Stochastic validation of ATM procedures by abstraction algorithms

Maria D. Di Benedetto, G. Di Matteo and A. D'Innocenzo
Department of Electrical and Information Engineering, Center of Excellence DEWS
University of L'Aquila, Italy
e-mail: mariadomenica.dibenedetto@univaq.it, giuliadimatteo@hotmail.it,
alessandro.dinnocenzo@ing.univaq.it

***Abstract*—In this paper we propose a methodology for formal reasoning based on stochastic hybrid systems theory and abstraction algorithms for stochastic dynamical systems, which provides a powerful framework to analyze stochastic models of ATM procedures. We propose the use of automatic tools for verifying probabilistic properties of ATM scenarios. In particular, we propose to use PCTL logic to define probabilistic properties of interest. We address a simple single-agent procedure of the A^3(Autonomous Aircraft Advanced) *ConOps* (Concept of Operations), describe a dynamical model for the aircraft deterministic dynamics and for the wind stochastic dynamics, and used MATLAB and PRISM tools in order to perform stochastic analysis of properties of interest of the addressed scenario.**

***Index Terms*—Air traffic management, Stochastic hybrid systems, Abstraction algorithms, Probabilistic model checking.**

I. Introduction

The introduction of new Air Traffic Management (ATM) procedures is a necessary condition for achieving safety and efficiency objectives requested by the increasing air traffic. Modeling, simulation and formal analysis and validation of new ATM procedures is an important and necessary step for the development of ATM systems. In the context of the iFly project, our research focuses on development of novel concepts and technologies for addressing the issues discussed above, in order to provide automatic tools for the ATM systems under development and standardization.

In the past, the Air Traffic Controller (ATC) and the pilots had access to different data, and the responsibility of changes in procedures and operations were totally delegated to the ATC. The introduction of the next generation of ATM systems forecasts the use of ground and on-board integrated surveillance systems, which guarantee a cooperation between the ATC and the pilots. Moreover, new technologies provide broadcast communication of an aircraft position in the airspace, thus enabling the possibility of decentralization of decision making from the ATC to the pilot. These are the enabling technologies for development of a plethor of applications, such as the Airborne Separation Assistance System (ASAS) [1], which aims to improve efficiency of air traffic management procedures by a decentralization of responsibility among the ATC and the pilots.

The more advanced ASAS application is the Airborne Self Separation (ASEP) [2], which aims to a total shift of responsibility to the pilots flying in a specified airspace. Within this airspace, the pilots are responsible of maintaining safety separation with the other aircraft using the on board surveillance system. The operative concept ConOps [3] consists of two planning phases and one validation phase. The first planning phase derives from the Autonomous Aircraft Advanced (A^3) concept [3], which contemplates a network of aircraft, each responsible of Airborne Self Separation with no ATC ground support. The second phase contemplates analysis and validation of results of the first phase, in order to improve the A^3 concept, including the ATC support when necessary. These new concepts are a potential solution to the increasing air traffic density expected in the future years, and forecast an increase of safe air traffic from three to six times the current air traffic. The main problem is providing guarantee that the new air traffic procedures are sufficiently safe.

In this paper we propose to apply a methodology for formal reasoning based on stochastic hybrid systems theory, that provides a powerful framework to analyze multi-agents stochastic models of ATM procedures. We propose the use of automatic tools for verifying probabilistic properties of ATM scenarios. In particular, we propose to use PCTL logic to define probabilistic properties of interest (we refer to [4] and references therein for a survey on PCTL). Recently, formal verification of stochastic models has been transformed from an academically attractive discipline to a research effort prone to yield industrially relevant applications, and tools for probabilistic model checking have been developed: we propose the use of PRISM [5], [6], [7] for automatic verification of PCTL properties on ATM procedures.

However, the dynamical analysis of high-dimensional, stochastic models poses a number of challenges. When direct analysis of the model under study is impaired by its sheer complexity, automatic verification and algorithmic control design procedures are essential. An approach that is successfully used to cope with the issue of computational complexity and scalability is that of *abstraction*: a system with smaller state space is sought, which is *equivalent* to the original system. System equivalence implies that some properties of the original (complex, possibly infinite dimensional) system are preserved by the (simple, possibly finite dimensional) abstraction. For this reason, the property of interest can be efficiently checked on the abstraction, in finite time and/or with a lower computational complexity. Figure 1 illustrates the main phases of our verification algorithm we propose.

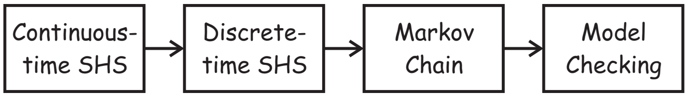


Fig. 1: Verification algorithm flow.

In the first block, a detailed continuous-time stochastic model of the ATM procedure (e.g. a stochastic model of the aircraft dynamics) is defined, using the mathematical framework of continuous time Stochastic Hybrid Systems (ct-SHS). This model can be discretized with respect to the time variable, thus obtaining a discrete time SHS (dt-SHS). We refer to [8] and references therein for the formal definition of discrete and continuous time SHSs. In the third block, a Markov Chain abstraction of the model is obtained using the abstraction procedure proposed in [9]. This abstraction procedure is essentially a partition of the state space, which depends on a tunable parameter δ (the width of the partition grid). The reason for using this abstraction is that it provides an approximation of the original system, and it can be used to perform automatic model checking using the tool PRISM. The results of the model checking verification directly apply to the original system, modulo an approximation error ϵ. This approximation error ϵ can be chosen a-priori, by modifying the parameter $\delta(\epsilon)$ of the abstraction procedure from dt-SHS to Markov Chain.

The paper is organized as follows. In Section II we describe the ATM scenario we considered to apply our methodology, i.e. the Hole in the clouds scenario illustrated in [3]. In Section III we describe a dynamical model for the aircraft deterministic dynamics, and for the wind stochastic dynamics. In Section IV we present our simulation results.

II. Scenario

We illustrate a simple procedure of the A^3(Autonomous Aircraft Advanced) *ConOps* (Concept of Operations), i.e. the Hole in the clouds scenario illustrated in [3].

In A^3 ConOps the concept of airspace has been redefined, introducing the concept of Performance Based Airspace (PBA). A^3 airspace is divided into 3 categories, as illustrated in Figure 2: the *Managed Airspace (MA)* is a high-density area; the *Unmanaged Airspace (UA)* is an area where ATC services are not accessible; the *Performance Based Airspace (PBA)* is an airspace whose boundaries are defined in time and space through MA and UA dynamic assignment.

In PBA autonomous aircraft are responsible for separation, according to the AFR (Autonomous Flight Rules). Operations are usually conducted under AFR or IFR (Instrument Flight Rules), while operations under VFR (Visual Flight Rules) are only admitted at specific altitudes. In PBA airspace aircraft have to guarantee self-separation and safe manoeuvres. Any conflict has to be avoided using appropriate manoeuvres: the final objective is safe cruise avoiding any conflict, e.g. *Protected Airspace Zones* (PAZ), *Restricted airspace areas* (RAA), or *Weather hazards areas* (WHA).

We consider an A^3 flight, defined as the flight between a departing Terminal Control Area (TMA) exit point, and an arriving TMA entry point, constrained by a Controlled Time of Arrival (CTA) at the arriving TMA entry point, as illustrated in Figure 4.

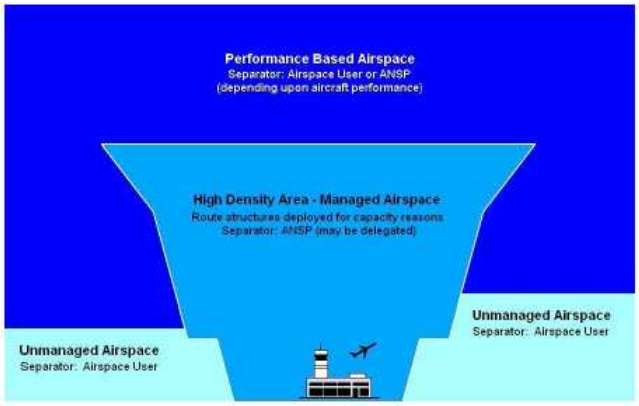

Performance Based Airspace
Separator: Airspace User or ANSP
(depending upon aircraft performance)
High Density Area - Managed Airspace
Route structures deployed for capacity reasons
Separator: ANSP (may be delegated)
Unmanaged Airspace
Separator: Airspace User
Unmanaged Airspace
Separator: Airspace User

Fig. 2: Airspace classification, from [3].

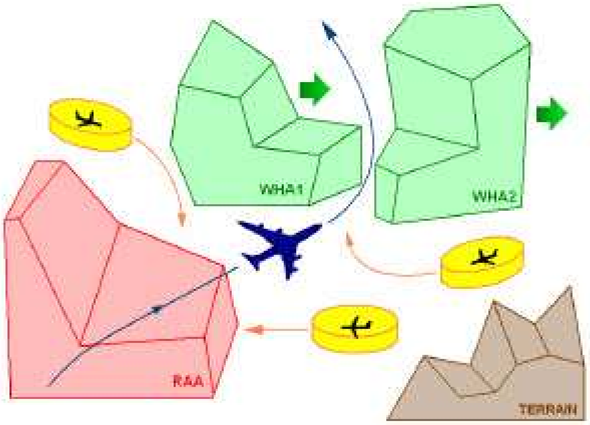


Fig. 3: Conflict environment in PBA, from [3].

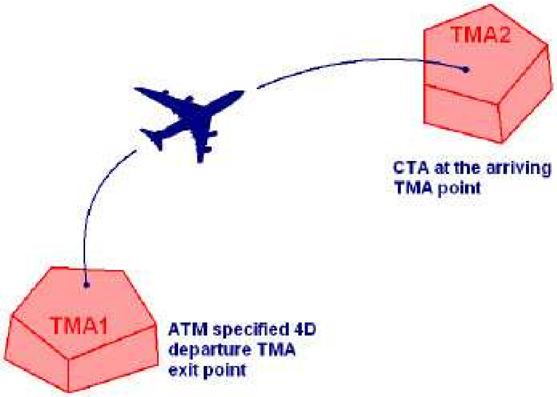

TMA2
CTA at the arriving TMA point
TMA1
ATM specified 4D departure TMA exit point

Fig. 4: A^3 flight, from [3].

During the flight, the aircraft follows its Business Trajectory (RBT) and maintains separation from other aircraft and conflicts, respecting constraints of imposed by Traffic

Flow Management. Given a Business Trajectory, we apply our methodologies to verify position and time of arrival to the arriving TMA without entering conflict areas, by taking into account stochastic wind disturbance on the aircraft dynamics. Figure 5 illustrates a scenario, where WHA conflicts are present.

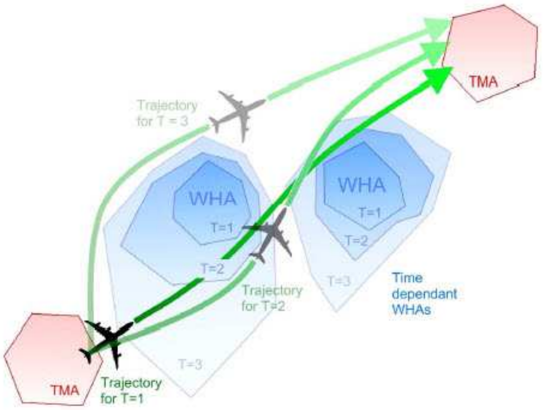


Fig. 5: WHA conflicts A^3 flight, from [3].

III. Aircraft dynamical model

We use a Point Mass Model (PMM) for modeling aircraft dynamics. We denote by X, Y horizontal position, by h altitude, by V true airspeed, by γ flight trajectory angle, and by ψ heading angle. Wind is considered as a disturbance on the aircraft dynamics, and is modeled by its speed $W = (w_1, w_2, w_3) \in \mathbb{R}^3$.

We use the following dynamical model from [10]:

$$\begin{aligned}
\dot{X} &= V cos(\psi) cos(\gamma) + w_1 \\
\dot{Y} &= V sin(\psi) cos(\gamma) + w_2 \\
\dot{h} &= V sin(\gamma) + w_3 \\
\dot{V} &= \tfrac{1}{m}[(T cos(\alpha) - D) - mg sin(\gamma)] \\
\dot{\psi} &= \tfrac{1}{mV}(L sin(\phi) + T sin(\alpha) sin(\phi)) \\
\dot{\gamma} &= \tfrac{1}{mV}[(L + T sin(\alpha)) sin(\phi) - mg cos(\gamma)]
\end{aligned} \tag{1}$$

where T denotes engine thrust, α attack angle, ϕ yaw angle, m aircraft mass and g gravity acceleration. L and D respectively denote lift and drag forces, which are functions of the state and the attack angle. Typically:

$$\begin{aligned}
L &= \tfrac{C_L S_\rho}{2}(1 + c\alpha)V^2, \\
D &= \tfrac{C_D S_\rho}{2}(1 + b_1\alpha + b_2\alpha^2)V^2,
\end{aligned}$$

where S denotes wing surface, ρ air density, and C_D, C_L, c, b_1, b_2 lift and drag aerodynamic coefficients that depend on the flight phase.

Figure 6 illustrates how forces act on the aircraft in the model described above.

From 1 we derive a 6-dimensional model of the aircraft $x = (x_1, x_2, x_3, x_4, x_5, x_6)^T \in \mathbb{R}^6$, with 3 constant inputs $u = (u_1, u_2, u_3)^T \in \mathbb{R}^3$ and 3 disturbance components $w = (w_1, w_2, w_3)^T \in \mathbb{R}^3$. By defining $x_1 = X$, $x_2 = Y$, $x_3 = h$, $x_4 = V$, $x_5 = \psi$, $x_6 = m$, $u_1 = T$, $u_2 = \phi$, $u_3 = \gamma$, and considering the consumption coefficient η, we obtain the following dynamics:

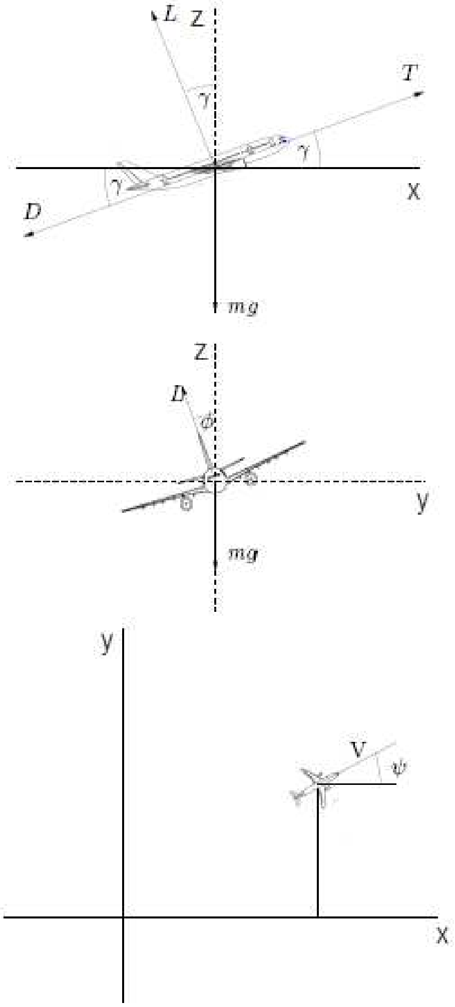


Fig. 6: Forces acting on the aircraft.

$$\dot{x} = \begin{bmatrix} x_4 cos(x_5) cos(u_3) + w_1 \\ x_4 sin(x_5) cos(u_3) + w_2 \\ x_4 sin(u_3) + w_3 \\ -\frac{C_D S_\rho}{2}\frac{x_4^2}{x_6} - g sin(u_3) + \frac{1}{x_6}u_1 \\ \frac{C_L S_\rho}{2}\frac{x_4}{x_6} sin(u_2) \\ -\eta u_1 \end{bmatrix} \tag{2}$$

State and input are subject to the following constraints: $x_3 > 0$, $x_4 \in [V_{min}, V_{max}]$, $x_6 \in [m_{min}, m_{max}]$, $u_1 \in [T_{min}, T_{max}]$, $u_2 \in [\phi_{min}, \phi_{max}]$, $u_3 \in [\gamma_{min}, \gamma_{max}]$. Values for state and input constraints and for parameters C_D, S and ρ can be found from the database BADA (Base of Aircraft DAta) [11].

Wind is modeled by a nominal component and a stochastic component $w = w_n + w_s$. The stochastic component is modeled by Gaussian random variables, i.e. by a random field $w_s : \mathbb{R} \times \mathbb{R}^3 \to \mathbb{R}^3$, where $w_s(t, P)$ represents wind in point $P \in \mathbb{R}^3$ at time $t \in \mathbb{R}$. We assume that $w_s(\cdot, \cdot)$ satisfies the following properties:

1) $w_s(t, P)$ is a Gaussian random variable with mean $\mu(t, P)$ and covariance matrix $\Sigma(t, P)$.
2) The random field is isotropic in x e y, i.e. the correlation structure does not change for rotations in the horizontal plane.
3) $w_{s1}(t, P), w_{s2}(t, P), w_{s3}(t, P)$ are independent $\forall t \in \mathbb{R}$, $\forall P \in \mathbb{R}^3$.

In the addressed scenario, we assume that altitude h, true airspeed V and heading angle ψ are constant, and that γ is equal to zero. Under these assumptions, equations (2) assume the following form:

$$\dot{x} = \begin{bmatrix} Vcos(\psi) + w_n cos(\beta) + w_{s1} \\ Vsin(\psi) + w_n sin(\beta) + w_{s2} \end{bmatrix} \tag{3}$$

where $x = (x_1, x_2)^T \in \mathbb{R}^2$ model the aircraft position in the plane $x_1 = X, x_2 = Y$, $(w_n cos(\beta), w_n sin(\beta))^T \in \mathbb{R}^2$ is the deterministic component of the wind where w_n e β are respectively wind velocity and direction, and $w_s = (w_{s1}, w_{s2})^T \in \mathbb{R}^2$ is the stochastic component of the wind. Using these dynamics, it is possible to derive a continuous time SHS describing the dynamics of the aircraft. Starting from this model and choosing a sampling time Δ, we derive a discrete time SHS by applying the classical Eulero-Maruyama discretization with constant step Δ.

From equations in (3) we obtain the deterministic component f that characterizes the dynamics of the aircraft for the dt-SHS model:

$$f = \begin{pmatrix} Vcos(\psi) + w_n cos(\beta) \\ Vsin(\psi) + w_n sin(\beta) \end{pmatrix}$$

We assume the aircraft is flying at cruise speed and at flight level 350. The table 7 (obtained from [11]) reports aircraft data in cruise, climb and descent phase.

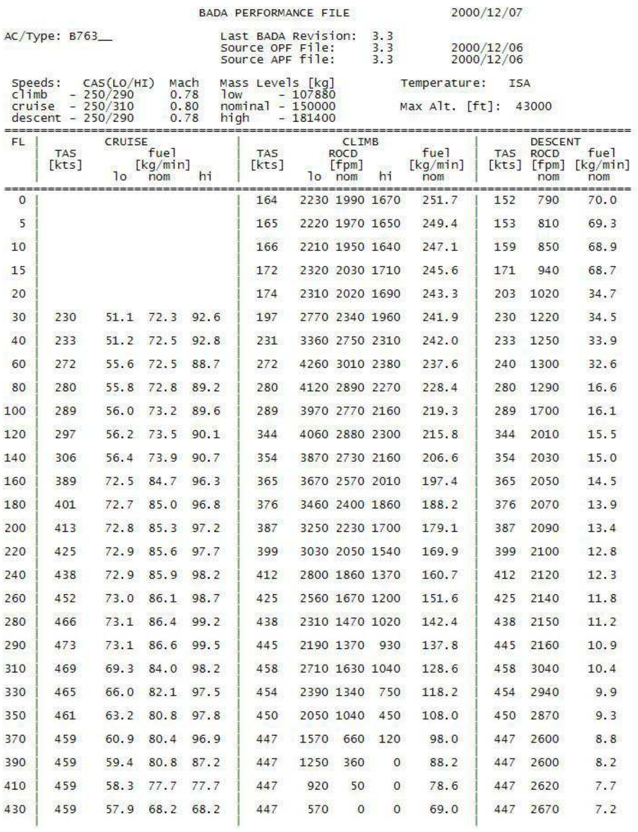
At flight level 350, in cruise phase, the corresponding true airspeed (TAS) is 461 kts (853.772 km/h). We assume that $\psi = \pi/4$, and that the wind has a constant deterministic component with speed 50 km/h and direction from North to East.

We resume the parameters used in our simulations: $V = 853,772 km/h = 0.2372 km/s$, $\psi = \pi/4$, $w_n = 50 km/h = 0.0139 km/s$, $\beta = 3\pi/8$.

IV. Simulation results

Given the aircraft and wind dynamics introduced in Section III, we consider the scenario illustrated in Figure 5 of Section II, and use our methodology for performing stochastic analysis of the original dtSHS through the Markov Chain abstraction.

We consider a simplified Performance Based Airspace (PBA) as a rectangular airspace defined by $a = 0, b = 10\ km., c = 0, d = 10\ km.$, as illustrated in Figure 8.



```
                         BADA PERFORMANCE FILE               2000/12/07

AC/Type: B763__                  Last BADA Revision:  3.3
                                 Source OPF File:     3.3        2000/12/06
                                 Source APF file:     3.3        2000/12/06

 Speeds:   CAS(LO/HI)  Mach   Mass Levels [kg]          Temperature:   ISA
 climb   - 250/290     0.78   low      - 107880
 cruise  - 250/310     0.80   nominal  - 150000         Max Alt. [ft]:  43000
 descent - 250/290     0.78   high     - 181400
```

FL	CRUISE				CLIMB					DESCENT		
	TAS [kts]	fuel [kg/min]			TAS [kts]	ROCD [fpm]			fuel [kg/min]	TAS [kts]	ROCD [fpm]	fuel [kg/min]
		lo	nom	hi		lo	nom	hi	nom		nom	nom
0					164	2230	1990	1670	251.7	152	790	70.0
5					165	2220	1970	1650	249.4	153	810	69.3
10					166	2210	1950	1640	247.1	159	850	68.9
15					172	2320	2030	1710	245.6	171	940	68.7
20					174	2310	2020	1690	243.3	203	1020	34.7
30	230	51.1	72.3	92.6	197	2770	2340	1960	241.9	230	1220	34.5
40	233	51.2	72.5	92.8	231	3360	2750	2310	242.0	233	1250	33.9
60	272	55.6	72.5	88.7	272	4260	3010	2380	237.6	240	1300	32.6
80	280	55.8	72.8	89.2	280	4120	2890	2270	228.4	280	1290	16.6
100	289	56.0	73.2	89.6	289	3970	2770	2160	219.3	289	1700	16.1
120	297	56.2	73.5	90.1	344	4060	2880	2300	215.8	344	2010	15.5
140	306	56.4	73.9	90.7	354	3870	2730	2160	206.6	354	2030	15.0
160	389	72.5	84.7	96.3	365	3670	2570	2010	197.4	365	2050	14.5
180	401	72.7	85.0	96.8	376	3460	2400	1860	188.2	376	2070	13.9
200	413	72.8	85.3	97.2	387	3250	2230	1700	179.1	387	2090	13.4
220	425	72.9	85.6	97.7	399	3030	2050	1540	169.9	399	2100	12.8
240	438	72.9	85.9	98.2	412	2800	1860	1370	160.7	412	2120	12.3
260	452	73.0	86.1	98.7	425	2560	1670	1200	151.6	425	2140	11.8
280	466	73.1	86.4	99.2	438	2310	1470	1020	142.4	438	2150	11.2
290	473	73.1	86.6	99.5	445	2190	1370	930	137.8	445	2160	10.9
310	469	69.3	84.0	98.2	458	2710	1630	1040	128.6	458	3040	10.4
330	465	66.0	82.1	97.5	454	2390	1340	750	118.2	454	2940	9.9
350	461	63.2	80.8	97.8	450	2050	1040	450	108.0	450	2870	9.3
370	459	60.9	80.4	96.9	447	1570	660	120	98.0	447	2600	8.8
390	459	59.4	80.8	87.2	447	1250	360	0	88.2	447	2600	8.2
410	459	58.3	77.7	77.7	447	920	50	0	78.6	447	2620	7.7
430	459	57.9	68.2	68.2	447	570	0	0	69.0	447	2670	7.2

Fig. 7: Data obtained from BADA [11].

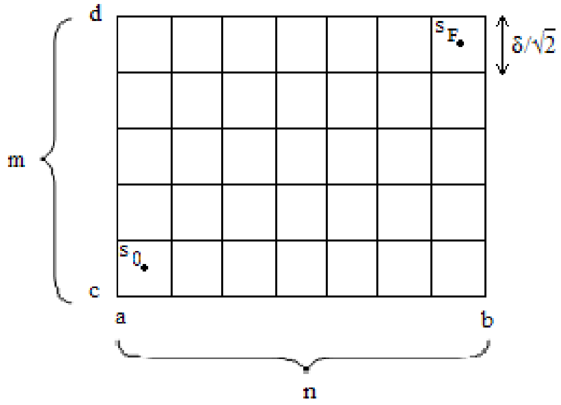


Fig. 8: Simplified and partitioned Performance Based Airspace (PBA).

The continuous time stochastic dynamics of the aircraft are discretized with sampling time $\Delta = 1s$ using the *Eulero-Maruyama* discretization, and the continuous state space of the aircraft dynamics have been restricted to the simplified PBA and partitioned using a grid of width $\delta/\sqrt{2}$, as illustrated in Figure 8. The parameter δ is the diameter of each partition cell. The number of cells is $n \cdot m$, and depends on a, b, c, d, δ as follows:

$$m = \frac{(d - c)}{(\delta/\sqrt{2})}, n = \frac{(a - b)}{(\delta/\sqrt{2})}.$$

We have chosen $\delta/\sqrt{2} = 0.2 km.$, which generates a 50×50 grid. Using the space and time discretizations defined above, we construct using the abstraction method defined in [9] a

Markov Chain abstraction of the original dtSHS. This Markov Chain is defined by a 2501×2501 stochastic matrix Π and an initial probability distribution Π_0, which in our case is given by the 2501×1 vector $\Pi(0) = [1\ 0\ 0 \ldots 0]^T$ (i.e. the initial position of the aircraft is the departure TMA with probability 1. The 2501^{st} state of the Markov Chain is an absorbing *sink state*, that models the state space region $\mathbb{R}^2 \setminus [a,b] \times [c,d]$. It is reasonable to model this whole region as a single unsafe state, since it models that the aircraft exits the PBA without reaching the arrival TMA. Moreover, the probability of entering this state is usually $\cong 0$.

A. Probability distribution evolution

Given Π, Π_0, we can compute the stochastic evolution at step $t \in \mathbb{N}$ of the 2501×1 probability vector $\Pi_t = \Pi^t \Pi_0$. As illustrated in Figure 10 we performed computation of Π_t at time steps $t = 1, 2, \cdots, T$, using MATLAB for constructing Π, Π_0 and plotting Π_t. The x-y plane represents the PBA, and the z axis is the probability that the aircraft belongs to each cell.

In Figure 9, it is clear that the effect of wind might bring the aircraft in the Weather hazards areas WHA, even if with a small probability. Computing the probability distribution Π_t for $t \in \{1, \cdots, T\}$, it is possible to compute the probability of entering the WHA area in the time interval $[0, T\Delta]$. In our case study, we considered $T = 70$.

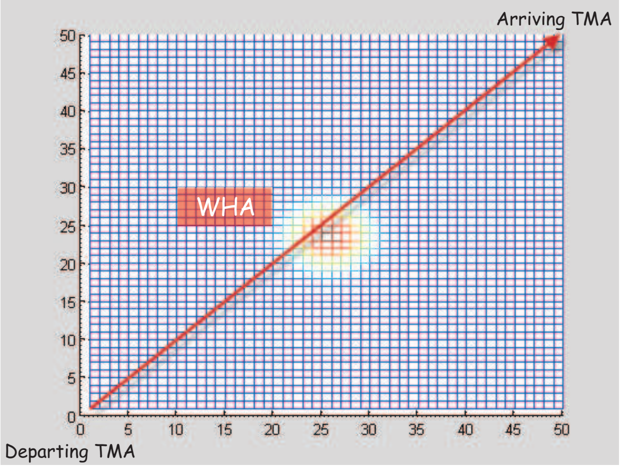


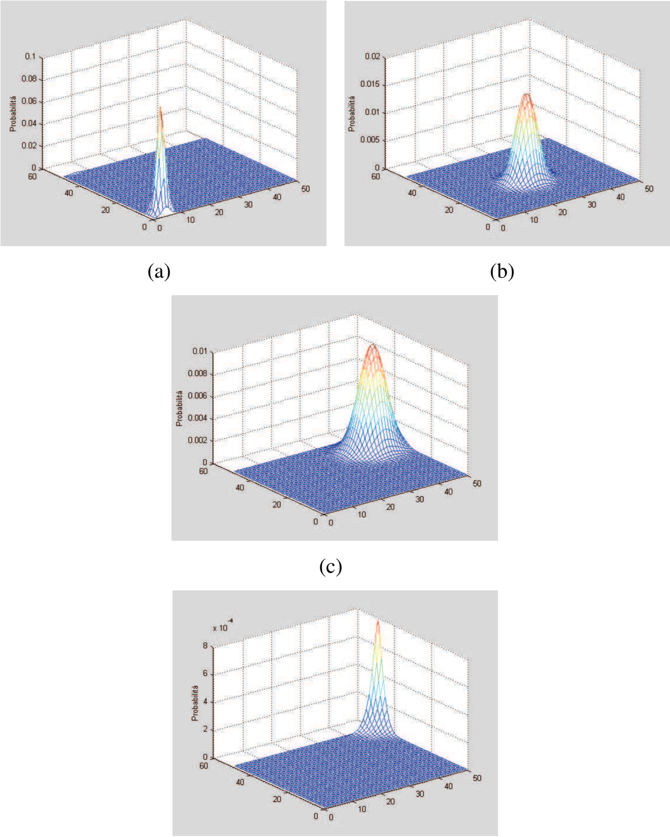
Fig. 9: Trajectory deviation at time 30 s. due to the wind deterministic component.

B. PCTL model checking

Given the Markov Chain abstraction and the WHA area defined by the set $[2,4]km \times [5,6]km$, we use the obtained matrix Π as an input to the tool PRISM [5], [6], [7], in order to perform model checking of the following PCTL properties.

1. Does the aircraft eventually reach the arriving TMA point, with probability greater than a value P? This formula can be expressed in PCTL by the unbounded until formula

$$\mathit{TRUE}\ \mathcal{U}\ \mathit{TMA}. \tag{4}$$



(a)

(b)

(c)

(d)

Fig. 10: Probability distribution evolution at times 5 s. (a), 30 s. (b), 50 s. (c), 70 s. (d).

Using PRISM with our abstraction we verified that, on our model, the property is satisfied with probability $P \geq 0.85$.

2. Does the aircraft eventually reach the arriving TMA point without passing through the WHA area, with probability greater than a value P? This formula can be expressed in PCTL by the unbounded until formula

$$\mathit{WHA}\ \mathcal{U}\ \mathit{TMA}. \tag{5}$$

Using PRISM with our abstraction we verified that, on our model, the property is satisfied with probability $P \geq 0.80$.

3. Does the aircraft reach the arriving TMA point within the Controlled Time of Arrival (CTA) and without passing through the WHA area, with probability greater than a value P? This formula can be expressed in PCTL by the bounded until formula

$$\mathit{WHA}\ \mathcal{U}_{\leq CTA}\ \mathit{TMA}. \tag{6}$$

Using PRISM with our abstraction we verified that, on our model, the property is satisfied with probability $P \geq 0.25$ for $CTA = 50s.$, with probability $P \geq 0.77$ for $CTA = 60s.$, and with probability $P \geq 0.80$ for $CTA = 70s.$.

The abstraction approximation introduces an error in the probabilistic evolution of the Markov Chain with respect to the original dtSHS, which depends on δ. In this paper, and using the bounds derived in [9], we considered a partition that introduces an error of 0.1 in the steady state probability of the abstraction. This is due to the limited resources of the hardware used for the simulations (a 1.8 GHz CPU takes 1 hour for constructing the Markov Chain abstraction). However, according to the results illustrated in [9], the precision of the abstraction can be arbitrarily chosen by decreasing δ and using faster CPUs. Moreover, executing model checking to our model through the tool PRISM is extremely fast on the abstraction Markov Chain (it takes a few seconds for each PCTL formula) even with a slow CPU. It is fundamental to stress that there exist no tools that perform stochastic model checking over a dtSHS: for this reason, our methodology is a technological enabler for applying automatic stochastic model checking to dtSHSs. Using model checking through our methodology, it is possible to determine the probability that the aircraft reaches the arriving TMA. The crew can use this value to decide whether to continue on the Business Trajectory, or to change the flight plan in order to avoid the WHA. For this reason, our methodology can be a useful tool to validate and apply new ATM concepts (in our case study, A^3 ConOps).

V. Conclusions

In this paper, we apply a methodology for formal reasoning based on stochastic hybrid systems theory and abstraction algorithms for dynamical systems, that provides a powerful framework to analyze multi-agents stochastic models of ATM procedures. We use PRISM model checker tools for verifying PCTL probabilistic properties of ATM scenarios. We applied our methodology to a simple single-agent ATM scenario, in the context of the concept A^3 ConOps. Future work aims to apply our methodology in a compositional framework, in order to overtake computational complexity issues in multi-agent systems.

Acknowledgments

This work was partially supported by European Commission under STREP project n.TREN/07/FP6AE/S07.71574/037180 IFLY.

References

[1] J.-M. Loscos, “Asas:towards new cooperation based on airborne spacing,” *Revue Technique de la DTI, ISNN 776-1239*, December 2005.

[2] “D6.1b qualitative risk assessment for asep-itp, v.1.0,” ASSTAR Projects, Tech. Rep., 01 February 2007.

[3] G. Cuevas, I. Echegoyen, J. Garcia, P. Casek, C. Keinrath, R. Weber, P. Gotthard, F. Bussink, and A. Luuk, “Autonomous Aircraft Advanced A^3 ConOps,” iFly, Tech. Rep., January 2009, deliverable 1.3.

[4] M. Kwiatkowska, G. Norman, and D. Parker, “Stochastic model checking,” in *Formal Methods for the Design of Computer, Communication and Software Systems: Performance Evaluation (SFM07)*, ser. Lecture Notes in Computer Science, M. Bernardo and J. Hillston, Eds. Springer, 2007, vol. 4486 (Tutorial Volume), pp. 220–270, to appear.

[5] A. Hinton, M. Kwiatkowska, G. Norman, and D. Parker, “PRISM: A tool for automatic verification of probabilistic systems,” in *Proc. 12th International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS’06)*, ser. LNCS, H. Hermanns and J. Palsberg, Eds., vol. 3920. Springer, 2006, pp. 441–444.

[6] M. Kwiatkowska, G. Norman, and D. Parker, “Prism: Probabilistic model checking for performance and reliability analysis,” *ACM SIGMETRICS Performance Evaluation Review*, 2009.

[7] “www.prismmodelchecker.org.”

[8] A. Abate, M. Prandini, J. Lygeros, and S. Sastry, “Probabilistic reachability and safety for controlled discrete time stochastic hybrid systems,” *Automatica*, 2007, accepted for publication.

[9] A. D’Innocenzo, A. Abate, M. Di Benedetto, and S. Sastry, “Approximate abstractions of discrete-time controlled stochastic hybrid systems,” in *Proceedings of the 47th IEEE Conference of Decision and Control*, Cancun, MX, December 2008, pp. 221–226.

[10] W. Glover and J. Lygeros, “Deliverable number d1.3: A multi-aircraft model for conflict detection and resolution algorithm evaluation,” HYBRIDGE, Tech. Rep., February 18 2004, project: Distributed Control and Stochastic Analysis of Hybrid Systems Supporting Safety Critical Real-Time Systems Designe.

[11] “www.eurocontrol.int/eec/public/standard_page/proj_bada.html.”