

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

[14] J. Kuchar and L. C. Yang, "Survey of conflict detection and resolution modeling methods," *AIAA Guidance, Navigat., Contr. Conf.*, pp. 1388–1397, Aug. 1997.

[15] M. G. Ballin and H. Erzberger, "An analysis of landing rates and separations at Dallas/Ft. Worth airport,", Tech. Rep., NASA Tech. Memo TM 110 397, July 1996.

[16] R. A. Paielli, "Empirical test of conflict probability estimation," in *2nd USA/Eur. Air Traffic Management Res. Development Seminar*, 1998.

[17] C. Tomlin, J. Lygeros, and S. Sastry, "Controller design for hybrid systems," Proc. IEEE, to be published.

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

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

[20] J. K. Kuchar, "A unified methodology for the evaluation of hazard alerting systems," Ph.D. dissertation, Massachusetts Inst. Technol., Cambridge, MA, 1995.

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

[22] M. Prandini, J. Lygeros, A. Nilim, and S. Sastry, "Randomized algorithms for probabilistic aircraft conflict detection," in *Proc. IEEE Conf. Decision, Contr.*, Phoenix, AZ, Dec. 1999, pp. 2444–2449.

[23] G. Chen, G. Chen, and S. Hsu, *Linear Stochastic Control Systems*. Boca Raton, FL: CRC, 1995.

[24] M. Vidyasagar, *A Theory of Learning and Generalization: With Applications to Neural Networks and Control Systems*. London, U.K.: Springer-Verlag, 1997.

[25] ——, "Statistical learning theory and its applications to randomized algorithms for robust controller synthesis," in *Proc. Plenary Lectures Mini-Courses Eur. Contr. Conf.*, Brussels, Belgium, July 1997, pp. 161–189.

[26] M. Prandini, "Adaptive LQG control: Optimality analysis and robust controller design," Ph.D. dissertation, Univ. Brescia, Italy, 1998.

[27] M. C. Campi and M. Prandini, "Randomized algorithms for the synthesis of adaptive controllers," in *Proc. MTNS'98 Conf.*, Padova, Italy, July 1998, pp. 723–726.

[28] R. A. Paielli and H. Erzberger, "Conflict probability estimation generalized to nonlevel flight," *Air Traffic Contr. Quart.*, vol. 7, no. 3, pp. 195–222, 1999.

[29] J. K. Kuchar, "Methodology for alerting-system performance evaluation," *J. Guidance, Contr., Dynam.*, 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 *Proc. IEEE Conf. Decision Contr.*, Phoenix, AZ, Dec. 1999, pp. 2438–2443.

[31] B. Oksendal, *Stochastic Differential Equations, An Introduction With Application*, 5th ed. New York: Springer-Verlag, 1998.

[32] J. Hu, J. Lygeros, M. Prandini, and S. Sastry, "A probabilistic framework for highway safety analysis," in *Proc. IEEE Conf. Decision Contr.*, Phoenix, AZ, Dec. 1999, pp. 3734–3739.

[33] R. Durrett, *Probability: Theory and Examples*, 2nd ed. Belmont, CA: Duxbury, 1996.

[34] J. Hu, "A study of conflict detection and resolution in free flight," M.S. thesis, Univ. California, Berkeley, Dec. 1999.

[35] J. Krozel and M. Peters, "Strategic conflict detection and resolution for free flight," in *Proc. IEEE Conf. Decision Contr.*, San Diego, CA, Dec. 1996, pp. 1822–1828.

[36] Y. Zhao and R. Schultz, "Deterministic resolution of a two aircraft conflict in free flight," in *Proc. AIAA Guidance, Navigat., Contr. Conf*, New Orleans, LA, 1997.

[37] J. Krozel, T. Mueller, and G. Hunter, "Free flight conflict detection and resolution analysis," in *Proc. AIAA Guidance, Navigat., Contr. Conf.*, San Diego, CA, 1996.

[38] N. Durand, J. M. Alliot, and J. Noailles, "Automatic aircraft conflict resolution using genetic algorithms," in *Proc. 11th Annu. ACM Conf. Appl. Computing ACM/SAC*, Philadelphia, PA, 1996.

[39] E. Frazzoli, Z. H. Mao, J. H. Oh, and E. Feron, "Resolution of conflicts involving many aircraft via semidefinite programming," AIAA J. Guidance, Contr., Dynam., to be published.

[40] P. K. Menon, G. D. Sweriduk, and B. Sridhar, "Optimal strategies for free-flight air traffic conflict resolution," *AIAA J. Guidance, Contr., Dynam.*, 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 Proc. IEEE Conf. Decision Contr., Sidney, Australia, to be published.

[42] J. Kosecka, C. Tomlin, G. Pappas, and S. Sastry, "Generation of conflict resolution maneuvers for air traffic management," in *Proc. IEEE Conf. Intell. Robot. Syst.*, 1997.

[43] Radio Tech. Commission Aeronaut., "Application of airborne conflict management: Detection, prevention and resolution,", Tech. Rep. Draft , 2000.

Maria Prandini received the Laurea degree in electrical engineering from the Politecnico of Milano, Italy, in 1994, and the Ph.D. 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 Postdoctoral Researcher with 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, and the M.S. degree in electrical engineering from the University of California at Berkeley in 1999. He is currently working toward the Ph.D. degree in electrical engineering at the University of California at Berkeley.

His research interests include stochastic and optimal control of hybrid systems, air traffic management, and geometric control.

John Lygeros received the B.Eng. degree in electrical and electronic engineering in 1990 and the M.Sc. degree in control and systems in 1991, both from Imperial College of Science Technology and Medicine, London, U.K., and the Ph.D. degree from the Electrical Engineering and Computer Sciences Department of the University of California, Berkeley in 1996.

From June 1996 to December 1999, he held Postdoctoral Research Appointments, first with the National Automated Highway Systems Consortium, Institute of Transportation Studies, University of California, Berkeley, then with the Laboratory for Computer Science, Massachusetts Institute of Technology, and finally with the Electrical Engineering and Computer Sciences Department, University of California, Berkeley. At the same time, he also held a part-time Research Engineer position at SRI International, Menlo Park, CA, from May 1998 to December 1999, and a Visiting Professor position at the Mathematics Department, Universite de Bretagne Occidentale, Brest, France. He is currently a University Lecturer at the Department of Engineering, University of Cambridge, U.K. His research interests include hierarchical, hybrid, and nonlinear control theory and their applications to large-scale systems such as highway systems, air traffic management, and power networks.

Dr. Lygeros ireceived the 1997 Eliahu Jury Award "for excellence in systems research," by the Electrical Engineering and Computer Sciences Department, University of California, Berkeley.

Shankar Sastry (F'94) received his Ph.D. degree from the University of California, Berkeley, in 1981.

He was on the faculty of the Massachusetts Institute of Technology (MIT), Cambridge, from 1980 to 1982, and Harvard University, Cambridge, MA, as a Chaired Gordon McKay Professor in 1994. He is currently a Professor of electrical engineering and computer sciences and a Professor of bioengineering at the University of California, Berkeley, on leave from his duties as Director of the Information Technology Office, DARPA, Washington, DC. He has held Visiting Appointments at the Australian National University, Canberra, University of Rome, Italy, Scuola Normale and University of Pisa, Italy, CNRS Laboratory LAAS, Toulouse (poste rouge), France, Professor Invite at Institut National Polytechnique de Grenoble (CNRS Laboratory VERIMAG), France. He was also 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 supervised 34 doctoral students to completion and over 50 masters students. He is the author of *Nonlinear Systems: Analysis, Stability and Control* (New York: Springer-Verlag, 1999) and a coauthor of *Adaptive Control: Stability, Convergence and Robustness* (Englewood Cliffs, NJ: Prentice Hall, 1989) and of *A Mathematical Introduction to Robotic Manipulation* (Boca Raton, FL: CRC, 1994). He has coedited *Hybrid Control—Vol. II, IV, and V* (Springer-Verlag lecture notes in computer science, 1995, 1997, and 1999, respectively), *Hybrid Systems: Computation and Control* (Springer-Verlag lecture notes in computer science, 1998), and *Essays in Mathematical Robotics* (Springer-Verlag IMA series). He has served as Associate Editor of *Journal of Mathematical Systems, Estimation and Control* and is currently 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*.

Dr. Sastry 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 American Automatic Control Council in 1990, an M.A. degree (*honoris causa*) from Harvard University, Cambridge, MA, in 1994, the Distinguished Alumnus Award of the Indian Institute of Technology, Bombay, in 1999, and the David Marr Prize for Best Paper in 1999 at the International Conference in Computer Vision. He was an Associate Editor of the IEEE TRANSACTIONS ON AUTOMATIC CONTROL, IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS, *IEEE Control Magazine*.