Overview of automation tools for avionics verification

S A Dyachenko¹,², D M Ilyashenko¹,² and E S Neretin¹,²
¹ Moscow Aviation Institute (National Research University), 4, Volokolamskoe shosse, Moscow, 125993, Russia
² Integration Center branch of IRKUT Corporation, 5, Aviatsionniy pereulok, Moscow, 125167, Russia
E-mail: dyachenkosa@mai.ru

Abstract. The existing verification automation tools for civil aircraft avionics are considered (particularly, formal methods, hardware-in-the-loop testing, outsource services). Key solutions presented on the market are described, their advantages and weak points are identified. Also, it is determined a promising field is the automation of graphic and aural information testing during the cockpit human-machine interface assessment (e.g. for cockpit display and flight warning systems). This type of verification is practically not automated nowadays. The architecture of a complex for the mentioned task is proposed, the methodology of verification using the described tool is considered either. The camera and microphone usage for monitoring of images and sound alerts which is being outputted to the crew allows applying of computer recognition algorithms during verification. It will allow reducing the time and cost expenses for human-machine interface testing as well as reducing the probability of mistakes could be made during fully manual testing. In addition, the proposed concept makes its application possible for any technical objects with a user interface in other industry areas (not just in aviation) such as space, automotive, shipbuilding, etc.

1. Introduction
According to the International Air Transport Organization (IATA) statistics, the loss of control in-flight (LOC-I) is the leading cause of accidents in terms of the victim’s number. In the period from 2015 to 2019, the 51% of total victims are caused by LOC-I although only 8% of crashes occurred due to this reason [1].

The IATA indicators are confirmed by the research of the leading civil aircraft developers. Boeing data from 2009 to 2018 show: the number of fatalities in aircraft crashes due to LOC-I more than double the victims caused the second largest factor—controlled flight into terrain (1183 versus 568; the total number of fatalities during the specified period is 2532) [2].

At the same time, the given values reflect a positive dynamic. Airbus claims thanks to the new technologies application, the annual death rate per million flights has been reduced from 0.6 fatalities in 1998 to 0.03 in 2017 [3] (i.e. the situation improved 20 times over the past two decades).

According to the accepted aviation terminology LOC-I category includes cases in which the crew was unable to maintain control in flight, so an irreparable deviation from the intended trajectory occurred [4]. Among the main causes of losing control related to LOC-I are:

- failure or malfunction of an aircraft system/component;
- deteriorated meteorological conditions;
• crew actions associated with low visibility;
• deliberate maneuver (leading, for example, to a stall);
• incorrect aircraft configuration (for example, the position of the flaps or slats inappropriate to the current stage of flight);
• events caused by icing of board elements, etc.

Most of them refer to weather deterioration or crew incorrect actions, so it is almost unpredictable circumstances. Thereby, their prevention is extremely difficult, this requires the development and integration of new facilities to increase the crew situational awareness in low visibility, as well as to automate piloting. However, the failure or malfunction of the onboard equipment can be caused by a mistake made in the aircraft development and not identified within the testing and certification on time.

A 2011 NASA research on 126 LOC-I crashes from 1979 to 2009 shows that more than half of the cases (54%) were due to adversities on board, most of which are related to equipment’s failure/error (33.3% of the total number of catastrophes).

Today the most of the avionics functions are performed by software which is growing over time in complexity and volume exponentially. So, special means are required to ensure its safety. Particularly, the strict requirements applied to the onboard software development process should be met. They are specified in the RTCA DO-178C [5] standard. Following the mentioned document, the software life cycle is consisted of:

• planning;
• development (including the development of requirements, software design, coding, and integration);
• integral sub-processes (verification, configuration management, quality assurance, interaction with the certification authorities);
• certification.

In addition to adherence to life cycle requirements, onboard software developers seek to minimize the human factor influence during its creation to improve code security. These goals can be achieved through the automation tools use which also reduces the time and financial costs associated.

2. Market overview in terms of onboard software verification automation

2.1. Formal methods verification solutions

Today, formal methods (in particular, model checking) are widespread among the approaches of avionics software verifying. They are based on the analysis of the system model which is expressed in temporal logic and described by a finite number of states. The system model, together with logical formulas, is fed to the input of a verifier program that automatically checks the truth of these formulas. Also, the use of Petri nets is widely applicable among the formal verification methods. In this case, modeling is carried out for events: it determines what actions take place in the system, what events preceded them, and what states the system will have after their execution [6]. Formal methods are implemented by software environments ANSYS SCADE [7], MATLAB & Simulink [8], MASIW developed by the Russian Academy of Sciences together with “GosNIIAS” [9], etc.

2.2. Hardware-in-loop verification solutions

To automate the verification using hardware-in-loop, another type of tools is used. Particularly, it is required to monitor and process information transmitted via data buses as well as to generate reports based on test results. An example of such tool is the diagnostic suite “FREGAT”
manufactured by JCS “UIMDB” [10]. The solution allows information signals simulation for various interfaces applied in aviation (discrete, RS-232, ARINC 429, ARINC 708, ARINC 717, MIL-STD-1553B, etc.) through embedded input/output devices. Inputting the signals and reading the system processing results in the output under test make it possible to assess the correctness of the embedded software. Besides, “FREGAT” provides displaying, documentation generation, and storing the received results.

Another solution for onboard systems verification (including hardware-in-loop testing) is provided by TechSAT GmbH which is the developer of the 2nd generation Avionics Development System (ADS2). It is a universal environment for prototyping, design, integration, and verification. Its use almost completely covers the development cycle under the SAE ARP4754A [11] standard. ADS2 integrates a software simulation environment and a hardware platform which jointly provide the following abilities:

- creating simulators of onboard equipment;
- implementing the exchange between models and real units via ARINC 429 [12], ARINC 664 [13], ARINC 825 [14], MIL-STD-1553B [15], Ethernet and other interfaces;
- data visualizing (plotting the graphs of signals);
- data recording;
- test sets design and execution.

At the same time, scalability, flexible configuration, and modular implementation of the complex are provided. The high-level architecture of the ADS2 test bench is shown in figure 1.

![Figure 1. High-level architecture of the ADS2 test bench.](image)

A significant advantage of the ADS2 solution is integration with popular computer-aided design systems (MATLAB & Simulink, ANSYS SCADE, LabVIEW, IBM Rational Rhapsody, IBM Rational DOORS, etc.), as well as with custom scripts written in the Python programming language.

The projects developed using ADS2 include the Airbus A350 cockpit, COMAC C919 electronic display system, MC-21 avionics, Boeing B787 remote data concentrator, Eurofighter Typhoon propulsion controller, etc. [16].
2.3. Outsource verification solutions

Also today, many developers of on-board equipment resort to the help of third-party companies for software verification. The main list of companies specializing in this includes ScienceSoft (The USA), A1QA (The USA), Advalange (Russia), Kualitatem (The USA), TestingXperts (The UK), Vector (The USA), BugRaptors (India), LDRA Technology (The USA), Aviaok International LLC (Russia), etc. [17–19].

They perform verification taking into account the requirements of industry standards, including setting up a test environment, generating test sets, integration and unit testing, generating test results, and assistance in software certification.

2.4. Market analysis resume

Formal methods allow to find and fix bugs at the early development phases. Additionally, the threshold for getting into work with formal verification software environments is quite low. The intuitive interface admits reducing the requirements for the skills of the tester (for example, no specific programming skills are required). However, this approach is aimed at checking the model's compliance with the requirements described in the formal language. Therefore, it does not take into account the features of the target platform.

Hardware-in-loop complexes are powerful tools used for real units testing. Also, an advantage is the available information exchange monitoring via data buses allows checking the correctness of the system operation. On the other hand, the key disadvantage is that checking all avionics equipment is impossible using these tools. The systems with a human-machine interface (HMI) (e.g. cockpit display system generating images on the screens or flight warning system outputting aural alerts to the crew) requires to carry out testing of visual/aural information besides data buses monitoring.

Third-party verification allows a company to reduce time and cost expenses, as well as to obtain a verification result without managing this process in fact. However, the customer is dependent on the outsourcing company, and the risk of corporate information leakage also increases.

Thus, the automation tools development for HMI testing is a vital task especially taking into account the high criticality of such systems (its failure can lead to catastrophic or hazardous situations in the case of cockpit display and flight warning systems respectively). The aviation HMI verification means are currently not available on the market. However, their implementation is possible through the use of computer processing methods (particularly, image and sound recognition algorithms).

At the current stage of technology development, a wide range of images and sound processing methods are used in IT (for authentication in software services using biometric data), medicine (for early detection and diagnosis of dangerous diseases), video surveillance (e.g. for tracking cars and citizens), etc. In aviation, such solutions are mainly used at airports to recognize the faces of passengers and the baggage in order to identify intruders and items prohibited for carriage. Though, the mentioned methods are not limited to these applications. By using it in the aircraft development process, it is possible to achieve significant advantages over current approaches.

Since the above methods are universal, they can be adapted to any HMI systems. That is, the use of automation tools for verifying graphic and sound information is not limited to the aviation industry only. Its application is possible for any technical objects with a user interface in other industry areas either: space, shipbuilding, automotive.

3. The architecture of the HMI verification automation complex

The proposed verification automation complex for avionics HMI systems in the cockpit includes the following components (see figure 2):

...
• avionics test bench containing cockpit environment;
• camera installed in front of displays to monitor the generated graphic information;
• microphone installed in front of speakers to record tone alarms and aural warnings;
• tester’s workstation (a personal computer) with software installed for processing data from the camera and microphone and generating test results.

![Diagram of avionics integration bench, camera, microphone, and tester’s workstation with software tools like TechSAT ADS, Microsoft Office, Python, and Textalk OCR]

Figure 2. The proposed architecture of the HMI verification automation complex.

The working principle of the complex is as follows. Before carrying out the tests, the verifier sets up the equipment and determines the expected results (symbols, text going to be displayed on the screens and sound warnings outputting through the speakers). Next, the signals for image/sound generation are set through the workstation or special analyzers (depends on the bench architecture) and gone to the system’s input. After, the camera and microphone automatically record the obtained image and sound which are fed to the input of the software installed on the computer. The software recognizes the received streams, compares them with the expected results, and generates a report based on the test performed.

The testing methodology using the described complex consists of the following steps:
• equipment setup;
• inputting of expected results;
• inputting of signals for generating graphics on the screens/sound warnings outputting;
• processing data from the camera/microphone;
• test results generating.

4. Conclusion
At the moment, there are many solutions on the market in terms of software verification of aviation onboard systems: formal methods suites for equipment models testing; hardware-in-loop tools for monitoring data buses; outsourcing services for testing embedded software. However, the available products do not provide HMI (graphics and sound) verification automation.

Their development can be based on computer recognition methods. Its application will allow significantly reduce the time and cost expenses for HMI systems (e.g. cockpit display and flight warning) verification, reduce the human factor influence (the probability of error) during testing, increase flight safety. In common, the reduction in verification time reduces the aircraft development cycle and rushes up the certification process.
The weak point of such a solution is in the following. The current standards applied in aviation does not permit the use of non-deterministic algorithms applied in image and sound recognition (neural nets, correlation functions) because they can bring mistakes in results. However, this drawback is leveled by the need to control the automatically obtained results by the operator. So, the operator could not be excluded from the verification process completely. However, manual actions that should be performed within fully manual testing are minimized.

The proposed described complex makes it possible to reduce the time and costs expenses for HMI systems verification by several times. Besides, the versatility of the computer recognition methods allows adapting the tool to any technical objects with a user interface that meets the demand in most industries (not just aviation).

References

[1] International Air Transport Association 2019 Safety Report (56th ed.) (Montreal: IATA) p 262
[2] Boeing 2019 Statistical Summary of Commercial Jet Airplane Accidents. Worldwide Operations 1959–2018 (50th ed.) (Chicago: Boeing) p 27
[3] Airbus 2018 Statistical Analysis of Commercial Aviation Accidents 1958–2017 (Toulouse: Airbus) p 32
[4] Commercial Aviation Safety Team of International Civil Aviation Organization 2011 Aviation Occurrence Categories. Definitions and Usage Notes (Montreal: ICAO) p 28
[5] RTCA 2011 DO-178 Software Considerations in Airborne Systems and Equipment Certification (Washington: RTCA) p 144
[6] Ivutin A N and Daragan E I 2011 The main approaches to real-time software verification Technical Science Proc. of Tula State University vol 2 (Tula: TSU Publishing) pp 563–567
[7] 2011 Methods for Industrial Critical Systems Proc. of 16th International Workshop FMICS (Trento: Springer) p 261
[8] 2014 Software Engineering and Formal Methods Proc. of 12th International Conference SEFM (Grenoble, Springer) p 394
[9] Buzdalov D V, Zelenov S V, Kornikhin E V, Petrenko A K, Strakh A V, Ugkenko A A and Khoroshilov A V 2014 The tools of integrated modular avionics systems design Proc. of the System Programming Institute of Russian Academy of Sciences vol 26(1) (Moscow: SPI RAS Publishing) pp 201–230
[10] Cherkashin S V, Shishkin V V and Dolbya N A 2009 Universal diagnostic system for onboard radio-electronic equipment Proc. of the Samara Scientific Center of Russian Academy of Sciences vol 11(3) (Samara: SSC RAS Publishing) pp 392–397
[11] Society of Automotive Engineers 2010 ARP4754A Guidelines for Development of Civil Aircraft and Systems (Warrendale: SAE) p 115
[12] Aeronautical Radio Incorporated 2004 ARINC Specification 429P1-17. MARK 33 Digital Information Transfer System (DITS). Part 1: Functional Description, Electrical Interface, Label Assignments and Word Formats (Annapolis: ARINC) p 309
[13] Aeronautical Radio Incorporated 2006 ARINC Specification 664P1-1. Aircraft Data Network. Part 1: Systems Concepts and Overview (Annapolis: ARINC) p 51
[14] Aeronautical Radio Incorporated 2011 ARINC Specification 825-2. General Standardization of CAN (Controller Area Network) Bus Protocol for Airborne Use (Annapolis: ARINC) p 170
[15] United States Department of Defense 1978 Interface Standard for Digital Time Division Command/Response Multiplex Data Bus MIL-STD-1553 (Washington: Department of Defense) p 51
[16] 2020 TechSAT ADS2 User Cases [Online] (accessed: 2020-11-06) URL https://techsat.com/wp-content/uploads/2020/01/TechSAT_PRS-ADS2-UseCases_20200117.pdf
[17] 2020 Top 20 Software Testing Companies in 2019 [Online] (accessed: 2020-08-14) URL https://medium.com/@andy_dassan/top-software-testing-companies-in-2019-c418b24f69d0
[18] 2020 Advalange [Online] (accessed: 2020-08-14) URL https://advalange.ru/testing
[19] 2020 LDRA Automating Software Verification, Requirements Traceability and Standards Compliance [Online] (accessed: 2020-08-14) URL https://ldra.com