Formal Verification of Autonomous Driving Systems *

Yuvaraj Selvaraj[1,2]

[1] Chalmers University of Technology, Gothenburg, Sweden
[2] Zenuity AB, Gothenburg, Sweden
yuvaraj.selvaraj@zenuity.com

1 Background

Significant progress has lately been made in the global automotive industry towards autonomous vehicles and software systems are becoming a critical part of every vehicle. Autonomous vehicles can potentially increase road safety and help reduce road traffic accidents. However, these are extremely complex safety critical systems, and human safety depends on their correctness. Incorrect behaviour of autonomous driving systems may have catastrophic consequences. An autonomous vehicle consists of many different hardware and software components (e.g. sensing, decision making, actuation, and control) that interact to solve the autonomous driving task. Factors like size, structure (level of interaction and communication between different systems), environment (the physical world in the case of autonomous vehicles), application domain etc., all contribute to a level of complexity that is incomprehensible to the human mind. This may lead to subtle but potentially dangerous bugs arising due to unforeseen edge cases, errors in the software design and/or implementation.

It is of utmost importance that all safety critical parts of an autonomous vehicle are veritably reliable and safe. This is a challenge for the development process due to the complexity that needs to be managed not only in the design but also in the verification and validation process. Coverage based testing is a widely adopted work flow in many large scale software development companies and also within the automotive industry. However, exhaustive testing for all possible use cases is not practical, and partly not even possible. Testing can never guarantee absence of unintended consequences nor provide sufficient certification evidence in all cases. Thus, there is a need for complementary methods to guarantee system safety, and the use of formal methods for this is becoming prevalent [6, 11].

The international standard ISO 26262 [7] derived from the generic functional safety standard IEC 61508 provides guidance on a risk based approach to manage, specify, develop, integrate, and verify safety critical systems in road vehicles,

* This research abstract includes content and preliminary results from a work that will be presented at FMICS 2019. Supported by FFI, VINNOVA under grant number 2017-05519, *Automatically Assessing Correctness of Autonomous Vehicles–AutoCAV*.

including various references to formal specification and verification. Adherence to the standard can potentially ensure that system quality is maintained, and unreasonable residual risk is avoided. The standard is based upon the V model of product development [5] and aims at achieving system safety through safety measures implemented at various levels of the development process. However, the standard addresses neither specific challenges inherent to autonomous driving systems, nor the development of safety critical software in an agile development work flow.

2 Problem and Research Questions

The development of large scale safety critical autonomous driving software in an agile way thus necessitates research to solve challenges arising from such inter-disciplinary problems. The challenges that need to be addressed are atleast two-fold: (i) The application of formal verification to autonomous driving systems, (ii) The application of formal verification techniques to large scale agile development of safety critical software.

Formal methods—with varying levels of formalisation—can be applied at various stages of the software development process. The choice of verification method and the expressive power of the formalism used to specify the properties affects the approaches to tackle the above mentioned challenges. Some of the known challenges include scalability, level of automation in the verification procedure, expressing the continuous and discrete behaviour of autonomous driving system, and a sound and complete tool support for efficient integration.

The main objective in this research is to establish formal verification as an efficient tool for autonomous driving software development in an industrial setting. This objective is aimed to address the two-fold challenge mentioned above and to this end, the following research questions need to be answered:

RQ1: What are the important factors that affect the formal verification of autonomous driving systems and what are the current challenges in the existing methods?

RQ2: What are the techniques/algorithms/tools needed to address the answer to ***RQ1*** and how can they be scaled?

RQ3: How can the different techniques/algorithms/tools be integrated effectively in the day-to-day software development process?

3 Methods

The method adopted involves a three part approach aimed to answer each of the three research questions.

- The first part aims to identify and evaluate the currently available techniques and tools to perform formal verification of software systems. This requires a literature study, a benchmark evaluation of suitable techniques and tools to identify important qualities of the different techniques and their challenges to address formal modelling and verification of autonomous driving software.
- The second part aims to find solutions to the challenges identified in the first part. This may require work targeted towards developing methods for efficient abstraction of systems for better scalability, investigation towards compositional techniques with assume/guarantee specifications, and improving automatic methods to generate and verify system models.
- The final part of the work is to develop an effective framework for the successful integration of the methods in agile software development. The work shall be focused on improving the interoperability of different formal techniques and tools and their seamless integration with conventional software development tools.

4 Current State of Research

The current state of research is towards the end of the first part of the work attempted to answer ***RQ1***. Initial work on addressing ***RQ2*** has also been started. In [14], we evaluate three formal verification methods and their respective formalisms to verify existing software in an autonomous driving vehicle: Supervisory Control Theory with Extended Finite State Machines [12, 16], Model Checking with Temporal Logic of Actions [10], Deductive Verification with contract based programming [1]. Insights from the case study show a need for multiple formal methods to prove correctness and the difficulty to capture the right level of abstraction to model and specify the formal properties for the verification objectives.

5 Related Work

A recent survey [11] on formal specification and verification of autonomous robotic systems is a comprehensive study of current state-of-the art literature focused on formal modelling, formal specification, and formal verification of robotic systems. It gives a summary on the challenges faced, current methods in tackling the challenges, and the limitations of existing methods. In [15], an overview of the challenges in designing, specifying and verifying cyber-physical systems, particularly semi-autonomous driving systems with human interaction is provided. [4] presents a model checking framework for verifying autonomous systems with a distinguished rational 'agent', confined to the system architecture level with autonomous driving as one example scenario. A survey on safety critical robotics [6] gives insights in problems being faced in the development of dependable robotics. The paper presents the main challenges and makes a link between concepts from dependability and robotics. An overview of different types of tool based verification methods for software and hardware systems can be found in [3]. [2]

provides a report on the survey of existing tools for formal verification. More than two decades ago, [9, 13] discussed different challenges in transferring formal methods technology into industrial applications. Although formal methods in industry have come a long way since then, some of the challenges remain significant even today. [8, 17] consider several aspects of safety critical systems and agile development.

References

1. Apt, K.R., de Boer, F.S., Olderog, E.: Verification of Sequential and Concurrent Programs. Texts in Computer Science, Springer (2009)
2. Armstrong, R.C., Punnoose, R.J., Wong, M.H., Mayo, J.R.: Survey of existing tools for formal verification. SANDIA REPORT SAND2014-20533 (2014)
3. Beckert, B., Hähnle, R.: Reasoning and verification: State of the art and current trends. IEEE Intelligent Systems **29**(1), 20–29 (2014)
4. Fisher, M., Dennis, L.A., Webster, M.P.: Verifying autonomous systems. Commun. ACM **56**(9), 84–93 (2013)
5. Forsberg, K., Mooz, H.: The relationship of system engineering to the project cycle. In: INCOSE International Symposium. vol. 1. Wiley Online Library (1991)
6. Guiochet, J., Machin, M., Waeselynck, H.: Safety-critical advanced robots: A survey. Robotics and Autonomous Systems **94**, 43–52 (2017)
7. ISO: Road vehicles – Functional safety. Tech. Rep. ISO 26262 (2011)
8. Kasauli, R., Knauss, E., Kanagwa, B., Nilsson, A., Calikli, G.: Safety-critical systems and agile development: A mapping study. In: 2018 44th Euromicro Conference on Software Engineering and Advanced Applications (SEAA). IEEE (2018)
9. Kemmerer, R.A.: Integrating formal methods into the development process. IEEE software **7**(5), 37–50 (1990)
10. Lamport, L.: Specifying systems: the TLA$^+$ language and tools for hardware and software engineers. Addison-Wesley Longman Publishing Co., Inc. (2002)
11. Luckcuck, M., Farrell, M., Dennis, L., Dixon, C., Fisher, M.: Formal specification and verification of autonomous robotic systems: A survey. arXiv preprint arXiv:1807.00048 (2018)
12. Ramadge, P.J., Wonham, W.M.: Supervisory control of a class of discrete event processes. SIAM journal on control and optimization **25**(1), 206–230 (1987)
13. Saiedian, H., Hinchey, M.G.: Challenges in the successful transfer of formal methods technology into industrial applications. Information and Software Technology **38**(5), 313–322 (1996)
14. Selvaraj, Y., Ahrendt, W., Fabian, M.: Verification of decision making software in an autonomous vehicle: An industrial case study (2019), (in press)
15. Seshia, S.A., Sadigh, D., Sastry, S.S.: Formal methods for semi-autonomous driving. In: 52nd ACM/EDAC/IEEE Design Automation Conference (DAC). IEEE (2015)
16. Skoldstam, M., Akesson, K., Fabian, M.: Modeling of discrete event systems using finite automata with variables. In: 2007 46th IEEE Conference on Decision and Control. pp. 3387–3392. IEEE (2007)
17. Wolff, S.: Scrum goes formal: Agile methods for safety-critical systems. In: Proceedings of the First International Workshop on Formal Methods in Software Engineering: Rigorous and Agile Approaches. pp. 23–29. IEEE Press (2012)