A HYBRID APPROACH FOR SAFETY ANALYSIS OF AIRCRAFT SYSTEMS

E. Villani[+], P. E. Miyagi*

[+] *Instituto Tecnológico de Aeronáutica, Department of Mechanical Engineering*
12228-900 São José dos Campos, São Paulo, BRAZIL
** Escola Politécnica, University of São Paulo*
Av. Prof. Mello Moraes, 2231 CEP 05508-900 São Paulo, BRAZIL
e-mail: evillani@ita.br, pemiyagi@usp.br

Abstract: This paper introduces the use of a hybrid modelling and simulation approach for the analysis of safety issues in aircraft systems. Traditionally, safety analysis in aircraft industry is performed without considering the system dynamics. In this paper the dynamics of the aircraft components are modelled using Petri nets and differential equations. Faults are incorporated in the model using probabilistic distributions functions. The reliability of the system under fault is then estimated by simulation. The approach is applied to the landing system of a military aircraft in order to compare two different control strategies for detecting and processing faults.

Keywords: Petri nets, hybrid systems, probabilistic behaviour, safety analysis.

1. INTRODUCTION

Safeness is one of the major concerns in the design of aircraft systems. Most of the aircraft components are provided with redundancy. The degree of redundancy of each component depends on a number of factors such as the kind and probability of fault, per hour of flight. Usually, in order to estimate how safe an aircraft system is, the probabilities of fault in each one of the system components are combined in a static approach using methods such as fault-tree analysis. These methods derive the total probability of fault in the system, which must be within pre-defined limits. The maximum allowed probability of fault depends on how the system deteriorates the level of flight quality and how it affects the aircraft operation and functionality (Stevens; Lewis, 1992).

An important limitation of the methods current under usage in the aeronautic industry is that they do not consider the system dynamics when analysing safeness. When the aircraft behaviour is taken into account, more precise and detailed information can be obtained. Among the issues to be investigated are:

- How a component fault affects the aircraft system behaviour?
- How does it influence the probability of fault in other components?
- How to estimate the probability of critical scenarios that combine a set of component faults?
- How does redundancy affect the system behaviour?
- How to estimate the probability of a wrong diagnostic and its consequences?

In this context, this paper proposes the use of hybrid system modelling and simulation techniques for analysing how the aircraft system dynamics may affect or influence the occurrence, detection and diagnosis of faults. The problem of fault modelling has already been approached by a number of works in many domains. However, most of them consider the problem either from a discrete (Ait-Ameur et al, 2003; Lundqvist, Asplund, 2003) or continuous (Matsuura et al, 2005) point of view. The use of hybrid simulation for the analysis of aircraft safeness is discussed in Pritchett et al (2000).

A hybrid approach is necessary when the system under analysis mixes both continuous and discrete behaviour (Alla, David, 2004). A number of aircraft systems are typically classified as hybrid. They incorporate continuous dynamics such as the continuous positioning of surfaces or the pressure evolution in a hydraulic system, as well as discrete sequence of events, such as switching between components in the case of fault, or executing the command sequences for landing and take-off.

The hybrid modelling formalism considered in this work is the Object-Oriented Differential Predicate Transition nets (OO-DPT net). It combines Petri net for the discrete part and differential equation systems for the continuous one. The object-oriented paradigm is incorporated in order to achieve modularity. The problem of design and analysing aircraft systems using DPT nets has already been presented before (Villani, Miyagi, Valette, 2003). However, previous works have assumed that the system is operating under nominal conditions, i.e., without modelling equipment redundancy and the possibility of fault occurrences. In this paper, faults are incorporated into the models by using probabilistic distribution functions.

The landing system of a military aircraft is presented as an example. Particularly, the approach is used to compare two different control strategies that combine the output of a set of redundant sensors in order to detect fault and avoid dangerous states.

This paper is organized as following. Section 2 describes the proposed approach and the modelling formalism. Section 3 describes the example and Section 4 presents some conclusion.

2. THE APPROACH FOR SAFETY ANALYSIS

2.1 Overview

The proposed approach for safety analysis is divided into the following steps: Step1 - System modelling, Step 2 - Specification of safety requirements, Step 3 – Simulation.

The approach is illustrated in Section 3 using as an example a landing system. Basically, in Step 1, the behaviour of the system under analysis is modelled using the OO-DPT net (presented in Section 2.2). For this purpose the designers must specify the list of all possible faults in each component. A fault may be a discrete event, such as an ON/OFF sensor is blocked in the OFF position. Or faults may also be a continuous activity, such as a leakage in the hydraulic system. In this case the fault is characterized by a set of parameters associated to continuous variables, such as the amount of the leakage per time unit. In the OO-DPT nets faults are modelled by probabilistic distribution functions, which are explained in Section 2.2. The model built in Step 1 must include not only the aircraft components and equipment but also the automatic control system. The strategies used for fault detection, diagnostic and treatment are also modelled as OO-DPT objects.

Step 2 consists of specifying the safety requirements. The system designers must determine what the critical situations are and how is the interaction with the pilot in the case fault. Particular emphasis must be given to the analysis of situations that may result in a wrong diagnostic made either by the automatic control system or the pilot.

Finally, in Step 3 the critical situations are translated into properties that the system must fulfil. An example is to establish a maximum probability to be in a critical state. The requirements are then analysed using Monte Carlo simulation.

2.2 The OO-DPT net

The OO-DPT net is the modelling formalism adopted in this work. It has been introduced in (Villani, Miyagi, Valette, 2005) and it is based on the incorporation of object-oriented (OO) concepts to the Differential Predicate-Transition (DPT) net, proposed in (Champagnat et al, 1998). The OO paradigm assures that an aircraft system can be specified by combining the model of the system components, such as sensors and actuators.

According to the definition of OO-DPT net, the model of a system is composed of the a set of objects organized in 'n' classes (C_1, C_2, …, C_n). Each class C_i is modelled by a DPT net, which defines an interface between differential equation systems and Petri net elements. Its main features are:

- Each object $O_{w.i}$ of the class C_i is represented by a token or a set of tokens in the DPT net of C_i.
- A set of variables (X_i) is associated with each object of the class C_i: they correspond to the attributes of the class. They are divided in X_{co_i}, X_{int_i}, X_{pb_i} and X_{im_i}. X_{co_i} are constant parameters and do not change their value during the object lifetime. X_{int_i} are internal variables and other objects cannot access their value. X_{pb_i} are public variables, other objects can read their value. X_{im_i} are image variables, they are public variables of other objects that are read by $O_{w.i}$.
- A differential equation system (F_{j_i}) is associated with each place (p_{j_i}): it defines the dynamic of a sub-set of X_i according to the time (θ), when a token of $O_{w.i}$ is in p_{j_i},.
- An enabling function (e_{j_i}) is associated with each transition (t_{j_i}): it triggers the firing of the enabled transitions according to the value of X_i.
- A junction function (j_{j_i}) is associated with each transition (t_{j_i}): it defines the value x_i associated with the tokens of the output places of t_{j_i} after the transition firing.

The communication among objects can be discrete or continuous. The continuous interactions are modelled by sharing the continuous variables among objects (e.g. $O_{w.i}$ and $O_{z.v}$). The value of the shared variables is determined by one object ($O_{w.i}$), where it is defined as a public variable (X_{pb_i}), and can be used in the junction function, the equation systems or the enabling function of another object ($O_{z.v}$), where it is defined as an image variable (X_{im_z}). The discrete interactions are method calls. Each class offers methods that are associated with its transitions and that can be requested by other classes. A method call is modelled as the fusion of two transitions: the transition t_{j_i} of the class C_i that offers the method and the transition t_{w_v} of the class C_v that calls the method. The method call happens when both transitions are enabled in their classes.

In order to model the occurrence of faults, probabilistic behaviour must be introduced into the OO-DPT net. The problem of modelling uncertainty in hybrid system has already been approached in many works of the literature (e.g. Pola et al. (2003)). Among the formalisms that model the discrete dynamics using Petri net is the Fluid Stochastic Petri net (Horton et al, 1996) and Dynamically Coloured Petri (Everdij, Blom, 2005). The introduction of uncertainty into models that merge Petri net and differential equation system is briefly approached in (Khalfaoui, 2003) and is based on Generalized Stochastic Petri net. In the proposed approach, probabilistic behaviour is introduced in junction functions. In this case, a probabilistic distribution (PD) is used to define the value of a variable after the firing of a transition. Each time the transition fires a new value is attributed to the variable according to the probabilistic distribution.

As an example, Figure 2 presents the OO-DPT net of the class *C_1 – Hydraulic Cylinder*, which models the behaviour of the cylinder illustrated in Figure 1.

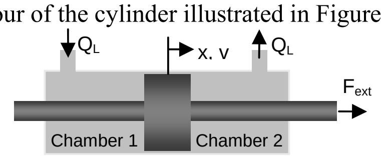

Figure 1. Hydraulic Cylinder.

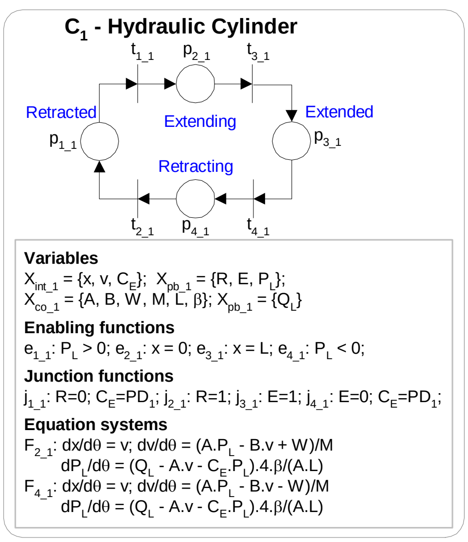

Figure 2. OO-DPT net of class C_1 – Hydraulic Cylinder.

The hydraulic cylinder can be completely extended (p_{3_1}) or retracted (p_{1_1}), or can be extending (p_{2_1}) or retracting (p_{4_1}). When moving, the dynamics of the cylinder is set according to a differential equation system (F_{2_1} and F_{4_1}) composed by 3 equations. The time is represented as 'θ'. The first equation relates the speed of the piston with its position. The second equation is the balance of the forces acting on the piston. The third equation is the balance of mass in Chambers 1 and 2 and it includes the effects of the fluid compressibility on the cylinder dynamics.
The class variables are:

- x, v – position and speed of the piston (x=0 when the piston is completely retracted);
- Q_L – amount of fluid entering Chamber 1 and leaving Chamber 2.
- R, E – auxiliary variables that indicate if the piston is completely extended or retracted;
- C_E – leakage coefficient;
- P_L - difference between the pressure in Chamber 1 and that in Chamber 2;
- V_{PL} – derivative of P_L;
- A, M – area and mass of the piston;
- B – damper coefficient;
- F_{ext} – external force;
- L – maximum value of x;
- β – compressibility coefficient of the fluid;

The communication of the class *C_1 - Hydraulic Cylinder* with other classes is made by sharing variables R, E and Q_L, and by reading variable P_L and V_{PL} from the class *C_3 – Electro-valve* (presented in Section 3).

The OO-DPT net of class C_1 models a fault as a certain amount of hydraulic fluid leakage. The pressure P_L and a leakage coefficient C_E, which is set according to the probabilistic distribution PD_1, determine the amount of leakage. The probabilistic distribution combines the probability of having different kinds of fault with the probability of having a certain amount of leakage when a certain fault occurs. A qualitative example of PD_1 is presented in Figure 3. The low values of C_E are the normal amount of leakage (when no fault occurs), the values on the centre of the graphic are related to problems in the cylinder seal, the high values of C_E are related to structural faults such as cracks in the actuating cylinder.

The next section presents the example considered in this paper: the landing system of a military aircraft.

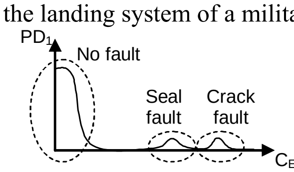

Figure 3. Probabilistic Distribution PD_1 of class C_1 – Hydraulic Cylinder.

3. THE EXAMPLE

The case-study considered in this paper is the landing-system of a military aircraft. It is composed of 3 landing sets (called A, B and C) containing each one a door and a landing-gear. The sequence that must be performed at landing consists of opening the doors of the 3 landing-gear compartments, extending the landing-gears and closing the doors. A similar sequence must be performed at take-off. The landing-gear and door movement is performed by a set of actuating hydraulic cylinders. For each door, a hydraulic cylinder opens and closes the door. For each landing gear, a hydraulic cylinder extends and retracts the landing gear. The hydraulic cylinders are moved by electro-valves. Figure 4 illustrated the hydraulic circuit for the doors.

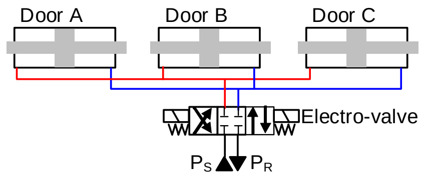

Figure 4. Door hydraulic circuit.

Discrete sensors inform the control system about the positions of the actuating cylinders, the pressure in the hydraulic system, among others. Each discrete sensor is provided with redundancy 3. Sensors signal are used for coordinating the landing system movements during landing and take-off. Furthermore, during cruise and other flight phases, they are constantly read and processed with a certain frequency in order to monitor the landing system and detect problems. In the case of fault the pilot is notified.

The purpose of this example is to analyze how the sensor redundancy augments the system safety and what is the best strategy for combining the signal from the redundant sensors. Because of the limited space, only the door sensors are considered in this paper. Each door has two different kinds of sensor, one indicating if the door is completely open (DO sensor) and one indicating if it is completely closed (DC sensor). Each of them has redundancy 3. In order to unambiguously refer to a sensor, the notation DO_{xy} is used to refer to the DO sensor 'x' (1, 2 or 3) of the door 'y' (A, B or C). Similarly, DC_{xy} is used to identify the DC sensors. The same configuration is adopted for the landing-gears.

Usually, the sensor outputs are processed in 4 levels. In Level 1, the outputs of three redundant sensors (e.g. DO_{1A}, DO_{2A} and DO_{3A}) are compared among them and a combined sensor output is generated. In Level 2, for each door the combined sensor output of door open (DO sensors) is compared with the combined sensor output of door closed (DC sensors) and a door output is provided. It indicates the current state of the door (open, closed, moving or fault). The output of Level 2 is also used in timing monitoring functions. When an order to open or close the door is emitted, the operation must be complete within a time interval. Otherwise, a fault is detected and informed to the pilot. The same approach is performed for each landing-gear. In Level 3, for each landing set, the landing-gear output is compared with the door output and a landing-set output is generated. At this level, faults are detected in situations such as if the door is closed and the landing-gear is moving. Finally, Level 4 compares the three landing-set outputs of Level 3 and a landing-system output indicating the current state of the landing system.

In this example, two different control strategies are considered for processing the sensors at Level 1 and Level 2. In the Level 1 of Strategy 1, if three or two output signals are 'ON', the combined sensor output is 'ON', otherwise the output is 'OFF'. Level 2 of Strategy 1 adopts Table 2 for defining its output.

Table 2 – Rules for Level 2 – Strategy 1.

Level 1 - DO	*Level 1 - DC*	***Level 2***
ON	OFF	*Open*
OFF	ON	*Closed*
OFF	OFF	*Moving*
ON	ON	*Fault*

In the case of Strategy 2, the following rules are considered for Level 1:

- If all the three output signals are 'ON', the combined sensor output is 'ON'. If all the three output signals are 'OFF', the combined sensor output is 'OFF'.
- If two output signals are 'OFF', and one is 'ON', the combined sensor output is 'OFF' and the identity of the sensor with output 'ON' is memorized. If on the next time the sensors are read, the output of this sensor is still different from the other two, this sensor is considered as fault, and from this moment on it is ignored by the control system.
- A similar approach is executed when two output signals are 'ON', and one is 'OFF'.
- If one sensor has been eliminated and the other two are 'OFF', the combined sensor output is 'OFF'. If one sensor has been eliminated and the other two are 'ON', the combined sensor output is 'ON'.
- If one sensor has been eliminated and the other two sensor signals are different from each other, the combined sensor output remains unchanged and an error is memorized. If on the next time the sensors are read, the two outputs are still different, the combined sensor output is fault.

The rules for defining the Level 2 output for Strategy 2 are presented in Table 3.

Table 3 – Rules for Level 2 – Strategy 2.

Level 1 - DO	*Level 1 - DC*	***Level 2***
ON	ON	*Fault*
ON	OFF	*Open*
ON	Fault	*Open*
OFF	ON	*Closed*
OFF	OFF	*Moving*
OFF	Fault	*Fault*
Fault	ON	*Closed*
Fault	OFF	*Fault*
Fault	Fault	*Fault*

This is typically a hybrid problem because an abnormal output of the system can be the result of a sensor fault or of a leakage in a hydraulic cylinder (continuous variable CE of Figure 2). In the second case, the leakage affects the cylinder dynamics, slowing it down or even reverting movement direction if the weight of the landing gear is against it. In this situation the control system must detect a fault and inform the pilot, which can then activate a backup system for the movement of the landing gear. The detection can be affected by a sensor fault and depends on the strategies at Levels 1 and 2.

3.1 System Modelling

Once that the focus of this example is on the processing of door sensor signals, the model of the system is limited to the doors hydraulic cylinders, electro-valve, sensors and controller. Similar objects are used to model the landing-gear hydraulic system.

The system model is composed of a set of 6 classes: *C_1 - Hydraulic Cylinder*, *C_2 - Discrete Sensor*, C_3 - *Hydraulic Electro-valve*, *C_4 – Level_1 Processor*, *C_5 – Level_2 Processor*, *C_6 - Door Sequence Controller*. The first 3 classes model the behavior of the physical components, while C_4, C_5 and C_6 model the control system. The model of class C_1 has already been presented in Section 3. There are three objects of class C_1: $O_{1.1}$ – Door A, $O_{2.1}$ - Door B, $O_{3.1}$ – Door C.

Model of Class C_2 – Discrete Sensor
Under normal operation, the discrete sensor is either ON (p_{2_2}) or OFF (p_{1_2}). The switching between ON and OFF takes place according to the value of the image variable S_{in}, which corresponds either to the public variable E or R of class *C_1 – Hydraulic Circuit*. However, each time the sensor must switch, a fault may occur and the sensor may go to states 'Blocked ON' (p_{7_2}) or 'Blocked OFF' (p_{6_2}). The fault corresponds to the firing of t_{4_2} or t_{5_2}, and happens with a probability of ($1-P_{OFF}$) and ($1-P_{ON}$), respectively. The probabilistic distribution PD_2 sets a random value between 0 and 1 to the variable 'rd', which determines the occurrence of a fault. There are 18 objects of class C_2: $O_{1.2}$ – DO_{1A}, $O_{2.2}$ – DO_{2A}, $O_{3.2}$ – DO_{3A}, $O_{4.2}$ – DO_{1B}, $O_{5.2}$ – DO_{2B}, $O_{6.2}$ – DO_{3B}, $O_{7.2}$ – DO_{1C}, $O_{8.2}$ – DO_{2C}, $O_{9.2}$ – DO_{3C}, $O_{10.2}$ – DC_{1A}, $O_{11.2}$ – DC_{2A}, $O_{12.2}$ – DC_{3A}, $O_{13.2}$ – DC_{1B}, $O_{14.2}$ – DC_{2B}, $O_{15.2}$ – DC_{3B}, $O_{16.2}$ – DC_{1C}, $O_{17.2}$ – DC_{2C}, $O_{18.2}$ – DC_{3C}.

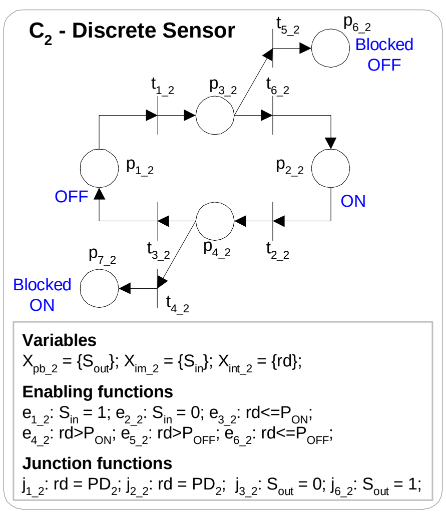

Figure 5. OO-DPT net of class C_2 – Discrete Sensor

Model of Class C_3 – Hydraulic Electro-Valve
This class relates the pressure on the hydraulic circuit (P_L) with the position of the electro-valve (Figure 4). The OO-DPT net is presented in Figure 6. The equation that defines the pressure PL combines the amount of fluid through the valve and the decreasing of pressure when passing through a restriction. No fault is considered for the electro-valve. The class variables are:

- V_{PL} – derivative of the pressure PL;
- Q_T – total amount of fluid to the door cylinders.
- Q_{LA}, Q_{LB}, Q_{LC} – amount of fluid entering Cylinders A, B and C.
- K_1, K_2 – constants of the electro-valve;
- P_S, P_R – Supply and return pressures of the hydraulic system.

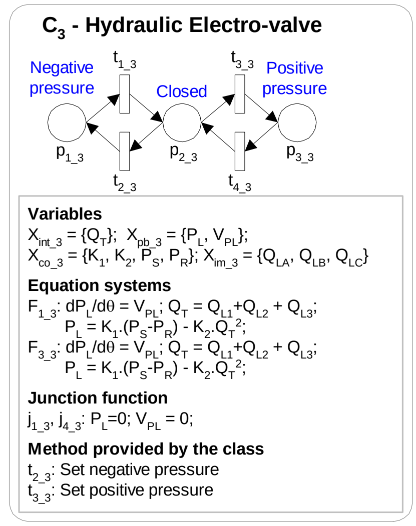

Figure 6. OO-DPT net of class C_3 – Hydraulic Electro-valve.

Model of Class C_4 –Level_1 Processor
This class represents the processing of sensor outputs at Level 1 and has mainly a discrete dynamics. It receives the output of 3 redundant sensors as image variables and provides an output to an object of class *C_5 – Level_2 Processor*. The OO-DPT model of this class varies according to the chosen strategy. Due to limited space, it has not been presented. There are 6 objects of this class: $O_{1.4}$ – DO Output_A, $O_{2.4}$ – DO Output_B, $O_{3.4}$ – DO Output_C, $O_{4.4}$ – DC Output_A, $O_{5.4}$ – DC Output_B, $O_{6.4}$ – DC Output_C.

Model of Class C_5 – Sensor Level_2
The model of this class is similar to the model of class C_4, but instead of processing the sensor signals, this class processes the output of the objects of class C_4 and warns the pilot in the case of fault. There are 3 objects of this class: $O_{1.5}$ – Door Output_A, $O_{2.5}$ – Door Output_B, $O_{3.5}$ – Door Output_C.

Model of Class C_6 – Door Controller
This class controls the operation of the electro-valve and the door opening and closing according to commands emitted by the pilot. It is important to observe that it uses the output of the objects of class C_5 to detect any fault. There is only one object of this class: $O_{1.6}$ – Door Controller.

3.2 Safety Requirements

The number of detected faults and wrong diagnostics are selected as a parameter to evaluate the two control strategies. The best strategy is the one that detects more faults and minimizes the wrong diagnostics. The following situation can be found and are particularly critical to the system operation:

- *Case 1*: Due to sensor faults, the control system indicates the doors are completely open when they are not yet. This is a particular critical situation because the control system will then follow the landing sequence, extending the landing-gear. A crash between the door and landing-gear may occur, damaging the landing system. Similarly, if the control system does not detect that one or more doors have a critical leakage, the pilot is not warned and therefore cannot activate the emergency system.

- *Case 2*: One of the objects $O_{1.5}$, $O_{2.5}$ or $O_{3.5}$ of class C_5 - *Sensor Level_2* of the control system warns the pilot of a fault in the sensors.
- *Case 3*: The object $O_{1.6}$ – Door Controller of the control system detects a fault in the hydraulic actuator (at least one of the doors is not open in maximum time interval) and warns the pilot.

3.3 Simulation

No software is available yet for simulating OO-DPT nets. As a consequence, Monte Carlo simulation has been performed by translating the OO-DPT net into MatLab language. The continuous dynamics is discretized with fixed time steps. Each class corresponds to a different subroutine. Furthermore, due to the confidential nature of aircraft system data, the results published in this section are not based on real values for model parameters and probabilities of faults. They must be considered only as an example of simulation output and not as the behaviour of a real military aircraft.

Referring to Figure 5, the probability of a discrete sensor remaining blocked 'ON' or 'OFF' is P_{ON} = P_{OFF} = 0.05. Referring to Figure 2, the PD_1, which assign a value to C_E, is the result of the sum 3 distribution, with the following weight and media:

- Case of no fault: weight=0.8, media=0.01
- Case of seal fault: weight=0.1, media=0.5
- Case of crack fault: weight=0.1, media=0.9

Using this data, simulation is performed by executing sequences of extending and retracting cycles. Table 4 presents the percentage of retracting and extending cycles that corresponds to Case 1, 2 or 3 (described in Section 2). The best strategy must minimize the occurrence of Case 1 (wrong diagnostic) and maximize the occurrence of Cases 2 and 3 (correct fault detections) According to this result the best strategy is Strategy 2.

Table 4 – Simulation results.

	Strategy 1	**Strategy 2**
Case 1	0.01	0.00
Case 2	7.28	7.19
Case 3	0.89	1.42

4. CONCLUSION

This paper presents the application of a hybrid approach for the safety analysis of aircraft systems. The proposed approach is based on OO-DPT net and faults are modelled by means of probabilistic distributions. The approach is applied to the landing system of a military aircraft. As an example the problem of analyzing sensor redundancy and comparing control strategies is detailed.

Results are currently obtained by Monte Carlo simulation. One of the points to be investigated in future work is the development of techniques and strategies to reduce the number of simulations, such as in the Dynamically Petri nets (Everjid, Blom, 2005). For this purpose the modelling flexibility may be restricted in order to satisfy the strong Markov properties. This is a particular critical point in aerospace application because, differently from the data used in Section 3.3, the probabilities of fault of aircraft components and equipment are extremely low, reaching 10^{-5} and even low.

ACKNOWLEDGES

The authors would like to thank the partial financial support of the governmental agencies FAPESP, CNPq and CAPES.

REFERENCES

Ait-Ameur, Y. et al. (2003) "Robustness analysis of avionics embedded systems." *Proceedings of the ACM SIGPLAN Conference on Language, Compiler, and Tool for Embedded Systems*, San Diego.

Alla, H.; David, R. (2004), *Discrete, Continuous, and Hybrid Petri Nets*. Springler Verlag.

Champagnat, R. et al. (1998), "Modelling and Simulation of a Hybrid System through Pr/Tr PN-DAE Model", *3rd International Conference on Automation of Mixed Processes*, Reims.

Everdij, M.H.C.; Blom, H.A.P., (2005) "Piecewise deterministic Markov processes represented by dynamically coloured Petri nets", *Stochastics*, vol.77, n.1, pp.1-29.

Horton, G.; Kulkarni, V.; Nicol, D.; Trivedi, K. "Fluid stochastic Petri nets" *ICASE Report 96-5*, Hampton, 1996.

Lundqvist, K.; Asplund, L. (2003) "A Ravenscar-Compliant Run-time Kernel for Safety-Critical Systems." *Real-Time System*, vol.24, n.1, pp.29-54.

Khalfaoui, S. (2003), "Méthode de Recherche des Scénarios Redoutés pour l'Evaluation de la Sûreté de Fonctionnement des Systèmes Mechatroniques du Monde Automobile", PhD Thesis, *Laboratoire d'Analyse et d'Architecture des Systèmes du CNRS*, Toulouse.

Matsuura, J.P.; Yoneyama, T.; Galvão, R.K.H., (2005) Learning bayesian networks for fault detection: application to the 747 longitudinal motion. *18th Int. Cong. Of Mechanical Engineering*, Ouro Preto.

Pola, G. et al. (2003), "Stochastic hybrid models: an overview". *IFAC Conference on Analysis and Design of Hybrid System (ADHS)*, St Malo.

Pritchett, A.R.; Lee, S.; Huang, D.; Goldsman, D. (2000) Hybrid-system simulation for national airspace system safety analysis, Proc. of the 2000 Winter Simulation Conference, Orlando.

Stevens, B.L.; Lewis, F.L. (1992), *Aircraft control and simulation*. Jonh Wiley & Son.

Villani, E.; Miyagi, P.E., (2003), "Petri Net and OO for the modular analysis of an aircraft landing system", *17th International Congress of Mechanical Engineering*, São Paulo.

Villani, E.; Miyagi, P.E.; Valette, R., (2005), "A Petri-Net based Object-Oriented Approach for the Modelling of Hybrid Productive Systems", *Non-linear Analysis*: *Theory and Methods*, vol.62, n.8, pp.1394-1418.