Better, Faster, Cheaper, and Safer Too – Is This Really Possible?

Iain Bate[1,2], Hans Hansson[2], Sasikumar Punnekkat[2]
[1]Department of Computer Science, University of York, York, UK
[2]Mälardalen Real-Time Research Centre, Mälardalen University, Västerås, Sweden
email: iain.bate@cs.york.ac.uk, hans.hansson@mdh.se, sasikumar.punnekkat@mdh.se

Abstract

Increased levels of automation together with increased complexity of automation systems brings increased responsibility on the system developers in terms of quality demands from the legal perspectives as well as company reputation. Component based development of software systems provides a viable and cost-effective alternative in this context provided one can address the quality and safety certification demands in an efficient manner. In this paper we present our vision, challenges and a brief outline of various research themes in which our team is engaged currently within two major projects.

1. Introduction

The aim of our work is to ensure functional safety, i.e., the absence of unreasonable risk due to hazards caused by abnormal (covering incorrect and unexpected behaviours brought about by systematic or random failures) of the electrical/electronic systems. Typically, the safety work is guided by domain specific standards that prescribe methods for ensuring functional safety at the appropriate safety integrity level and require the developer to provide elaborate evidences to support their safety claims.

Even though several standards have been in place for a while, the cost of obtaining certification is significant, with estimates such as 30% of lifecycle costs [4] and 25-75% of development costs [13] are spent on certification, and the cost of verifying a single line of code is between USD 150-250 [7]. Kesseler's review of nine guidance/standards documents [10], covering DO-178B, DO-278B, AC120-76A, ESA DRD 920, IEC 61508, IEC 60880-2, FDA 1252, and ISO-15408, found that the current state of certification practices hampers innovation when integrated systems use COTS components.

In an industrial automation scenario, there is an increasing need for globally acceptable, integrated and intelligent safety solutions, which can essentially reduce the plant downtime as well as limit liability exposure [1]. In essence this means that not only the first time certification of systems should be made cheaper, it should also be easier to change the systems without incurring disproportionate cost compared to the size of the change, the systems and components developed should be reusable across product lines and in different domains, and Maintenance Free Operating Periods (MFOPs) should be lengthened without the risk of unscheduled maintenance. At the same time the industries should be able to introduce new technologies such as Wireless Sensor Networks (WSN) and Wireless Communications whilst still being able to produce an acceptable safety case. Relevant standards include IEC 62061, IEC 60204 and ISO 13849 for machine safety or IEC 61511 for process safety, NFPA 79 for industrial machinery, the ANSI B11 series for machine tools, S2 for the semiconductor industry, and RIA 15.06 for robots.

The research outlined in this paper is motivated by several important and clearly perceivable trends: (1) The increase in software based solutions which has led to new legal directives in several application domains, e.g. automotive, as well as a growth in safety certification standards. (2) The need for more information to increase the efficiency of production, reduce the cost of maintaining sufficient inventory, and enhance the safety of personnel by the use of more sensors within the factory. (3) The rapid increase in complexity of software controlled products and production systems, mainly due to the flexibility and ease of adding new functions made possible by the software. As a result the costs for certification-related activities increase rapidly. (4) Modular safety arguments and safety argument contracts have in recent years been developed to support the needs of incremental certification. (5) Component-Based Development (CBD) approaches, by which systems are built from pre-developed components, have been introduced to improve both reuse and the maintainability of systems. CBD has been in the research focus for some time and is gaining industrial acceptance, though few approaches are targeting the complex requirements of the embedded domain.

Our aim is to enhance existing CBD frameworks by extending them to include dependability aspects so that the design and certification of systems can be addressed together more efficiently. This would allow reasoning about the design and safety aspects of parts of the systems (referred to as components) in relative isolation, without consideration of their interfaces and emergent behaviour, and then deal with these remaining issues in a more structured manner without having to revert to the current holistic practices. The majority of research on such compositional aspects has concentrated on the functional properties of systems with a few efforts dealing with timing properties, e.g. based on rely guarantees [5] and Time-Triggered Protocols. However, much less work has considered non-functional properties, including dependability properties such as safety, reliability and availability.

This paper presents on-going research performed within two larger research projects: (1) the EU/ARTEMIS project SafeCer (2011-2015; safecer.eu) - a European project with more than 30 partners in six different countries, which aims at adapting processes, developing tools, and demonstrating applicability of composable certification within the domains: Automotive, Avionics, Construction Equipment, Healthcare, and Rail, as well as addressing cross-domain reuse of safety-relevant components; (2) the Swedish national project SYNOPSIS (tinyurl.com/MDH-SYNOPSIS), is a project at Mälardalen University (2011-2016) sharing the SafeCer objective of composable certification, but emphasizing more the scientific basis than industrial deployment.

2. Composable Safety-certification

Our overarching objective is to increase efficiency and reuse in development and certification of safety-relevant embedded systems by providing process and technology that enable composable qualification and certification, i.e. qualification/certification of systems/subsystems based on reuse of already established arguments for and properties of their parts. The scope of the work is the software parts of the system, however the software safety case must reside and clearly link to the wide systems safety case. This structure is reflected in most current safety standards, e.g. DO178B, ARP4751 etc..

Composable certification can be realised only if the nature of the interfaces between parts of the systems is sufficiently well understood, particularly regarding the incompleteness of contracts, the emergent behaviour when components interact and the existence of components whose behaviour may have insufficient evidence that they meet the failure-related targets (e.g. reliability and safety). A key aspect of our strategy is the use of safety contracts, enforced by health monitoring, to ensure these emergent behaviours cannot lead to hazards. Stress-based testing and model checking will supplement more conventional forms of verification to check that the properties hold. The basis for the testing work will be search-based techniques capable of not only dealing with large and complicated systems, but more importantly they can work without knowledge of the internal structure and detailed behavioural interface information of components [12].

The approach that is to be taken is to use the Goal Structuring Notation (GSN) for arguing the safety of systems. The main purpose of a goal structure is to show how goals (claims about the system) are successively broken down into sub-goals until a point is reached where claims can be supported by direct reference to available evidence (solutions). As part of this decomposition, using the GSN it is also possible to make clear the argument strategies adopted (e.g. adopting a quantitative or qualitative approach), the rationale for the approach (i.e. assumptions, justifications) and the context in which goals are stated (e.g. the system scope or the assumed operational role). To prevent the need for a single argument for the whole system, decomposition is supported by allowing arguments to be split into smaller parts. For further details on GSN see [8].

An advantage of GSN is that it makes clear how certain parts of the safety case relate to different aspects of the systems design, verification and certification. Figure 1 presents a typical approach to arguing about timing. The argument concerns an engine control computer and is taken from [9]. The arguer contends that the system will exhibit deterministic timing behaviour in the presence of credible faults (goal G22 and context C10) are met. An argument is made over detection of the faults (goal G24) and recovery from the faults in bounded time (G25). The detection of faults (goal G24) is then decomposed into value (goal G26) and timing (goal G27) fault detection. Recovery (goal G25) is split between attempting shutdown and recovery (goal G28), and if that doesn't work the resource is assumed to have permanently failed so it is taken out of service (goal G29).

The GSN clearer shows which parts of the argument, and consequently the evidence, that would need to be revisited. For example the credible faults and their relationship to the rest of the argument are explicitly identified. In the context of factory automation the faults of interest would be derived from a combined of hazard and failure analysis. It would include mechanical wear, failures due to the harsh environment, and unexpected interference sources. If the system is then changed or reuse, then these would have to be refreshed. However the implications would be clearer in terms of the impact on the rest of the safety case. More recently [2], GSN has been extended to deal with modular arguments and the relationships between how systems are design and certified explored.

2.1 Challenges

Composable certification poses several challenges at various levels:

System level Challenges: From an industrial perspective reducing cost of system design and certification, reducing development cycles and shortening the lead-time for re-validation and re-certification are extremely important challenges, which we address by providing a means for developing systems in a compositional way taking explicit account of both the functional and behavioural (including dependability) nature of interfaces, thus supporting component reuse, incremental certification, and reuse of safety-arguments. Agile mechanisms to produce traceability for evidence across artefacts are also essential. Our system oriented research is specifically focusing on (1) Requirements and Processes for co-certification, i.e., intertwined development and certification, and (2) Safety Argumentation (context, assumptions, limitations and confidence analysis of contracts).

Disciplinary level Scientific Challenge: Managing complexity and scalability are important challenges when introducing any new technique. Bottom-up composition of parametrized specifications in conjunction with top-down checked models and verification efforts form our strategy to meet these challenges. In particular our dis-

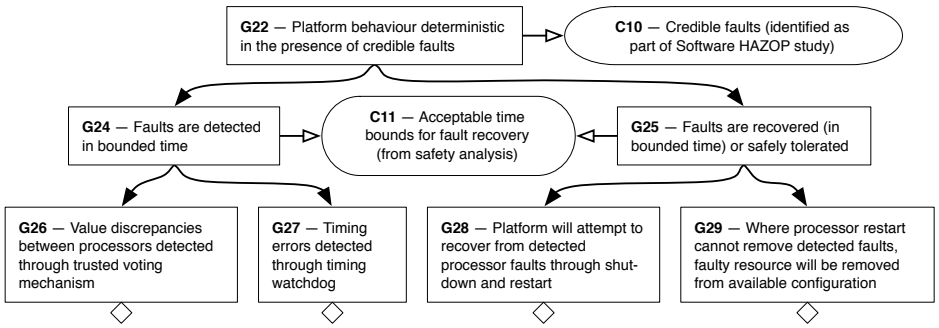

Figure 1. Typical health monitoring argument (taken from [9])

ciplinary research focuses on (1) achieving predictability of behavioural interfaces through static contract analysis, parameterization and virtualization, and (2) hypothesis-centric certification through model-checking and specialised testing efforts.

Demonstrator Challenges: The developed methods and technology need to be validated in industrial settings considering relevant domain specific safety standards. The diversity of standards and domain specific practices constitutes a challenge for large scale deployment of research results. Specific focus will be on (1) development/integration of proposed techniques and processes in a Certification tool framework, and (2) demonstration of applicability across multiple domains as well as cross-domain reusability through industrial case studies.

Progress beyond the state-of-the-art : Our research addresses many important research questions, including how to develop safety arguments that support compositional approaches based on well-defined interfaces, and how to generate the corresponding evidence in a similar compositional way, where possible automatically.

The certification of systems often require a safety argument that the system's hazards are sufficiently understood and mitigated, and supporting evidence to demonstrate the validity of that argument. Traditionally, certification standards have been process-oriented, i.e. where a hazard analysis is performed to identify the severity and risks associated in functional failure for determining a Safety Integrity Level (SIL), which in turn is used to choose and customize the process applied. Such approaches are rather insufficient and inefficient for CBD as well as for the perceived changes in standards, and in particular not applicable in product-line engineering [6].

Safety arguments need further attention to deal with compositional approaches with components instead of decompositional approaches to form modules. The challenge here is that traditional safety cases are normally concerned with establishing the hazards associated with the system, building an understanding of how the hazards may occur, and then mitigating them so the risk is acceptable considering their severity. The last of these is normally satisfied by arguing that the construction of the system from components is acceptable. In contrast, a compositional approach would argue about the construction of the system and then how each component contributes towards the hazards. This inversion is challenging for a number of reasons including the danger that the argument of hazard mitigation is then spread through the safety case, and even then only implicit, rather than being explicitly tackled early on.

There is additionally a need for research into arguments as to why a system is sufficiently safe despite contracts being incomplete. The challenges here are accepting that the contracts are incomplete, working out where to put the contracts, understanding their limitations and ensuring these limitations do not lead to the large re-certification activities that we are trying to avoid. Major technological breakthroughs in these areas are essential to enable efficient certification of future safety-relevant embedded systems. We aim to enhance the state-of-the-art, the state of industrial practice, and possibly the state of certification standards through scientific innovations in the above themes, as well as have a significant impact on cost and time to market factors.

2.2 A reuse scenario

As an illustration of the potential benefits of the process, technology and tools that we are developing, consider the following reuse scenario:

Company X has a broad range of variants of its machine equipment product. According to the relevant safety standards, the safety of each variant needs to be certified for its intended use. Furthermore, assume that product variant V has been certified and that the company has received an order for a new variant V', which only differs from V in that it contains one new subsystem and will be used in a slightly different context.

The traditional approach to safety-certification would be to formulate a separate set of safety cases, collect evidence by verification, and present an argumentation for the safety of V' that is independent of V.

Our approach is different, in that we focus on the differences (the delta) between V and V', and reuse a large part of the safety cases, evidence and argumentation from the certification of V in the certification of V', as illustrated by Figure 2.

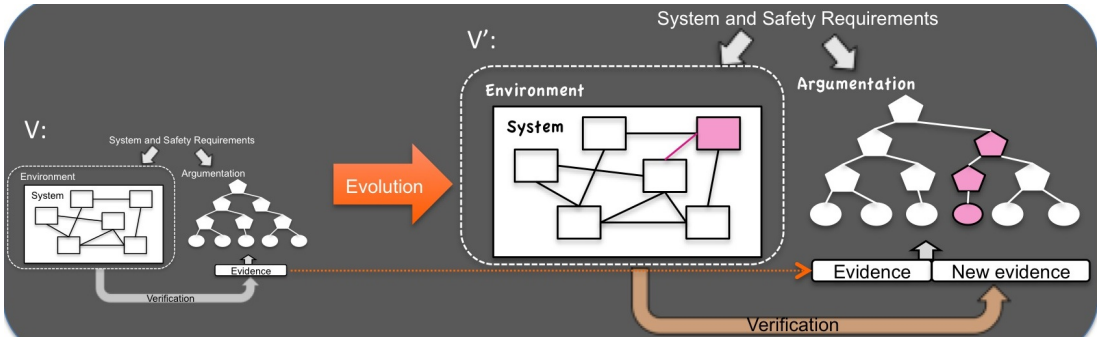

Figure 2. Composable certification - the product-line scenario.

3. Approach and Preliminary Results

To enable composable certification as outlined above, we are performing research into

Component models - in particular extending software component models with safety-contracts that enable specification of guarantees (the properties that hold for the component) and assumptions (the conditions under which these properties hold).

Safety-argumentation - providing better support for contracts, understanding confidence in the argument and evidence and arguing safety in a compositional way.

Verification - determining the limitations of the contracts and the likelihood of failures that might lead to hazards occurring using combinations of model checking and stress testing.

Traceability - improving the traceability between parts of the argument and between the argument and the component design such that the impact of changes can be better understood leading to better focus on regression testing.

The research is still at an early stage. To date we have

- Produced a Generic Process Model that is a representation of different certification standards and safety practices
- Identified what should go into component contracts to support dependability. This has not only included the important properties of systems but also the confidence and reliability that is needed for the information across the interface.
- Developed preliminary arguments that support compositional approaches and highlight the confidence issues with evidence.
- Developed health monitoring strategies for an ABS system that we are now modeling and verifying by model checking to understand under what conditions hazards can still occur.
- Developed strategies for the certification of wireless communication technologies that includes a harmonisation of design techniques, anomaly detection, and safety argument / analysis. Preliminary details can be found in [3, 11].

Preliminary results from our research efforts as well as the growing interest and involvement of industrial partners gives us stronger conviction on the feasibility of our hypothesis that it is possible to develop safe systems in a cost efficient manner using the composable system development and certification approach. A few years from now we expect to be in a position to answer YES to the challenging title of this paper.

References

[1] ARC White paper. Profisafe: Networked safety for process and factory automation.

[2] I. Bate and T. Kelly. Architectural considerations in the certification of modular systems. *Reliability Engineering and System Safety*, 81:303–324, 2003.

[3] I. Bate, Y. Wu, and J. Stankovic. Developing safe and dependable sensornets. In *EUROMICRO Conference on Software Engineering and Advanced Applications*, 2011.

[4] R. Cleaveland. Formal certification of aerospace embedded software. In *National Workshop on Aviation Software Systems: Design for Certifiably Dependable Systems*, 2006.

[5] W. Coleman and C. Jones. A structural proof of the soundness of rely/guarantee rules. *Journal of Logic and Computation*, 17:807–841, 2007.

[6] I. Habli. *Model-Based Assurance of Safety-Critical Product Lines*. PhD thesis, Dept of Comp.Sc., U. of York, 2009.

[7] I. Habli and T. Kelly. Challenges of establishing a software product line for an aerospace engine monitoring system. In *11th International Software Product Line Conference*, pages 193–202, 2007.

[8] T. Kelly. *Arguing Safety - A Systematic Approach to Safety Case Management*. PhD thesis, Department of Computer Science, University of York, 1999.

[9] T. Kelly, I. Bate, J. McDermid, and A. Burns. Building a preliminary safety case: An example from aerospace. In *Australian Workshop on Industrial Experience with Safety Critical Systems and Software*, October 1997.

[10] E. Kesseler. Air transport, from privilege to commodity: A COTS enabled paradigm shift. Technical report, Natl Aerospace Laboratory, 2003.

[11] T. H. Lim, I. Bate, and J. Timmis. Multi-modal routing to tolerate failures. In *7th International Conference on Intelligent Sensors, Sensor Networks and Information Processing*, pages 211–216, December 2011.

[12] P. McMinn. Search-based software test data generation: a survey. *Software Testing, Verification and Reliability*, 14:105156, 2004.

[13] N. Storey. *Safety-Critical Computer Systems*. Addison-Wesley, 1996.