Dynamic Assurance of Safety

Summary Report of a Visit to the NASA Ames Research Centre, Moffett Field, California USA, 12-22 August 2015

Ibrahim Habli, University of York

Summary

Many software-intensive systems used in critical industries such as aviation, power and healthcare require the assurance of safety properties. Existing standards, processes, techniques and practices place a great emphasis on how this can be achieved throughout the design and development stages. However, there is little guidance on how safety assurance should be maintained through life, i.e. during the operational, maintenance and decommissioning stages. Many assumptions about the environment and the system performance and use, particularly for complex and novel autonomous systems, that are made during the design and development stages might turn out to be incorrect during operation. From a safety point of view, this can threaten the validity of the safety case and weaken confidence in the actual safety of the system.

This research visit has investigated this issue in collaboration with Dr Ewen Denney and Dr Ganesh Pai in the Intelligent Systems Division at the NASA Ames Research Center. It also explored ways in which the safety case of a system can be treated as a dynamic artefact, which is continually updated based on operational data in order to maintain and assess the validity of the safety justification.

Introduction

The development and use of software-intensive systems in safety-critical applications are heavily guided by safety standards [1]. These standards form the basis on which the systems are approved and certified. Historically, these standards tended to be highly prescriptive, i.e. in terms of the specific certification requirements they impose and the means used for meeting these requirements [2].

For the last twenty years, a new approach to software certification has emerged, placing more emphasis on the safety case argument and evidence that developers have to generate and communicate in order to satisfy high-level, i.e. less prescriptive, certification goals [3]. This approach has been established, in part, as a way of responding to the challenges involved in certifying systems that use novel technologies, e.g. unmanned aircraft systems, or emerging software engineering techniques, e.g. automated and model-based development, where

the rapid changes in these technologies and techniques have outpaced the ability to prescribe one-size-fits-all certification measures.

The validity of the safety case, defined prior to deployment, relies on predictions and assumptions about the behaviour of the system and the interactions the system has with its environment (i.e. other systems, users and processes). However, increasingly, safety-critical systems are interconnected and dynamically reconfigurable (e.g. networked medical devices) and are given more authority and becoming more autonomous (e.g. driverless cars). As such, the learning, emergent and adaptive behaviour and use of these systems pose a significant challenge to the assumptions and predictions made in the safety case, regardless of whether the documentation of the safety argument is explicit (i.e. in goal-based certification) or implicit (i.e. in prescriptive certification).

That is, both the system, e.g. through the learning functions of a UAS flight control system, and its environment, e.g. through usage scenarios of infusion pumps not envisaged by the designers, will evolve after deployment. The mismatch between the actual operation of the system and our initial understanding of it may undermine the evidence used, and invalidate the claims made, in the safety case.

Research Study

During this visit, the research has explored the notion of Dynamic Safety Cases [4], aiming to transform the safety reasoning into a dynamic and live artefact that is checked, validated and updated based on actual feedback data. The work has considered four activities that are associated with a Dynamic Safety Case, as shown in Figure 1, namely:

- **Identify** the sources of uncertainty, highlighted in the safety case, that have the potential to weaken confidence in safety;
- **Monitor** by collecting the operational data that correspond to these sources of uncertainty;
- **Analyse** the impact on the safety reasoning and evidence, considering techniques such as Bayesian Belief Networks; and
- **Respond** by making changes, when necessary, to the system, its environment and the safety case.

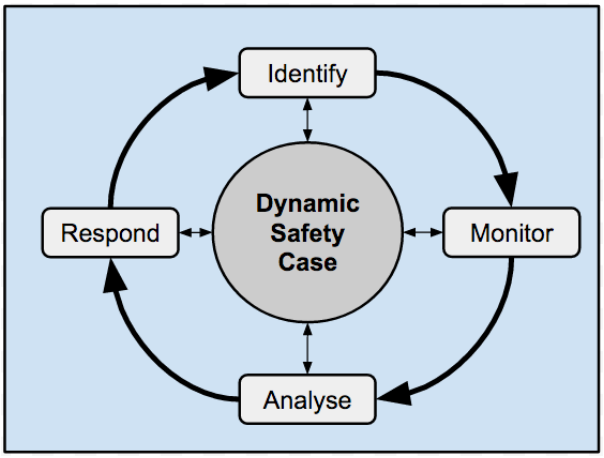

Figure 1: Dynamic Safety Case Activities

Discussion/Findings:

The initial objective of the visit was to assess the use of probabilistic uncertainty analysis for the assurance of unmanned aircraft systems, as part of a dynamic way for assuring safety. However, throughout the process and discussions, we discovered that many challenges lie in the collection and choice of the data used for a probabilistic approach such as Bayesian Belief Networks (BBNs).

This has led to the need to elaborate on the lifecycle discussed earlier. There is the potential for BBNs to form the engine for updating the belief about the safety of complex systems such as unmanned aircraft systems. However, this should be framed within a wider framework that considers data collection and assurance (e.g. relevance, quality and quantity of available data) and means for responding to changes on safety confidence.

The visit has highlighted a number of challenges to the above model:

- Concerning sources of uncertainty, how can we decide on the most important subset of these? How do we capture and analyse the variable nature of uncertainty? Should this be performed analytically or empirically?
- Concerning monitoring, what can we learn from the field of smart sensors and health monitoring?
- Concerning analysis, what can we learn from the world of AI and machine learning?
- Concerning responses, what are the rules for updating the safety argument and evidence dynamically? How much automation is possible and desirable? Do we need a new theory for argument refactoring?

Systems Engineering Context and Future Steps

There is a great deal of interest within the systems engineering community in the concept of resilience [5]. Resilient software systems are not expected to merely

resist changes and certain classes of failures, but dynamically adapt, learn and evolve in the face of these events. In other words, resilience is the ability of a system to proactively manage, rather than, resist uncertainty. For safety-critical systems, dynamic safety cases, with their emphasis on adaptive and continuous assurance, have the potential to provide structured means for reasoning about resilience, particularly for systems or environments that exhibit a high degree of uncertainty (e.g. unmanned autonomous systems or healthcare settings).

Future steps include the investigation of the relationship between Dynamic Safety Cases and Safety Management Systems, particularly leading and lagging indicators, and the development of formalism as a basis for partial automation.

References

[1] Hatcliff J, Wassyng A, Kelly T, Comar C, Jones P. Certifiably safe software-dependent systems: challenges and directions. InProceedings of the on Future of Software Engineering 2014 May 31 (pp. 182-200). ACM.

[2] Hawkins R, Habli I, Kelly T, McDermid J. Assurance cases and prescriptive software safety certification: A comparative study. Safety science. 2013 Nov 30;59:55-71.

[3] McDermid JA. Software safety: where's the evidence?. InProceedings of the Sixth Australian workshop on Safety critical systems and software-Volume 3 2001 Jul 1 (pp. 1-6). Australian Computer Society, Inc..

[4] Denney E, Pai G, Habli I. Dynamic Safety Cases for Through-life Safety Assurance. In37th International Conference on Software Engineering (ICSE 2015)–New Ideas and Emerging Results (NIER) 2015 May.

[5] Hollnagel E, Woods DD, Leveson N. Resilience engineering: Concepts and precepts. Ashgate Publishing, Ltd.; 2007.