The landing gear case study: challenges and experiments

Frédéric Boniol[1] · **Virginie Wiels**[1] · **Yamine Aït-Ameur**[2] · **Klaus-Dieter Schewe**[3]

DOI 10.1007/s10009-016-0431-4

1 Introduction

Embedded critical systems need to be validated very thoroughly; it usually results in very long and onerous test phases. Formal techniques, in particular formal specification lan-guages and associated proof tools, could be an advantageous alternative, or at least a good complement and allow a signif-icant reduction of test phases. However, for these techniques to be used in practice, one issue to consider is their efficiency and scalability on complex industrial systems.

Case studies have played an essential role in the history of formal methods. They have allowed to illustrate the application of formal techniques for modelling and verification, to compare different methods in terms of expressivity, performance and easiness of use. They have also permitted to enact the progress made by these methods.

Dagstuhl seminar 9523 is about the famous Steam Boiler case study in 1995 had a lot of impact on the formal methods community. This case study allowed the assessment of formal techniques, the comparison of different formal techniques, the identification of areas for future work [2,3].

As formal methods have made a lot of progress since 1995, ABZ'2014 has proposed a complex case study, representative of industrial needs. The proposed case study is a landing gear control system. It is composed of three parts: a pilot interface, a mechanical and hydraulic parts, and a digital part. This system is a representative of critical embedded systems. The action to be done at each time depends on the state of all the physical devices composing the system and on their temporal behaviour. When considering such systems, the challenge is first to model and to program the software part controlling the landing and the retraction sequence, and second to prove safety requirements taking into account the physical behaviour of hydraulic devices.

The case study attracted a lot of interest. 11 selected papers were presented at the ABZ'2014 conference and published in [7]. They used different formal techniques: Event-B [1], ASM [8], Fiacre [10]. They also proposed different kinds of verification: proof, model checking, test generation, run-time monitoring, and simulation. One of the main conclusions of the ABZ'2014 case study track was that formal methods are now powerful enough to model and to verify complex case studies as the landing gear system. The associated tools proved powerful enough to support such modelling and verification activities. Such an assessment is good news both for industrial engineers and for the academic community. However, a lot of difficulties remain.

This special issue presents an extended version of six of the models that were presented at the ABZ'2014 conference.

The next section gives an overview of the case study. For a more detailed presentation, the reader should refer to [6]. Section 3 discusses the main challenges of this case study. Section 4 introduces then the six experiments presented in

✉ Virginie Wiels
virginie.wiels@onera.fr

Frédéric Boniol
frederic.boniol@onera.fr

Yamine Aït-Ameur
yamine@enseeiht.fr

Klaus-Dieter Schewe
klaus-dieter.schewe@scch.at

1 ONERA, 2 avenue E. Belin, BP 74025, 31055 Toulouse, France

2 IRIT-ENSEEIHT, 2 rue C. Camichel, BP 7122, 31071 Toulouse, France

3 Software Competence Center Hagenberg GmbH, Softwarepark 21, 4232 Hagenberg, Austria

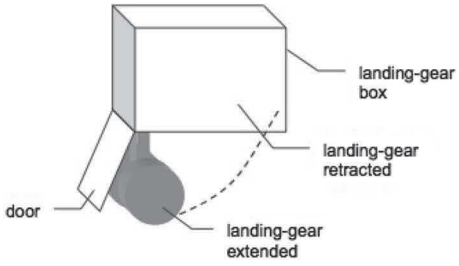

Fig. 1 Landing set

this special issue. The last section gives few words of conclusion.

2 The landing gear case study: brief overview

The landing system is in charge of manoeuvring landing gears and associated doors. It is composed of 3 landing sets: front, left and right. Each landing set contains a door, a landing-gear and associated hydraulic cylinders. A simplified schema of a landing set is presented in Fig. 1.

The system is controlled digitally. From a high level point of view, a basic landing sequence are: (1) open the doors of the landing gear boxes, (2) extend the landing gears and (3) close the doors. Similarly, after taking off, the corresponding basic retraction sequence to be performed are: (1) open the doors, (2) retract the landing gears and (3) close the doors. From a more concrete point of view, these two basic sequences can be interleaved in an intricate way: the pilot can interrupt each sequence at any time and at any point to start the opposite sequence as often as he/she wishes, leading to an infinite number of possible scenarios. This is one source of complexity of the system. The second source of complexity is failure management.

2.1 Architecture of the system

The landing gear system is composed of three parts: (1) a mechanical part which contains all the mechanical devices and the three landing sets, (2) a digital part including the control software, (3) and a pilot interface.

The pilot interface To command the retraction and outgoing of gears, an Up/Down handle is provided to the pilot. When the handle is switched to "Up" the retracting landing gear sequence is executed, when the handle is switched to "Down" the extending landing gear sequence is executed.

Three lights inform the pilot of the current position of the gears and the doors, and of the current health state of the system and its equipments: (1) one green light "gears are locked down", (2) one orange light "gears manoeuvring", (3) one red light "landing gear system failure". No light is on when the gears are locked up and when no failure has been observed.

The mechanical and hydraulic parts The architecture of the hydraulic part is described in Fig. 2. Each landing set contains three latching boxes: one for locking the gear in the up position, when the gear is retracted, a second one for locking the gear in the down position, when the gear is extended, and a third one for locking the door in the closed position. Note that there is no latching box for the open position, meaning that the door is not mechanically locked when it is open.

The landing gears and doors motion is performed by a set of actuating cylinders:

- For each door, a cylinder opens and closes the door.
- For each landing gear, a cylinder retracts and extends the landing gear.

Hydraulic power is provided to the cylinders from an external hydraulic circuit through a set of electro-valves:

- One general electro-valve to supply the specific electro-valves with hydraulic power from the aircraft hydraulic circuit.
- Four electro-valves to set pressure, respectively, on the portion of the hydraulic circuit related to
  - door opening,
  - door closing,
  - landing gear extending,
  - landing gear retracting.

Each electro-valve is activated by an electrical order coming from the digital part. In the specific case of the general electro-valve, this electrical order goes through an analogical switch to prevent abnormal behaviour of the digital part (e.g. abnormal activation of the general electro-valve).

Note that the three doors (resp. gears) are controlled simultaneously by the same electro-valve. It is thus not possible to control the doors (resp. gears) separately.

A set of discrete sensors inform the digital part about the state of the equipments:

- Front/right/left gear is locked/not locked in the extended position.
- Front/right/left gear is locked/not locked in the retracted position.
- Front/right/left gear shock absorber is on ground/in flight.
- Front/right/left door is open/not open.
- Front/right/left door is locked/not locked in the closed position.
- Hydraulic circuit (after the general electro-valve) is pressurised/not pressurised.
- The analogical switch between the digital part and the general electro-valve is closed/open.

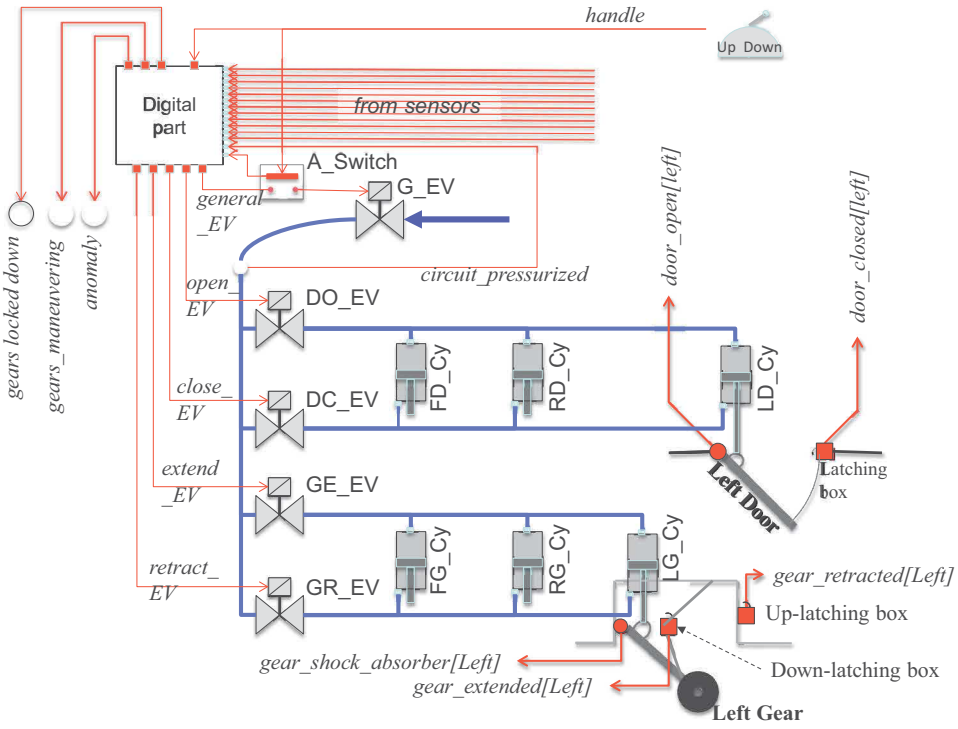

Fig. 2 Architecture of the hydraulic part

To mitigate sensor failures, each sensor is triplicated. It delivers simultaneously three discrete values describing the same situation.

The digital part The digital part is composed of two identical computing modules (see Fig. 3). Each one executes in parallel the same control software. This software is in charge of controlling gears and doors, detecting anomalies, and informing the pilot about the global state of the system and anomalies (if any). It is part of a retroaction loop with the physical system, and produces commands for the distribution elements of the hydraulic system with respect to the sensors values and the pilot orders. The two computing modules receive the same input (sensor values and pilot orders).

From these input, each module computes five electrical orders (one for each electro-valve). These corresponding electrical orders outgoing from the two modules are physically produced on the same electrical line. The implicit composition of two output is an electrical "OR" as shown in Fig. 3. As a consequence, if the two different computing modules send two different values (*true* and *false*) on the same line (for instance in case of failure of one of the two computing modules), then only the *true* value is transmitted to the corresponding electro-valve.

Similarly the two modules produce global boolean state variables to the cockpit (one for each cockpit light). These output are synthesised by each module from sensors data and from the situation awareness. Similarly to electrical orders provided to the electro-valves, the boolean state variables

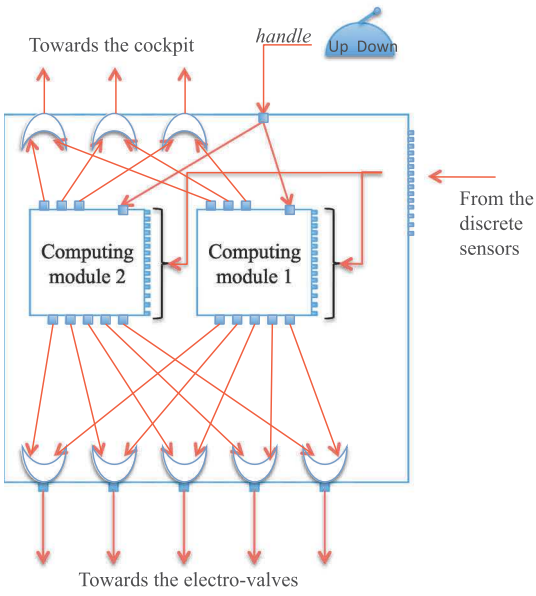

Fig. 3 Digital architecture

from the two modules are composed following a logical "OR" operation.

2.2 Mechanical and hydraulic equipment

The analogical switch (between the digital part and the general electro-valve) The aim of this switch is to protect the system against abnormal behaviour of the digital part.

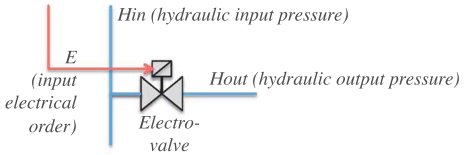

Fig. 4 An electro-valve equipment

To prevent inadvertent order to the electro-valves, the general electro-valve can be stimulated only if this switch is closed. The switch is closed each time the "Up/Down" handle is moved by the pilot, and it remains closed for 20 s. After this duration, the switch automatically becomes open. Because of inertial reasons, the transition from the two states closed and open takes a given amount of time: (1) 0.8 s from open to closed, and (2) 1.2 s from closed to open. In the closed position, the switch transmits the electrical order from the digital part to the general electro-valve. In the open position, no electrical order is sent to the electro-valve. In that case, the oil pressure in the hydraulic circuit becomes down.

In addition to this normal behaviour, the analogical switch can fail at any time. However, the most likely failures to consider are permanent failures: the switch remains blocked in the closed or in the open position.

Electro-valves All the electro-valves are supposed to have the same behaviour. As shown in Fig. 4, an electro-valve is an hydraulic equipment with two hydraulic ports *Hin* and *Hout*, and an electrical port E with the following behaviour:

- if $E = false$ (the voltage of the electrical order is down), then $Hout = 0$ (no pressure on the hydraulic output side, the hydraulic circuit is open);
- if $E = true$ (the voltage of the electrical order is high), then $Hout = Hin$ (the hydraulic circuit is closed).

Note that the electrical order must be sustained to *true* (i.e, at the high voltage) to maintain the electro-valve in the closed position. The electrical order is not a discrete event, but can be seen as an analogical signal.

Because of inertial reasons, when E rises from *false* to *true* (i.e., the electro-valve switches from open to closed), the pressure grows up continuously from 0 to *Hin*. One can suppose that the rise in pressure approximatively follows a linear law, and that the total duration of the transition phase is 1 s. In the same way, when E falls to *false*, the pressure goes down linearly from *Hin* to 0. The total duration of the pressure drop is 3.6 s.

Cylinders Cylinders are pure hydraulic equipments. As shown in Fig. 5, they begin to move when they receive hydraulic pressure, and they stop to move when the pressure goes down or when they reach the end of their race.

Gear cylinders are locked in high or down position by means of a latching box mechanism (the latching boxes are physically on the gears, one for each position). When a gear cylinder is locked in high (resp. down) position and when it receives pressure from the high (resp. down) hydraulic circuit, first it is unlocked from the high (resp. down) position, then it moves to the down (resp. high) position, and finally it is locked in the down (resp. high) position.

Door cylinders are locked (by means of two latching boxes on each door) only in closed position. Doors remain open by maintaining pressure in extension circuit. When a door cylinder is locked in closed position and when it receives pressure from the extension hydraulic circuit, first it is unlocked from the closed position, then it moves to the open position, and finally it is maintained in the open position as long as the pressure is maintained in the hydraulic extension circuit.

All these operations are done automatically with the hydraulic pressure only. No electrical part is involved in cylinders. These operations take a certain amount of time, depending on the position of the cylinder in the aircraft and in the hydraulic circuit. The durations are given in Table1. The values are only mean values. The true durations can vary around these values up to 20 %.

Note that it is possible to stop and to inverse the motion of any cylinder at any time.

2.3 Software specification

The aim of the software part of the system is twofold: (1) to control the hydraulic devices according to the pilot orders and to the mechanical devices positions; (2) to monitor the system and to inform the pilot in case of anomaly.

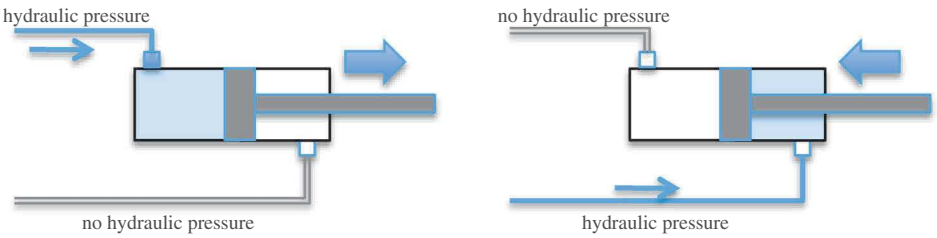

Fig. 5 Extension and retraction of a cylinder

Table 1 Durations of the operations

Duration (in s) of …	Front gear	Front door	Right/left gear	Right/left door
Unlock in down position	0.8	–	0.8	–
From down to high position	1.6	1.2	2	1.6
Lock in high position	0.4	0.3	0.4	0.3
Unlock in high position	0.8	0.4	0.8	0.4
From high to down position	1.2	1.2	1.6	1.5
Lock in down position	0.4	–	0.4	–

Expected scenarios in normal mode When the command line is working (in normal mode), the landing system reacts to the pilot orders by actioning or inhibiting the electro-valves of the appropriate cylinders. Two basic scenarios are considered: the outgoing sequence, and the retraction sequence.

Outgoing sequence The outgoing of gears is decomposed in a sequence of elementary actions. When the gears are locked in retracted position, and the doors are locked in closed position, if the pilot sets the handle to "Down", then the software should have the following sequence of actions:

1. stimulate the general electro-valve isolating the command unit to send hydraulic pressure to the manoeuvring electro-valves,
2. stimulate the door opening electro-valve,
3. once the three doors are in the open position, stimulate the gear outgoing electro-valve,
4. once the three gears are locked down, stop the stimulation of the gear outgoing electro-valve,
5. stop the stimulation of the door opening electro-valve,
6. stimulate the door closure electro-valve,
7. once the three doors are locked in the closed position, stop the stimulation of the door closure electro-valve,
8. and finally stop stimulating the general electro-valve.

Retraction sequence In the same way, the retraction of gears is decomposed in a sequence of elementary actions. When the gears are locked in down position, and the doors are locked in closed position, if the pilot sets the handle to "Up", then the software should have the following sequence of actions:

1. stimulate the general electro-valve isolating the command unit, to send hydraulic pressure to the manoeuvring electro-valves,
2. stimulate the door opening electro-valve,
3. once the three doors are in the open position, if the three shock absorbers are relaxed, then stimulate the gear retraction electro-valve and go to step 4, else (if one of the three shock absorbers is not relaxed) go to step 5,
4. once the three gears are locked up, stop the stimulation of the gear retraction electro-valve,
5. stop the stimulation of the door opening electro-valve,
6. stimulate the door closure electro-valve,
7. once the three doors are locked in the closed position, stop the stimulation of the door closure electro-valve,
8. and finally stop stimulating the general electro-valve.

The previous sequences can be interrupted by counter orders (e.g., a retraction order during the let down sequence) at any time. In that case, the scenario stops and restarts in the counter-sequence from the point where it was interrupted. For instance, if an outgoing sequence is interrupted in the door closure phase (step 6 of the outgoing sequence) by an "Up" order, then the stimulation of the door closure electro-valve is stopped, and the retraction sequence is executed from step 2: the door opening electro-valve is stimulated and the doors begin opening again. Afterwards, the scenario continues up to the final step or to a new interruption.

Timing constraints Because of inertia of the oil pressure and to prevent shock waves in the hydraulic circuit, the outgoing and retraction sequences have to meet three timing constraints. First, stimulations of the general electro-valve and the manoeuvring electro-valve must be separated by at least 200 ms. Second, orders to stop the stimulation of the general electro-valve and the manoeuvring electro-valve must be separated by at least 1s. And third, two contrary orders (closure vs. opening doors, extension vs. retraction gears) must be separated by at least 100 ms.

Health monitoring The second objective of the control software is to detect anomalies and to inform the pilot. Anomalies are caused by failures on hydraulic equipment, electrical components, or computing modules. Whenever an anomaly is detected, the system is globally considered as invalid. An anomaly signal is sent to the pilot interface. The effect of this action is to put the red light "landing gear system failure" on.

2.4 Failures

Potential failures Regarding the mechanical and hydraulic equipment, the most likely failures to consider are permanent failures:

– each electro-valve can fail and end up blocked either in the closed or in the open state;
– each cylinder can fail and be blocked in its last position (down, high, or any where between these two positions);
– similarly to electro-valves, the analogical switch is a mechanical device which can fail and be blocked either in the closed or in the open state;
– finally an hydraulic leak can happen anywhere in the circuit resulting in permanent pressure drop.

Regarding the sensors, let us recall that each sensor is composed of three redundant single discrete sensors. Each of them can fail in two different ways: (1) the single sensor is permanently blocked on one of its two values (*true* or *false*); or (2) it behaves erroneously varying randomly between *true* and *false*.

Finally, regarding the computing modules, each of them can fail in two different ways: (1) permanent failure, in that case the module does not send any order to the electro-valves and to the lights; (2) the module behaves erroneously by sending random values to the actuators.

All the faults are supposed independent.

Note that we do not consider total loss of the electrical power resulting in the simultaneous loss of the two computing modules. Similarly, we do not consider failures of the lights.

Probabilities For the sake of simplicity, all the failures are supposed to follow exponential distributions:

– $\lambda_{\text{mh}} = 10^{-3}$ for the mechanical and hydraulic equipment (including the electro-valves, the cylinders and the analogical switch), meaning that the equipment will probably fail around 10^3 flight hours;
– $\lambda_{\text{s}} = 10^{-3}$ for each single sensor;
– and $\lambda_{\text{cm}} = 10^{-4}$ for each computing module.

Because of the independent hypothesis between faults, the probability of having n failures at the same time is the product of the probability of each fault. For instance, the probability for losing the two computing modules is 10^{-8}; the probability for losing two single sensors of the same sensor if 10^{-6}, etc.

2.5 Requirements

The landing system is a critical system. It has to meet a set of safety requirements. These requirements are divided into two parts: *normal mode* requirements, and *failure mode* requirements.

By *normal mode*, we mean any scenario involving no failure. By *failure mode* we mean any scenario involving combinations of failures with probability greater than 10^{-7}.

Normal mode requirements

– (R_{11} (resp. R_{12})) When the command line is working, if the landing gear command handle has been pushed DOWN (resp. UP) and stays DOWN (resp. UP), then the gears will be locked down (resp. retracted) and the doors will be seen closed less than 15 s after the handle has been pushed;
– (R_{21} (resp. R_{22})) When the command line is working, if the landing gear command handle remains in the DOWN (resp. UP) position, then retraction (resp. outgoing) sequence is not observed.
– (R_{31}) When the command line is working, the stimulation of the gears outgoing or the retraction electro-valves can only happen when the three doors are locked open.
– (R_{32}) When the command line is working, the stimulation of the doors opening or closure electro-valves can only happen when the three gears are locked down or up.
– (R_{41} (resp. R_{42})) When the command line is working, opening and closing doors electro-valves (resp. outgoing and retraction gears electro-valves) are not stimulated simultaneously.
– (R_{51})When the command line is working, it is not possible to stimulate the manoeuvring electro-valve (opening, closure, outgoing or retraction) without stimulating the general electro-valve.

Failure mode requirements

– (R_{61} (resp. R_{62})) If one of the three doors is still seen locked in the closed (resp. open) position more than 7 s after stimulating the opening (resp. closure) electro-valve, then the red light “landing gear system failure” is on.
– (R_{63} (resp. R_{64})) If one of the three gears is still seen locked in the down (resp. up) position more than 7 s after stimulating the retraction (resp. outgoing) electro-valve, then the red light “landing gear system failure” is on.
– (R_{71} (resp. R_{72})) If one of the three doors is not seen locked in the open (resp. closed) position more than 7 s after stimulating the opening (resp. closure) electro-valve, then the red light “landing gear system failure” is on.
– (R_{73} (resp. R_{74})) If one of the three gears is not seen locked in the up (resp. down) position more than 10 s after stimulating the retraction (resp. outgoing) electro-valve, then the red light “landing gear system failure” is on.
– (R_{81} (resp. R_{82})) When at least one computing module is working, if the landing gear command handle has been DOWN (resp. UP) for 15 s, and if the gears are not locked down (resp. retracted) after 15 s, then the red light “landing gear system failure” is on.

3 Main challenges

The landing gear case system is a critical cyber physical systems which offers several challenges for formal modelling and verification methods.

Size of the state space A first (classical) challenge is the combinatorial explosion. The system involves more than 70 components (from sensors to computing units). All these components evolve in parallel in an asynchronous way. Each component has about four behavioural modes (including functional and dysfunctional modes). The global behaviour of the system is the result of interleaving the behaviour of all the components, which leads to a huge state space. For instance, let us consider a redundant sensor. This sensor is composed of three single sensors. These three single sensors are asynchronous. They have their own inertia and they can change (from *true* to *false* for instance) with a small delay. The number of potential states of the redundant sensor is then 2^3. There are 18 redundant sensors in the system (3 per gear, 2 per door, 1 for the hydraulic circuit, 1 for the analogical switch, and 1 for the handle). The number of potential states of all the sensors is then 8^{18}. Adding the behaviour of the mechanical components (electro-valves, cylinders, analogical switch and handle), the whole state space contains more than 2^{70} states. Adding now the potential failures, the state space including dysfunctional scenarios goes over 2^{100} states. The first challenge is to overcome this combinatorial explosion: how to explore such a huge state space, or how to abstract it in a safe way according to requirements to be verified.

Handling failure modes The second challenge is the handling of failure modes. As said above, the landing system involves a great number of digital, mechanical and hydraulic components. Each component can fail in several ways following several probabilistic distributions. As explained previously, requirements to be verified on the system mix functional and dysfunctional behaviours in an intricate way. The verification of the system requires an interleaving of functional and probabilistic dysfunctional modelling.

Handling time A third challenge is about time. 12 of the 19 requirements of the case study involve time. Two of them (R_{11} and R_{12}) relate to end-to-end latencies, from the pilot's handle to the lights which notify the situation to the crew through all the digital, mechanical and hydraulic components. The other ones specify a maximal response time for the system to inform the pilot in case of anomalies. Verification of these requirements needs to take into account the real-time behaviour of each component in an accurate way. Knowing that time is another cause of combinatorial explosion, the third challenge is how to best model time.

4 Experiments

This special issue contains six different modelling and verification of the case study.

The first paper "Aircraft Landing Gear System: Approaches with Event-B to the Modelling of an Industrial System" by Wen Su and Jean-Raymond Abrial [13], proposes three Event-B modelling of the case study. Modelling a complex system is not an easy automatable task, the three proposed models follow three different approaches and uses different techniques for verification and validation (simulation, model checking, theorem proving, constraint solving). The paper brings valuable insights on modelling complex systems in general and modelling the landing gear system in particular.

An Event-B modelling and verification of the case study is also proposed by the second paper "Modelling a Landing Gear System in Event-B" by Amel Mammar and Régine Laleau [12]. The construction of the model is incremental with respect to the three challenges stated before: it first models the system without considering time and potential failures, then adds time and failures step by step.

The third paper "Validation of the ABZ Landing Gear System using ProB" by Dominik Hansen, Lukas Ladenberger, Harald Wiegard, Jens Bendisposto and Michael Leuschel [11], is another B modelling which puts a special emphasis on visualisation of the model of the landing gear system. Visualisation tools are used to provide different views of the system, the paper describes the help it provides for the development and validation of the model.

The fourth paper "The Landing Gear System in Multi-Machine Hybrid Event-B" by Richard Banach [5] proposes an hybrid modelling of the case study. Hybrid modelling is ideally suited for the landing gear system which combines mechanical, hydraulic and software parts. No verification tool is yet available but the hybrid modelling of the case study is a valuable help to understand how the continuous movement of the mechanical devices interferes with the digital components.

A model checking approach is proposed by the fifth paper "Environment-driven Reachability for Timed Systems" by Ciprian Teodorov et al. [9]. This paper addresses the state-space explosion issue using a context-aware verification approach. This approach brings a significant reduction of the state-space based on a specific handling of the environment of the system. The decomposition into different contexts allows an exploration of combinations of nominal behaviour and failures.

The sixth and last paper "Rigorous development process of a safety-critical system: from ASM models to Java code" by Paolo Arcaini, Angelo Gargantini and Elvinia Riccobene [4], proposes a refinement-based development of the landing gear system using Abstract State Machine. Each refinement step can be proved correct using SMT-based approach. A java

implementation is produced, its conformance with respect to the specification is checked using two approaches: model-based testing and runtime verification.

5 Conclusion

The case study track at ABZ'2014 was a very interesting and lively session. In this special issue, the authors have led further their modelling and verification of the landing gear system. The six papers present interesting aspects of the system and insights on formal modelling and verification approaches. We hope for many more work to come in the future on this case study, from the different existing formal communities.

References

1. Abrial, J.: Modeling in Event-B—system and software engineering. Cambridge University Press, Cambridge (2010)
2. Abrial, J.-R., Börger, E., Langmaack, H. (eds.): Formal Methods for Industrial Applications, Specifying and Programming the Steam Boiler Control (the book grow out of a Dagstuhl Seminar, June 1995). Lecture Notes in Computer Science, vol. 1165. Springer, London (1996)
3. Abrial, J., Börger, E., Langmaack, H.: The steam boiler control specification problem. http://www.informatik.uni-kiel.de/~procos/dag9523/dag9523.html (1996)
4. Arcaini, P., Gargantini, A., Riccobene, E.: Rigorous development process of a safety-critical system: from asm models to java code. Int. J. Softw. Tools Technol. Transf. doi:10.1007/s10009-015-0394-x (2016)
5. Banach, R.: The landing gear system in multi-machine hybrid event-b. International J. Softw. Tools Technol. Transf. doi:10.1007/s10009-015-0409-7 (2016)
6. Boniol, F., Wiels, V.: The landing gear system case study. In: ABZ case study, Communications in Computer information science, vol. 433. Springer, Switzerland (2014)
7. Boniol, F., Wiels, V., Ait Ameur, Y., Schewe, K.-D.: ABZ 2014: the landing gear case study. Proceedings of case study track, held at the 4th international conference on abstract state machines, alloy, B, TLA, VDM, and Z, Toulouse, June 2–6, 2014. Communications in Computer Information Science, vol. 433. Springer, Switzerland (2014)
8. Börger, E., Stärk, R.F.: Abstract state machines: A method for high-level system design and analysis. Springer, New York (2003)
9. Teodorov, C., Dhaussy, P., Le Roux, L.: Environment-driven reachability for timed systems. Safety verification of an aircraft landing gear system. Int. J. Softw. Tools Technol. Transf. doi:10.1007/s10009-015-0401-2 (2016)
10. Farail, P., Gaufillet, P., Peres, F., Bodeveix, J.P., Filali, M., Berthomieu, B., Rodrigo, S., Vernadat, F., Garavel, H., Lang, F.: FIACRE: an intermediate language for model verification in the TOPCASED environment. In: European congress on embedded real-time software (ERTS), SEE, Toulouse (2008)
11. Hansen, D., Ladenberger, L., Wiegard, H., Bendisposto, J., Leuschel, M.: Validation of the abz landing gear system using prob. Int. J. Softw. Tools Technol. Transf. doi:10.1007/s10009-015-0395-9 (2016)
12. Mammar, A., Laleau, R.: Modelling a landing gear system in event-b. Int. J. Softw. Tools Technol. Transf. doi:10.1007/s10009-015-0391-0 (2016)
13. Su, W., Abrial, J.R.: Aircraft landing gear system: approaches with event-b to the modelling of an industrial system. Int. J. Softw. Tools Technol. Transf. doi:10.1007/s10009-015-0400-3 (2016)