Chapter 10

Free Flight Collision Risk Estimation by Sequential Monte Carlo Simulation

Henk A.P. Blom
National Aerospace Laboratory NLR

Jaroslav Krystul
University of Twente

G.J. (Bert) Bakker, Margriet B. Klompstra
National Aerospace Laboratory NLR

Bart Klein Obbink
National Aerospace Laboratory NLR

10.1 Introduction

10.1.1 Safety Verification of Free Flight Air Traffic

Technology allows aircraft to broadcast information about its own-ship position and velocity to surrounding aircraft, and to receive similar information from surrounding aircraft. This development has stimulated the rethinking of the overall concept for future Air Traffic Management (ATM), e.g., to transfer responsibility for conflict prevention from ground to air. As the aircrews thus obtain the freedom to select their trajectory, this conceptual idea is called free flight [57]. It changes ATM in such a fundamental way, that one could speak of a paradigm shift: centralised control becomes distributed, responsibilities transfer from ground to air, fixed air traffic routes are removed, and appropriate new technologies are brought in. Each aircrew has the responsibility to timely detect and solve conflicts, thereby assisted

by navigation means, surveillance processing, and conflict resolution systems. Due to the potentially many aircraft involved, the system is highly distributed. This free flight concept idea has motivated the study of multiple operational concepts and implementation choices [33], [37], [41], [44], [54]. One of the key outstanding issues is the safety verification of free flight design, and in particular when air traffic demand is high.

For en-route traffic, the International Civil Aviation Organisation (ICAO) has established thresholds on the acceptable probability of a mid-air collision. Hence, the en-route free flight safety verification problem consists of estimating the collision probability of free flight operations, and subsequently comparing this estimated level with the ICAO established thresholds [34]. The civil aviation community also has established some approximate models to estimate (an upper-bound of) the risk of collision between aircraft flying within a given parallel route structure [32], [38], [40]. Additional methods have been exploited to develop some valuable extensions of this approach, e.g., using fault trees see [22] and using stochastic analysis and Monte Carlo (MC) simulation [3], [4], [29]. Andrews et al. [1] have shown how statistical data in combination with a fault tree of the functionalities of the advanced operation can serve to predict how reliability of free flight supported systems impact contributions to collision risk of an advanced operation [24], neglecting other contributions to collision risk. The challenge is to analyse the risk of collision between aircraft in free flight without the limitation of a fixed route structure. We aim to improve this situation by developing a novel approach toward collision risk assessment for advanced air traffic designs. An initial shorter paper on this development is [7].

10.1.2 Probabilistic Reachability Analysis

In air traffic, a mid-air collision event happens at the moment in time that the physical shapes of two airborne aircraft hit each other. Such event can be represented as a moment in time that the joint state of aircraft involved hit a certain subset of their joint state space. With this, the problem to estimate the probability of collision between two aircraft within a finite time period is to analyse the probability that this collision subset is reached by their joint aircraft state within that time period. In systems theory, the estimation of the probability of reaching a given subset of the state space within a given time period is known as a problem of probabilistic reachability analysis, e.g., see [49].

Hu et al. [39], Prandini and Hu [56], and Chapter 5 of this volume apply probabilistic reachability analysis for the development of a grid based computation to evaluate the probability that two aircraft come closer to each other than some established minimum separation criteria. The numerical challenge of this problem, however, differs from free flight collision risk estimation on the following aspects:

- The collision subset is more than three orders smaller in volume than the conflict subset is.
- A safety directed model of an air traffic operation includes per aircraft also the states of the technical systems and the pilot models, which increases the size

of the state space by many orders in magnitude.

- There are multiple aircraft, not just two, inducing a non-zero probability of a chain collision.

If we would follow the numerical approach of Chapter 5 of this volume to estimate collision risk in free flight operations, then these aspects would imply a blow up of the number of grid points to a practically unmanageably large number.

In most safety critical industries, e.g., nuclear, chemical, etc., reachability analysis is addressed by methods that are known as dynamical approaches towards Probabilistic Risk Analysis (PRA). For an overview of these dynamical methods in PRA, see [50]. These dynamical PRA methods make explicitly use of the fact that between two discrete events the dynamical evolution satisfies an ordinary differential equation. Essentially this means that these dynamical PRA methods apply to the class of stochastic hybrid system models that do not involve Brownian motion. In the hybrid systems control community these are known as piecewise deterministic Markov process [9], [17].

For proper safety modelling of air traffic operations, however, it is needed to incorporate Brownian motion in the piecewise deterministic Markov process models, e.g., to represent the effect of random wind disturbances on aircraft trajectories [55]. The class of systems which incorporates Brownian motion within piecewise deterministic Markov processes, has been defined as a Generalised Stochastic Hybrid System (GSHS) [10]. GSHS is the class of non-linear stochastic continuous-time hybrid dynamical systems, having a hybrid state consisting of two components: a continuous valued state component and a discrete valued state component. The continuous state evolves according to an SDE whose vector field and drift factor depend on both hybrid state components. Switching from one discrete state to another discrete state is governed by a probability law or occurs when the continuous state hits a pre-specified boundary. Whenever a switching occurs, the hybrid state is reset instantly to a new state according to a probability measure which depends itself on the past hybrid state. GSHS contain, as a subclass, the switching diffusion process, the probabilistic reachability of which is studied in Chapter 5 of this volume. Important complementary dynamics is induced by the interaction between the hybrid state components.

10.1.3 Sequential Monte Carlo Simulation

Shah et al. [58] explain very well that the advantage of using MC simulation in evaluating advanced operations is its capability to identify and evaluate emergent behaviour, i.e., novel behaviour which is exhibited by complex safety critical systems and emerges from the combined dynamical actions and reactions by individual systems and humans within the system. This emergent behaviour typically cannot be foreseen and evaluated by examining the individuals behaviour alone. Shah et al. [58] explain that agent based MC simulation is able to predict the impact of revolutionary changes in air transportation; it integrates cognitive models of technology behaviour and description of their operating environment. Simulation of these individual models acting together can predict the results of completely new transforma-

tions in procedures and technology. Their MC simulations reach up to the level of novel emerging hazardous events. For safety risk assessment however, it is required to go further with the MC simulations up to the level of emerging catastrophic events. In en-route air traffic these catastrophic events are mid-air collisions.

A seemingly simple approach toward the estimation of mid-air collision probability is to run many MC simulations with a free flight stochastic hybrid model and count the fraction of runs for which a collision occurs. The advantage of a MC simulation approach is that this does not require specific assumptions or limitations regarding the behaviour of the system under consideration. A key problem is that in order to obtain accurate estimates of rare event probabilities, say about 10^{-9} per flying hour, it is required to simulate 10^{11} flying hours or more. Taking into account that an appropriate free flight model is large, this would require an impractically huge simulation time.

Del Moral and co-workers [13], [14], [18] developed a sequential MC simulation approach for estimating small reachability probabilities, including a characterisation of convergence behaviour. The idea behind this approach is to express the small probability to be estimated as the product of a certain number of larger probabilities, which can be efficiently estimated by the MC approach. This can be achieved by introducing sets of intermediate states that are visited one set after the other, in an ordered sequence, before reaching the final set of states of interest. The reachability probability of interest is then given by the product of the conditional probabilities of reaching a set of intermediate states given that the previous set of intermediate states has been reached. Each conditional probability is estimated by simulating in parallel several copies of the system, i.e., each copy is considered as a particle following the trajectory generated through the system dynamics. To ensure unbiased estimation, the simulated process must have the strong Markov property. Hence, we extend the approach of [13]–[14] for application to free flight, and illustrate its application to free flight scenarios.

10.1.4 Development of MC Simulation Model

For the modelling of accident risk of safety-critical operations in nuclear and chemical industries, the most advanced approaches use Petri nets as model specification formalism, and stochastic analysis and Monte Carlo simulation to evaluate the specified model, e.g., see [50]. Since their introduction as a systematic way to specify large discrete event systems that one meets in computer science, Petri nets have shown their usefulness for many practical applications in different industries, e.g., see [16]. Various Petri net extensions and generalisations and numerous supporting computer tools have been developed, which further increased their modelling opportunities. Nevertheless, literature on Petri nets appeared to fall short for modelling the class of GSHS [10] that was needed to model air traffic safety aspects well [55].

Cassandras and Lafortune [12] provide a control systems introduction to Petri nets and a comparison with other discrete event modelling formalisms like automata. Both Petri nets and automata have their specific advantages. Petri net is more power-

ful in the development of a model of a complex system, whereas automata are more powerful in supporting analysis. In order to combine the advantages offered by both approaches, there is need for a systematic way of transforming a Petri net model into an automata model. Such a transformation would allow using Petri nets for the specification and automata for the analysis. For a timed or stochastic Petri net with a bounded number of tokens and deterministic or Poisson process firing, such a transformation exists [12]. In order to make the Petri net formalism useful in modelling air traffic operations, we need an extension of the Petri net formalism including a one-to-one transformation to and from GSHS. Everdij and Blom [26]–[28] have developed such extension in the form of (Stochastically and) Dynamically Coloured Petri Net, or for short (S)DCPN.

Jensen [42] introduced the idea of attaching to each token in a basic Petri net (i.e., with logic transitions only), a colour which assumes values from a finite set. Tokens and the attached colours determine which transitions are enabled. Upon firing by a transition, new tokens and attached colours are produced as a function of the removed tokens and colours. Haas [36] extended this colour idea to (stochastically) timed Petri nets where the time period between enabling and firing depends of the input tokens and their attached colours. In [36], [42] a colour does not change as long as the token to which it is attached remains at its place. Everdij and Blom [26], [27] defined a Dynamically Coloured Petri Net (DCPN) by incorporating the following extensions: (1) a colour assumes values from a Euclidean state space, its value evolves as solution of a differential equation and influences the time period between enabling and firing; (2) the new tokens and attached colours are produced as random functions of the removed tokens and colours. An SDCPN extends an DCPN in the sense that colours evolve as solutions of a stochastic differential equation [28].

This chapter is organised as follows. Section 10.2 develops the sequential MC simulation approach toward probabilistic reachability analysis of a GSHS model of free flight air traffic. Section 10.3 explains how an initial GSHS model has been developed for a specific free flight air traffic concept of operation. Section 10.4 applies the sequential MC simulation approach of Section 10.2 to the GSHS model of Section 10.3. Section 10.5 draws conclusions.

10.2 Sequential MC Estimation of Collision Risk

10.2.1 Stochastic Hybrid Process Considered

Throughout this and the following sections, all stochastic processes are defined on a complete stochastic basis $(\Omega,\mathcal{F},\mathbb{F},\mathsf{P},\mathsf{T})$ with $(\Omega,\mathcal{F},\mathsf{P})$ a complete probability space, and $\mathbb{F}$ an increasing sequence of sub-σ-algebra's on the positive time line $\mathsf{T}=\mathbb{R}_+$, i.e., $\mathbb{F}\triangleq\{\mathcal{J},(\mathcal{F}_t,t\in\mathsf{T}),\mathcal{F}\}$, $\mathcal{J}$ containing all P-null sets of $\mathcal{F}$ and $\mathcal{J}\subset\mathcal{F}_s\subset\mathcal{F}_t\subset\mathcal{F}$ for every $s<t$.

We assume that air traffic operations are represented by a stochastic hybrid pro-

cess $\{x_t, \theta_t\}$ which satisfies the strong Markov property. In [10], [11], [46] and in Chapter 2 of this volume, this property has been shown to hold true for the processes generated as execution of a GSHS. For an N-aircraft free flight traffic scenario the stochastic hybrid process $\{x_t, \theta_t\}$ consists of Euclidean valued components $x_t \triangleq \text{Col}\{x_t^0, x_t^1, \ldots, x_t^N\}$ and discrete valued components $\theta_t \triangleq \text{Col}\{\theta_t^0, \theta_t^1, \ldots, \theta_t^N\}$, where x_t^i assumes values from $\mathbb{R}^{n_i}$, and θ_t^i assumes values from a finite set (M^i). Physically, $\{x_t^i, \theta_t^i\}$, $i = 1, \ldots, N$, is the hybrid state process related to the i-th aircraft, and $\{x_t^0, \theta_t^0\}$ is a hybrid state process of all non-aircraft components. The process $\{x_t, \theta_t\}$ is $\mathbb{R}^n \times M$-valued with $n = \sum_{i=0}^N n_i$ and $M = \bigotimes_{i=0}^N M^i$. In order to model collisions between aircraft, we introduce mappings from the Euclidean valued process $\{x_t\}$ into the relative position and velocity between a pair of two aircraft (i, j). The relative horizontal position is obtained through the mapping $y^{ij}(x_t)$, the relative horizontal velocity is obtained through the mapping $v^{ij}(x_t)$. The relative vertical position is obtained through the mapping $z^{ij}(x_t)$, and relative vertical rate of climb is obtained through the mapping $r^{ij}(x_t)$. The relation between the position and velocity mappings satisfies:

$$dy^{ij}(x_t) = v^{ij}(x_t)\,dt \tag{10.1}$$

$$dz^{ij}(x_t) = r^{ij}(x_t)\,dt. \tag{10.2}$$

A collision between aircraft (i, j) means that the process $\{y^{ij}(x_t), z^{ij}(x_t)\}$ hits the boundary of an area where the distance between aircraft i and j is smaller than their physical size. Under the assumption that the length of an aircraft equals the width of an aircraft, and that the volume of an aircraft is represented by a cylinder the orientation of which does not change in time, then aircraft (i, j) have zero separation if $x_t \in D^{ij}$ with:

$$D^{ij} = \left\{x \in \mathbb{R}^n; |y^{ij}(x)| \leq (l_i + l_j)/2 \text{ AND } |z^{ij}(x)| \leq (s_i + s_j)/2\right\}, \quad i \neq j \tag{10.3}$$

where l_j and s_j are length and height of aircraft j. For simplicity we assume that all aircraft have the same size, by which (10.3) becomes:

$$D^{ij} = \left\{x \in \mathbb{R}^n; |y^{ij}(x)| \leq l \text{ AND } |z^{ij}(x)| \leq s\right\}, \quad i \neq j \tag{10.4}$$

Although all aircraft have the same size, notice that in (10.4), D^{ij} still depends of (i, j). If x_t hits D^{ij} at time τ^{ij}, then we say a collision event between aircraft (i, j) occurs at τ^{ij}, i.e.,

$$\tau^{ij} = \inf\{t > 0\,;\, x_t \in D^{ij}\}, \quad i \neq j \tag{10.5}$$

Next we define the first moment τ^i of collision with any other aircraft, i.e.,

$$\tau^i = \inf_{j \neq i}\{\tau^{ij}\} = \inf_{j \neq i}\{t > 0\,;\, x_t \in D^{ij}\} = \inf\{t > 0\,;\, x_t \in D^i\}, \tag{10.6}$$

with $D^i \triangleq \cup_{j \neq i} D^{ij}$. From this moment τ^i on, we assume that the differential equations for $\{x_t^i, \theta_t^i\}$ stop evolving.

An unbiased estimation procedure of the risk would be to simulate many times aircraft i amidst other aircraft over a period of length T and count all cases in which the realization of the moment τ^i is smaller than T. An estimator for the collision risk of aircraft i per unit T of time then is the fraction of simulations for which $\tau^i < T$.

10.2.2 Risk Factorisation Using Multiple Conflict Levels

Cérou et al. [13]–[14] developed a novel way of speeding up Monte Carlo simulation to estimate the probability that an $\mathbb{R}^n$-valued strong Markov process x_t hits a given "small" subset $D \in \mathbb{R}^n$ within a given time period $(0,T)$. This method essentially consists of taking advantage of an appropriately nested sequence of closed subsets of $\mathbb{R}^n$: $D = D_m \subset D_{m-1} \subset \ldots \subset D_1$, and then start simulation from outside D_1, and subsequently simulate from D_1 to D_2, from D_2 to D_3, ..., and finally from D_{m-1} to D_m. Krystul and Blom [45], [47] extended this Interacting Particle System (IPS) approach to switching diffusions. For probabilistic reachability analysis of an air traffic design, this IPS approach is now further extended to the class of SHS the execution of which satisfies the strong Markov property as adressed in Chapter 2 of this volume.

Prior to a collision of aircraft i with aircraft j, a sequence of conflicts ranging from long term to short term always occurs. In order to incorporate this explicitly in the MC simulation, we formalise this sequence of conflict levels through a sequence of closed subsets of $\mathbb{R}^n$: $D^{ij} = D^{ij}_m \subset D^{ij}_{m-1} \subset \ldots \subset D^{ij}_1$ with for $k = 1, \ldots, m$:

$$\begin{aligned} D^{ij}_k = \big\{ x \in \mathbb{R}^n; |y^{ij}(x) + \Delta v^{ij}(x)| \le d_k \ \text{AND} \\ |z^{ij}(x) + \Delta r^{ij}(x)| \le h_k, \ \text{for some } \Delta \in [0, \Delta_k] \big\}, \end{aligned} \tag{10.7}$$

for $i \neq j$, with d_k, h_k and Δ_k the parameters of the conflict definition at level k, and with $d_m = l$, $h_m = s$ and $\Delta_m = 0$, and with $d_{k+1} \le d_k$, $h_{k+1} \le h_k$ and $\Delta_{k+1} \le \Delta_k$. If x_t hits D^{ij}_k at time τ^{ij}_k, then we say the first level k conflict event between aircraft (i,j) occurs at moment τ^{ij}_k, i.e.,

$$\tau^{ij}_k = \inf\{t > 0 \,;\, x_t \in D^{ij}_k\}. \tag{10.8}$$

Similarly as we did for reaching the collision level by aircraft i, we consider the first moment τ^i_k that aircraft i reaches conflict level k with any of the other aircraft, i.e.,

$$\tau^i_k = \inf_{j \neq i}\{\tau^{ij}_k\} = \inf_{j \neq i}\{t > 0 \,;\, x_t \in D^{ij}_k\} = \inf\{t > 0 \,;\, x_t \in D^i_k\}, \tag{10.9}$$

with $D^i_k \stackrel{\Delta}{=} \cup_{j \neq i} D^{ij}_k$.

Next, we define $\{0,1\}$-valued random variables $\{\chi^i_k, k = 0, 1, \ldots, m\}$ as follows:

$$\begin{aligned} \chi^i_k &= 1, \text{ if } \tau^i_k < T \text{ or } k = 0 \\ &= 0, \text{ else.} \end{aligned}$$

By using this χ^i_k definition we can write the probability of collision of aircraft i with any of the other aircraft as a product of conditional probabilities of reaching the next conflict level given the current conflict level has been reached:

$$\begin{aligned} \mathbb{P}\left(\tau^i_m < T\right) = \mathbb{E}\left[\chi^i_m\right] = \mathbb{E}\left[\prod_{k=1}^m \chi^i_k\right] = \prod_{k=1}^m \mathbb{E}\left[\chi^i_k \,|\, \chi^i_{k-1} = 1\right] \\ = \prod_{k=1}^m \mathbb{P}\left(\tau^i_k < T \,|\, \tau^i_{k-1} < T\right) = \prod_{k=1}^m \gamma^i_k, \end{aligned} \tag{10.10}$$

with $\gamma_k^i \triangleq \mathbb{P}\left(\tau_k^i < T \,|\, \tau_{k-1}^i < T\right)$.

With this, the problem can be seen as one to estimate the conditional probabilities γ_k^i in such a way that the product of these estimators is unbiased. Because of the multiplication of the various individual γ_k^i estimators, which depend on each other, in general such a product may be heavily biased. The key novelty in [13] was to show that such a product may be evaluated in an unbiased way when $\{x_t\}$ makes part of a larger stochastic process that satisfies the strong Markov property. This approach is explained next.

10.2.3 Characterisation of the Risk Factors

Let us denote $E' = \mathbb{R}^{n+1} \times M$, and let $\mathcal{E}'$ be the Borel σ-algebra of E'. For any $B \in \mathcal{E}'$, $\pi_k^i(B)$ denotes the conditional probability of $\xi_k \triangleq (\tau_k, x_{\tau_k}, \theta_{\tau_k}) \in B$ given $\chi_l^i = 1$ for $1 \le l \le k$.

Define $Q_k^i = (0,T) \times D_k^i \times M$, $k = 1, \ldots, m$. Then the estimation of the probability for ξ_k to arrive at the k-th nested Borel set Q_k^i is characterised through the following recursive set of transformations

$$\begin{array}{ccccc} & & \overset{\text{prediction}}{} & & \overset{\text{conditioning}}{} \\ \pi_{k-1}^i(\cdot) & \longrightarrow & p_k^i(\cdot) & \longrightarrow & \pi_k^i(\cdot) \\ & & \downarrow & & \\ & & \gamma_k^i & & \end{array}$$

where $p_k^i(B)$ is the conditional probability of $\xi_k \in B$ given $\chi_l^i = 1$ for $0 \le l \le k-1$.

Because $\{x_t, \theta_t\}$ is a strong Markov process, $\{\xi_k\}$ is a Markov sequence. Hence the one step prediction of ξ_k satisfies a Chapman-Kolmogorov equation:

$$p_k^i(B) = \int_{E'} p_{\xi_k \,|\, \xi_{k-1}}(B|\xi)\, \pi_{k-1}^i(d\xi) \text{ for all } B \in \mathcal{E}'. \tag{10.11}$$

Next we characterise the conditional probability of reaching the next level

$$\begin{aligned} \gamma_k^i &= \mathbb{P}\left(\tau_k^i < T \mid \tau_{k-1}^i < T\right) \\ &= \mathbb{E}\left[\chi_k^i \mid \chi_{k-1}^i = 1\right] \\ &= \int_{E'} 1_{\{\xi \in Q_k^i\}}\, p_k^i(d\xi), \end{aligned} \tag{10.12}$$

and the conditioning satisfies:

$$\pi_k^i(B) = \frac{\int_B 1_{\{\xi \in Q_k^i\}}\, p_k^i(d\xi)}{\int_{E'} 1_{\{\xi' \in Q_k^i\}}\, p_k^i(d\xi')} \text{ for all } B \in \mathcal{E}'. \tag{10.13}$$

With this, each of the m terms γ_k^i in (10.10) is characterised as a solution of a sequence of "filtering" kind of equations (10.11)–(10.13). However, an important difference with "filtering" equations is that (10.11)–(10.13) are ordinary integral equations, i.e., they have no stochastic term entering them.

10.2.4 Interacting Particle System Based Risk Estimation

Based on this theory, an Interacting Particle System (IPS) simulation algorithm is explained next for an arbitrary hybrid state strong Markov process model of air traffic. The transformations (10.11)–(10.13) lead to the IPS algorithm of [13] to estimate $\mathbb{P}(\tau_m^i < T)$, where $\overline{\gamma}_k^i$, $\overline{p}_k^i$ and $\overline{\pi}_k^i$ denote the numerical approximations of γ_k^i, p_k^i and π_k^i respectively. When simulating from D_{k-1}^i to D_k^i, a fraction $\overline{\gamma}_k^i$ of the Monte Carlo simulated trajectories only will reach D_k^i within the time period $(0,T)$.

IPS Step 0. Initial sampling for $k = 0$.

- For $l = 1,\ldots,N_p$ generate initial state value outside Q_1^i by independent drawings (x_0^l, θ_0^l) from $p_{x_0,\theta_0}(\cdot)$ and set $\xi_0^l = (0, x_0^l, \theta_0^l)$.
- For $l = 1,\ldots,N_p$, set the initial weights: $\omega_0^l = 1/N_p$.
- Then $\overline{\pi}_0^i = \sum_{l=1}^{N_p} \omega_0^l \, \delta_{\{\xi_0^l\}}$.

IPS Iteration cycle: For $k = 1,\ldots,m$ perform step 1 (prediction), step 2 (assess fraction), step 3 (conditioning), and step 4 (resampling).

IPS Step 1. Prediction of $\pi_{k-1}^i \longrightarrow p_k^i$, based on (10.11);

- For $l = 1,\ldots,N_p$ simulate a new path of the hybrid state Markov process, starting at ξ_{k-1}^l until the k-th set Q_k^i is hit or $t = T$ (the first component of ξ_k^l counts time).
- This yields new particles $\{\hat{\xi}_k^l, \omega_{k-1}^l\}_{l=1}^{N_p}$.
- $\overline{p}_k^i$ is the empirical distribution associated with the new cloud of particles: $\overline{p}_k^i = \sum_{l=1}^{N_p} \omega_{k-1}^l \delta_{\xi_k^l}$.

IPS Step 2. Assess fraction γ_k^i, based on (10.12);

- The particles that do not reach the set Q_k^i are killed, i.e., we set $\hat{\omega}_k^l = 0$ if $\hat{\xi}_k^l \notin Q_k^i$ and $\hat{\omega}_k^l = 0$ if $\hat{\omega}_k^l = w_{k-1}^l$ if $\hat{\xi}_k^l \in Q_k^i$.
- Approximation: $\gamma_k^i \approx \overline{\gamma}_k^i = \sum_{l=1}^{N_p} \hat{\omega}_k^l$. If all particles are killed, i.e., $\overline{\gamma}_k^i = 0$, then the algorithm stops without $\mathbb{P}(\tau^i < T)$ estimate.

IPS Step 3. Conditioning of $p_k^i \longrightarrow \pi_k^i$, based on (10.13);

The non-killed particles form a set S_k^i, i.e., iff $\hat{\xi}_k^l \in Q_k^i$, then particle $\{\hat{\xi}_k^l, \hat{\omega}_k^l\}$ is stored in S_k^i.

Renumbering the particles in S_k^i yields a set of particles $\{\tilde{\xi}_k^l, \tilde{\omega}_k^l\}_{l=1}^{N_{S_k}}$ with N_{S_k} the number of particles in S_k^i. Hence, we also have $\overline{\gamma}_k^i = \sum_{l=1}^{N_{S_k}} \tilde{\omega}_k^l$.

IPS Step 4. Resampling of π_k^i

Draw N_p particles ξ_k^l independently from the empirical measure $\overline{\pi}_k^i = \sum_{l=1}^{N_{S_k}} \tilde{\omega}_k^l \, \delta_{\{\tilde{\xi}_k^l\}}$ each of which gets weight $\omega_k^l = \frac{1}{N_p}$.

After step 4, the new set of particles is $\{\xi_k^l, \omega_k^l\}_{l=1}^{N_p}$. If $k < m$ then repeat steps 1, 2, 3, 4 for $k := k+1$. Otherwise, stop with $\mathbb{P}(\tau^i < T) \approx \prod_{k=1}^{m} \overline{\gamma}_k^i$.

REMARK 10.1 Cérou et al. [13]–[14] have proven, under certain conditions, how to manage the simulations from D_{k-1}^i to D_k^i, such that the product of these fractions $\overline{\gamma}_k^i$ forms an unbiased estimate of the probability of x_t to hit the set D^i within the time period $(0,T)$, i.e.,

$$\mathbb{E}\left[\prod_{k=1}^{m} \overline{\gamma}_k^i\right] = \prod_{k=1}^{m} \gamma_k^i = \mathbb{P}(\tau^i < T),$$

and there also is some bound on the expected estimation error, i.e.,

$$\left(\mathbb{E}(\prod_{k=1}^{m} \overline{\gamma}_k^i - \prod_{k=1}^{m} \gamma_k^i)^q\right)^{\frac{1}{q}} \leq \frac{a_q b_q}{\sqrt{N_p}},$$

for some finite constants a_q and b_q, which depend on the simulated scenario and the sequence of intermediate levels adopted. ■

10.2.5 Modification of IPS Resampling Step 4

A well known problem with particle systems is the possibility of particle depletion or impoverishment. In order to reduce the sensitivity of the above algorithm on these points, we modify step 4 in two ways: (1) we reduce the chance of impoverishment by not throwing away any particle; and (2) we make copies of particles, but avoid that these copies take away too much weight from the original particles.

Modified IPS Step 4

Resample N_p particles from S_k^i as follows:

If $\frac{1}{2}N_p \leq N_{S_k} \leq N_p$, then copy the N_{S_k} particles, i.e., $\xi_k^l = \tilde{\xi}_k^l$ and set $\omega_k^l = \tilde{\omega}_k^l \, \frac{N_{S_k}}{\overline{\gamma}_k^i \, N_p}$ for $l = 1, \ldots, N_{S_k}$; the total weight of these particles is $\frac{N_{S_k}}{N_p}$. Subsequently, draw $N_p - N_{S_k}$ particles ξ_k^l independently from the empirical measure $\overline{\pi}_k^i = \sum_{l=1}^{N_{S_k}} \tilde{\omega}_k^l \, \delta_{\{\tilde{\xi}_k^l\}}$ and set $\omega_k^l = \frac{\sum_{l=1}^{N_{S_k}} \tilde{\omega}_k^l}{\overline{\gamma}_k^i \, N_p} = \frac{1}{N_p}$; the total weight of this is $1 - \frac{N_{S_k}}{N_p}$.

Else, i.e., if $N_{S_k} < \frac{1}{2}N_p$, then copy the N_{S_k} particles, i.e., $\xi_k^l = \tilde{\xi}_k^l$, and set $\omega_k^l = \frac{1}{2}\tilde{\omega}_k^l / \overline{\gamma}_k^i$ for $l = 1, \ldots, N_{S_k}$; the total weight of these particles is $\frac{1}{2}$. For the remaining weight of $\frac{1}{2}$, we independently draw $N_p - N_{S_k}$ particles ξ_k^l from the empirical measure $\overline{\pi}_k^i = \sum_{l=1}^{N_{S_k}} \tilde{\omega}_k^l \delta_{\{\tilde{\xi}_k^l\}}$ and set $\omega_k^l = \frac{\frac{1}{2}\sum_{l=1}^{N_{S_k}} \tilde{\omega}_k^l}{\overline{\gamma}_k^i (N_p - N_{S_k})} = \frac{\frac{1}{2}}{N_p - N_{S_k}}$.

10.3 Development of a Petri Net Model of Free Flight

In order to apply the IPS algorithm toward the assessment of collision risk of free flight, we need to develop a MC simulator of these operations such that the simulated trajectories constitute realizations of a hybrid state strong Markov process. Everdij and Blom [26]–[28] have developed a Stochastically and Dynamically Coloured Petri Net (SDCPN) formalism that ensures the specification of a free flight MC simulation model which is of the appropriate class. This section explains how the SDCPN formalism has been used to develop a MC simulation model of a particular free flight design.

The specific free flight design for which we wish to estimate the collision risk by sequential MC simulation is the Autonomous Mediterranean Free Flight (AMFF) operation [43]. AMFF has been developed to study the introduction of autonomous free flight operation in Mediterranean airspace. In parallel to the current study, the safety of the AMFF operation has been addressed in [53], following a fault tree analysis approach. These results show that application of AMFF seems feasible to accommodate low en-route traffic conditions over the Mediterranean. However, this study also concludes that the fault tree approach has limited analysis capabilities in showing whether AMFF can safely accommodate a higher traffic density. For this, a need was identified to use a more advanced safety risk assessment approach that considers complex situations involving dynamic interactions between multiple human and systems. The current study addresses this for AMFF under relatively high traffic densities.

For the development of a Petri net model of the AMFF operation, two key challenges have to be addressed: a syntactical challenge of developing a model that is consistent, complete, and unambiguous; and a semantics challenge of representing the AMFF operation sufficiently well. This section shows how the (S)DCPN formalism has been used to address the syntactical challenge. Addressing the semantics challenge falls outside the scope of this study.

10.3.1 Specification of Petri Net Model

In using the (S)DCPN formalism [26], [27], [28] in modelling more and more complex multi-agent hybrid systems, it was found that the compositional specification power of Petri nets reaches its limitations. More specifically, the following

problems were identified:

1. For the modelling of a complete Petri net for complex systems, a hierarchical approach is necessary in order to be able to separate local modelling issues from global or interaction modelling issues.

2. Often the addition of an interconnection between two low-level Petri nets leads to a duplication of transitions and arcs in the receiving Petri net.

3. The number of interconnections between the different low level Petri nets tends to grow quadratically with the size of the Petri net.

Everdij et al. [30] explained which Petri net model specification approaches from literature solve problem 1, and developed novel approaches to solve problems 2 and 3. Together, these approaches are integrated into a compositional specification approach for SDCPN, which is explained below.

In order to avoid problem 1, the compositional specification of an SDCPN for a complex process or operation starts with developing a Local Petri Net (LPN) for each agent that exists in the process or operation (e.g., air traffic controller, pilot, navigation and surveillance equipment). Essential is that these LPNs are allowed to be connected with other Petri net parts in such a way that the number of tokens residing in an LPN is not influenced by these interconnections. We use two types of interconnections between nodes and arcs in different LPNs:

- Enabling arc (or inhibitor arc) from one place in one LPN to one transition in another LPN. These types of arcs have been used widely in Petri net literature.

- Interaction Petri Net (IPN) from one (or more) transition(s) in one LPN to one (or more) transition(s) in another LPN.

In order to avoid problems 2 and 3, high level interconnection arcs have been introduced that allow, with well-defined meanings, arcs to initiate and/or to end on the edge of the box surrounding an LPN [30]. The meaning of these interconnections from or to an edge of a box allows several arcs or transitions to be represented by only one arc or transition.

10.3.2 High Level Interconnection Arcs

As an illustration of how high level interconnection arcs avoid duplication of arcs and transitions within an LPN and duplication of arcs between LPNs, we give three examples of these high level interconnection arcs. See [30] for a complete overview of these high level interconnection arcs.

In the first example, Figure 10.1, an enabling arc starts on the edge of an LPN box and ends on a transition in another LPN box, means that enabling arcs initiate from all places in the first LPN and end on duplications of this transition in the second LPN. The duplicated transitions should have the same guard or delay function and the same firing function and their input places should have the same colour type. This

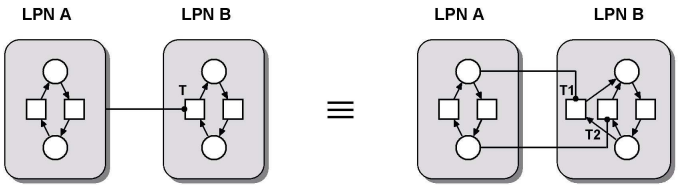

FIGURE 10.1: High level enabling arc starts at the edge of an LPN box.

high level interconnection arc is not defined for inhibitor or ordinary arcs instead of enabling arcs.

In the second example, Figure 10.2, an enabling arc ends on the edge of an LPN box.[1] This means that for each transition in the receiving LPN a copy of this enabling arc should be in place. Figure 10.2 shows an example of this high level interconnection arc. This type of high level arc can also be used with inhibitor arcs instead of enabling arcs. It cannot be used with ordinary arcs, due to the restriction that the number of tokens in an LPN should remain the same.

In the third example, Figure 10.3, an ordinary arc starts on the edge of an LPN box and ends on a transition inside the same box. This means that ordinary arcs start from all places in the LPN box to duplications of this transition. The duplicated transitions should have the same guard or delay function and the same firing function and their set of input places should have the same set of colour types. Figure 10.3 illustrates how this avoids both the duplication of transitions and arcs within an LPN, and the duplication of arcs between LPNs.

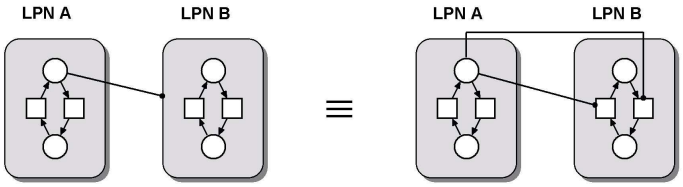

FIGURE 10.2: High level enabling arc ends at the edge of an LPN box.

10.3.3 Agents and LPNs to Represent AMFF Operations

In the Petri net modelling of AMFF operations for the purpose of an initial collision risk assessment, the following agents are taken into account:

[1]Figures 10.2, 10.3, 10.4, and 10.5 are from [8], with can kind permission of Springer Science and Business Media.

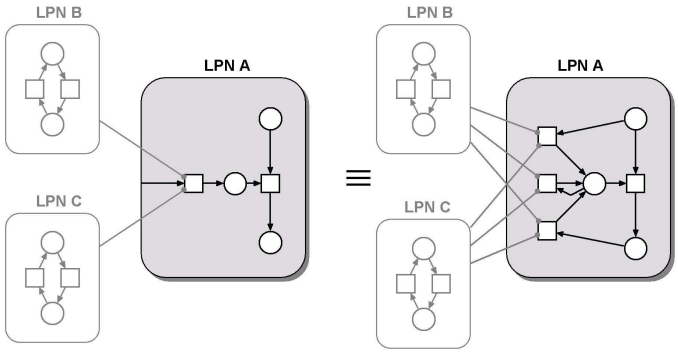

FIGURE 10.3: High level ordinary arc starts on the edge of an LPN box and ends on a transition inside the same LPN box.

- Aircraft
- Pilot-Flying (PF)
- Pilot-Not-Flying (PNF)
- Airborne Guidance, Navigation and Control (AGNC)
- Airborne Separation Assistance System (ASAS)
- Communication, Navigation and Surveillance (CNS)

It should be noticed that our initial model representing AMFF operations, does not yet incorporate other relevant agents such as Airborne Collision Avoidance System (ACAS), Airline Operations Centre (AOC), Air Traffic Control (ATC), or an environmental model. This should be taken into account when interpreting the simulation results obtained with this initial model.

Per agent, particular LPNs and IPNs have been developed and subsequently the interactions between these LPNs and IPNs have been specified. A listing of LPNs per agent reads as follows:

- Aircraft LPNs:
  - Type
  - Evolution mode
  - Systems mode
  - Emergency mode
- Pilot-Flying (PF) LPNs:
  - State situation awareness

  - Intent situation awareness
  - Goal memory
  - Current goal
  - Task performance
  - Cognitive mode
- Pilot-Not-Flying (PNF) LPNs:
  - Current goal
  - Task performance
- Airborne Separation Assistance System (ASAS) LPNs:
  - Processing
  - Alerting
  - Audio alerting
  - Surveillance
  - System mode
  - Priority switch mode
  - Anti-priority switch mode
  - Predictive alerting (of other aircraft)
- Airborne Guidance, Navigation and Control (AGNC) LPNs:
  - Indicators failure mode for PF
  - Engine failure mode for PF
  - Navigation failure indicator for PF
  - ASAS failure indicator for PF
  - Automatic Dependent Surveillance-Broadcast (ADS-B) receiver failure indicator for PF
  - ADS-B transmitter failure indicator for PF
  - Indicator failure mode for PNF
  - Guidance mode
  - Horizontal guidance configuration mode
  - Vertical guidance configuration mode
  - Flight Management System (FMS) flight plan
  - Airborne Global Positioning System (GPS) receiver
  - Airborne Inertial Reference System (IRS)
  - Altimeter

  - Horizontal position processing
  - Vertical position processing
  - ADS-B transmission
  - ADS-B receiver
- Communication, Navigation and Surveillance (CNS) LPNs:
  - Global GPS / satellites
  - Global ADS-B ether frequency
  - Secondary Surveillance Radar (SSR) mode-S frequency

The actual number of LPNs in the whole model then equals $38N + 3$, where N is the number of aircraft. In addition there are 35 IPNs per aircraft; hence the number of IPNs equals $35N$. Brownian motion enters in each of the aircraft evolution mode LNPs. In this initial model these Brownian motions are assumed to be independent.

10.3.4 Interconnected LPNs of ASAS

The approach taken in developing the AMFF concept of operation [43] is to avoid much information exchange between aircraft and to avoid dedicated decision-making by artificial intelligent machines. Although the conflict detection and resolution approach developed for AMFF has its roots in the modified potential field approach [37], it has some significant deviations from this. The main deviation is that conflict resolution in AMFF is intentionally designed not to take the potential field of all aircraft into account. The resulting AMFF design can be summarised as follows:

- All aircraft are supposed to be equipped with Automatic Dependent Surveillance-Broadcast (ADS-B), which is a system that periodically broadcasts own aircraft state information, and continuously receives the state information messages broadcasted by aircraft that fly within broadcasting range ($\sim$ 100 Nm).
- To comply with pilot preferences, conflict resolution algorithms are designed to solve multiple conflicts one by one rather than according to a full concurrent way, e.g., see [37].
- Conflict detection and resolution are state-based, i.e., intent information, such as information at which point surrounding aircraft will change course or height, is supposed to be unknown.
- The vertical separation minimum is 1000 ft and the horizontal separation minimum is 5 Nm. A conflict is detected if these separation minima will be violated within 6 minutes.
- The conflict resolution process consists of two phases. During the first phase, one of the aircraft crews should make a resolution maneuver. If this does not work, then during the second phase, both crews should make a resolution maneuver.

- Prior to the first phase, the crew is warned when an ASAS alert is expected to occur if no preventive action would be timely implemented; this prediction is done by a system referred to as P-ASAS (Predictive ASAS).
- Conflict co-ordination does not take place explicitly, i.e., there is no communication on when and how a resolution maneuver will be executed.
- All aircraft are supposed to use the same resolution algorithm, and all crew are assumed to use ASAS and to collaborate in line with the procedures.
- Two conflict resolution maneuver options are presented: one in vertical and one in horizontal direction. The pilot decides which option to execute.
- ASAS related information is presented to the crew through a Cockpit Display of Traffic Information (CDTI).

ASAS is modelled through the SDCPN depicted in Figure 10.4. The ADS-B information received from other aircraft is processed by the *LPN ASAS surveillance*. Together with the information about its own aircraft state information (from AGNC), the *LPN ASAS processing* uses this information to perform conflict detection and resolution functionalities. Subsequently, the *LPN ASAS alerting* and the *LPN P-ASAS processing* informs the PF and PNF through *ASAS audio alerting* about any aircraft that is in potential ASAS conflict with its own aircraft, and suggests resolution options including a prioritization. Three complementary LPNs represent non-nominal behaviour modes, each combination of which has a specific influence on the ASAS alerting LPN:

- *ASAS system mode* may be working, failed, or corrupted (failed or corrupted mode also influences the ASAS processing LPN).
- *ASAS priority switching mode*; under emergency, the PF switches this from "off" to "on."
- *ASAS anti-priority switch*; this is switched from "off" to "on" when own ADS-B is not working.

10.3.5 Interconnected LPNs of "Pilot Flying"

This subsection illustrates the specific Petri net model developed for the Pilot Flying. For the semantical basis behind this type of model, see [2], [5], [6], [15], [59]. A graphical representation of all LPNs the Pilot-Flying consists of, is given in Figure 10.5.

The Human-Machine-Interface where sound or visual clues might indicate that attention should be paid to a particular issue, is represented by a LPN that does not belong to the Pilot-Flying as agent and is therefore not depicted in the figure. Similarly, the arcs to or from any other agent are not shown in Figure 10.5. Because of the

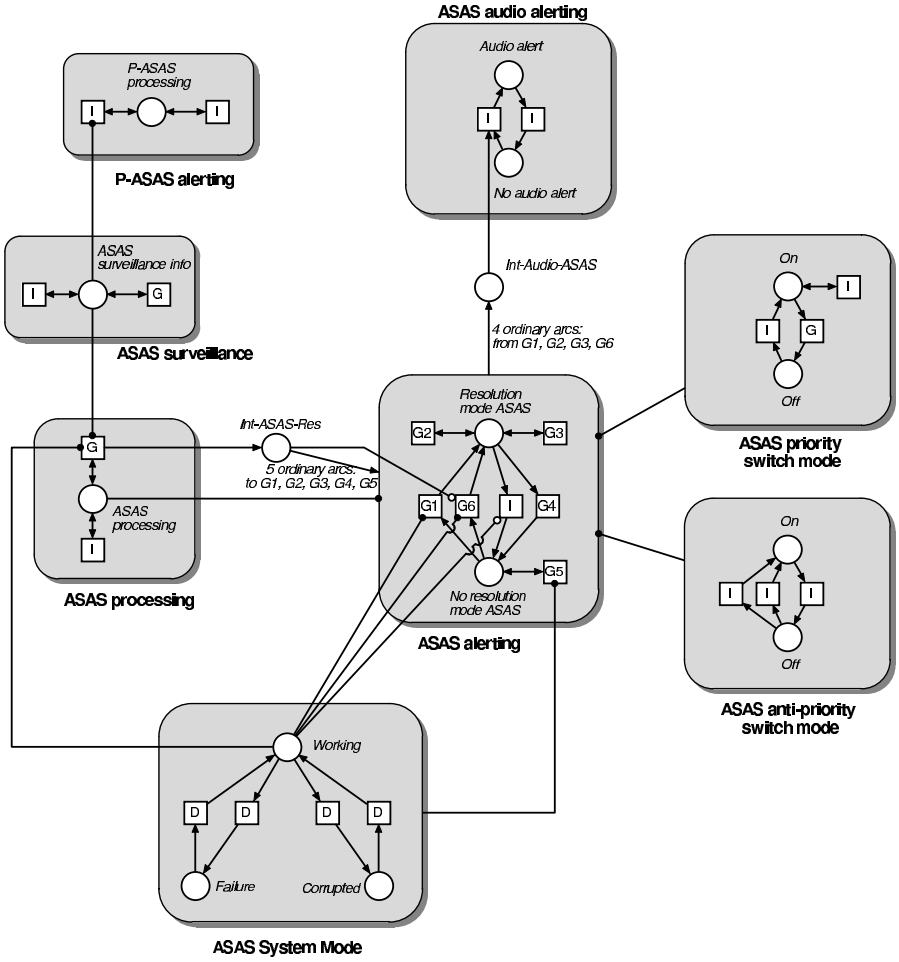

FIGURE 10.4: The agent ASAS in AMFF is modelled by eight LPNs, a number of ordinary and enabling arcs, and two IPNs (with one place each).

very nature of Petri nets, these arcs can easily be added during the follow-up specification cycle. To get an understanding of the different LPNs, a good starting point might be the LPN "Current Goal" (at the bottom of the figure) as it represents the objective the Pilot-Flying is currently working on. Examples of such goals are "Collision Avoidance," "Conflict Resolution," and "Horizontal Navigation." For each of these goals, the pilot executes a number of tasks in a prescribed or conditional order, represented in the LPN "Task Performance." Examples of such tasks are "Monitoring and Decision," "Execution," and "Execution Monitoring." If all relevant tasks for the current goal are considered executed, the pilot chooses another goal, thereby using his memory (where goals deserving attention might be stored, represented by

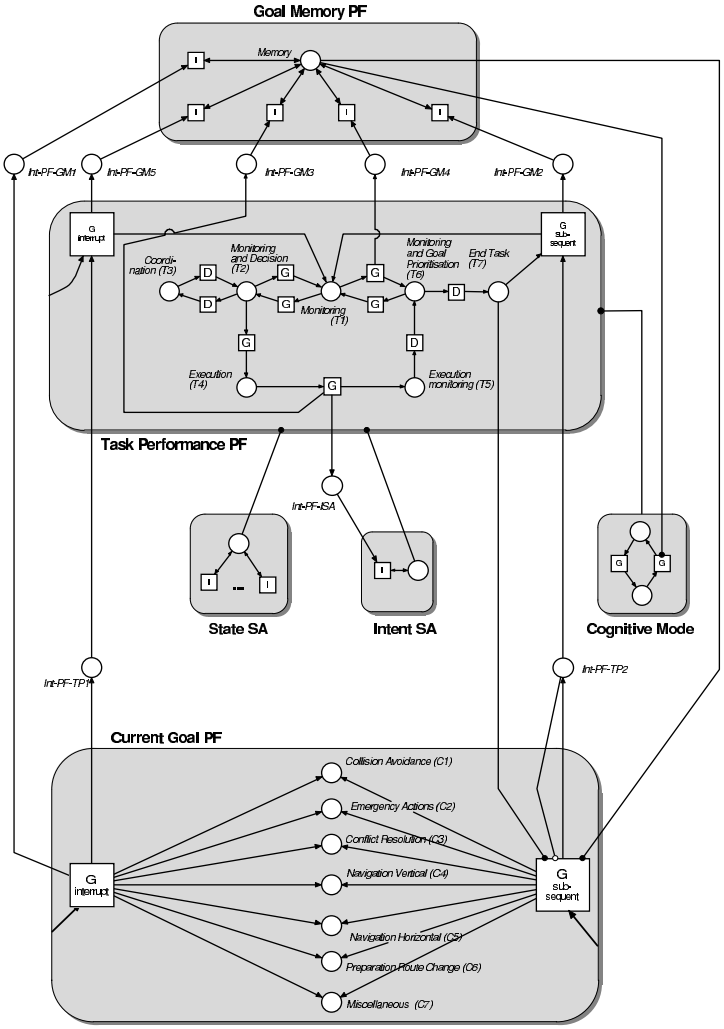

FIGURE 10.5: The agent Pilot-Flying in AMFF is modelled by six LPNs, and a number of ordinary and enabling arcs and some IPNs, consisting of one place and input and output arcs. The interconnections with other agents are not shown.

the LPN "Goal Memory") and the Human-Machine-Interface. His memory where goals deserving attention might be stored is represented as the LPN "Goal Memory" in Figure 10.5.

So, the LPNs "Current Goal," "Task Performance," and "Goal Memory" are important in the modelling of which task the Pilot-Flying is executing. The other three LPNs are important in the modelling on how the Pilot-Flying is executing the tasks. The LPN "State SA", where SA stands for Situation Awareness, represents the rel-

Table 10.1: Dimensional analysis of agent PF.

Pilot-Flying (PF) LPNs and IPNs	**Number of places**	**Maximum colour state space**
Pilot Flying (PF) LPNs:		
State Situation Awareness	1	$\mathbb{R}^3$
Intent Situation Awareness	1	$\mathbb{R}^5$
Goal memory	1	$\mathbb{R}$
Current goal	7	$\mathbb{R}$
Task performance	7	$\mathbb{R}^2$
Cognitive mode	2	$\mathbb{R}$
Pilot Flying (PF) internal IPNs:		
Int-PF-GM1	1	$\mathbb{R}$
Int-PF-GM2	1	$\mathbb{R}$
Int-PF-GM3	1	$\mathbb{R}$
Int-PF-GM4	1	$\mathbb{R}$
Int-PF-GM5	1	$\mathbb{R}$
Int-PF-TP1	1	$\mathbb{R}^2$
Int-PF-TP2	1	$\mathbb{R}$
Int-PF-ISA	1	$\mathbb{R}$
Pilot Flying (PF) external IPNs:		
Ext-PF-Audio-PF	5	
Ext-PF-PNF	1	$\mathbb{R}$
Ext-PF-PASAS	1	$\emptyset$
Ext-PF-SSA-1	1	$\emptyset$
Ext-PF-SSA-2	1	$\mathbb{R}$
Ext-PF-SSA-3	1	$\mathbb{R}$
Ext-PF-SSA-4	1	$\mathbb{R}$
Ext-PF-SSA-5	1	$\mathbb{R}$
PRODUCT	490	$\mathbb{R}^{28}$

evant perception of the pilot about the states of elements in his environment, e.g., whether he is aware of an engine failure. The LPN "Intent SA" represents the intent, e.g., whether he intends to leave the free flight airspace. The LPN "Cognitive mode" represents whether the pilot is in an opportunistic mode, leading to a high but error-prone throughput, or in a tactical mode, leading to a moderate throughput with a low error probability.

10.3.6 Model Verification, Parameterisation, and Validation

The compositionally specified SDCPN model enables a systematic implementation, verification and validation of the resulting Monte Carlo simulator. This is done

through the following systematic steps:

- Software code testing. This is done through conducting the following sequence of testing: random number generation, statistical distributions, common functions, each LPN implementation, each agent implementation, interactions between all agents, full MC simulation.
- Numerical approximation testing. This is needed to learn choosing an appropriate numerical integration step size and an appropriate number of particular MC simulations.
- Graphical user interface testing. This is to verify that the input and output of data works well.
- Parameterisation. This is done through a search for literature and statistical sources, and complemented by expert interviews. The fusion of these different pieces of information is accomplished following a Bayesian approach.
- Initial model validation through studying MC simulator behaviour and sensitivities to parameter changes under dedicated scenarios.
- Overall validation, which is directed to the evaluation of differences between model and reality and what effect these differences have at the assessed risk level.

The last validation step typically is done at a later stage in the risk assessment process, with the help of active participation of operational experts [25], [31].

10.3.7 Dimensions of MC Simulation Model

Now, we analyse the dimensions of the joint state space of the resulting MC simulation model. In Table 10.1 and Table 10.2, this is done for the agents PF and ASAS respectively, including all LPNs and all IPNs that end on one of these LPNs. The second column gives the number of places in the LPN or IPN. The third column gives the maximum state space of the colour used within an LPN or IPN. We also perform this analysis to the LPNs and IPNs of the other agents. The resulting number of product places and product state spaces is given in Table 10.3. This table brings into account that of each agent, except global CNS, there is one per aircraft.

The product places of the global CNS agent form the θ_t^0 state space M^0. The corresponding state space is empty, which means that there is no x_t^0. The product places of the other agents form the state space $\bigotimes_{i=1}^{N} M^i$ of the process components θ_t^i, $i = 1,\ldots,N$. Per aircraft, the number of product places is $|M^i| \approx 0.777 \times 10^{12}$. The colours attached to the places in the other agents form the process components x_t^i, $i = 1,\ldots,N$, each of which assumes values in $\mathbb{R}^{126+21N}$.

Each of the scenarios considered in the next subsection has eight aircraft, so $N = 8$. This means that the number of product places equals $\approx 16 \times (0.777 \times 10^{12})^8 \approx 2.13 \times 10^{96}$, and that the product of the colour state spaces equals $\mathbb{R}^{2352}$.

Table 10.2: Dimensional analysis of agent ASAS.

ASAS LPNs and IPNs	**Number of places**	**Maximum colour state space**
ASAS LPNs:		
Processing	1	$\mathbb{R}^{13+12N}$
Alerting	2	$\mathbb{R}^{7}$
Audio alerting	2	$\emptyset$
Surveillance	1	$\mathbb{R}^{11+9N}$
System mode	3	$\emptyset$
Priority switch mode	2	$\emptyset$
Anti-priority switch mode	2	$\emptyset$
Predictive alerting	1	$\mathbb{R}^{3}$
ASAS internal IPNs:		
Int-ASAS-Resolution	1	$\emptyset$
Int-ASAS-Audio	1	$\emptyset$
ASAS external IPNs:		
Ext-ASAS-PF	1	$\mathbb{R}^{3}$
Ext-PASAS-PNF	1	$\emptyset$
Ext-ASASProc-PNF	1	$\emptyset$
Ext-ASASurv-ADSB-Global	1	$\mathbb{R}$
Ext-ASASprio-PNF	1	$\emptyset$
PRODUCT	48	$\mathbb{R}^{38+21N}$

10.4 Simulated Scenarios and Collision Risk Estimates

The IPS algorithm of Section 10.2 is now applied to three hypothetical AMFF air traffic scenarios. The first scenario has eight aircraft that fly at the same flight level and their flight plans cause them to fly through the same point in airspace at the same moment in time. The second scenario has one aircraft flying through a virtual infinite airspace of randomly distributed aircraft, with a density 3 times as high as in a current high capacity en route area. The third scenario is the same as the second, except that the aircraft density is four times lower. Prior to describing these scenarios and simulation results, we explain the parameterisation of the IPS algorithm used.

10.4.1 Parameterisation of the IPS Simulations

The main safety critical parameter settings of the free flight enabling technical systems (GPS, ADS-B and ASAS) are given in Table 10.4; in the table, global ADS-B down refers to frequency congestion/overload of the data transfer technology used

Table 10.3: Dimensional analysis of complete SDCPN.

Agent	**Number of product places**	**Maximum colour product state space**
Aircraft	24^N	$\mathbb{R}^{13N}$
Pilot Flying (PF)	490^N	$\mathbb{R}^{28N}$
Pilot-not-Flying (PNF)	7^N	$\mathbb{R}^{3N}$
AGNC	$(15 \times 2^{16})^N$	$\mathbb{R}^{45N}$
ASAS	48^N	$\mathbb{R}^{37N+21N^2}$
Global CNS	16	$\emptyset$
PRODUCT	$\approx 16 \times (0.777 \times 10^{12})^N$	$\mathbb{R}^{126N+21N^2}$

for ADS-B. The IPS conflict levels k are defined by parameter values for lateral conflict distance d_k, conflict height h_k, and time to conflict Δ_k. These values have been determined through two steps. The first was to let an operational expert make a best guess of proper parameter values. Next, during initial simulations with the IPS some fine tuning of the number of levels and of parameter values per level has been done. The resulting values are given in Table 10.5.

10.4.2 Eight Aircraft on Collision Course

In this simulation eight aircraft start at the same flight level, some 135 Nm (250 km) out of each other, and fly in eight 45 degrees differing directions with a ground speed of 466 knots (= 240 m/s), all aiming to pass through the same point in airspace. By running ten times the IPS algorithm the collision risk is estimated ten times. The number of particles per IPS simulation run is 12,000. The total simulation time took about 20 hours on two machines, and the load of computer memory per machine was about 2.0 gigabyte. For the first eight IPS runs, the estimated fractions $\overline{\gamma}_k^i$ are given in Table 10.6 for each of the conflict levels, $k = 1, \ldots, 8$, and aircraft $i = 1$. It can be seen that the first and sixth IPS runs have zero particles that reach the last (8th level). Hence the first and sixth IPS runs yield $\overline{\gamma}_8^i = 0$. This is a clear example of particle depletion.

The IPS estimated mean probability for one aircraft to collide with any of the other seven aircraft equals 2.2×10^{-5}. The minimum and maximum values now are respectively a factor 250 lower and a factor 4 higher than the mean value. We also verified that this risk value was not sensitive at all to the failure rates of the ASAS related technical systems.

In [37] a similar eight aircraft encounter scenario has been simulated some hundred times, for varying initial aircraft positions, without noticing any collision event. At a collision probability value of 2.2×10^{-5}, the chance to count at least one collision would be less than 1%. As such the current results agree quite well with the

Table 10.4: Parameter values of free flight enabling technical systems.

Model Parameter	**Probability**
Global GPS down	1.0×10^{-5}
Global ADS-B down	1.0×10^{-6}
Aircraft ADS-B receiver down	5.0×10^{-5}
Aircraft ADS-B transmitter down	5.0×10^{-5}
Aircraft ASAS system mode corrupted	5.0×10^{-5}
Aircraft ASAS system mode failure	5.0×10^{-5}

Table 10.5: IPS conflict level parameter values.

k	1	2	3	4	5	6	7	8
d_k (Nm)	4.5	4.5	4.5	4.5	2.5	1.25	0.5	0.054
h_k (ft)	900	900	900	900	900	500	250	131
Δ_k (min)	8	2.5	1.5	0	0	0	0	0

Table 10.6: Fractions counted during eight IPS runs of scenario 1.

Level	1^{st} IPS	2^{nd} IPS	3^{rd} IPS	4^{th} IPS	5^{th} IPS	6^{th} IPS	7^{th} IPS	8^{th} IPS
1	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
2	0.528	0.529	0.539	0.533	0.537	0.538	0.536	0.539
3	0.426	0.429	0.424	0.431	0.421	0.428	0.426	0.418
4	0.033	0.036	0.035	0.037	0.039	0.031	0.044	0.039
5	0.175	0.180	0.183	0.181	0.142	0.157	0.181	0.147
6	0.267	0.158	0.177	0.144	0.255	0.138	0.295	0.146
7	0.150	0.268	0.281	0.427	0.645	0.208	0.253	0.295
8	0.000	0.009	0.233	0.043	0.455	0.000	0.006	0.815
Product of fractions	0.0	5.58×10^{-7}	1.67×10^{-5}	4.01×10^{-6}	9.33×10^{-5}	0.0	8.00×10^{-7}	4.48×10^{-5}

fact that in these earlier simulations for an eight aircraft scenario no collision has been observed. We also verified that the novel simulation results for an eight aircraft scenario agreed quite well with the expectation of the designers of the AMFF operational concept.

Table 10.7: Fractions counted during eight IPS runs of scenario 2.

Level	1st IPS	2nd IPS	3rd IPS	4th IPS	5th IPS	6th IPS	7th IPS	8th IPS
1	0.922	0.917	0.929	0.926	0.925	0.925	0.925	0.921
2	0.567	0.551	0.560	0.559	0.554	0.551	0.561	0.556
3	0.665	0.666	0.674	0.676	0.672	0.673	0.664	0.670
4	0.319	0.331	0.323	0.321	0.328	0.321	0.334	0.331
5	0.370	0.367	0.371	0.379	0.363	0.345	0.366	0.343
6	0.181	0.158	0.162	0.171	0.164	0.181	0.148	0.191
7	0.130	0.209	0.174	0.145	0.162	0.170	0.214	0.215
8	0.067	0.005	0.094	0.066	0.002	0.150	0.015	0.019
Product of fractions	6.42×10^{-5}	6.76×10^{-6}	1.11×10^{-4}	6.99×10^{-5}	2.57×10^{-6}	1.75×10^{-4}	1.99×10^{-5}	2.98×10^{-5}

10.4.3 Free Flight Through an Artificially Constructed Airspace

In this simulation the complete airspace is divided into packed containers. Within each container a fixed number of seven aircraft ($i = 2,\ldots,8$) fly at arbitrary position and in arbitrary direction at a ground speed of 466 Nm/hr. One additional aircraft ($i = 1$) aims to fly straight through a sequence of connected containers, at the same speed, and the aim is to estimate its probability of collision with any of the other aircraft per unit time of flying.

Per container, the aircraft within it behave the same. This means that we have to simulate each aircraft in one container only, as long as we apply the ASAS conflict prediction and resolution also to aircraft copies in the neighbouring containers. In principle this can mean that an aircraft experiences a conflict with its own copy in a neighbouring container. This also means that the size of a container should not go below a certain minimum size.

By changing container size we can vary traffic density. To choose the appropriate traffic density, our reference point is the highest number (17) of aircraft counted at 23rd July 1999 in an en-route area near Frankfurt of size 1 degree $\times$ 1 degree $\times$ FL290-FL420. This comes down to 0.0032 a/c per Nm^3. For our simulation we assume a 3 times higher traffic density, i.e., 0.01 a/c per Nm^3. This resulted in choosing containers having a length of 40 Nm, a width of 40 Nm, and a height of 3000 feet and with 8 aircraft flying in such a container.

By running the IPS algorithm ten times (+ one extra later on) over 25 minutes (5 minutes to allow convergence and 20 minutes to estimate collision probability) the collision probability per unit time of flying has been estimated. The number of particles per IPS simulation run is 10,000. The total simulation time took about 300 hours on two machines, and the load of computer memory per machine was about 2.0 gigabyte. For the first eight IPS runs, the estimated fractions $\overline{\gamma}_k^i$ are given in Table 10.7 for each of the conflict levels, $k = 1,\ldots,8$, for aircraft $i = 1$.

The estimated mean probability of collisions per 20 minutes of aircraft flight equals 5.22×10^{-5}, which is equal to a probability of collisions per aircraft flight hour of 1.6×10^{-4}, with minimum and maximum values respectively a factor four lower and higher. We also verified that this risk value was not sensitive at all to the failure rates of the ASAS related technical systems.

One should be aware that this value has been estimated for the simulation model of the intended AMFF operation. Hence the question is what this means for the intended AMFF operation itself? By definition a simulation model of AMFF differs from the intended AMFF operation. If it can be shown that the combined effect of these differences on the risk level is small, then the results obtained for the simulation model may be considered as a good representation of the accident risk of the intended operation. In order to assess the combined effect of these differences there is need to perform a bias and uncertainty assessment [25].

In order to better learn understanding of what causes the collision risk of the simulation model to be relatively high, we performed an extra IPS run, and memorised in static memory for each particle the ancestor history at each of the eight levels. This allowed us to trace back what happened for the particles that hit the last level set (i.e., collision). There appeared to be five different collision events. Evaluation of these five collision events showed that all five happened under nominal safety critical conditions. Four of the five collisions were due to a growing number of multiple conflicts that could not be solved in time under the operational concept adopted. The fifth collision was of another type: at quite a late moment finally a conflict between two aircraft was solved with a maneuver by one of the two aircraft. However because of this maneuver there was a sudden collision with a third nearby aircraft.

These detailed evaluations of the five collision events of the 11th IPS run also showed that a significant increase of collision risk is caused by the relatively small height (4000 ft) of a container. Because of this small height it happened that an aircraft in one container came in conflict with a copy of its own in a neighbouring container, and in such a situation there was an undesired limitation in conflict resolution options, and thus an undesired artificial increase in collision risk.

The results in this section seem to indicate that the key factor in the increased risk of collision for encounters with homogeneous traffic in the background — as opposed to the eight encountering aircraft only scenario — are the multiple conflicts. Under the far higher traffic densities than what the AMFF operational concept was designed for, it is not always possible to timely solve a sufficiently high fraction of those multiple conflicts. On the basis of this finding one would expect that the collision risk would decrease faster than linear with a decrease in traffic density. The validity of this expectation is verified by the next scenario.

10.4.4 Reduction of the Aircraft Density by a Factor Four

Now we enlarge the length and width of each container by a factor two. This means that the traffic density has gone down by a factor four. Hence the density is now $\frac{3}{4}$ of the density counted on 23rd July 1999 in the en-route area near Frankfurt. This still is a factor 2.5 higher than current average density above Europe. At the

Table 10.8: Fractions counted during four IPS runs of scenario 3.

Level	1st IPS	2nd IPS	3rd IPS	4th IPS
1	0.755	0.750	0.752	0.749
2	0.295	0.292	0.286	0.285
3	0.476	0.475	0.497	0.487
4	0.263	0.258	0.266	0.267
5	0.321	0.315	0.300	0.328
6	0.068	0.088	0.082	0.096
7	0.156	0.367	0.290	0.254
8	0.011	0.059	0.021	0.005
Product of fractions	1.07×10^{-6}	1.61×10^{-5}	4.31×10^{-6}	1.07×10^{-6}

same time simulated flying time has been increased to 60 minutes (with 10 minutes prior flying to guarantee convergence).

By running four times the IPS algorithm the collision risk is estimated four times. The number of particles per IPS simulation run is 10,000. The total simulation time took about 280 hours on two machines, and the load of computer memory per machine was about 2.0 gigabyte. For these IPS runs, the estimated fractions $\overline{\gamma}_k^i$ are given in Table 10.8 for each of the conflict levels, $k = 1,\ldots,8$, for aircraft $i = 1$. The estimated mean probability of collision per aircraft flight hour equals 5.64×10^{-6}, with minimum and maximum values respectively a factor five lower and higher. This is about a factor 30 lower than the previous scenario with a four times higher aircraft density. Thus, for the model there is a steep decrease of collision probability with decrease of traffic density, and this agrees well with the expectation at the end of the previous section.

10.4.5 Discussion of IPS Simulation Results

Because of the IPS simulation approach we were able to estimate collision risk for complex multiple aircraft scenarios. The large increase in handling complexity of multiple aircraft encounter situations is a major improvement over what was feasible before for two aircraft flying in a parallel route structure [4], [29]. Inherent to the IPS way of simulation, the dynamic memory of the computers used appeared to pose the main limitation on the full exploitation of the novel sequential MC simulation approach. This also prevented performing a bias and uncertainty assessment for the differences between the simulation model and the AMFF operation. As long as such a bias and uncertainty assessment has not been performed, any conclusion drawn from the simulation apply to the simulation model only, and need not apply to the intended AMFF operation.

The simulations performed for a model of AMFF allow free flight operational concept developers to learn characteristics of the simulation model. Because of the

sequential MC simulation based speed up, these simulations can show events that have not been observed before in MC simulations of an AMFF model. Under far higher traffic densities than what the AMFF operational concept has been designed for, the simulations of the model shows it is not always possible to timely solve multiple conflicts. As a result of this, at high traffic levels there is a significant chance that multiple conflicts are clogging together, and this eventually may cause a non-negligible chance of collision between aircraft in the simulation model. It has also been shown that by lowering traffic density, the chance of collision for the model rapidly goes down.

10.5 Concluding Remarks

We studied collision risk estimation of a free flight operation through a sequential Monte Carlo simulation. First a Monte Carlo simulation model of this free flight operational concept has been specified in a compositional way using the Stochastically and Dynamically Coloured Petri Net (SDCPN). Subsequently a novel sequential MC simulation method [13], [14] has been extended for application to collision risk estimation in air traffic, and has subsequently been applied to an SDCPN model of free flight.

The results obtained show that the novel simulation model specification and collision risk estimation method allow to speed up the Monte Carlo simulations for much more complex air traffic encounter situations than what was possible before, e.g., [4], [29]. Moreover, for the simulation model of the free flight operational concept considered, behaviour has been made visible that was expected by free flight concept designers, but could not be observed in straightforward Monte Carlo simulations of free flight concepts (e.g., Hoekstra [37]): the rare chance of clogging multiple conflicts at far higher traffic density levels than where the particular concept has been designed for. Hence, further attention has to be drawn toward the development and incorporation in the particular operational concept design of advanced methods in handling multiple conflicts. For example, Hoekstra [37] studied a conflict resolution approach that performs better than the one adopted in the AMFF concept. In addition, there are some complementary developments that aim to develop complex conflict resolution solvers with some guaranteed level of performance [20], [51] under nominal conditions, and ways to incorporate situation awareness views by human operators (pilots and/or controllers) in these combinatorial conflict resolution problems [21].

The initial collision risk estimation results obtained with our sequential MC simulation of free flight provides valuable feedback to the design team and allows them to learn from Monte Carlo simulation results they have never seen before. This allows them to significantly improve their understanding of when and why multiple conflicts are not solved in time anymore in the simulation model. Subsequently the

operational concept designers can use their better understanding for adapting the free flight design such that it can better bring into account future high traffic levels.

In its current form the sequential MC simulation approach works well, but at the same time poses very high requirements on the availability of dynamic computer memory and simulation time. The good message is that in literature on sequential MC simulation, e.g., see [18], [19], [23], [35], [48], [52], complementary directions have been developed which remain to be explored for application to free flight collision risk estimation. These potential improvements of sequential MC simulation will be studied in follow-up research.

Acknowledgement. The authors thank Mariken Everdij (NLR) for valuable discussions and a thorough review of a draft version of this chapter.

References

[1] J.W. Andrews, J.D. Welch, H. Erzberger. Safety analysis for advanced separation concepts. In *Proceedings of USA/Europe ATM R&D Seminar*, Baltimore, MD, 27–30 June 2005.

[2] H.A.P. Blom, J. Daams and H.B. Nijhuis. Human cognition modeling in Air Traffic Management safety assessment. In *Air Transportation Systems Engineering*, edited by G.L. Donohue and A.G. Zellweger, Vol. 193 in Progress in Astronautics and Aeronautics, Paul Zarchan, Editor-in-Chief, Chapter 29, pages 481–511, 2001.

[3] H.A.P. Blom, G.J. Bakker. Conflict probability and incrossing probability in air traffic management. In *Proc. IEEE Conf. on Decision and Control*, Las Vegas, December 2002.

[4] H.A.P. Blom, G.J. Bakker, M.H.C. Everdij, M.N.J. van der Park. Collision risk modeling of air traffic. In *Proceedings of European Control Conference*, Cambridge, UK, 2003.

[5] H.A.P. Blom, S.H. Stroeve, M.H.C. Everdij, M.N.J. Van der Park. Human cognition performance model to evaluate safe spacing in air traffic. *Human Factors and Aerospace Safety*, Vol. 2, pages 59–82, 2003.

[6] H.A.P. Blom, K.M. Corker, S.H. Stroeve. On the integration of human performance and collision risk simulation models of runway operation. In *Proceedings of the 6th USA/Europe Air Traffic Management R&D Seminar*, Baltimore, MD, 27–30 June 2005.

[7] H.A.P. Blom, G.J. Bakker, B. Klein Obbink, M.B. Klompstra. Free flight safety risk modelling and simulation. In *Proceedings of International Conference on Research in Air Transportation (ICRAT)*, Belgrade, 26–28 June 2006.

[8] H.A.P. Blom, J. Lygeros. *Stochastic Hybrid Systems: Theory and Safety Critical Applications*, LNCIS series, Springer, Berlin, July 2006.

[9] M.L. Bujorianu, J. Lygeros. Reachability questions in piecewise deterministic Markov processes. In *Proceedings of Hybrid Systems Computation and Control*, Eds. O. Mahler, A. Pnuelli, LNCIS number 2623, Springer, Berlin, pages 126–140, 2003.

[10] M.L. Bujorianu. Extended stochastic hybrid systems. In *Proceedings of Hybrid Systems Computation and Control*, Eds. O. Mahler, A. Pnuelli, LNCIS number 2993, Springer, Berlin, pages 234–249, 2004.

[11] M.L. Bujorianu, J. Lygeros. Towards a general theory of stochastic hybrid systems. In [8], pages 3–30, 2006.

[12] C.G. Cassandras, S. Lafortune. *Introduction to Discrete Event Systems*, Kluwer Academic Publishers, Boston, 1999.

[13] F. Cérou, P. Del Moral, F. Le Gland and P. Lezaud. Genetic genealogical models in rare event analysis, Publications du Laboratoire de Statistiques et Probabilites, Toulouse III, 2002.

[14] F. Cérou, P. Del Moral, F. Le Gland, P. Lezaud. Limit theorems for the multilevel splitting algorithms in the simulation of rare events. In *Proceedings of Winter Simulation Conference*, Orlando, FL, 2005.

[15] K. Corker. Cognitive Models & Control: Human & System Dynamics in Advanced Airspace Operations, Eds: N. Sarter and R. Amalberti, *Cognitive Engineering in the Aviation Domain*, Lawrence Earlbaum Associates, Hillsdale, New Jersey, 2000.

[16] R. David, H. Alla. Petri Nets for the modeling of dynamic systems - A survey, *Automatica*, Vol. 30, No. 2, pages 175–202, 1994.

[17] M.H.A. Davis. *Markov Models and Optimization*, Chapman & Hall, London, 1993.

[18] P. Del Moral. *Feynman-Kac Formulae. Genealogical and Interacting Particle Systems with Applications*, Springer Verlag, New York, 2004.

[19] P. Del Moral, P. Lezaud. Branching and interacting particle interpretations of rare event probabilities. In [8], pages 277–324, 2006.

[20] D.V. Dimarogonas, S.G. Loizou, K.J. Kyriapoulos. Multirobot navigation functions II: towards decentralization. In [8], pages 209–256, 2006.

[21] E. De Santis, M.D. Di Benedetto, S. Di Gennaro, A.D.'Innocenzo, G. Pola. Critical observability of a class of hybrid systems and application to air traffic management. In [8], pages 141–170, 2006.

[22] DNV. Safety assessment of P-RNAV route spacing and aircraft separation, Final report TRS 052/01, Eurocontrol, 2003.

[23] A. Doucet, N. de Freitas and N. Gordon. *Sequential Monte Carlo Methods in Practice*, Springer-Verlag, New York, NY, 2001.

[24] H. Erzberger. Transforming the NAS: The next generation air traffic control system. In *Proceedings of the 24th Int. Congress of the Aeronautical Sciences (ICAS)*, Yokohoma, Japan, 2004.

[25] M.H.C. Everdij, H.A.P. Blom. Bias and uncertainty in accident risk assessment, NLR report CR-2002-137, National Aerospace Laboratory NLR, 2002.

[26] M.H.C. Everdij, H.A.P. Blom. Petri nets and hybrid state Markov processes in a power-hierarchy of dependability models. In *Proceedings of IFAC Conference on Analysis and Design of Hybrid Systems*, Saint-Malo Brittany, France, pages 355–360, June 2003.

[27] M.H.C. Everdij, H.A.P. Blom. Piecewise deterministic Markov processes represented by Dynamically Coloured Petri Nets, *Stochastics*, Vol. 77, pages 1–29, 2005.

[28] M.H.C. Everdij, H.A.P. Blom. Hybrid Petri nets with diffusion that have into-mappings with generalised stochastic hybrid processes. In [8], pages 31–64, 2006.

[29] M.H.C. Everdij, H.A.P. Blom, G.J. (Bert) Bakker. Modeling lateral spacing and separation for airborne separation assurance using Petri nets. Forthcoming in: *Simulation, Transactions of the Society for Modeling and Simulation International*, Vol. 82, 2006.

[30] M.H.C. Everdij, M.B. Klompstra, H.A.P. Blom, B. Klein Obbink. Compositional specification of a multi-agent system by stochastically and Dynamically Coloured Petri Nets. In [8], pages 325–350, 2006.

[31] M.H.C. Everdij, H.A.P. Blom, S.H. Stroeve. Structured assessment of bias and uncertainty in Monte Carlo simulated accident risk. In *Proceedings of the 8th Int. Conf. on Probabilistic Safety Assessment and Management (PSAM8)*, New Orleans, LA, May 2006.

[32] FAA/Eurocontrol. A concept paper for separation safety modeling, Cooperative R&D Action Plan 3 report. Available at http://www.faa.gov/asd/ia-or/pdf/cpcomplete.pdf, 20 May 1998.

[33] FAA/Eurocontrol. Principles of Operations for the Use of ASAS, Cooperative R&D Action Plan 1 report, Version 7.1, 2001.

[34] FAA/Eurocontrol. Safety and ASAS applications, Co-operative R&D Action Plan 1 report, version 4.1, 2004.

[35] P. Glasserman. *Monte Carlo Methods in Financial Engineering*, Stochastic Modeling and Applied Probability, Vol. 53, Springer, New York, NY, 2003.

[36] P.J. Haas. *Stochastic Petri Nets, Modeling, Stability, Simulation*, Springer-Verlag, New York, 2002.

[37] J. Hoekstra. *Designing for Safety, the Free Flight Air Traffic Management concept*, PhD Thesis, Delft University of Technology, November 2001.

[38] D.A. Hsu. The evaluation of aircraft collision probabilities at intersecting air routes. *Journal of Navigation*, Vol. 34, pages 78–102, 1981.

[39] J. Hu, M. Prandini, S. Sastry. Probabilistic safety analysis in three dimensional aircraft flight. In *Proceedings of the 42nd IEEE Conference on Decision and Control (CDC)*, Maui, 2003.

[40] ICAO. Manual on airspace planning methodology for the determination of separation minima, ICAO Doc. 9689-AN/953, 1998.

[41] ICAO. Airborne separation assistance system (ASAS) circular, Draft, version 3, SCRSP, WGW/1 WP/5.0, International Civil Aviation Organization, May 2003.

[42] K. Jensen. *Coloured Petri Nets: Basic Concepts, Analysis Methods and Practical Use*, Vol. 1, Springer, London, UK, 1992.

[43] B. Klein Obbink. MFF airborne self separation assurance OSED, Report MFF R733D. Available at http://www.medff.it/public/index.asp, April 2005.

[44] J. Krozel. Free flight research issues and literature search. Under NASA contract NAS2-98005, 2000.

[45] J. Krystul, H.A.P. Blom. Monte Carlo simulation of rare events in hybrid systems, Hybridge Report D8.3. Available at http://www.nlr.nl/public/hosted-sites/hybridge, 2004.

[46] J. Krystul, H.A.P. Blom. Generalised stochastic hybrid processes as strong solutions of stochastic differential equations, Hybridge report D2.3. Available at http://www.nlr.nl/public/hostedsites/hybridge/, 2005.

[47] J. Krystul, H.A.P. Blom. Sequential Monte Carlo simulation of rare event probability in stochastic hybrid systems. In *Proceedings of the 16th IFAC World Congress*, Prague, Czech Republic, June 4-8, 2005.

[48] J. Krystul, H.A.P. Blom. Sequential Monte Carlo simulation for the estimation of small reachability probabilities for stochastic hybrid systems. In *Proceedings of IEEE-EURASIP Int. Symposium on Control, Communications and Signal Processing*, Marrakech, Morocco, March 13-15, 2006.

[49] A.B. Kurzhanski, P. Varaiya. On reachability under uncertainty. *SIAM Journal on Control and Optimization*, Vol. 41, pages 181–216, 2002.

[50] P.E. Labeau, C. Smidts and S. Swaminathan. Dynamic reliability: towards an integrated platform for probabilistic risk assessment. *Reliability Engineering and System Safety*, Vol. 68, pages 219–254, 2000.

[51] A. Lecchini, W. Glover, J. Lygeros, J. Maciejowski. Monte Carlo optimisation for conflcit resolution in air traffic control. In [8], pages 257–276, 2006.

[52] F. Le Gland, N. Oudjane. A sequential particle algorithm that keeps the particle system alive. In [8], pages 351–400, 2006.

[53] MFF. MFF Final safety case, Report MFF D734, ed. 1.0. Available at http://www.medff.it/public/index.asp, November 2005.

[54] NASA. Concept definition for distributed air-/ground traffic management (DAG-TM), Version 1.0, Advanced Air Transportation Technologies project, Aviation System Capacity Program, National Aeronautics and Space Administration, NASA, 1999.

[55] G. Pola, M.L. Bujorianu, J. Lygeros, M.D. Di Benedetto. Stochastic hybrid models: an overview with applications to air traffic management. In *Proceedings of IFAC Conf. Analysis and Design of Hybrid Systems (ADHS)*, Saint-Malo, Brittany, France, 2003.

[56] M. Prandini, J. Hu. A stochastic approxmation method for reachability computations. In [8], pages 107–139, 2006.

[57] RTCA. Free Flight Implementation, Task Force 3 Final Technical Report, Washington DC, 1995.

[58] A.P. Shah, A.R. Pritchett, K.M. Feigh, S.A. Kalarev, A. Jadhav, K.M. Corker, D.M. Holl, R.C. Bea. Analyzing air traffic management systems using agent-based modeling and simulation. In *Proceedings of the 6th USA/Europe Seminar on Air Traffic Management Research and Development*, Baltimore, MD, 27-30 June 2005.

[59] S.H. Stroeve, H.A.P. Blom and M.N. van der Park. Multi-agent situation awareness error evolution in accident risk modeling. In *Proceedings of the 5th USA/Europe Seminar on Air Traffic Management Research and Development*, Budapest, Hungary, 23-27 June 2003.