Aircraft Navigation Systems Safety Assessment via Probabilistic Model Checking

Gabriel Duarte Pasa[1] and Valdivino Alexandre de Santiago Júnior[2(✉)]

[1] Embraer, Avenida Brigadeiro Faria Lima, 2170, Putim, São José dos Campos, SP 12227-901, Brazil
gabriel.pasa@embraer.com.br

[2] Instituto Nacional de Pesquisas Espaciais (INPE), Avenida dos Astronautas, 1758, Jardim da Granja, São José dos Campos, SP 12227-010, Brazil
valdivino.santiago@inpe.br

Abstract. The safety assessment process is a mandatory step in the development and certification of safety-critical systems such as the ones in the aerospace industry. In this work we show how Probabilistic Model Checking, a Formal Verification method, can help to assess the safety of navigation systems for a civil commercial transport category aircraft. The process involves a top-down approach identifying functions and its respective failure modes. Each failure event is associated with a hazard level, with an inverse relationship between the maximum acceptable probability and the event severity. Fault Tree Analysis (FTA) is the most commonly used method to quantify each event's probability, but probabilistic models are also accepted as means of compliance demonstration, as per ARP-4761. Results show that the use of Probabilistic Model Checking as a means to complement non-formal methods is valuable, where we were able to evaluate the probability of several failure modes described in FAA's AC 20-138D, making use of Continuous-Time Markov Chains (CTMCs) with up 4.3 million of reachable states and 60 million of transitions.

Keywords: Safety assessment · Aerospace systems · Probabilistic model checking · Aircraft navigation

1 Introduction

The safety assessment process is part of the development of modern systems in several industries such as aerospace, biomedical, nuclear, and automotive. With the rapid advent of technology over the past years, those safety-critical systems are becoming more and more complex, which brought the industries to a collective effort to build guidelines for safe systems design. In the aeronautical industry, the Part 25 airworthiness standards introduce the fail-safe design concept which considers the effect of single and multiple failure modes and defines some basic

© Springer Nature Switzerland AG 2021
O. Gervasi et al. (Eds.): ICCSA 2021, LNCS 12952, pp. 465–480, 2021.
https://doi.org/10.1007/978-3-030-86973-1_33

objectives pertaining those failures, such as that a single failure, regardless of its probability, shall not prevent continued safe flight and landing [7].

The standard guidelines for conducting safety assessment on civil airborne systems is defined in the SAE ARP-4761 [15]. The methods presented in the document are considered by the major regulation authorities as accepted means of compliance with the safety related airworthiness requirements and those methods are directly related with the ones defined in the SAE ARP-4754A [16], guidelines for development of civil aircraft and systems. The process is based on a top-down approach, starting from the aircraft functions, and Failure Mode and Effect Analysis (FMEA) and Failure Hazard Analysis (FHA), assessing how the aircraft system and equipment failure modes may lead to aircraft level failure conditions. The ARP-4761 [15] suggests a few different methods for performing the safety assessment but the most widely used is the Fault Tree Analysis (FTA).

Even though the methodology presented in the ARP-4761 [15] is what is commonly accepted by the regulation authorities as means of compliance with the airworthiness requirements, this does not mean that alternative methods can not be considered. The AC 25.1309-1B [7] and ARP-4761 [15] are intended to provide guidance to the product development engineer, although they are not mandatory. As the aeronautical industry evolves with new methodologies, techniques, and tools, so does the means of compliance with the airworthiness standards. An alternative to the traditional FTA is the use of probabilistic models on the safety assessment process, and even though these methods are not yet widely applied, they are predicted in the ARP-4761 [15].

Probabilistic Model Checking is a Formal Verification method which is roughly composed of two concepts: a probabilistic model, such as Discrete-Time Markov Chain (DTMC), Continuous-Time Markov Chain (CTMC), and properties formalised via probabilistic temporal logics, such as Probabilistic Computation Tree Logic (PCTL) and Continuous Stochastic Logic (CSL) [12]. It is an interesting field which has been used to solve problems in several domains such as biology [5], power supply systems [6], communication and networks protocols [11], performance of systems [17], and many more.

When compared to the FTA methodology, a Probabilistic Model Checking analysis brings the possibility to express a temporal sequence of events or state-dependent behaviour of systems. In addition, the introduction of formal-based safety assessment methods brings the possibility of automating certain parts of the safety assessment process.

In this article, we hypothesise that Formal Verification methods can effectively contribute to the process of assessing the safety of aircraft navigation systems, providing an alternative to complement the study carried out using classic but non-formal methods. Hence, we show how Probabilistic Model Checking can help in the safety assessment for commercial transport category aircraft. Our focus is set on a subset of functions related to aircraft positioning and navigation, for which the failure modes and hazard classifications are described in the AC 20-138D [8]. We derived several CTMC models and formalised properties via CSL using the PRISM Probabilistic Model Checker tool [13].

This article is organised as follows. In Sect. 2, we present background while in Sect. 3 we show in detail the methodology we devised to address the problem stated. Results and analysis are in Sect. 4, and in Sect. 5 we present related work. Conclusions and future research directions are presented in Sect. 6.

2 Background

According to [1], "Model checking is an automated technique that, given a finite-state model of a system and a formal property, systematically checks whether this property holds for (a given state in) that model". It has been successfully used for quite some time by academia and some industry sectors. On the other hand, Probabilistic Model Checking [1,12] is an extension of Model Checking relying on probabilistic models (e.g. DTMC, CTMC) and probabilistic temporal logics (e.g. PCTL, CSL). One of the main motivations to develop Probabilistic Model Checking is to relax the rigid notions related to Functional Model Checking (a model satisfies or does not satisfy a formal property), because many systems are subject to various phenomena of a stochastic nature, and trying to figure out whether the system has a 99% chance of failing is often more realistic.

Markov Chains are used to model randomly changing processes, where transitions between the different possible states are only dependent on the current state (i.e. the system has no memory). In this article we consider CTMC as our probabilistic model. A CTMC makes transitions from state to state, independent of the past and similarly to a DTMC, but once entering a state remains in that state, independent of the past, for an exponentially distributed amount of time before changing state again. Moreover, a continuous random variable X has an exponential distribution, with parameter (event rate) λ, if its probability density function is given by ($\lambda > 0$):

$$f(x;\lambda) = \begin{cases} \lambda e^{-\lambda x}, & \text{if } x \geq 0 \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

We can define a CTMC in accordance with the Formal Methods community as follows.

Definition 1: *A (labelled) CTMC is a tuple* $\mathcal{C} = (S, \iota, \mathbf{R}, L)$*, where:*

- S *is a finite set of states,*
- $\iota \in S$ *is the initial state,*
- $\mathbf{R} : S \times S \longrightarrow \mathbb{R}_{\geq 0}$ *is the transition rate matrix,*
- $L : S \to 2^{AP}$ *is a labelling function which assigns to each state* $s \in S$ *the set* $L(s)$ *of atomic propositions that are valid in the state (AP = set of atomic propositions).*

In the above definition, $\mathbf{R}$ assigns rates to each pair of states in the CTMC (used as parameters of the exponential distribution). A transition can only occur between states s and s' if $\mathbf{R}(s, s') > 0$ and, in this case, the probability of this transition being triggered within t time-units equals $1 - e^{-\mathbf{R}(\mathbf{s},\mathbf{s}')\cdot\mathbf{t}}$ (Cumulative Distribution Function).

The probabilistic temporal logic we selected was CSL [12]. CSL extends the classical Computation Tree Logic (CTL) [3] with two probabilistic operators: path ($\mathcal{P}$) and steady-state ($\mathcal{S}$) operators. The syntax of CSL is defined using state and path formulas.

3 Methodology

3.1 Failure Conditions and Hazard Classifications

As per AC 25.1309-1B [7], failure conditions are classified into five classes according to the criticality, ranging from "No safety effect" to "Catastrophic", as described in Table 1. The more severe a failure condition is, the lower is the acceptable probability for it to happen. When demonstrating compliance with the safety objectives, one must classify each mapped failure condition into one of those categories and then demonstrate that the probability of it occurring, per flight hour, is lower than the respective threshold.

Note that AC 25.1309-1B [7] mentions that for "Minor" failure conditions, a quantitative analysis is not required and a numerical range is provided as a reference only (and as such will not be analysed in this work). Naturally, the "No safety effect" conditions do not require a quantitative analysis either.

Table 1. Relationship between probability and severity of failure condition as per AC 25.1309-1B [7].

Failure condition classification	No safety effect	Minor	Major	Hazardous	Catastrophic
Effect on occupants	Inconvenience	Physical discomfort	Physical distress, possibly including injuries	Serious or fatal injury to a small number of passengers or cabin crew	Multiple fatalities
Allowable qualitative probability	No Probability Requirement	Probable	Remote	Extremely Remote	Extremely Improbable
Allowable quantitative probability	No Probability Requirement	$<10^{-3}$	$<10^{-5}$	$<10^{-7}$	$<10^{-9}$

The scope of this work will encompass the aircraft positioning and navigation failure conditions described in the AC 20-138D [8] (refer to Table 2), which fall into two categories: "loss of navigation" and "misleading information". Each failure condition is applicable to a certain type of operation, mostly related to approach procedures (navigation leading to landing).

Table 2. Hazard classifications for the subset of aircraft positioning and navigation functions being evaluated (adapted from FAA AC 20-138D [8]).

	Advisory Vertical Guidance	Enroute/Terminal Area/Non-precision Approach (LNAV or RNP 0.3)	Non-precision Approach with Vertical Guidance (LNAV/VNAV)	LP/LPV Approach	GLS Approach
Loss of Navigation	No Effect	Major	Major	Major	Major
Misleading Information	Minor	Major	Major	Hazardous	Hazardous

The "Advisory Vertical Guidance" function will not be evaluated since it is attributed with at most a "Minor" failure condition, which does not require a quantitative analysis.

The "Enroute/Terminal Area/Non-precision Approach" function is essentially the capacity for the aircraft to determine its lateral position (latitude and longitude) with the appropriate level of position uncertainty (which depends on the type of operation and flight phase). Lateral Navigation (LNAV) is a mode of operation where the aircraft lateral trajectory is typically controlled by the Flight Management System (FMS). Required Navigation Performance (RNP) 0.3 is a similar mode, but it has a stricter position uncertainty limit of 0.3 nautical miles.

The "Non-precision Approach with Vertical Guidance" function is a type of approach procedure with both lateral (LNAV) and Vertical Navigation (VNAV). The vertical guidance is usually a glide path from cruise altitude towards the runway, constructed by the FMS based on the barometric altitude, which is obtained from the Air Data System (ADS).

The "LP/LPV Approach" is another type of approach procedure where the guidance is provided by the Global Navigation Satellite System (GNSS). LP refers to "Localiser Performance" which provides lateral guidance only. Localiser Performance with Vertical Guidance (LPV) is similar to LP but with vertical guidance as well, both provided by the GNSS. In order to achieve the precision required for this type of operation, the GNSS system must be compatible with Satellite-Based Augmentation System (SBAS). In SBAS, geostationary satellites (in addition to the regular satellites from the GNSS constellation) broadcast augmentation information to the receivers, which improves the accuracy and integrity of the position solution.

The "GNSS Landing System" (GLS) approach is similar to LPV, and it uses the GNSS data to calculate both lateral and vertical guidance for the final approach segment. However, instead of using satellite-based, it uses ground-based augmentation to improve the positioning accuracy. A ground station located near the airport runways constantly receives the GNSS satellite signals and calculates the local position errors (it can do it well since the station is fixed at a known position). This correction information is broadcast to nearby receivers via a Very High Frequency (VHF) datalink. This datalink requires another receiver (not the GNSS receiver).

Each aircraft function relies on a specific set of equipment and as such will only be affected by failures in those equipment. The navigation failure modes are also typically attributed to specific operations and flight phases (for instance, a GLS approach failure will only occur during approach), so it is important to also model the flight phase transitions.

3.2 High-Level System Architecture

The high-level avionics system architecture considered in this work is depicted in Fig. 1.

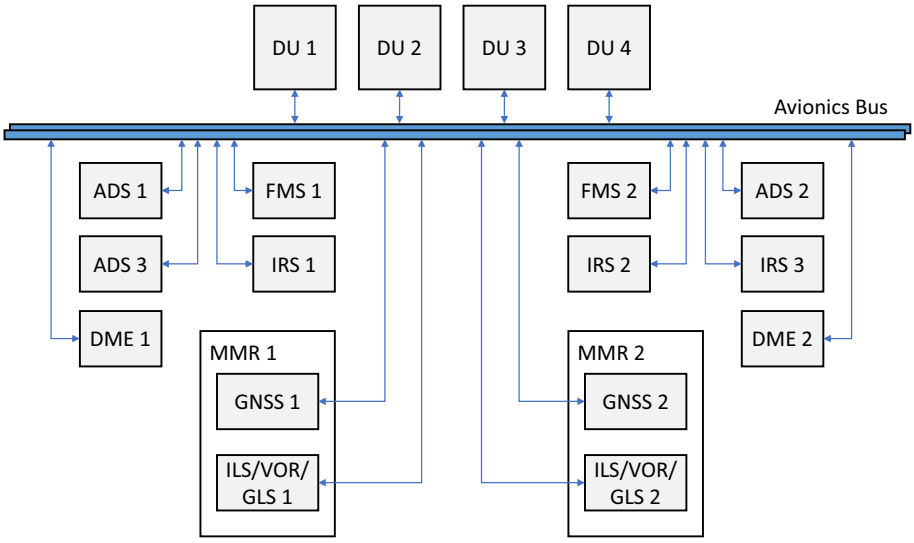

Fig. 1. High-level avionics architecture, including Display Units (DU), Air Data System (ADS), Flight Management System (FMS), Inertial Reference System (IRS), Distance Measuring Equipment (DME), Multi-Mode Receiver (MMR), Global Navigation Satellite System (GNSS), Instrument Landing System (ILS), VHF Omnidirectional Range (VOR), GNSS Landing System (GLS).

The Flight Management System (FMS) is the main navigation computer. It provides high level aircraft commands (desired heading, airspeed, vertical speed) so that the aircraft follows the flight plan (sequence of waypoints from the origin to destination). In order to navigate, the FMS reads data from several sensors, such as the GNSS, Inertial Reference System (IRS), Air Data System (ADS), Distance Measuring Equipment (DME), and Multi-Mode Receiver (MMR).

The display units (DU) are the cockpit displays which provide information for the pilots, such as altitude, airspeed, attitude (roll, pitch, and yaw angles), as well as information from the FMS. The IRS and ADS provide critical information for continued safe flight (i.e. attitude, airspeed, altitude, angle of attack) and as such are triple redundant.

In the scope of this work, only the navigation-related systems are modelled. For brevity and generality, some low-level implementation-specific failure modes are abstracted away, such as avionics bus or display failures.

3.3 Flight Phases and Exposure Times

Many failure conditions are dependent on specific flight phases (either more likely or only possible in specific conditions). This can be seen in the accident statistics shown in Fig. 2 where, for example, the final approach and landing phases combined account for 49% of fatal accidents, while only representing 4% of the total flight time, in a typical 1.5-h flight.

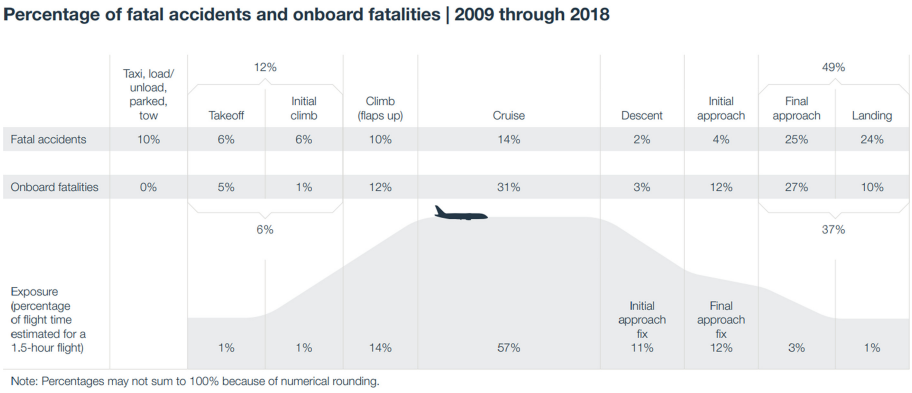

Fig. 2. Commercial airplanes statistics on fatal accidents, on-board fatalities, and exposure times per flight phase. Adapted from [2].

Since the failure conditions evaluated in this work (Table 2) are dependent on the flight phase, a typical flight is modelled based on the exposure times shown in Fig. 2.

3.4 Continuous-Time Markov Chain Models

In order to perform the quantitative safety assessment analysis, a CTMC model is used to map the relevant aircraft flight phases and behaviour of the multiple avionics systems involved in providing the selected subset of aircraft functions being evaluated. The CTMC approach is suitable for the problem at hand since safety assessment usually deals with failure rates (not static probabilities) and the safety objectives are usually expressed in terms of probability per flight hour (which is intrinsically a rate too).

The different systems depicted in the high-level avionics architecture (Fig. 1) were modelled in PRISM as per the diagram in Fig. 3. All systems start as healthy and may present one out of two possible failure modes: they may become failed (annunciated fault) or may present erroneous information (non-annunciated fault). Once failed, the system shall continue failed until the end of the flight. Once erroneous, the system may eventually transition to failed.

The failure mode rates (for annunciated and non-annunciated faults) for each system are listed in Table 3. The values used for those rates are naturally very important for the result of the safety assessment analysis. In our case they were estimated based on experience, but usually these are provided by the equipment manufacturer or, if performing a preliminary safety assessment, these could be used to define reliability requirements for the systems under development, guaranteeing compliance with the safety objectives.

A sample of PRISM code implementing the logic in Fig. 3 can be seen in Listing 1.1. Note that all CTMC models developed in PRISM and also the properties formalised in CSL can be accessed online [14]. Due to formatting reasons, we changed some variable names in the listings below.

The flight phases are modelled as in Fig. 4 and PRISM code is in Listing 1.2. The rate of each transition occurring is the inverse of the average time in each flight phase, following the flight profile depicted in Fig. 2.

Table 3. Equipment failure mode rates.

System	Failure Rate (per hour)	Erroneous Rate (per hour)
FMS	1E-6	1E-7
GNSS	1E-4	1E-6
IRS	1E-6	1E-8
DME	1E-5	1E-6
ADS	1E-6	1E-8
GLS	1E-5	1E-7

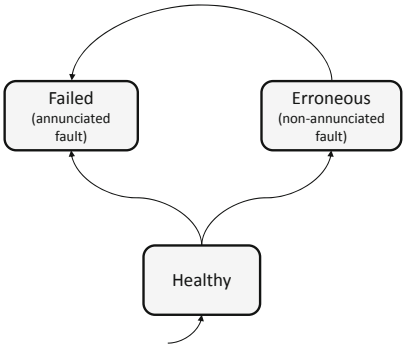

Fig. 3. Equipment state transitions.

```
// --------------------------------
// Inertial Reference System (IRS)
// --------------------------------
const double irs_failure_rate = 1E-6;
const double irs_erroneous_rate = 1E-8;

module irs1
  irs1_fail: bool init false;
  irs1_err: bool init false;

  [] true -> irs_failure_rate : (irs1_fail'=true)&(irs1_err'=false);
  [] (!irs1_fail) -> irs_erroneous_rate : (irs1_err'=true);
endmodule
module irs2 = irs1[irs1_fail=irs2_fail, irs1_err=irs2_err] endmodule
module irs3 = irs1[irs1_fail=irs3_fail, irs1_err=irs3_err] endmodule
```

Listing 1.1. Sample of PRISM code implementing the IRS. All other systems are implemented similarly.

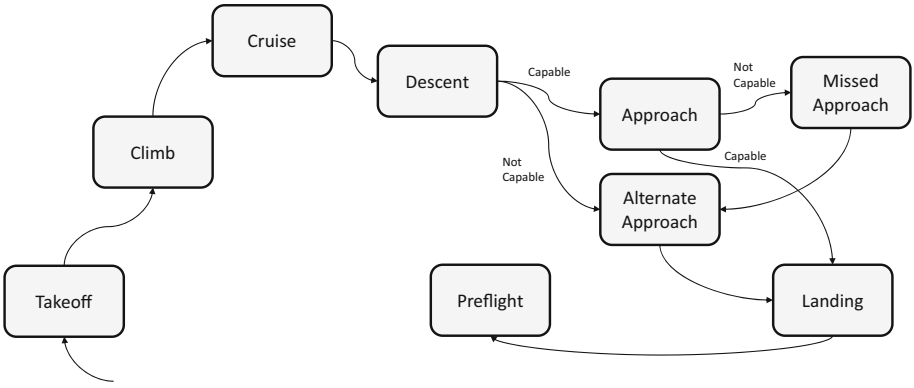

Fig. 4. Flight phases state transitions.

```
// Typical flight duration (in hours)
const double flight_duration = 1.5;

// Mean times spent on each flight phase (in hours)
const double time_in_takeoff = 0.01 * flight_duration;
const double time_in_climb = 0.15 * flight_duration;
const double time_in_cruise = 0.57 * flight_duration;
const double time_in_descent = 0.11 * flight_duration;
const double time_in_approach = 0.15 * flight_duration;
const double time_in_landing = 0.01 * flight_duration;
const double time_in_missed_approach = time_in_climb;

module flight_phase
  // Phases of flight: 0 = pre-flight; 1 = takeoff;
  // 2 = climb; 3 = cruise; 4 = descent;
  // 5 = approach; 6 = alternate approach;
  // 7 = landing; 8 = missed approach
  phase: [0..8] init 1;

  // Takeoff to climb
  [] (phase=1) -> 1/time_in_takeoff: (phase'=2);
  // Climb to cruise
  [] (phase=2) -> 1/time_in_climb: (phase'=3);
  // Cruise to descent
  [] (phase=3) -> 1/time_in_cruise: (phase'=4);
  // Descent to approach / alternate approach
  [] (phase=4) & (!loss_of_navigation) -> 1/time_in_descent: (phase'=5);
  [] (phase=4) & (loss_of_navigation) -> 1/time_in_descent: (phase'=6);
  // Approach to landing / missed approach
  [] (phase=5) & (!loss_of_navigation) -> 1/time_in_approach: (phase'=7);
  [] (phase=5) & (loss_of_navigation) -> 1E5: (phase'=8);
  // Alternate approach to landing
  [] (phase=6) -> 1/time_in_approach: (phase'=7);
  // Missed approach to alternate approach
  [] (phase=8) -> 1/time_in_missed_approach: (phase'=6);
  // Landing to pre-flight
  [] (phase=7) -> 1/time_in_landing: (phase'=0);
endmodule
```

Listing 1.2. PRISM code implementing the flight phase state transitions.

The initial state is takeoff which is followed by climb, cruise, and descent. Pilots may then choose to either continue with the planned approach or switch to an alternate approach (a suitable alternative is always assumed to exist), depending on the equipment failure states, which will determine the aircraft capability to continue with the initially planned approach procedure.

If the aircraft is not capable before starting the approach, the pilots will divert to the alternate approach as soon as the descent is finished. However, if the aircraft becomes incapable during the approach, it will go into a missed approach (go around) maneuver, which is then followed by the alternate approach. After the approach (or alternate approach), the aircraft will go into landing and finally will move to the pre-flight phase (parked, load/unload, towing, etc.), at which point it will no longer be in flight.

Four different CTMC models were created, each one related to a different operation. In Table 4, we show the characteristics of the models. Note that the CTMC model for the LNAV/VNAV operation has around 4.3 million of reachable states and about 60 million of transitions, showing the degree of detail we embedded in our solution.

Table 4. Characteristics of the CTMC models.

Operation	# Reachable states	# Transitions
LNAV/RNP 0.3	159,787	1,719,943
LNAV/VNAV	4,331,609	59,590,374
LP/LPV	16,771	146,329
GLS	158,011	1,692,511

3.5 Safety Properties

The definition of capable depends on the type of operation being performed (different types of approach procedures will require different systems to be operational). In PRISM code this is represented by the " loss_of_navigation " formula. The "misleading_information" formula will determine if the systems indications are erroneous to a point where the pilots could be mislead, which may be a more severe failure condition (refer to Table 2).

One architectural assumption is that each instance of the triple redundant systems (IRS and ADS) may be accessed by either FMS (i.e. FMS 1 and FMS 2 may use IRS 1, IRS 2, and IRS 3). The other systems (GNSS, GLS, DME) are primarily used by the FMS on the same side (i.e. FMS 1 uses GNSS 1, FMS 2 uses GNSS 2). The display unit presenting the Primary Flight Display (PFD) 1 is fed by FMS 1 and the PFD 2 is fed by FMS 2.

The "Enroute/Terminal Area/Non-precision Approach (LNAV or RNP 0.3) operation is assumed to be able to be performed with a single navigation source (valid information in a single PFD). This is expected to be viable since this operation has at most a "Major" hazard classification as per Table 2. The capability formulas for this operation mode can be seen in Listing 1.3. Note that the FMS uses the onside GNSS, onside DME or any IRS to compute the lateral guidance.

```
formula enroute_terminal_approach = (phase >= 2 & phase <=5);

// Loss of navigation
formula all_irs_fail = (irs1_fail) & (irs2_fail) & (irs3_fail);
formula pfd1_not_av = (fms1_fail) | (gnss1_fail & dme1_fail & all_irs_fail);
formula pfd2_not_av = (fms2_fail) | (gnss2_fail & dme2_fail & all_irs_fail);
formula loss_of_navigation = (pfd1_not_av) | (pfd2_not_av);

// Misleading Information
formula all_irs_err = (irs1_err) & (irs2_err) & (irs3_err);
formula pfd1_err = (gnss1_err | dme1_err | all_irs_err) | (fms1_err);
formula pfd2_err = (gnss2_err | dme2_err | all_irs_err) | (fms2_err);
formula misleading_information = (pfd1_err) | (pfd2_err);
```

Listing 1.3. PRISM code implementing the Enroute/Terminal Area/Non-precision Approach (LNAV or RNP 0.3) capability formulas.

CSL representation of the "loss of navigation" safety property for the LNAV/RNP 0.3 mode can be seen in Eq. 2, while the "misleading information" property is seen in Eq. 3:

$$\mathcal{P}_{<10^{-5}\times T}[\Diamond(\text{enroute_terminal_approach} \wedge \text{loss_of_navigation})] \quad (2)$$

$$\mathcal{P}_{<10^{-5}\times T}[\Diamond(\text{enroute_terminal_approach} \wedge \text{misleading_information})] \quad (3)$$

Notice that in Eq. 2 and Eq. 3 there is a factor T which stands for the average flight time. It is included since the safety objectives are given as a probability per flight hour. Note that one can check this flight time by calculating it via cost/reward in PRISM, as shown in Listing 1.4, and via a cost/reward-based property, as shown in Eq. 4.

```
rewards "flight_hours"
    phase > 0 : 1;
endrewards
```

Listing 1.4. PRISM code implementing the flight time cost/reward.

$$T \coloneqq R\{\text{“flight_hours”}\} =?[C] \quad (4)$$

The “Non-precision Approach with Vertical Guidance (LNAV/VNAV)” operation is also assumed to be able to be performed with a single navigation source. CSL representation of the “loss of navigation” safety property for the LNAV/VNAV mode can be seen in Eq. 5, while the “misleading information” property is seen in Eq. 6:

$$\mathcal{P}_{<10^{-5}\times T}[\Diamond(\text{phase} = 8)] \quad (5)$$

$$\mathcal{P}_{<10^{-5}\times T}[\Diamond(\text{phase} = 5 \wedge \text{misleading_information})] \quad (6)$$

The “LP/LPV Approach” operation is assumed to require independent navigation sources in both PFD 1 and PFD 2. This was a necessity since this approach type has lower minimum altitudes and as such has a more severe misleading information hazard classification (“Hazardous” instead of “Major”). In this mode, the FMS only requires the GNSS (to calculate the current position and trajectory) and the IRS (to calculate the attitude). The capability formulas are given in Listing 1.5.

```
// Loss of navigation
formula all_irs_fail = (irs1_fail) & (irs2_fail) & (irs3_fail);
formula pfd1_not_av = (gnss1_fail) | (fms1_fail) | (all_irs_fail);
formula pfd2_not_av = (gnss2_fail) | (fms2_fail) | (all_irs_fail);
formula loss_of_navigation = (pfd1_not_av) & (pfd2_not_av);

// Misleading Information
formula all_irs_err = (irs1_err) & (irs2_err) & (irs3_err);
formula pfd1_err = (gnss1_err) | (fms1_err) | (all_irs_err);
formula pfd2_err = (gnss2_err) | (fms2_err) | (all_irs_err);
formula misleading_information = (pfd1_err) & (pfd2_err);
```

Listing 1.5. PRISM code implementing the LP/LPV Approach capability formulas.

CSL representation of the “loss of navigation” safety property for the LP/LPV Approach mode can be seen in Eq. 7, while the “misleading information” property is seen in Eq. 8:

$$\mathcal{P}_{<10^{-5}\times T}[\Diamond(\text{phase} = 8)] \quad (7)$$

$$\mathcal{P}_{<10^{-7}\times T}[\Diamond(\text{phase} = 5 \wedge \text{misleading_information})] \tag{8}$$

Similarly, the "GLS Approach" operation is assumed to require independent navigation sources in the PFDs. In this mode, the FMS requires the onside GNSS and the onside GLS (to calculate the current position and trajectory) and any IRS (to calculate attitude). CSL formulation of the "loss of navigation" safety property can be seen in Eq. 9, while the "misleading information" property is seen in Eq. 10:

$$\mathcal{P}_{<10^{-5}\times T}[\Diamond(\text{phase} = 8)] \tag{9}$$

$$\mathcal{P}_{<10^{-7}\times T}[\Diamond(\text{phase} = 5 \wedge \text{misleading_information})] \tag{10}$$

4 Results and Analysis

In order to demonstrate compliance with the safety properties defined for each failure mode, for each type of operation, the properties were evaluated in PRISM and the results are shown in Table 5. The "Objective" column shows the maximum acceptable probability limit and it is obtained by the probability per flight hour (as defined in Table 1) multiplied by the average flight time $T = 1.5$ h.

The "Probability" column shows the probability for each property as calculated by PRISM. These values should be always lower than the objective in order to be compliant.

Table 5. Safety properties results as evaluated by PRISM.

Operation	Failure mode	Classification	Objective	Probability	Compliant
LNAV/RNP 0.3	Loss of Navigation	Major	1.5E-5	2.97E-6	✓
	Misleading Information	Major	1.5E-5	6.24E-6	✓
LNAV/VNAV	Loss of Navigation	Major	1.5E-5	4.50E-7	✓
	Misleading Information	Major	1.5E-5	6.24E-6	✓
LP/LPV	Loss of Navigation	Major	1.5E-5	6.82E-9	✓
	Misleading Information	Hazardous	1.5E-7	3.71E-12	✓
GLS	Loss of Navigation	Major	1.5E-5	8.23E-9	✓
	Misleading Information	Hazardous	1.5E-7	4.41E-12	✓

As shown in Table 5, all safety objectives were met. Note that the compliance is heavily reliant on the failure rates assumed for each system, and during development of the aircraft these values should be reassessed as changes are made to the components.

We went further in our experiments and decided to increase the flight time, T, in order to realise whether the compliance was still achieved. Figure 5 shows the results for the Loss of Navigation failure mode and considering the LNAV/RNP 0.3 operation with $T = 18$ h, which is currently considered the longest flight in the world [4]. Note that now the "Objective" probability is 18E-5 but, again, the final probability is still lower than the "Objective" and hence the compliance is still met. Due to space constraints, we omit other experiments that we have performed with different values of flight time, but the results were similar.

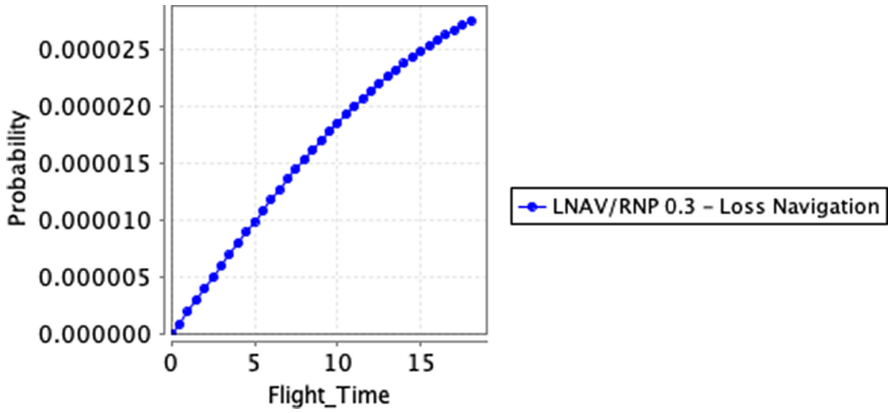

Fig. 5. LNAV/RNP 0.3: Loss of Navigation, $T = 18$ h.

5 Related Work

In this section, we present some relevant studies related to our research. Gomes et al. [9] presented an approach for systematic safety assessment using Probabilistic Model Checking. CTMC models are constructed automatically from Simulink (block) diagrams and a set of tabular data about the underlying system failure modes and hazard analysis. Their safety objectives are defined as steady-state properties, as opposed to transient, which is only viable in models with infinite paths (no terminal state). In these infinite path models, failures are introduced and must later be repaired, adding additional assumptions about the repair strategy and latent failures.

In [10], the authors show how functional models used during design or for qualitative safety assessment may be directly reused for a quantitative analysis. A simple case study is provided, using the PRISM Probabilistic Model Checker.

One major difference between our work and all these previous studies is the detail of the CTMC models. Even though we derived the models manually, the level of details of our models are considerably higher, and hence we believe that the results we obtained are more realistic.

6 Conclusions

The safety assessment process is of extreme importance to the aerospace industry, due to its safety-critical nature. This process, especially for very large and complex systems, is laborious, time-consuming, and expensive. Therefore the industry has much to gain by optimising it, for instance, by adopting formal and possibly more automated model-based methods.

In this article, we wondered whether Formal Verification methods can contribute to the process of assessing the safety of aircraft navigation systems, providing an alternative to complement the study carried out using classical but

non-formal methods. By relying on Probabilistic Model Checking (CTMC models and safety properties formalised in CSL), we could validate this hypothesis with consistent results and dealing with models with up to 4.3 million of reachable states and 60 million of transitions. CTMC models can represent arbitrary asynchronous stochastic transition systems, which are very flexible (for instance, allowing for the modelling of flight phases, pilot decisions as well as multiple equipment failure modes). A Probabilistic Model Checker such as PRISM allows for the Formal Verification of the system's safety properties, which can be performed iteratively from early stages of development when the impacts of design changes are smaller.

This article does not feature a complete assessment of all aircraft functions due to the high complexity of this type of modelling. One future direction is trying to integrate classical methods, such as FTA, FMEA, with Probabilistic Model Checking where the latter participates in part of the process. Other direction is trying to address the issue of scalability related to Probabilistic Model Checking via compositional techniques.

References

1. Baier, C., Katoen, J.P.: Principles of Model Checking, p. 975. The MIT Press, Cambridge (2008)
2. Boeing: Statistical summary of commercial jet airplane accidents - Worldwide operations, 1959–2019 (2020). https://bit.ly/3cGqgfb. Accessed 14 June 2021
3. Clarke, E.M., Emerson, E.A., Sistla, A.P.: Automatic verification of finite-state concurrent systems using temporal logic specifications. ACM Trans. Program. Lang. Syst. **8**(2), 244–263 (1986)
4. CNN: The world's longest flight is back - and now it's even longer (2020). https://cnn.it/3sXBnXD. Accessed 14 June 2021
5. Dannenberg, F., Kwiatkowska, M., Thachuk, C., Turberfield, A.J.: DNA walker circuits: computational potential, design, and verification. In: Soloveichik, D., Yurke, B. (eds.) DNA 2013. LNCS, vol. 8141, pp. 31–45. Springer, Cham (2013). https://doi.org/10.1007/978-3-319-01928-4_3
6. Dioto, M., Eras, E.R., de Santiago Júnior, V.A.: On the feasibility of probabilistic model checking to analyze battery sustained power supply systems. In: Misra, S., Gervasi, O., Murgante, B., Stankova, E., Korkhov, V., Torre, C., Rocha, A.M.A.C., Taniar, D., Apduhan, B.O., Tarantino, E. (eds.) ICCSA 2019. LNCS, vol. 11620, pp. 743–757. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-24296-1_59
7. FAA: AC 25.1309-1B - System Design and Analysis. Federal Aviation Administration (FAA) (2002)
8. FAA: AC 20–138D - Airworthiness Approval of Positioning and Navigation Systems. Federal Aviation Administration (FAA) (2014)
9. Gomes, A., Mota, A., Sampaio, A., Ferri, F., Buzzi, J.: Systematic model-based safety assessment via probabilistic model checking. In: Margaria, T., Steffen, B. (eds.) Leveraging Applications of Formal Methods, Verification, and Validation, pp. 625–639. Springer, Heidelberg (2010)
10. Güdemann, M., Ortmeier, F.: Probabilistic model-based safety analysis. Electron. Proc. Theoretical Comput. Sci. **28**, 114–128 (Jun 2010)

11. Konur, S., Fisher, M.: Formal analysis of a vanet congestion control protocol through probabilistic verification. In: 2011 IEEE 73rd Vehicular Technology Conference (VTC Spring), pp. 1–5 (2011). https://doi.org/10.1109/VETECS.2011.5956327
12. Kwiatkowska, M., Norman, G., Parker, D.: Stochastic model checking. In: Bernardo, M., Hillston, J. (eds.) SFM 2007. LNCS, vol. 4486, pp. 220–270. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-72522-0_6
13. Kwiatkowska, M., Norman, G., Parker, D.: PRISM 4.0: verification of probabilistic real-time systems. In: Gopalakrishnan, G., Qadeer, S. (eds.) CAV 2011. LNCS, vol. 6806, pp. 585–591. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-22110-1_47
14. Pasa, G.D., Santiago Júnior, V.A.: PRISM models and properties: Aircraft navigation systems safety assessment via probabilistic model checking (2021). https://bit.ly/3gsWons. Accessed 14 June 2021
15. SAE: ARP4761 - Guidelines and Methods for Conducting the Safety Assessment Process on Civil Airborne Systems and Equipment. SAE International (1996)
16. SAE: ARP4754A - Guidelines for Development of Civil Aircraft and Systems. SAE International (2010)
17. de Santiago Júnior, V.A., Tahar, S.: Time performance formal evaluation of complex systems. In: Cornélio, M., Roscoe, B. (eds.) SBMF 2015. LNCS, vol. 9526, pp. 162–177. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-29473-5_10