Air-Traffic Control in Approach Sectors: simulation examples and optimisation

Andrea Lecchini[1], William Glover[1], John Lygeros[2], and Jan Maciejowski[1]

[1] Control Lab, Department of Engineering, University of Cambridge, CB2 1PZ
Cambridge, UK {al394, wg214, jmm}@eng.cam.ac.uk,
WWW home page: http://www-control.eng.cam.ac.uk/
[2] Department of Electrical Engineering, University of Patras, Rio, 26500 Patras,
Greece lygeros@ee.upatras.gr,
WWW home page: http://www.sml.ee.upatras.gr/lygeros

Abstract. In this contribution we consider the approach to the runway as a case study of our research on conflict resolution for Air-Traffic Control with stochastic models. We simulate the approach for landing and optimise the maneuver through a simulation based optimisation strategy.

1 Introduction

In the current organisation of Air-Traffic Management the centralised Air-Traffic Control is in complete control of the air-traffic and ultimately responsible for safety. The main objective of Air-Traffic Control is to maintain safe separation between aircraft by issuing proper instructions to the pilots. A *conflict* is defined as the situation of loss of minimum safe separation between two aircraft. If it is possible, Air-Traffic Control tries also to fulfil the (possibly conflicting) requests of aircraft and airlines; for example, desired paths to avoid turbulence or desired time of arrivals to meet schedule. In order to improve performance of Air-Traffic Control, mainly in anticipation of increasing levels of traffic, research effort has been spent in the last decade on creating tools for conflict detection and resolution. A review of research work in this area of Air-Traffic Control is presented in [1].

Uncertainty is introduced in air-traffic by the action of the wind field, incomplete knowledge of the physical coefficients of the aircraft and unavoidable imprecision in the execution of Air-Traffic Control instructions. In conflict detection the objective is to evaluate conflict probability over a certain future horizon starting from the current positions and flight plans of the aircraft. In conflict resolution the objective is to calculate suitable maneuvers to avoid a predicted conflict. A number of conflict resolution algorithms have been proposed for a deterministic setting, for example [2–4]. In a stochastic setting, research has concentrated mainly on conflict detection [5–8]. The main reason for this is the complexity of stochastic prediction models which, even if it does not make it impossible to estimate conflict probability through Monte Carlo methods, it makes the quantification of the effects of possible control actions intractable.

Air-traffic conflict resolution involves several hybrid aspects related either to the nature of the system and to the control problem. The system itself contains continuous dynamics, arising from the physical motion of the aircraft, discrete dynamics, arising from the logic embedded in the Flight Management System, and stochastic dynamics, arising from the effect of wind on the aircraft tracks and uncertainty in the physical parameters of the aircraft (for example the mass). Other hybrid aspects, from the point of view of Air-Traffic Control, are the fact that aircraft follow a nominal path that is a sequence of straight lines and that the motion of aircraft can not be freely adjusted. For example descending aircraft follow a prespecified speed profile and therefore "descent" can be seen as a discrete state with only a "1/0" value. Moreover, in conflict resolution, there are two rather separate problems one has to solve: (i) coordination between aircraft (e.g. selecting a landing sequence), which is typically a discrete combinatorial problem, and (ii) selecting the parameters of the resolution maneuver within the constraints imposed by the coordination, which is typically an optimisation problem over a continuous set.

We are currently investigating — see also [9] — on the use a Monte Carlo approach for conflict resolution in order to extend to this task the advantages of the Monte Carlo framework, in terms of flexibility and complexity of the prediction models that can be used. To this end, we adopt a Monte Carlo Markov Chain randomised optimisation method introduced originally in [10, 11].

Here we illustrate our approach in the solution of a typical Air-Traffic Control situation involving aircraft approaching the runway in Approach Sectors. In Section 1 we give a general formulation of the problem. The Monte Carlo Markov Chain optimisation procedure is described in Section 2. In Section 3 and 4 respectively we illustrate Air-Traffic Control in Approach Sectors and the air-traffic simulator. A simulation example with optimisation is presented in Section 5. Section 6 contains conclusion and future objectives.

2 Penalty formulation of an expected value optimisation problem with constraints

In this paper we formulate conflict resolution as a constrained optimisation problem. Given a set of aircraft involved in a conflict, the conflict resolution maneuver is determined by a parameter ω which defines the nominal paths of the aircraft The actual execution of the maneuver is affected by uncertainty. Therefore, the sequence of actual positions of the aircraft during the resolution maneuver (for example: the sequence of positions every 6 seconds which is a typical time interval between two successive radar sweeps) *a-priori* of its execution is a random variable denoted by X. A conflict is defined as the event that the positions of two aircraft during the execution of the maneuver are too close. The objective is to select ω in order to maximise the expected value of some measure of performance associated to the execution of the resolution maneuver while ensuring a small probability of conflict. In this section we introduce the formulation of the problem in a general fashion.

Let X be a random variable whose distribution depends on some parameter ω. The distribution of X is denoted by $p_\omega(x)$ with $x \in \mathbf{X}$. The set of all possible values of ω is denoted by $\mathbf{\Omega}$. We assume that a constraint on the random variable X is given in terms of a feasible set $\mathbf{X_f} \subseteq \mathbf{X}$. We say that a realisation x, of random variable X, violates the constraint if $x \notin \mathbf{X_f}$. Moreover, we assume that for a realisation $x \in \mathbf{X_f}$ some definition of performance of x is given. In general performance can depend also on the value of ω, therefore performance is measured by a function $\mathrm{perf}(\omega, x)$, $x \in \mathbf{X_f}$, $\omega \in \mathbf{\Omega}$. We assume that $\mathrm{perf}(\omega, x)$ takes values in $(0, 1]$. The probability of satisfying the constraint is denoted by $\mathrm{P}(\omega)$

$$\mathrm{P}(\omega) = \int_{x \in \mathbf{X_f}} p_\omega(x) dx \,. \tag{1}$$

The probability of violating the constraint is denoted by $\bar{\mathrm{P}}(\omega) = 1 - \mathrm{P}(\omega)$. The expected performance for a given $\omega \in \mathbf{\Omega}$ is denoted by $\textsc{Perf}(\omega)$, where

$$\textsc{Perf}(\omega) = \int_{x \in \mathbf{X_f}} \mathrm{perf}(\omega, x) p_\omega(x) dx \,. \tag{2}$$

Ideally one would like to maximise the performance over all ω, subject to a bound on the probability of constraint satisfaction. Given a bound $\bar{\mathbf{P}} \in [0, 1]$, this corresponds to solving the constrained optimization problem

$$\textsc{Perf}_{\max|\bar{\mathbf{P}}} = \sup_{\omega \in \mathbf{\Omega}} \textsc{Perf}(\omega) \tag{3}$$

$$\text{subject to } \bar{\mathrm{P}}(\omega) < \bar{\mathbf{P}}. \tag{4}$$

Clearly, a necessary condition for the problem to have a solution is that there exists $\omega \in \mathbf{\Omega}$ such that $\bar{\mathrm{P}}(\omega) \leq \bar{\mathbf{P}}$, or, equivalently,

$$\bar{\mathrm{P}}_{\min} = \inf_{\omega \in \mathbf{\Omega}} \bar{\mathrm{P}}(\omega) < \bar{\mathbf{P}}. \tag{5}$$

This optimization problem is generally difficult to solve, or even to approximate by randomised methods. Here we approximate this problem by an optimisation problem with penalty terms. We show that with a proper choice of the penalty term we can enforce the desired maximum bound on the probability of violating the constraint, provided that such a bound is feasible, at the price of sub-optimality in the resulting expected performance.

Let us introduce the function $u(\omega, x)$ defined as

$$u(\omega, x) = \begin{cases} \mathrm{perf}(\omega, x) + \Lambda & x \in \mathbf{X_f} \\ \\ \quad 1 & x \notin \mathbf{X_f}, \end{cases} \tag{6}$$

where $\Lambda > 1$. The parameter Λ represents a reward for constraint satisfaction. The expected value of $u(\omega, x)$ is given by

$$U(\omega) = \int_{x \in \mathbf{X}} u(\omega, x) p_\omega(x) dx \qquad \omega \in \mathbf{\Omega} \,. \tag{7}$$

Instead of the constrained optimization problem (3)–(4) we solve the unconstrained optimization problem:

$$U_{\max} = \sup_{\omega \in \mathbf{\Omega}} U(\omega). \tag{8}$$

Assume the supremum is attained and let $\bar{\omega}$ denote the optimum solution, i.e. $U_{\max} = U(\bar{\omega})$. For $\bar{\omega}$ we would like to obtain bounds on the probability of violating the constraints and the level of suboptimality of $\textsc{Perf}(\bar{\omega})$ over $\textsc{Perf}_{\max\,|\bar{\mathbf{P}}}$. A basic bound on the probability of violating the constraint at $\bar{\omega}$ is the following.

Proposition 1. $\bar{\mathrm{P}}(\bar{\omega})$ *satisfies*

$$\bar{\mathrm{P}}(\bar{\omega}) \leq \frac{1}{\Lambda} + \frac{\Lambda - 1}{\Lambda} \bar{\mathrm{P}}_{\min} . \tag{9}$$

Proof. The optimisation criterion $U(\omega)$ can be written in the form

$$\begin{aligned} U(\omega) &= \int_{x \in \mathbf{X_f}} (\mathrm{perf}(\omega, x) + \Lambda) p_\omega(x) dx + \int_{x \notin \mathbf{X_f}} p_\omega(x) dx \\ &= \textsc{Perf}(\omega) + \Lambda - (\Lambda - 1)\bar{\mathrm{P}}(\omega) . \end{aligned}$$

By the definition of $\bar{\omega}$ we have that $U(\bar{\omega}) \geq U(\omega)$ for all $\omega \in \mathbf{\Omega}$. We therefore can write

$$\textsc{Perf}(\bar{\omega}) + \Lambda - (\Lambda - 1)\bar{\mathrm{P}}(\bar{\omega}) \geq \textsc{Perf}(\omega) + \Lambda - (\Lambda - 1)\bar{\mathrm{P}}(\omega) \qquad \forall \omega$$

which can be rewritten as

$$\bar{\mathrm{P}}(\bar{\omega}) \leq \frac{\textsc{Perf}(\bar{\omega}) - \textsc{Perf}(\omega)}{\Lambda - 1} + \bar{\mathrm{P}}(\omega) \qquad \forall \omega . \tag{10}$$

Since $0 < \mathrm{perf}(\omega, x) \leq 1$, $\textsc{Perf}(\omega)$ satisfies

$$0 < \textsc{Perf}(\omega) \leq P(\omega) . \tag{11}$$

Therefore we can use (11) to obtain an upper bound on the right-hand side of (10) from which we obtain

$$\bar{\mathrm{P}}(\bar{\omega}) \leq \frac{1}{\Lambda} + \frac{\Lambda - 1}{\Lambda} \bar{\mathrm{P}}(\omega) \qquad \forall \omega \in \mathbf{\Omega}.$$

We eventually obtain (9) by taking a minimum to eliminate the quantifier on the right-hand side of the above inequality.

Proposition 1 suggests a method for choosing Λ to ensure that the solution $\bar{\omega}$ of the optimization problem will satisfy $\bar{\mathrm{P}}(\bar{\omega}) \leq \bar{\mathbf{P}}$. The following immediate corollaries make this observation more explicit.

Corollary 1. *Any*

$$\Lambda \geq \frac{1 - \bar{\mathrm{P}}_{\min}}{\bar{\mathbf{P}} - \bar{\mathrm{P}}_{\min}} \tag{12}$$

ensures that $\bar{\mathrm{P}}(\bar{\omega}) \leq \bar{\mathbf{P}}$.

Typically such a bound will not be useful in practice, since the value of $\bar{\mathrm{P}}_{\mathrm{min}}$ will be unknown. If we know that there exists a parameter $\omega \in \mathbf{\Omega}$ for which the constraints are satisfied almost surely a tighter (and potentially more useful) bound can be obtained.

Corollary 2. *If there exists $\omega \in \mathbf{\Omega}$ such that $\bar{\mathrm{P}}(\omega) = 0$, then any*

$$\varLambda \geq \frac{1}{\bar{\mathbf{P}}} \tag{13}$$

ensures that $\bar{\mathrm{P}}(\bar{\omega}) \leq \bar{\mathbf{P}}$.

For cases where the existence of such an ω cannot be guaranteed, it suffices to know $\bar{\mathrm{P}}(\omega)$ for some $\omega \in \mathbf{\Omega}$ with $\bar{\mathrm{P}}(\omega) < \bar{\mathrm{P}}$ to obtain a bound.

Corollary 3. *If there exists $\omega \in \mathbf{\Omega}$ for which $\hat{\mathrm{P}} = \bar{\mathrm{P}}(\omega)$ is known, then any*

$$\varLambda \geq \frac{1 - \hat{\mathrm{P}}}{\bar{\mathbf{P}} - \hat{\mathrm{P}}} \tag{14}$$

ensures that $\bar{\mathrm{P}}(\bar{\omega}) \leq \bar{\mathbf{P}}$.

The last bound will of course be more conservative than those of the previous two corollaries. In addition to bounds on the probability that $\bar{\omega}$ satisfies the constraints, we would also like to obtain a bound on how far the performance $\textsc{Perf}(\bar{\omega})$ is from the ideal performance $\textsc{Perf}_{\max|\bar{\mathbf{P}}}$. The following proposition provides a basic bound in this direction.

Proposition 2. *The performance of the maximiser, $\bar{\omega}$, of $U(\omega)$ satisfies*

$$\textsc{Perf}(\bar{\omega}) \geq \textsc{Perf}_{\max|\bar{\mathbf{P}}} - (\varLambda - 1)(\bar{\mathbf{P}} - \bar{\mathrm{P}}_{\mathrm{min}}). \tag{15}$$

Proof. By definition of $\bar{\omega}$ we have that $U(\bar{\omega}) \geq U(\omega)$ for all $\omega \in \mathbf{\Omega}$. In particular, we know that

$$\textsc{Perf}(\bar{\omega}) \geq \textsc{Perf}(\omega) - (\varLambda - 1)\left[\bar{\mathrm{P}}(\omega) - \bar{\mathrm{P}}(\bar{\omega})\right] \qquad \forall \omega \,:\, \bar{\mathrm{P}}(\omega) \leq \bar{\mathbf{P}}\,.$$

Taking a lower bound of the right-hand side, we obtain

$$\textsc{Perf}(\bar{\omega}) \geq \textsc{Perf}(\omega) - (\varLambda - 1)\left[\bar{\mathbf{P}} - \bar{\mathrm{P}}_{\mathrm{min}}\right] \qquad \forall \omega \,:\, \bar{\mathrm{P}}(\omega) \leq \bar{\mathbf{P}}\,.$$

Taking the maximum and eliminating the quantifier on the right-hand side we obtain the desired inequality.

Clearly to minimise the gap between the optimal performance and the performance of $\bar{\omega}$ we need to select $\varLambda$ as small as possible.

3 Simulation-based optimisation

In this section we recall a simulation-based procedure, to find approximate optimisers of $U(\omega)$. The only requirement for applicability of the procedure is to be able to obtain realisations of the random variable X with distribution $p_\omega(x)$ and to evaluate $u(\omega, x)$ pointwise. This optimisation procedure is in fact a general procedure for the optimisation of expected value criteria. It has been originally proposed in the Bayesian statistics literature [10].

The optimisation strategy relies on extractions of a random variable $\varOmega$ whose distribution has modes which coincide with the optimal points of $U(\omega)$. These extractions are obtained through Monte Carlo Markov Chain (MCMC) simulation [12]. The problem of optimising the expected criterion is then reformulated as the problem of estimating the optimal points from extractions concentrated around them. In the optimisation procedure, there exists a tunable trade-off between estimation accuracy of the optimiser and computational effort. In particular, the distribution of $\varOmega$ is proportional to $U(\omega)^J$ where J is a positive integer which allows the user to increase the “peakedness” of the distribution and concentrate the extractions around the modes at the price of an increased computational load. If the tunable parameter J is increased during the optimisation procedure, this approach can be seen as the counterpart of Simulated Annealing for a stochastic setting. Simulated Annealing is a randomised optimisation strategy developed to find tractable approximate solutions to complex deterministic combinatorial optimisation problems [13]. A formal parallel between these two strategies has been derived in [11].

The MCMC optimisation procedure can be described as follows. Consider a stochastic model formed by a random variable $\varOmega$, whose distribution has not been defined yet, and J conditionally independent replicas of random variable X with distribution $p_\varOmega(x)$. Let us denote $h(\omega, x_1, x_2, \ldots, x_J)$ the joint distribution of $(\varOmega, X_1, X_2, X_3, \ldots, X_J)$. It is straightforward to see that if

$$h(\omega, x_1, x_2, \ldots, x_J) \propto \prod_j u(\omega, x_j) p_\omega(x_j) \tag{16}$$

then the marginal distribution of $\varOmega$, say $h(\omega)$, satisfies

$$h(\omega) \propto \left[\int u(\omega, x) p_\omega(x) dx\right]^J = U(\omega)^J \,. \tag{17}$$

This means that if we can extract realisations of $(\varOmega, X_1, X_2, X_3, \ldots, X_J)$ then the extracted $\varOmega$’s will be concentrated around the optimal points of $U(\varOmega)$ for a sufficiently high J. These extractions can be used to find an approximate solution to the optimisation of $U(\omega)$.

Realisations of the random variables $(\varOmega, X_1, X_2, X_3, \ldots, X_J)$, with the desired joint probability density given by (16), can be obtained through Monte Carlo Markov Chain simulation. The algorithm is presented below. In the algorithm, $g(\omega)$ is known as the instrumental (or *proposal*) distribution and is freely chosen by the user; the only requirement is that $g(\omega)$ covers the support of $h(\omega)$.

MCMC algorithm

Given $\omega(k)$, $x_j(k)$, $j = 1, \ldots, J$ realisations of random variable $X(k)$ with distribution $p_{\omega(k)}(x)$, and $u_J(k) = \prod_{j=1}^{J} u(\omega(k), x_j(k))$:

1 Extract

$$\tilde{\Omega} \sim g(\omega)$$

2 Extract

$$\tilde{X}_j \sim p_{\tilde{\Omega}}(x) \quad j = 1, \ldots, J$$

and calculate

$$\tilde{U}_J = \prod_j u(\tilde{\Omega}, \tilde{X}_j)$$

3 Extract the new state of the chain as

$$[\omega(k{+}1),\ u_J(k{+}1)] = \begin{cases} [\tilde{\Omega},\ \tilde{U}_J] & \text{with probability } \rho(\omega(k), u_J(k), \tilde{\Omega}, \tilde{U}_J) \\ [\omega(k),\ u_J(k)] & \text{with probability } 1 - \rho(\omega(k), u_J(k), \tilde{\Omega}, \tilde{U}_J) \end{cases}$$

where

$$\rho(\omega, u_J, \tilde{\omega}, \tilde{u}_J) = \min\left\{1, \frac{\tilde{u}_J}{u_J}\frac{g(\omega)}{g(\tilde{\omega})}\right\}$$

This algorithm is a formulation of the Metropolis-Hasting algorithm for a desired distribution given by $h(\omega, x_1, x_2, \ldots, x_J)$ and proposal distribution given by

$$g(\omega) \prod_j p_\omega(x_j)\,.$$

In this case, the acceptance probability for the standard Metropolis-Hastings algorithm is

$$\frac{h(\tilde{\omega}, \tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_J)}{h(\omega, x_1, x_2, \ldots, x_J)} \frac{g(\omega) \prod_j p_\omega(x_j)}{g(\tilde{\omega}) \prod_j p_\omega(\tilde{x}_j)}\,.$$

By inserting (16) in the above expression one obtains the probability $\rho(\omega, u_J, \tilde{\omega}, \tilde{u}_J)$. Under minimal assumptions, the Markov Chain $\Omega(k)$ is uniformly ergodic with stationary distribution $h(\omega)$ given by (17). Results that characterise the convergence rate to the stationary distribution can be found for example in [12].

A general guideline to obtain faster convergence is to concentrate the search distribution $g(\omega)$ where $U(\omega)$ assumes nearly optimal values. The algorithm represents a trade-off between computational effort and the "peakedness" of the target distribution. This trade-off is tuned by the parameter J which is the power of the target distribution and also the number of extractions of X at each step of the chain. Increasing J concentrates the distribution more around the optimisers of $U(\omega)$, but also increases the number of simulations one needs to perform at each step. Obviously if the peaks of $U(\omega)$ are already quite sharp, this implies some advantages in terms of computation, since there is no need to

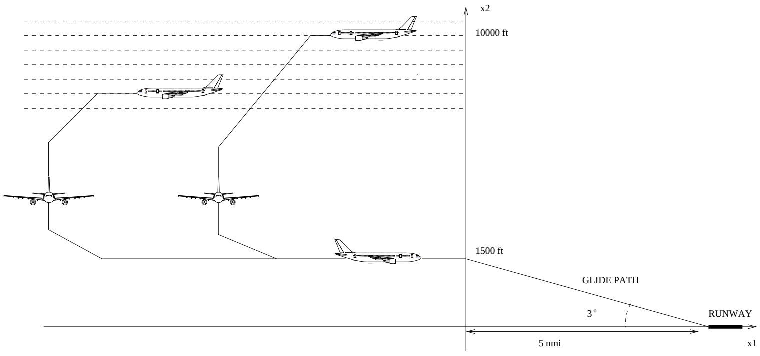

Fig. 1. Schematic representation of approach maneuver: elevation view.

increase further the peakedness of the criterion by running more simulations. For the specific $U(\omega)$ proposed in the previous section, a trade-off exists between its peakedness and the parameter Λ, which is related to probability of constraint violation. In particular, the greater Λ is the less peaked the criterion $U(\omega)$ becomes, because the relative variation of $u(\omega, x)$ is reduced, and therefore more computational effort is required for the optimisation of $U(\omega)$.

4 Air-Traffic Control in Terminal Airspace and Approach Sectors

Terminal Airspace and Approach Sectors are perhaps the most difficult scenarios in Air-Traffic Control. The management of traffic, in this case, includes tasks such as determining landing sequences and issuing of "vector" maneuvers to avoid collisions, holding the aircraft in "stacks" in case of congested traffic, etc. Here, we give a schematic representation of the problem as described in [14].

During most of the flight, aircraft stay at cruising altitudes, above 30000 *ft*. In the current organisation, the traffic at these altitudes has an *en-route* structure, which facilitates the action of Air-Traffic Control. Aircraft follow prespecified corridors at different *flight levels*. Two adjacent flight levels are separated by 100 *ft*; for example, the altitude of 30000 *ft* is denoted by FL300.

Towards the end of the flight, aircraft enter the Terminal Airspace where air-traffic controllers guide them from cruising altitudes to the entry points of the Approach Sector, between FL50 and FL150. Ideally, aircraft should enter the Approach Sector in a sequence properly spaced in time. Air-traffic controllers of the Approach Sector are then responsible for guiding the aircraft towards the proper runway. The tasks of Air-Traffic Control in the Approach Sectors include:

1) Maintain safe separation between aircraft. This is the most important requirement for safety, in any sector, during all parts of the flight. Aircraft must always maintain a minimum level of separation. A *conflict* between two aircraft is defined as the situation of loss of minimum safe separation between them. Safe

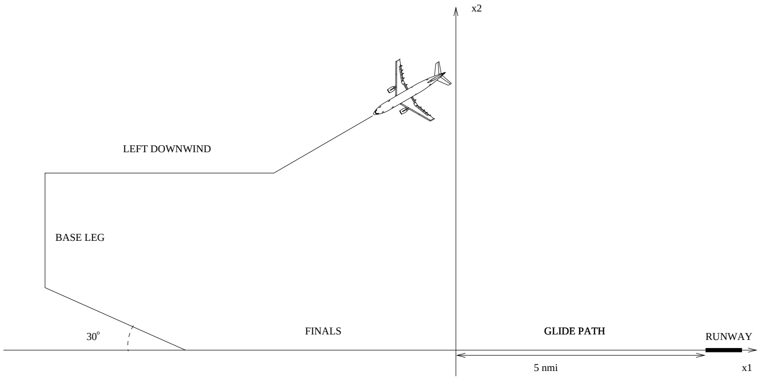

Fig. 2. Schematic representation of approach maneuver: plan view.

separation is defined by a protected zone centred around each aircraft. The level of accepted minimum separation can vary with the density of the traffic and the region of the airspace. A largely accepted shape of the protected zone is defined by a vertical cylinder, centred on the aircraft with having radius 5 *nmi* and height 2000 *ft*, so that aircraft which do not have 5 *nmi* of horizontal separation must have 1000 *ft* of vertical separation.

2) Descend aircraft from entry altitude to intercept localiser. Once aircraft have entered the Approach Sector, Air-Traffic Control must guide them from the entry altitude (FL50 to FL150) to FL15. This is the altitude at which they can intercept the *localiser*, i.e. the radio beacons which will guide them onto the runway. The point at which the aircraft will actually start the descent towards the runway is an important variable which has to be carefully chosen since it can affect the rest of the maneuver and the coordination with other aircraft. The reason is that aircraft fly following prespecified speed profiles which depend on the altitude; they fly faster at high altitudes and slower at low altitudes. This implies that aircraft, flying at lower altitudes, are slower in joining the landing queue.

3) Sequence aircraft towards the runway. The air-traffic controllers must direct the aircraft towards the runway in a properly spaced queue. This is done by adjusting the waypoints (corners) of a standard approach route (STAR) — see Figures 1 and 2. Typically the route is composed of four legs. During their descent, aircraft are first aligned, on one of the two sides of the runway, in the direction of the runway but with opposite heading. This leg is called the *left/right downwind leg*, since aircraft are expected to land against the wind. Aircraft then they perform a turn of approximately 90°, to approach the localiser. This second segment is called the *base leg*. Aircraft perform an additional turn in order to intercept the plane of the localiser with an angle of incidence of approximately 30°. The reason is that 30° is a suitable angle for pilots to perform the final turn in the direction of the runway as soon as possible when the localiser has been intercepted. It is required that aircraft intercept the localiser plane at least 5

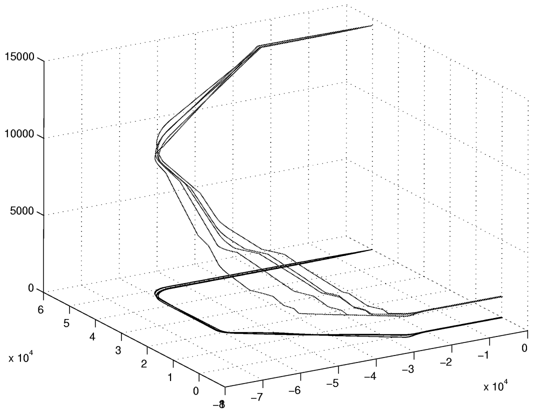

Fig. 3. Several trajectory realisations of an approach maneuver (altitude is expressed in feet and plan coordinates are expressed in meters).

nmi from the beginning of the runway and at an altitude of $1000 - -1500$ *ft*, so that they can follow a $3^\circ - -5^\circ$ glide path to the runway.

This approach geometry (which is referee to as the "trombone") is advantageous to air-traffic controllers as it allows them great flexibility in spacing aircraft by adjusting the length of the downwind leg.

5 Simulation of the air-traffic

We have developed an air-traffic simulator that simulates adequately the behaviour of a set of aircraft for Air-Traffic Control purposes [15, 16]. Realistic models of current commercial aircraft have been implemented according to [17]. The simulator contains also realistic stochastic models of the wind disturbance [18]. The models contain continuous dynamics, arising from the physical motion of the aircraft, discrete dynamics, arising from the logic embedded in the Flight Management System, and stochastic dynamics, arising from the effect of the wind and incomplete knowledge of physical parameters (for example the aircraft mass, which depends on fuel, cargo and number of passengers). The simulator has been coded in Java and can be used in different operation modes either to generate accurate data, for validation of the performance of conflict detection and resolution algorithm, or to run faster simulations of simplified models.

The nominal path for each aircraft is entered in the simulator as a sequence of waypoints. The actual trajectories of the aircraft are then a perturbed version of the nominal path, depending on the particular realisations of wind disturbances and uncertain parameters. In Figures 3 and 4 several trajectory realisations corresponding to the same nominal path are displayed. In this example, the aircraft, initially at 15000 *ft*, performs the approach maneuver described in the previous section. In addition to stochastic wind terms, uncertainty about the mass of the

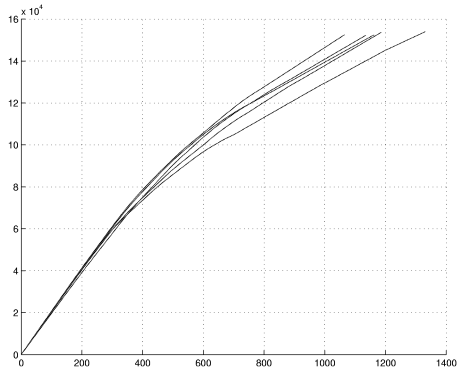

Fig. 4. Travelled distance (meters) versus time (seconds) for several trajectory realisations of an approach maneuver.

aircraft is introduced as an uniform distribution between two extreme values. The figures suggest that the resulting uncertainty in the position of aircraft is of the order of magnitude of a few kilometres.

6 Simulation of arrivals and optimisation

In this section, we optimise an approach maneuver with coordination between two aircraft. We consider Aircraft One (A1) and Aircraft Two (A2) approaching the runway as illustrated in Figure 5. In the figure, the glide path towards the runway starts at the origin of the reference frame and coordinates are expressed in meters. The aircraft are initially in level flight. The parameters of the approach maneuver are the distance, from initial position, of the start of the final descent (ω_1) and the length of the downwind leg (ω_2).

The initial position of A1 is [0 50000] and altitude 10000 *ft*. The approach maneuver of this aircraft is fixed to $\bar{\omega}_1 = 30000$ and $\bar{\omega}_2 = 50000$. The initial position of A2 is [0 50000] and altitude 10000 *ft*. The parameters of its approach maneuver will be selected using the optimization algorithm. The range of the optimisation parameters is $\omega_2 \in [35000,\ 60000]$ and $\omega_1 \in [0,\ \omega_2]$. The motion of the two aircraft is affected by the same uncertainty as in the simulation example of Section 5.

We assume that the performance of the approach maneuver is measured by the arrival time of A2 at the start of the glide path (T_2). The measure of performance is given by perf $= e^{-a \cdot T_2}$ with $a = 5 \cdot 10^{-4}$. The constraint is that the trajectory of A2 is not in conflict with the trajectory of A1. Aircraft 2 must also reach the altitude of 1500 *ft* before the start of the glide path. We optimise initially with an upper bound on probability of constraint violation given by $\bar{\mathbf{P}} = 0.3$. It is easy to see that there exists a maneuver in the set of optimisation parameters that gives negligible conflict probability. Therefore, based on inequality (13), we

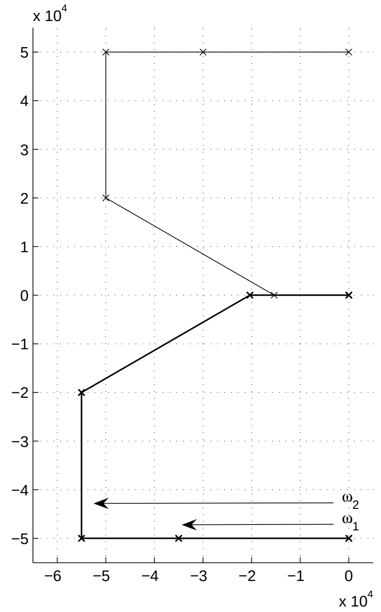

Fig. 5. Approach maneuvers for A1 (thin) and A2 (bold).

select $\Lambda = 3.5$ in the optimisation criterion.

The results of the optimisation procedure are illustrated in Figures 6-9 for different values of J and proposal distribution g. Each figure shows the scatter plot of the accepted parameters during MCMC simulation. In all cases the first 10% of accepted parameters was discarded as a *burn in* period to allow convergence of the chain to its stationary distribution. Figure 6 illustrates the case $J = 5$ and proposal distribution g uniform over the parameter space. In this case, the ratio between accepted and proposed parameters during MCMC simulation was 0.36. A region characterised by a low density of accepted parameters can be clearly seen in the figure. These are parameters which correspond to a conflicting maneuver where the aircraft are performing an almost symmetrical approach. The figure also shows two distinct “clouds” of accepted maneuvers. They correspond to a discrete choice that the air traffic controller has to make: either land A2 before A1 (bottom right cloud) or land A1 before A2 (top left cloud).

Figure 7 illustrates the case $J = 50$ and g uniform. In this case the ratio between accepted and proposed states was 0.08. The case $J = 50$ is illustrated also in Figure 8. In this case, however, the proposal distribution g was a sum of 100 Gaussian distributions $N(\mu, \sigma^2 I)$ with variance $\sigma^2 = 10^5\, m^2$. The means of Gaussian distributions were 100 parameters chosen from those accepted in the MCMC simulation for $J = 5$ and belonging to the cloud corresponding to “A2 arrives before A1”. This appears to be the more promising cloud because of the higher density of points; recall that the distribution of accepted points is

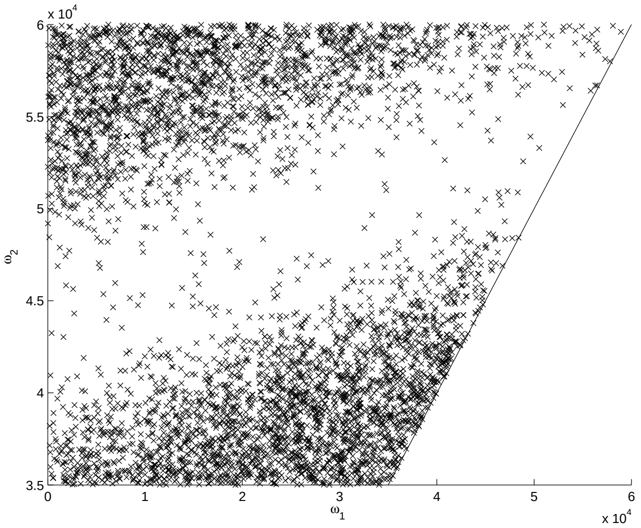

Fig. 6. Accepted states (50000) during MCMC simulation ($J = 5$, g uniform).

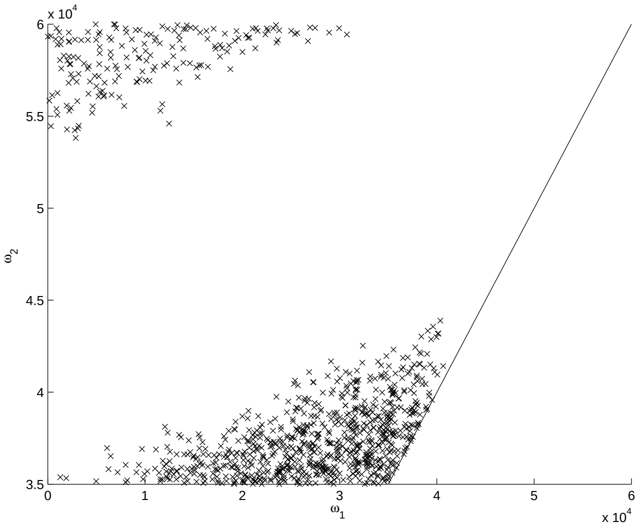

Fig. 7. Accepted states (1000) during MCMC simulation ($J = 50$, g uniform).

concentrated around the maximisers of $U(\omega)$. The choice of this proposal distribution gives clear computational advantages since less computational time is spent searching over regions of non optimal parameters. For this choice of g the ratio between accepted and proposed parameters increased to 0.2. Figure 9 illustrates the case $J = 100$ and proposal distribution constructed as before from states accepted for $J = 50$. In this case the ratio between accepted and

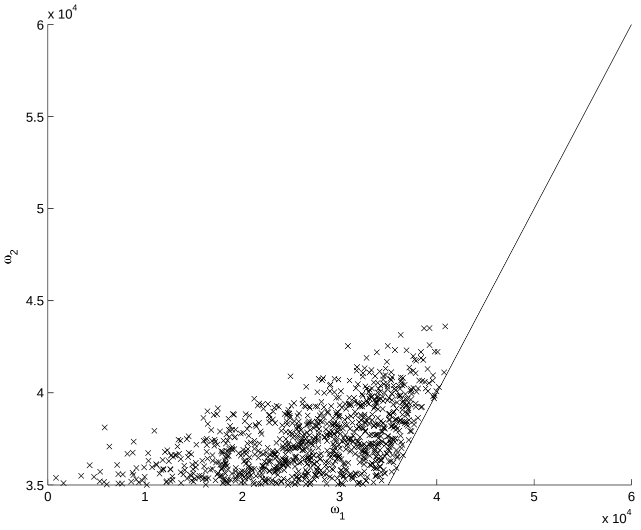

Fig. 8. Accepted states (1000) during MCMC simulation ($J = 50$, g sum of Gaussian distributions).

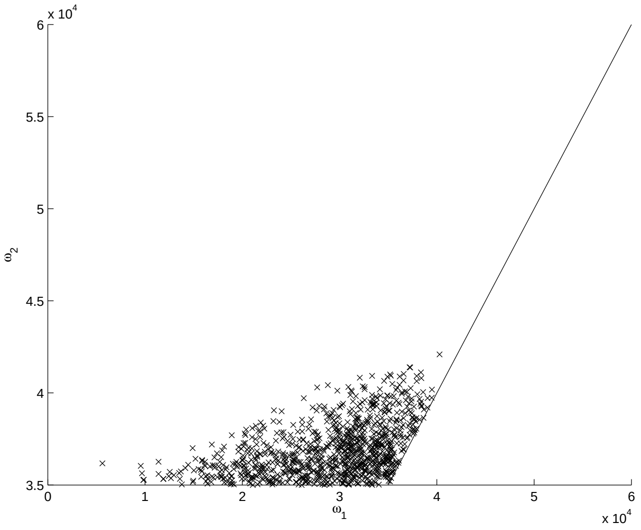

Fig. 9. Accepted states (1000) during MCMC simulation ($J = 100$, g = sum of Gaussian distributions).

proposed parameters was 0.5. Figure 9 indicates that a nearly optimal maneuver is $\omega_1 = 35000$ and $\omega_2 = 35000$. The probability of conflict for this maneuver, estimated by 1000 Monte-Carlo runs, was zero.

7 Conclusions

In this paper we illustrated a Monte Carlo approach to air traffic conflict resolution in a stochastic setting. The main motivation for our approach is to enable the use of realistic stochastic hybrid models of aircraft flight; Monte Carlo methods appear to be the only ones that allow such models. We have formulated conflict resolution as the optimisation of an expected value criterion with probabilistic constraints. Here, a penalty formulation of the problem has been considered which guarantees constraint satisfaction but delivers a suboptimal solution. A side effect of the optimization procedure is that structural differences between maneuvers (e.g. the sequencing choices in the landing example considered here) are highlighted as "clouds" of maneuvers accepted by the algorithm.

Our current research is concerned with overcoming the suboptimality imposed by the need to provide constraint satisfaction guarantees. A possible way is to use the Monte Carlo Markov Chain procedure presented in Section 3 to obtain optimisation parameters that satisfy the constraint and then to optimise over this set in a successive step.

Acknowledgement: This work was supported by the European Commission under project HYBRIDGE IST-2001-32460 and EUROCONTROL under contract C20051E/BM/03. The authors would like to thank EUROCONTROL Experimental Centre for having provided places to 'Air-Traffic Control Familiarisation Course' from which the case study considered in this paper has been inspired.

References

1. Kuchar, J., Yang, L.: A review of conflict detection and resolution methods. IEEE Transactions on Intelligent Transportation Systems **1** (2000) 179–189
2. Frazzoli, E., Mao, Z., Oh, J., Feron, E.: Aircraft conflict resolution via semi-definite programming. AIAA Journal of Guidance, Control, and Dynamics **24** (2001) 79–86
3. Hu, J., Prandini, M., Sastry, S.: Optimal Coordinated Maneuvers for Three-Dimensional Aircraft Conflict Resolution. AIAA Journal of Guidance, Control and Dynamics **25** (2002)
4. Tomlin, C., Pappas, G., Sastry, S.: Conflict resolution for Air Traffic Management: a case study in multi-agent hybrid systems. IEEE Transactions on Automatic Control **43** (1998) 509–521
5. Paielli, R., Erzberger, H.: Conflict probability estimation for free flight. Journal of Guidance, Control and Dynamics **20** (1997) 588–596 Available from World wide Web: http://www.ctas.arc.nasa.gov/publications/papers/.
6. Irvine, R.: A geometrical approach to conflict probability estimation. In: 4th USA/Europe Air Traffic Management R&D seminar, Santa Fe (2001) Available from World Wide Web: http://atm2001.eurocontrol.fr/finalpapers/pap137.pdf.
7. Krystul, J., Bagchi, A., Blom, H.: Risk decomposition and assessment methods. Technical Report WP8, Deliverable D8.1, HYBRIDGE (2003) Available from World Wide Web: http://www.nlr.nl/public/hosted-sites/hybridge/.

8. Hu, J., Prandini, M., Sastry, S.: Aircraft conflict detection in presence of spatially correlated wind perturbations. In: AIAA Guidance, Navigation and Control Conf., Austin, Texas, USA (2003)
9. Lecchini, A., Glover, W., Lygeros, J., Maciejowski, J.: Air Traffic Control with an expected value criterion. Technical Report WP5, Deliverable D5.2, HYBRIDGE (2004) Accepted for presentation at IFAC World Congress 2005. Available from World Wide Web: http://www.nlr.nl/public/hosted-sites/hybridge.
10. Mueller, P.: Simulation based optimal design. In: Bayesian Statistics 6, J.O. Berger, J.M. Bernardo, A.P. Dawid and A.F.M. Smith (eds.), Oxford University Press (1999) 459–474
11. Mueller, P., Sanso, B., De Iorio, M.: Optimal Bayesian design by inhomogeneous Markov chain simulation. Technical report (2003) Available from World Wide Web: http://www.ams.ucsc.edu.
12. Robert, C., Casella, G.: Monte Carlo Statistical Methods. Springer-Verlag (1999)
13. van Laarhoven, P., Aarts, E.: Simulated Annealing: Theory and Applications. D.Reidel Publishing Company (1987)
14. EUROCONTROL Experimental Centre: Air-Traffic Control Familiarisation Course. (2004)
15. Glover, W., Lygeros, J.: A multi-aircraft model for conflict detection and resolution algorithm validation. Technical Report WP1, Deliverable D1.3, HYBRIDGE (2003) Available from World Wide Web: http://www.nlr.nl/public/hosted-sites/hybridge/.
16. Glover, W., Lygeros, J.: A stochastic hybrid model for air traffic control simulation. In: Hybrid Systems: Computation and Control, 7th International Workshop. Volume 2993 of Lecture Notes in Computer Science., Philadelphia, PA, USA, Springer (2004) 372–386
17. EUROCONTROL Experimental Centre: User Manual for the Base of Aircraft Data (BADA) — Revision 3.3. (2002) Available from World Wide Web: `http://www.eurocontrol.fr/projects/bada/`.
18. Cole, R., Richard, C., Kim, S., Bailey, D.: An assessment of the 60 km rapid update cycle (ruc) with near real-time aircraft reports. Technical Report NASA/A-1, MIT Lincoln Laboratory (July 15, 1998)