

Methods and Tools for the Engineering and Assurance of Safe Autonomous Systems

Edited by

Elena Troubitsyna¹, Ignacio J. Alvarez², Philip Koopman³, and Mario Trapp⁴

1 KTH Royal Institute of Technology - Stockholm, SE, elenatro@kth.se

2 Intel - Hillsboro, US, ignacio.j.alvarez@intel.com

3 Carnegie Mellon University - Pittsburgh, US, koopman.cmu@gmail.com

4 TU München, DE, mario.trapp@cit.tum.de

Abstract

Autonomous systems rely increasingly on Artificial Intelligence (AI) and Machine Learning (ML) for implementing safety-critical functions. It is widely accepted that the use of AI/ML is disruptive for safety engineering methods and practices. Hence, the problem of safe AI for autonomous systems has received a significant amount of research and industrial attention over the last few years. Over the past decade, multiple approaches and divergent philosophies have appeared in the safety and ML communities. However, real-world events have clearly demonstrated that the safety assurance problem cannot be resolved solely by improving the performance of ML algorithms. Hence, the research communities need to consolidate their efforts in creating methods and tools that enable a holistic approach to safety of autonomous systems. This motivated the topic of our seminar – exploring the problem of engineering and safety assurance of autonomous systems from an interdisciplinary perspective. As a result, the discussions of achievements and challenges spanned over a broad range of technological, organizational, ethical and legal topics summarized in this document.

Seminar April 7–12, 2024 – <http://www.dagstuhl.de/24151>

2012 ACM Subject Classification Artificial intelligence, Dependable and fault-tolerant systems and networks, Embedded systems, Safety critical systems, Software and its engineering

Keywords and phrases ai, safety assurance, safety-critical autonomous systems, simulation-based verification and validation, software engineering

Digital Object Identifier 10.4230/DagRep.14.4.1

Edited in cooperation with Troubitsyna, Elena

1 Executive Summary

1.1 Introduction

The examples of modern autonomous systems include self-driving cars, UAV (drones), underwater vehicles, various industrial and home service robots. In general, autonomous systems are intended to operate without human intervention over prolonged time periods, perceive their operating environment and adapt to internal and external changes.

For example, a self-driving car gathers information from camera and lidar to detect, e.g., pedestrians on the road and plan collision avoidance maneuvers, slowing down or breaking, i.e., avoid hazards. The perception functions process the inputs of various sensors and generate the internal model of the operating environment. By relying on this model, the decision functions plan and execute the actions required to achieve the goals of the mission.



Except where otherwise noted, content of this report is licensed under a Creative Commons BY 3.0 Unported license

Methods and Tools for the Engineering and Assurance of Safe Autonomous Systems, *Dagstuhl Reports*, Vol. 14, Issue 04, pp. 1–17

Editors: Ignacio J. Alvarez, Philip Koopman, Mario Trapp, and Elena Troubitsyna



DAGSTUHL
REPORTS

Dagstuhl Reports
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

In general, they follow the generic “sense-understand-decide-act” behavioral pattern, which is also traditionally adopted in robotics.

Both sensing and decision making usually rely on Artificial Intelligence (AI), in particular Machine Learning (ML). While AI and ML algorithms have already been used in robotics for several decades, their use in safety-critical systems is fairly new and currently not appropriately addressed by safety engineering neither from technological, nor from organizational and legal points of view.

The problem of safe AI has received a significant amount of research and industrial attention over the last few years, but there has been a divergence in the approaches taken by the safety and the ML communities. Moreover, it has become clear that the safety assurance problems cannot be resolved by improving the ML algorithms alone. Hence, the research communities should consolidate their efforts in creating methods and tools enabling a holistic approach to safety of autonomous systems.

This motivated the topic of our seminar – exploring the problem of engineering and assuring safety of autonomous systems from an interdisciplinary perspective. A group of experts from avionics, automotive, machine learning, simulation, verification and validation and safety engineering reviewed the current academic state-of-the-art, industry practices and standardization to determine the latest achievement and challenges in developing and safety assurance for autonomous systems over a broad range of technological, organizational, ethical and legal perspectives.

As a result, the discussions of achievements and challenges in developing and assuring safety of autonomous systems spanned over a broad range of technological, organizational, ethical and legal topics.

1.1.1 Organisation of the seminar

The seminar brought together researchers and practitioners from different disciplines and application domains. Since, currently, the innovation in autonomous systems is strongly led by industry, a significant number of participants were industrial engineers, who not only shared their best practices but also identified unsolved research problems. In constructive debates, we discussed the results of applying and experimenting with various techniques for engineering safe autonomous systems and identified open research challenges.

To facilitate an open discussion between the participants, and analyze the problem of engineering safe autonomous systems from different points of view, before the seminar, we identified the following general discussion themes:

- Role of formal methods in engineering and assurance of safe autonomous systems
- Regulatory, assurance and standards for safety-critical autonomous systems
- Safety of AI-based system versus normal technical system safety
- Safety and security interactions
- Risk acceptance for autonomous systems

This report presents the summaries of the discussions focused on the specific topics within these themes.

We would like to acknowledge the supporting contributors – the session chairs and scribes that helped to collect the information for this report: Magnus Albert (SICK AG - Waldkirch, DE), Ensar Becic (National Transportation Safety Board, US), Nicolas Becker (Stellantis France - Poissy, FR), Simon Burton (Gerlingen, DE), Radu Calinescu (University of York, GB), Betty H. C. Cheng (Michigan State University - East Lansing, US), Krzysztof Czarnecki (University of Waterloo, CA), Niels De Boer (Nanyang TU - Singapore, SG), Lydia

Gauerhof (Bosch Center for AI - Renningen, DE), Jérémie Guiochet (LAAS - Toulouse, FR), Hans Hansson (Mälardalen University - Västerås, SE), Aaron Kane (Edge Case Research - Pittsburgh, US), Lars Kunze (University of Oxford, GB), Jonas Nilsson (NVIDIA Corp. - Santa Clara, US), Nick Reed (Reed Mobility - Wokingham, GB), Jan Reich (Fraunhofer IESE - Kaiserslautern, DE), Martin Rothfelder (Siemens - München, DE), Philippa Ryan (University of York, GB), Fredrik Sandblom (Zenseact AB - Gothenburg, SE), Stefano Tonetta (Bruno Kessler Foundation - Trento, IT), Kim Wasson (Joby Aviation - Santa Cruz, US), and William H. Widen (University of Miami - Coral Gables, US). In the spirit of Chatham House Rules that prevailed in the meeting, we are not attributing any particular written text to any particular person.

2

Table of Contents

| | |
|--|----|
| Executive Summary | 1 |
| Introduction | 1 |
| Role of Formal Methods in Engineering and Assurance of Safe Autonomous Systems | 5 |
| Regulatory, assurance and standards for safety-critical autonomous systems | 6 |
| Safety of AI-based system versus conventional technical system safety | 11 |
| Safety interface with security | 14 |
| Risk perception for autonomous systems | 15 |

2.1 Role of Formal Methods in Engineering and Assurance of Safe Autonomous Systems

2.1.1 Formal methods for AI-enabled systems

Model-driven and formal approaches are often used in the development of safety-critical systems. Hence, it is natural to discuss the perspectives of using formal methods in engineering safe autonomous systems. We observed that traditional formal methods, such as model checking, theorem proving, and static analysis provide not an “absolute” but rather a context-dependent proof of correctness and system safety. Moreover, they typically model certain core system functionality related to safety, i.e., do not aim at specifying the entire system behaviour. An open challenge is to understand the context surrounding the formal model and systematically represent it. Hence, an important research direction is to develop the mechanisms of safe context-aware incorporation of partial specifications. They should contribute to handling the complexity of autonomous systems.

There are different views on the goal of applying formal methods for engineering AI-enabled systems. It is commonly agreed that the scope of formal methods should be broadened to adapt them to systems engineering. We identified two most likely roles that formal methods will play in designing safe autonomy. On the one hand, formal models can be used to improve robustness of AI components, e.g., assess quality of data sets and facilitate generation of synthetical data. On the other hand, they can treat an AI component as a black box and focus on modeling system-level safety properties under uncertainty. To achieve the latter, the advances in probabilistic reasoning are required to express robustness of an AI component as a part of the entire system specification. We commonly agreed that formal methods do not seem to be a good fit for the high-dimensionality problems, i.e., it is unlikely that formal modeling of neural networks would become feasible. However, we already have had successful experiments with formal verification of decision making and path planning components. Hence, we concluded that formal methods can facilitate improving safety of the overall autonomous system by rigorously modeling monitors and specifying safety wrappers.

An important aspect to be considered is how to validate a formal model, i.e., ensure its appropriateness. In the context of ML-based components, we need to understand how to bridge a reality gap. Typically, a formal model describes the requirements to be fulfilled by a component semantically, while Deep Neural Network (DNN) recognises objects as a collection of pixels. Establishing a mapping between semantics of safety requirements and their actual implementation is an open challenge. Hence, further research is required to understand how to efficiently decompose system-level safety properties into properties to be fulfilled by neural network architecture and data sets.

An important aspect in formal modeling of autonomous systems is representing an open unconstrained operational environment. Since AI algorithms are data-driven and make decisions even in the situations that cannot be foreseen, verifying safety of such decisions currently constitutes an unsolved challenge. We concluded that it could be addressed by generation of a formal model based on the data about system behavior and then verifying its safety, i.e., using formal modeling for re-engineering and verification rather than verification-driven engineering of AI-enabled systems. Overall, we believe that formal methods will play a significant role in safety assurance although their use would be adapted and broadened to incorporate reasoning about system safety that is enabled by AI components.

2.1.2 Static and dynamic analysis of safety cases

A safety case presents and communicates risk as well as provides analysis. Hence, it plays an important role in the design and certification of safety-critical systems. It is typically used to support risk-informed decision making processes. Safety cases bring a vast amount of work products related to the design and development choices together into a cohesive narrative. Safety management through the lifecycle of an autonomous system is highly dynamic, with changing environments and continuous system improvement occurring through development and operation. In order to properly present this dynamic safety argument, safety cases for autonomous systems must also be managed in a dynamic way.

Dynamic updates to a safety case come in many forms. Safety performance evidence from field data and other ongoing analysis is updated throughout development and during system operation. Additionally, technical analysis and processes may be updated as part of continuous improvement of the system or expanding the operational domain or capabilities.

Dynamic analysis of safety cases is desired to manage efficient evaluation of safety cases in the face of constantly evolving safety evidence and argument. Automated and dynamic evaluation of safety cases can provide many benefits, including:

- Keeping pace with the quick release cycles of modern autonomous systems;
- Provide up-to-date status reporting of the safety assurance argument;
- Provide impact analysis capabilities to manage changes and incremental assurance;
- Minimize the cost and effort required by traditionally slow and highly manual evaluation.

A primary challenge in building methods and tools to support the dynamic evaluation of safety cases is that the structure and usage of safety cases is not standardized. Different domains, different uses (e.g., internal assessment, certification, and messaging to external stakeholders), and relevant stakeholders affect the desired format and needs of a safety case. There is a clear divergence between research focused on using argument structures as models for evaluation and computation and other work which treats the safety case primarily as a human communication tool. Research towards evaluation over the safety argument structure itself is often based on Goal Structuring Notation and other graphical argument notations, adding semantics for evaluation and analysis directly to the argument structure. From the “safety case as communications” perspective, these analyses are better treated as separate analysis concepts, which can be integrated into the safety case as evidence where needed to support communicating the desired properties.

The needs of safety case tooling and analysis methods heavily depend on the usage. We identified two main challenges in dynamic safety case evaluation: defining and aligning the industry with consistent safety-case related terminology and processes to support the value of common methods and tools and navigating the scope overlap between safety case analysis and other related safety and program-management based analysis.

2.2 Regulatory, assurance and standards for safety-critical autonomous systems

2.2.1 Safety metrics for ML-based functionality: requirements engineering aspect

The use of ML-based components to implement safety-critical functions differentiates autonomous safety-critical systems from “nominal” safety-critical systems. However, measuring the safety of ML components is notoriously difficult. First and foremost, safety is a system property, not an ML component property. Hence, defining “safety metric for ML” is a

contradiction per se. Thus, it is more appropriate to develop performance metrics for ML components. Correspondingly, there are two challenges associated with this problem: decomposition/composition of safety metrics and their interpretation. To address the former, a systematic methodology is needed to decompose and compose system safety metrics into ML component performance metrics and vice versa. This involves deriving a set of necessary and sufficient performance level requirements and metrics from the system level requirements and metrics. It also means that individual performance-level metrics must be composed into system-level metrics. It requires developing techniques enabling inference of system's safety from lower level performance metrics.

To address the latter, a comprehensive framework of safety metrics should be established. The metrics must be both measurable and interpretable, especially given the complex and sometimes unpredictable nature of ML systems. The metrics should be capable of measuring the completeness and appropriateness of requirements decomposition as well as assessing the satisfaction of specific performance requirements.

An open research challenge is developing mechanisms enabling a combination of black-box metrics, which reflect a component's or system's behavior without knowledge of its internal realization, and white-box metrics, which reflect the specific internal realization of a component or system.

We also underscored the need to address the problem of measuring residual uncertainties and how they can be handled by runtime monitoring. This allows for real-time adjustments and improvements, enhancing the safety and performance of the ML system.

A key point is to create a common understanding among different stakeholders that metrics should be assigned to requirements rather than components. We believe that this shift in focus from general components' properties to specific performance requirements is crucial.

Overall, while there are many challenges in defining and measuring safety metrics for ML and no universal solution. It is clear that these obstacles can be overcome with a systematic, engineering-focused approach and an emphasis on continuous monitoring and improvement.

2.2.2 Safety metrics for ML-based functionality: from AI-component to system-level metrics

The problem of measuring safety of ML-based components is currently considered to be a central one for assuring safety of autonomous systems. Hence, it was discussed in two parallel sessions. Though the sessions focused on different aspects, they had a common starting point – an acknowledgement that safety is a system-level property, which is not straightforward to decompose into component-level metrics. As a result it poses a question: what are the differentiating factors between leading and lagging safety metrics while defining safety at the component versus the system level. This highlighted the industry's inclination to treat AI training data as a competitive edge, complicating the evaluation of AI model performance and necessitating traceability in metrics.

Currently, there is no consolidated view on the problem of safety metrics of system architecture, which is correlated to the problem of modeling probabilistic nature of AI. The automotive industry tends to make emphasis on threshold values for widely agreed functional safety indicators (FuSa - ISO 26262) and safety of the intended functionality (SOTIF - ISO 21448). However, any attempt to define safety metrics for AI-based systems is cursed with dimensionality problems. SOTIF is notoriously ambiguous in the ML safety assessment. The discussion consensus was that explainability, repeatability and testability should be prioritized in safety metrics, despite the inherent challenge of establishing causality

in current ML approaches. Furthermore, there is a clear need for real-time metrics post vehicle deployment to monitor the internal AI-based systems performance and the overall vehicle safety (positive risk balance).

We also discussed the use of lagging metrics in the development of Autonomous Systems and the concept of positive risk balance, particularly within operational design domains (ODDs). The experts identified the lack of specification and consistency in ODDs in industry as a critical issue. Moreover, there is also the need to develop a common understanding of risk acceptance during ODD exits.

The most pressing problems in the field today are the definition of metrics for ML-based components of safe autonomous systems that are suitable for legal inclusion. Such metrics should enable and support regulation of the mass deployment of automated driving systems. This requires further immediate work on the relationship between ML performance metrics and overall vehicle safety, the explainability of ML metrics, and the need for cross-domain consensus on metric definitions. In-depth deliberations underscored the importance of focusing on measurable behaviors, the challenges of probing design defects in ML systems, and the societal implications of accepting AI-based solutions as potentially defective systems. The main take-away was a call to develop robust metrics that genuinely imply safety and address the complexities of translating AI-based component-level metrics to vehicle-level safety.

2.2.3 Explainability and understandability

The explainability of a complex system is recognised as a critical element in establishing its trustworthiness (for a variety of audiences). It is also essential for its efficient development. The concept of explainability has different aspects including error tracing or liability assignment. It was recognised that the approach may look and feel very different if the audience for explainability is an engineer, a safety investigator, senior management, the media or the public. Similarly, responsibility for communication of system explainability may reside with different roles within an organization depending on the audience.

The approach to explainability may also differ depending on its time sensitivity and purpose. Data collection and analysis for explainability following a safety critical incident is likely to be very different to that applied for focused engineering development activities or for achieving runtime explainability. In all cases, the resources (personnel, computing etc.) devoted to explainability have to be proportionate relative to the requirements. Further work is required to define what proportionate resource allocation means for achieving explainability of an autonomous system.

The concept of ‘Explainability by design’ was referenced as an important technique for satisfying these requirements. This implies using a system architecture that facilitates access to critical information to support the required explainability with modular or layered architectures likely to enable explainability more readily than ‘black box’ systems.

Explainability is a rather new concept that requires further deliberation and analysis. Among the key challenges to be addressed is the question of how to balance explainability-by-design (i.e., deliberate architectural design to maximize/optimize/set explainability) with performance of the end-to-end system (faster, more efficient). We also recognised that it is worth investigating different “layers” of explainability, i.e., would it be sufficient to merely consider, e.g., “reasonable reproducibility” or an ability to identify a root cause. Each “layer” has an impact on the practical resources and time that it would be required to provide an explanation.

We also concluded that currently there is no consensus on minimum requirements for

explainability in the context of its audience, purpose and time sensitivity. This work would require creating a consensus on a vocabulary for explainability that should be grounded in the common understanding of the possible deviations from the intended functionality and/or functional limitations

Finally, there is a clear need for work on integrating explainability of AI/ML into a broader category of explainability for autonomy/complex systems as well as the broader topic of safety governance and decision-making (including the safety assessment).

2.2.4 Bootstrapped safety approaches

The main concept behind bootstrapping is to gain increased confidence and range in the metric “number of miles without a crash”. This is achieved through predicting/extrapolating future performance through live testing as follows:

- Run X miles with safety driver (without crashes)
- Assume X miles with confidence interval
- Add Y miles without safety driver
- Extend/extrapolate number of miles without a crash

We observed this approach being used in industrial settings and hence, it is worth to analyze its ethical and technical issues. On the ethical side, we had a debate about whether it was appropriate or responsible to deploy the car without a safety driver when effectively performing live-testing. Some participants argued that the car could not be sold without demonstrating that it can be used without a driver, whilst others felt this was for marketing and a safety driver should always be used when building confidence/assurance. Additionally, the participants discussed how the risk could be communicated to the public or senior management effectively.

From a technical perspective it was noted that there would be many challenges to the validity of statistical data gathered this way. To analyse the problem, we assumed that the software version was identical across all the tests (although this may not be the case in reality). Obviously, a single crash upsets the statistics and to counter this the manufacturer may argue that particular crash does not count (e.g., it’s an extremely rare event or not the fault of the autonomous vehicle) or has been fixed in the next software build. It would also be difficult to aggregate different individual drives due to environmental conditions, level of traffic, confidence of the safety driver etc.

A proposed solution was to use a strong safety case as a “bayesian prior” and attempt to falsify the claims via the use of run-time monitoring. It was assumed that the safety case had a static structure and that appropriate/useful Safety Performance Indicators (SPIs) can be quantified, measured and monitored. At a sufficiently low-level, the developers can gather evidence that supports or counters different claims in the case more concretely. This leads to other challenges though, such as inter-relationships between evidence and how to aggregate confidence throughout a safety case.

A relevant topic in this respect is supplementing the evidence with simulations (which has its own challenges such as performance, repeatability and closeness to reality) and also engineering in safety mitigations and redundancies to reduce risk further.

2.2.5 Managing Uncertainty

Uncertainty has complex and multi-dimensional nature in the design, operation, and assurance of software-intensive safety-critical products and systems. Several taxonomies have been

proposed already. However, there is a need for the new one that would capture the aspects introduced by ML.

Complexity of managing uncertainty has increased as a result of introducing ML- based solutions for managing autonomous vehicles in open environments (aka public roads). Many projects adopt a technical perspective, but the problem is much broader, e.g., it includes assurance/organisational uncertainty. Useful existing taxonomies on uncertainty needs updating to capture this.

Autonomous systems broaden the causes of uncertainty in the design of safety-critical systems. The primary source of uncertainty is the system’s operating environment. We agreed that “unknown unknowns” are always going to be unavoidable for autonomous systems deployed in the real world and a lack of knowledge may lead to an unjustified sense of confidence in assurance. Different types of uncertainties can interact in unexpected, complex ways. Managing uncertainty involves finding trade-offs between safety, performance, cost, etc.

An open issue is to find appropriate mechanisms for managing uncertainty. For example, in controlled environments, such as factories, higher level (non-AI) safety-functions could be used to reduce the safety risks of underlying AI-based solutions. However, such solutions are not likely to be easily adaptable to a more open environment. The problem of managing uncertainty should also be considered from an ethical point of view.

2.2.6 Assurance and standards

Standards play an important role in engineering and assurance of safety-critical systems. They are based on the best practices and operational history of safety-critical systems. There are general, cross domain standards, like, e.g., IEC 61508 – a generic standard applicable to all kinds of electrical, electronic, and programmable electronic safety-related systems. There is also a large number of industry-specific standards, e.g., ISO 26262 addressing functional safety for road vehicles in automotive domain or DO-178C applicable to airborne systems and equipment and in particular, provides guidelines for the development of aviation software. A newer UL 4600 standard focuses on the safety of Autonomous Systems (AS). It aims at providing a framework for evaluating the safety of autonomous systems, including both hardware and software.

Developing comprehensive standards for autonomous systems is complex due to a number of factors. On the one hand, since autonomous systems rely on AI for its functioning, there should be some kind of general alignment between the standards specifically addressing AI and broader standards for autonomous systems. Since AI is a rapidly developing technology, balancing maturity versus freedom to innovate is one of the open issues. On the other hand, defining the appropriate scope and guidelines for application of standards for safety in the area of AS is also challenging because it should provide sufficient technical guidance and ease of checking at the same time.

Currently, the development of AS is driven by industry. Enabling cross-industry learning and sharing best engineering practices is desirable but not an easily attainable task. Ideally, it would be beneficial to distill the principles governing both product and process-related aspects of engineering and assuring safety as well as create technical guidelines addressing all stages of engineering of autonomous systems.

An important aspect is also training and education in safety standards. Standards should provide a technical guidance that can be implemented using various technologies and processes. Hence, compliance to the standards should go beyond merely checking-off safety requirements and focus on creating a safety management system. In particular, in the

context of autonomous systems there should be significant advances in creating guidelines addressing technologies and processes associated with recording and analysing data sets and connecting with testing and deployment data to enable continuous safety monitoring and improvement.

Ensuring the safety of autonomous systems is an evolving challenge, given the complexity and unpredictability of real-world environments. Advances in artificial intelligence and machine learning introduce new variables that traditional safety frameworks must adapt to address. Continuous development and refinement of standards, alongside collaboration between regulatory bodies, industry stakeholders, and research institutions, are essential for the safe integration of autonomous systems into society.

2.3 Safety of AI-based system versus conventional technical system safety

2.3.1 How AI safety differs from conventional technical system safety? Towards building a comprehensive fault model

For the conventional systems, safety and development assurance processes operate on the system-level functions. They support implementation choices within a system hierarchy. All these processes are based on a requirements tree. The process rigor and demonstration burdens levied on implementations are based on whether these implementations can cause or contribute to functional hazards, and upon the severity of those hazards. The choice to use AI/ML in an implementation creates as-yet open assurance challenges relative to the use of conventional hardware or software.

For conventional hardware and software, fault models have been developed that provide the basis for compelling validation and verification approaches. These are validated theories of operation that indicate what must be controlled and demonstrated. The assurance process seeks predictability of system behavior, and hardware and software submit to physical and logical laws respectively that allow assessment of their contribution to this prediction. Further, the physical and logical properties of hardware and software guide what to evaluate and test in order to identify and repair faults.

Learned AI/ML models resist both the physical and logical rationales for correct performance, and we as yet lack a theory of their operation sufficient to provide a basis for compelling validation and verification methods. Without such a knowledge and experience basis to indicate what to test and control, we lack a clear path to establishing confidence in associated system behavior. Our current inability to fully understand and predict the performance of these implementations therefore limits their application to functions for which we will accept the associated failure conditions. Such low-criticality applications should, however, be used as learning opportunities.

In order to achieve defensible process controls for the development of AI/ML implementations, we require a validated theory of what to control and why, based on an understanding of the contribution of the controlled factors to the performance of the implementation. This theory must also address the sources of incorrect performance, and the assessment of performance itself will need grounding, definition, and validated methods. It will be necessary, but not sufficient, to address properties of both the data and the inferences. A body of experience in low-criticality deployments should complement desktop analysis in evolving and validating such theory.

2.3.2 How AI safety differs from conventional technical system safety? Closing traceability gap

The use of AI technologies, in particular ML, for the realization of safety-critical functionality is challenging previous approaches to the safety assurance of software-based systems. In some respects, as safety is considered a system-level property, then the use of AI/ML for the implementation of individual components should not radically change systems safety engineering practices. On the other hand, the complex nature of the tasks implemented by AI/ML, as well as the strong reliance on training and verification data, requires adjustments to safety-critical development and assurance processes.

Requirements engineering for complex systems and environments was identified as a particularly relevant area because it covers a number of aspects related to the tasks for which AI is used as well as the specific impact of data-driven ML techniques. There was a feeling for the need for a well-defined scientific approach to the safety assurance of ML-based functions, including a foundational understanding of the objectives, challenges and demonstrable effectiveness of various approaches to ML safety. New approaches to requirements elicitation and analysis of AI-based, complex autonomous systems are required. These approaches should bridge the “semantic gap” between the engineer’s understanding of the task and environment (including the relevance of certain semantic features) and the actual syntactic features of the feature space that are used by the AI/ML components to make decisions. For many complex systems, this gap might only be closed through an iterative process of system validation and requirements refinement.

A general theme through all of the discussions related to the impact of AI on systems safety engineering was that in general more rigor must be applied to existing processes to manage the increased complexity of the environment, task and technology. However, specific additional methodologies are also needed to fill in technology-specific gaps. An example of which is filling traceability gaps between functional requirements specifications and data. We strongly believe that an organizational change management approach is required. Such an approach should improve existing systems engineering capabilities and introduce new capabilities needed for a new generation of systems and engineers.

We also discussed the perspectives of using AI in safety engineering processes. This requires addressing a number of fundamental issues related to the role of human judgment and reliance on automation in the safety analysis and engineering of complex systems. AI can be seen as an additional tool that should be used wherever helpful, but caution is needed to avoid automation complacency (blind reliance on the results of the AI-based analysis).

2.3.3 Safety architectures incorporating low-integrity ML components: quality assurance perspective

Integrity is an underlying concept of many standards governing safety. It comprises engineering systems that exhibit acceptable hazardous hardware failure rates, that have a sufficiently low probability of design flaws leading to hazardous failures, and are suitable for the application, i.e. safety of the intended functionality. The automotive domain highlighted the latter aspect in the SOTIF- ISO 21448 standard but still has no guidance as to how safe is safe enough (amount of residual risk).

The problem of ML integrity was discussed by experts from different domains – automotive, public transport and aviation – and different technologies – perception, testing, architecture and safety engineering. We focused on identifying the ML-specific challenges including algorithms implementing the models, parameters, data, training and testing.

The pace of innovation in ML field is increasing and many ML technologies do not have safety-related integrity arguments. We discussed the question: How can we build systems relying on AI or ML yet achieve a higher Safety Integrity Level (SIL). A related question is how to integrate no/low-integrity ML components into systems with high SIL?

The promising approaches focusing on quality-assurance include demonstrating safety of ML-based systems by thorough testing, applying rigor in training and testing with regard to data quality.

From the architectural perspective, safety envelopes and diversity were identified as the most promising approaches. As a basis for the discussion, we used a simplified generic architecture that should detect an obstacle with certain true positive and negative rates, and have some medium integrity level (e.g., ASIL B, SIL 2, ...). This functionality together with the safety targets is determined by the application requirements.

The architecture included two diverse sensors, path detectors and a fusion component. We identified the following necessary conditions for achieving the required SIL. First of all, error characteristics for sensing components should be well-understood and the fusion component must be designed to meet the perception level targets. Hence, application-specific ODD should be an important parameter. Moreover, there might as well be a checker which checks the output of the fusion against the data from the sensors.

Secondly, the system should be able to detect an "out of bounds" condition and have a strategy for encountering such conditions. We assume that an additional monitor (model scope monitoring) might be required. This may be implemented by using ML-based models.

Moreover, we identified the need for a metric for architectures with ML-based components that should be similar to the one, which the classical functional safety provides for hardware architectures. An assessment scheme is needed to gain confidence.

In all domains, in particular aviation, an architecture that safeguards ML components with classical components seems to be a way to gain confidence. However, this significantly reduces the advantages gained by the generalization capabilities of trained ML-based systems. Some experts believe that, in the foreseeable future, the integrity of such systems will be established experimentally rather than by a state-of-the-art for process rigor in training and testing. We also underscored that care should be taken that there is no claim of diversity with no proof of sufficient dissimilarity.

2.3.4 Safety architectures integrating low-integrity ML components: employing redundancy and safety monitoring

Another session on the same topic focused on discussing the problem of designing appropriate safety monitors for ML components. We discussed that currently, the engineers rely only on test evidences to argue about safety of ML components and there is no established framework for the process-based arguments. The experts agreed that the Potential architectural solution will be functionality-specific, i.e., depending on whether they are used for monitoring perception or planning. It is unlikely that it would be possible to design rule-based monitors for some ML components. Moreover, safety monitors might over-constrain the system.

An open challenge in designing monitors for ML components is arbitration, i.e., defining the principles for resolving disagreements between the monitor and the ML component. One potential solution could be based on a comparison to digital twins/HD-map in run-time. Another interesting solution is to rely on plausibility checks, e.g. by using dynamical models and relying on temporal consistency. Developing approaches supporting co-design of ML components and monitors also constitutes an interesting direction for future research. A

blue-sky research direction is to design AI-based safety monitors, though, in this case, it would be even more challenging to argue about system safety.

Finally, we discussed that in classical safety-critical systems, redundant architectures that combine components with low reliability result in creating reliable fault tolerant systems. An interesting research direction is to investigate whether redundant architectures combining multiple low-integrity ML components could also increase system integrity. Further research is needed to understand how to identify common cause failures and argue about integrity of redundant ML-based architectures

2.3.5 Test planning, validation and coverage

Since autonomous systems operate in an open dynamically changing environment, the question of verification, i.e., has the system been sufficiently tested becomes especially challenging. An overall goal of developing a system that would be “testable by design” has several challenges. Since autonomous systems continuously evolve, the development should be automated, i.e., test cases should be automatically generated from system specification. However, specifications themselves might be incomplete or inconsistent and hence, testing would inherit all the problems of them. One approach to address this issue is the use of AI to assist in test case generation. AI can analyze specifications to identify gaps and inconsistencies, suggesting potential test cases. However, this introduces challenges related to explainability and traceability.

An important aspect to be considered is also creating oracles that are appropriate for testing safety-related behaviour. An oracle in testing is the mechanism used to determine whether a system’s behavior is correct. Key questions include when to stop testing and how to define pass/fail criteria. These criteria should go beyond simple failure rates to include quantitative properties that comprehensively assess the system’s performance. For example, instead of just counting failures, the oracle could evaluate the severity of failures and their impact on overall system safety and functionality.

Some underlying metrics might be orthogonal to each other. For example, some safe behaviour might result in a significant performance degradation and hence, the question is how to define a trade-off between them. Another aspect is addressing uncertainty: how to appropriately approximate it and align with high-level system safety properties?

Since the system continuously changes and operates in an open environment, it is hard to define coverage criteria. There are different alternatives to focus on: requirements coverage, code coverage or structural coverage. The open challenge is how to ensure that verification coverage criteria are aligned with the validation criteria which focus on scenario and real-world coverage.

The discussion concluded that development and testing should be an iterative process. Based on test results as well as field data, specifications and test cases should be continuously updated, i.e., evolve in response to new information. An iterative process is essential for addressing emerging issues, adapting to new requirements and continual safety improvement.

2.4 Safety interface with security

Autonomous systems extensively rely on networking in their operation, which motivated the need to analyze the relationships between safety and security. We focused on discussing the critical aspects of automotive cybersecurity, the role of existing standards, and the interactions between safety and security functions within organizations

The discussion emphasized that safety and security are often siloed within companies, yet their integration is crucial, especially in safety-critical systems. The non-compositional nature of security versus the compositional nature of safety presents unique challenges in creating systems that are both secure and safe. There is a strong need to create methodologies that integrate safety and security from the design phase. This includes establishing practices that consider the dynamic nature of security threats while maintaining the integrity and reliability required for safety.

We also identified the key challenge for co-engineering safety and security – balancing the need for frequent security updates with the safety requirement for system. Hence, it is crucial to create management systems that can swiftly respond to new security threats without compromising system safety. We emphasized the need for systems that can adapt to evolving security threats. This includes designing architectures that can handle transient attacks. It is important to develop understanding of the long-term implications of such vulnerabilities on safety.

Automated driving is the most actively developed domain of autonomous systems and, hence, the discussion naturally addressed the issues in automotive cybersecurity. We highlighted vulnerabilities in automotive systems such as the CAN-bus and OBD-port. The necessity for robust cybersecurity measures to protect against and mitigate these threats was underscored.

There was a consensus on the inadequacy of current standards (ISO 26262, ISO 21434, UNECE R155) in providing clear methodologies for integrating security risks into safety assessments. The absence of precise application methods in standards was identified as a significant gap. As a result, the organizations have developed their own protocols, creating inconsistencies and potential vulnerabilities. Research should focus on creating unified, actionable guidelines that can be universally applied to assess the impact of the cyberattacks on safety and mitigate cybersecurity induced safety risks.

Since the use of ML is essential for the design of autonomous systems, it is also important to understand the impact of data poisoning and adversarial attacks on system safety. A key challenge in this respect is to reconcile the need for ongoing system updates to address new security threats with the traditional safety perspective of minimal changes for ensuring stability.

We emphasized the necessity for creating an unified approach to safety and security in autonomous systems. The discussions pointed towards the development of integrated safety-security architectures and the importance of continuous updates and management of both safety and security measures. There is a need for greater collaboration between safety and security teams within organizations. Ensuring these teams can work together effectively requires organizational changes and a better understanding of the interdependencies between safety and security.

2.5 Risk perception for autonomous systems

2.5.1 Ethical Issues in safety-critical autonomous systems

There is a broad range of ethical issues associated with engineering, safety assurance and deployment of autonomous systems. They include the personal ethics of engineers (conduct, due-diligence), ethics of corporate decisions (to deploy when things may not be fully assured), ethics built into the actual system (to reflect societal norms), and ethical consensus (via standards bodies).

We discussed personal and professional ethics and engineering consensus via standards. It was noted that there are many autonomous vehicles standards, but some of these are of poor quality. We observed that there were conflicting interests in standardisation committees between engineers and corporate voices. Therefore, it is important to be involved with standardisation groups, and thus secure some improvement of the quality of the standards.

Possible standards misuse was also discussed, whether that was doing activities simply to comply with requirements (using a checklist, rather than following processes to add value), or using them to influence policy makers and legislation. For example, SAE J3016 standard (Taxonomy And Definitions For Terms Related To Driving Automation Systems For On-Road Motor Vehicles) had been used to influence policy based on the erroneous assumption that safety risk increased in step with levels of autonomy (rather than highest risk in the middle). This significantly diminishes the benefits of the standard.

An integration of different aspects, e.g., such as ethics and security, which had previously been in separate silos was also discussed. We observed that the use of the term “non-functional” properties implies that they are less important than “functional” properties. It was felt that siloing such issues was not going to be effective for autonomous systems. However, engineering mechanisms to deal with them are not well addressed in safety-specific standards (e.g., bias, fairness, transparency). Additionally, standards may not address the increased level of risk that comes with autonomy and AI. Therefore, potentially there might be a need for greater integrity levels.

There is also a controversial issue of whether ethics in the industry was in alignment with societal norms, and what level of risk was considered acceptable. The experts noted that, in the past, much higher levels of risk were tolerated (e.g., in the early days of steam railways) and that different industries/domains have different expectations. This can also vary in different cultures. We discussed that humans tend to think about short term effects rather than long term, which can make discussions about reductions of accidents over several years difficult. “Goal zero”, i.e., no accidents was discussed as to whether it should be aspirational even if not possible in reality. In practice, there will be cost trade-offs, and addressing liability may be prioritised over maximising safety. For example, compliance to standards or regulations may be more important than improved design. We emphasized that there is a clear need for the well-grounded safety culture that would allow for avoiding blame and improving safety reporting.

2.5.2 How Can We Define “Safe Enough” for an Autonomous Vehicle?

A broadly accepted quantitative metric for assessing the safety of machine learning-based systems for autonomous driving has yet to be established. An adequate definition of safe enough for such systems must address the limitations of the evaluation methods. It must also encompass known, unknown, and uncertain factors present for pre-deployment assurance, post-deployment monitoring and incident evaluation. There is also a clear need to establish the metrics that enable public understanding and potential acceptance of the technology.

Using a reference human driver as the model for Automated Driving System (ADS) performance is intuitive, and might be effective for post-incident system evaluation. However, the feasibility of developing a complete model of a reference human driver for pre-deployment assurance is questionable.

A pure statistical average safety metric is likely to be insufficient. Additional considerations required will likely need to address at least the potential for risk transfer across different population demographics, the potential for negligent driving incidents, and mitigation of specific hazards identified both before and after deployment. Pre-deployment prediction of

ADS safety will likely suffer from significant uncertainty. An important question to ask when making a deployment decision is not only whether expected safety levels will be acceptable, but also whether the uncertainty of that expectation has such a large range that there is too high a risk of unacceptable safety outcomes, which might require additional data collection to reduce uncertainty pre-deployment.

The inherent uncertainty in determining ADS risk, along with the difficulty of creating explainable ML decision-making, might make it impractical to take a quantitative-only evidence approach for predicting ADS safety before deployment.

Exhaustive scenario testing demonstrating the capacity of an ADS to make appropriate decisions offers one approach to pre-deployment safety evaluation. However, the lack of a standardized set of testing scenarios makes evaluating scenario completeness difficult, with the potential for missing scenarios contributing to deployment risk uncertainty.

Comprehending ML decision-making is challenging, but the recording of tangible ML outputs and the inferences of “why” they might have occurred can still have substantial value. Understanding where in the ML process an error occurs (e.g., perception, decision-making) helps OEMs identify areas of improvement, and is critical in post-deployment monitoring and incident evaluation.

Post-deployment safety should include an absence of credible unacceptable risk, and in particular OEM should mitigate any identified specific risks. However, there should be an expectation that some level of uncertainty will remain as well as the potential for operational environment changes. As such, the assessment of safe enough must continue throughout the ADS lifecycle. This continuing assessment should rely upon incident information as well as continued ADS data collection.

Access to recorded data will be essential for crash investigators and regulators, as well as for continued assurance by OEM. The benefits of recorded data extend beyond observable incidents. A minor property damage crash has obvious and observable safety implications, but a misclassification of a detected object or a poor path plan that does not lead to an observable incident can also have substantive safety implications. Regulators might assess system capabilities beyond any incidents, and OEMs might identify problematic system areas for improvement.

Open issues and topics worth exploring further in this area include: determining concrete acceptance criteria for deployment and continued operation; the types of data and data collection mechanisms needed; comparative benefits and issues with obtaining data before vs. after deployment; defining a maturity framework for Automated Driving System development and deployment (e.g., akin to a Technology Readiness Level scale); explaining black-box decision-making functions within an ADS; defining and characterizing acceptable risk uncertainties; developing a publicly understandable but accurate safety metric for communicating with general audiences; and withstanding political and/or business-motivated decisions that are counter to acceptable safety.