Holistic Perspectives on Safety of Automated Driving Systems – Methods for Provision of Evidence

Magnus Gyllenhammar, Gabriel Rodrigues de Campos, and Martin Törngren, *Senior Member, IEEE*

***Abstract*—In recent years, the enormous investments in Automated Driving Systems (ADSs) have distinctly advanced ADS technologies. Despite promises made by several high profile automakers, it has however become clear that the challenges involved for deploying ADS have been drastically underestimated. This paper focuses on the challenge of providing sufficient evidence to support the safety claims of ADSs. The provisioning of such evidence clearly relates both to technical maturity of ADS systems (including actual experiences from deploying such systems), and on the development of methodologies for reasoning about ADS safety claims. Contrary to previous generations of automotive systems, common design, development, verification and validation methods for safety critical systems do not suffice to cope with the increased complexity and operational uncertainties of an ADS. Therefore, the aim of this paper is to provide an understanding of existing methods focusing on the development of a safe ADS and, most importantly, identifying the associated challenges and gaps. We present eight challenges, collectively distinguishing ADSs from safety critical systems in general, and discuss the existing methods in the light of these eight challenges. Based on this discussion, a set of research gaps are identified.**

***Index Terms*—Automated driving system, safety, safety assurance, holistic safety, evidence provision, research gaps**

I. Introduction

AUTOMATED DRIVING SYSTEMS (ADSs) promise enormous benefits to society in terms of increased comfort, safety and efficiency of the transportation systems. To achieve such benefits, it is essential to provide evidence that adequately supports the safety claims of the system, not least to ensure public acceptance. However, such evidence has, so far, proven difficult to compile, especially due to: (a) the uncertainties imposed by the operating environment and the ADS's interactions with other road users; (b) the fact that the ADS itself tends to require a complex set of interwoven functions and subsystems in order to take on the entire strategic, tactical and operational responsibilities of the driving task; and (c) the diversity and rareness of road accidents, resulting in high dependability requirements on the system to perform better than, or on par with, human drivers. These three aspects complicate the construction of a *complete* and *predictive* safety case.

M. Gyllenhammar and G. Rodrigues de Campos are with Zenseact

M. Gyllenhammar and M. Törngren are with the Mechatronics division at KTH, Royal Institute of Technology

The research has been supported by the Strategic vehicle research and innovation (FFI) programme in Sweden via the SALIENCE4CAV project (ref 2020-02946) and the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation.

Of course, safety cases were always required to be both complete and predictive. The differences for ADSs, however, are: (i) the practical infeasibility of collecting sufficient evidence from in-traffic testing alone [1, 2, 3]; (ii) the vastness of scenarios arising from (a) and (b); (iii) the reliance on machine learning in some safety critical parts of the system; and (iv) the industrial shift into agile development processes with frequent releases, in order to accommodate for shifting operational conditions (e.g. corresponding to more data being available on (ii)) and user needs.

There already exist several well established methods to design, develop, verify and validate dependable and safety critical systems in general. Notably, in the automotive industry, ISO 26262 [4] outlines a set of processes agreed to ensure sufficient functional safety of the Electrical and Electronic (E/E) systems of the vehicle. However, the ability of these methods and processes to provide sufficient safety evidence in relation to aspects (a) – (c) and (i) – (iv) remains to be shown. Below, examples of the plethora of methods available for providing safety evidence for ADSs are listed. Note, however, that this list is neither complete nor exhaustive, but represents a selection of particular pertinent topics for the benefit of our argument:

- **scenario-based testing, verification and validation**, addressing (i) and (ii), have been proposed to solve the validation and testing challenges [5, 6, 7, 8, 9, 10], where the scenario models can also be derived from naturalistic data or field tests;
- **statistical extrapolations** of non-crash scenarios, also denoted near misses, derived from collected data, addressing (i), can provide a means for leveraging field data to argument the safety of the system at higher integrity levels [11];
- **precautionary safety** [12, 13], ameliorating (a), is proposed for ensuring safe tactical decision-making despite uncertainty of adverse events;
- **formal methods and rules** [14, 15, 16], mitigating (ii) and potentially (iii) as well as (iv), have been suggested to solve the problem of safe decision-making;
- **Machine learning (ML) or Artificial Intelligence (AI)-based components**, i.e. (iii), the safety of which have been discussed in [17, 18, 19, 20, 21];
- **dynamic assurance cases** [19, 22, 23, 24], ameliorating the impact of unknowns from (a) and (b) in order to reduce the gaps in (i), (ii) and (iv), have been suggested

to cope with changes in the operational environment of autonomous systems and as a means to reduce the residual risk of operating the system through monitoring;
- **shifting portions of the assurance task from design-time to run-time**, addressing (ii) and coping with (a) as well as (b), has been proposed as an effective means to maintain assurance of the system while allowing for improved performance. Supportive concepts including ConSerts [25, 26], and Digital Dependability Identities (DDIs) [27, 28, 29]; and
- **Dynamic Safety Management (DSM) and Dynamic Risk Assessment (DRA)**, allowing the system to dynamically address (a) – (c) and support solving (i) and (ii), have been suggested as a means to allow the tactical decisions of the ADS to be made using appropriate run-time measures of risk [26, 30, 31].

It is important to highlight that none of the methods above is able to simultaneously address all of (a) – (c) and (i) – (iv). This makes it difficult to grasp the current progress towards a safe ADS and what gaps remain to be filled. Therefore, we aim to elucidate the abilities and shortcomings of existing methods in the state-of-the-art with respect to the particular challenges encountered when providing evidence for safety of an ADS. To support this discussion, eight challenges have been identified, in the light of which the aforementioned selection of methods, among others, are discussed. For the sake of clarity, the methods are separated into four main categories: design techniques, verification and validation methods, run-time risk assessment and run-time (self-)adaptation, as illustrated in the mind-map in Fig. 2.

The paper lays a foundation for a holistic perspective of safety of ADSs, highlighting what areas are presently addressed in the literature as well as what challenges remain to be solved. Furthermore, we discuss which methods can help to reduce the gap with respect to each of the eight identified challenges.

The contributions of our paper can be summarised as follows:

- **A holistic perspective** of methods contributing with safety evidence of ADSs;
- **Eight challenges** for providing safety evidence for ADSs;
- **A state-of-the-art review** of existing methods, in the light of the aforementioned challenges; and
- **Research gaps**, based on the three contributions above.

The layout of the paper is illustrated in Fig. 1. We start by defining the addressed research questions in Sec. II, followed by the delimitation of this paper in Sec. II-B, and the presentation of preliminaries and definitions in Sec. II-D. Sec. III, presents the challenges in providing safety evidence for ADSs. Design techniques are discussed in Sec. IV, and verification and validation methods are presented in Sec. V. These sections are followed by Sec. VI, where run-time risk assessment concepts are outlined, while Sec. VII covers methods for run-time (self-)adaptation. The results are presented in Sec. VIII, where promising methods to address each of the presented challenges are discussed. Each method's ability to address the challenges, or lack thereof, is collected in TABLE I.

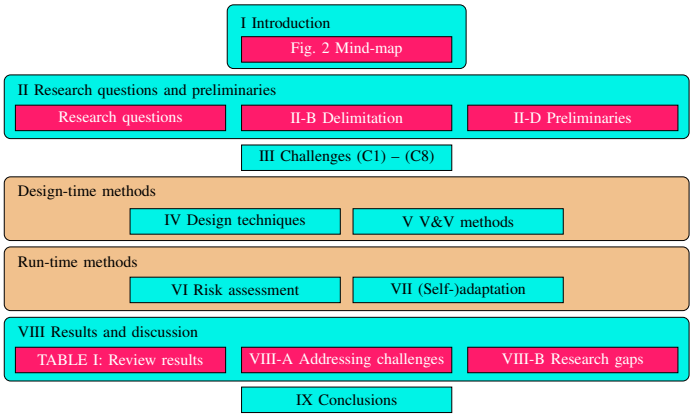

Fig. 1. The layout of the paper. Sections are illustrated in blue and subsections and visuals/contributions in pink. The yellow boxes group the different sections corresponding to the four method categories: design techniques and verification and validation methods and run-time risk assessment and (self-)adaptation respectively.

Future research avenues are given in Sec, VIII-D, while some concluding remarks are provided in Sec. IX.

II. Research Questions and Preliminaries

This paper aims at providing a foundation for a holistic perspective on safety of ADS. In order to provide a state-of-the-art review on existing methods, and identify pertinent technical challenges and research gaps on this topic, the paper is articulated around the following research questions:

RQ1: What are the present challenges for providing safety evidence for an ADS?
RQ2: What methods exist in the literature that support such evidence provision?
RQ3: How do these methods address, and how are they affected by, the challenges from *RQ1*?
RQ4: Based on the results from *RQ3*, what are the gaps in the state-of-the-art of *RQ2* considering the challenges of *RQ1*?

The answer to the first research question, *RQ1*, is given in Sec. III. Methods present in the literature are presented throughout sections IV – VII, addressing question *RQ2*. To answer *RQ3*, TABLE I collects our assessment of each considered method's ability to address the challenges of Sec. III. Lastly, the research gaps, answering *RQ4*, are presented in Sec. VIII-B.

A. A Note on the Structure and Content of the Paper

The mind-map in Fig. 2 is provided as a means to structure the content of and guide the reader through the discussions of this paper. It should be noted that the categories and the provided mind-map are not claimed to be exhaustive nor complete. Rather, it provides one way of organising the methods discussed and their interconnections. The methods depicted are collected under four main categories: design techniques, V&V methods, run-time risk assessment and run-time adaptation, each represented with its own section in this paper (Sections IV – VII). In more detail, the first two categories correspond to

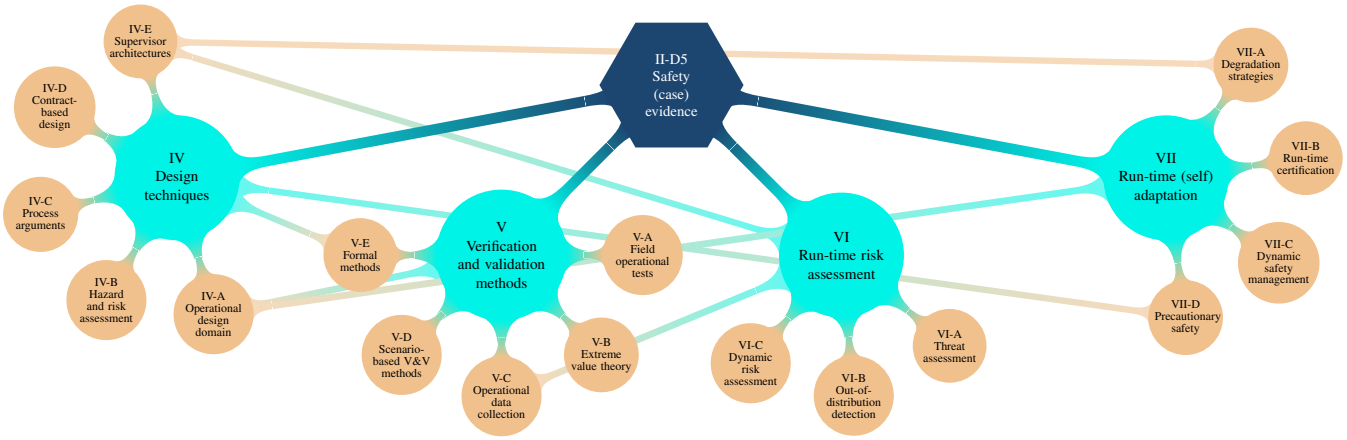

Fig. 2. A mind-map of the concepts discussed in this paper, grouped into four major themes, each supplying methods or evidence to the assurance case of an ADS. Note that this mind-map neither claims to be complete nor exhaustive, but represents a visual illustration of the concepts discussed in the paper and their interconnections.

activities and methods commonly included before deployment of the system. For example, one can see the design techniques as collecting the activities on the left side of the V-model (c.f. Fig 4), whereas the V&V methods correspond to the activities in the right leg. The run-time concepts do, however, not strictly fit within a classic ”waterfall” development cycle and are not covered by the activities depicted in the V-model. The two run-time categories instead aim to collect methods supporting evaluation, evidence provision as well as adaptation of the ADS in run-time.

Comparing to the Cyber-Physical Systems (CPS) framework by the National Institute of Standards and Technologies (NIST) [32, Fig. 4], the scope of this paper covers methods both within the conceptualisation and realisation facets. The design techniques covered in this paper fit nicely into the conceptualisation facet and the V&V methods are situated in the realisation facet, as are the run-time concepts (corresponding to *operating* the CPS/ADS). The aim of this paper is to elucidate the evidence that each method provides toward the assurance of the ADS, thus effectively providing a link between the first two facets, conceptualisation and realisation, and the third facet, assurance, of the CPS framework [32, Fig. 4].

It is worthwhile pointing out that several of the methods discussed in this paper could be positioned into two or even four of the categories, and a judgement of the the authors have been exercised in order to position each method in the section where the most relevant aspects for provisioning of safety evidence can be appropriately highlighted. For example, *supervisor architectures* are allocated to design techniques to highlight architectural aspects even though such architectural considerations are integral for both effective monitoring (pertaining to methods of run-time risk assessment) as well as run-time adaptation. Further, the application of supervisors is a design decision but also strongly supports the validation of the system.

B. Delimitation

The research and development efforts for the successful introduction and productification of ADSs are immense and include a wide variety of obstacles [33]. Nevertheless, in this paper we limit ourselves to the challenges pertaining to *safety* in the sense of functional safety (as per, e.g. ISO 26262 [4]) and nominal safety (e.g.Safety Of The Intended Functionality (SOTIF) [34]). We highlight technical areas and methods that provide evidence supporting the safety claims of an ADS.

In the interest of length and clarity of this paper and its contributions, we consider the following areas as *out of scope*:

- collaborative and communicating systems, for example: vehicle-to-vehicle (V2V), vehicle-to-infrastructure (V2I) and vehicle-to-everything (V2X) communications, despite the indicated importance for safe deployment of ADSs, e.g., [35],
- cyber-security,
- human-machine interface and interaction, including safety considerations on hand-over procedures and mode confusion, as analysed in, e.g., [36],
- the physical platform, on which the ADS operates, is assumed to be reliable (note however, that architectural patterns for e.g. fault tolerance are discussed),
- safety issues of automation complacency [37, 38],
- the question whether an advanced-driver-assistance-system can be scaled into a full ADS, and
- the impact of safety culture and the way the developing organisation is organised, as, for example, in the case of the Boeing 737 Max accidents [39].

Furthermore, aspects pertaining to tracking, organisation, and traceability of arguments and evidence of the safety case will also be excluded of the discussion of this work.

The delimitation of this paper could also be seen in the light of the levels provided in the CPS framework [32, Fig. 1], where we cover the innermost levels: device and system, but consider systems-of-systems and human interaction (at least in terms of the human being the user of the system, human traffic participants would naturally be in scope) as out of scope.

Note that the methods and references included in our work have been collected through exploratory search across multiple domains, stemming from the authors' own areas of expertise. Considering the high volume of research in this topic, the studied reference list could naturally be extended and complemented. The presented selection is nevertheless considered to be pertinent and representative of the existing solutions, and the arguments and conclusions made in this paper believed to be valid.

C. Related Work

Our paper focuses on a holistic perspective on technical methods providing safety evidence for ADSs. While autonomous vehicle technology and safety (assurance), in particular, have been approached in many different research works, no other work, to the best of the authors' knowledge, has conducted a holistic analysis of the existing methods providing safety evidence.

There are, however, several works pertinent to the topic of our work and worth mentioning here.

For instance, Nair et al. [40] review the state-of-the-art of safety evidence provision for safety certification across multiple application domains. While Nair et al. [40] use the term *evidence provision* to cover the following three different aspects: the information constituting evidence, how to structure the evidence as well as how to assess the evidence; we in our paper focus our discussions on the first and last aspect and, at least partly, leave the discussion of how to structure the evidence for future work. Further, whereas our paper aims to elucidate and discuss the contributions of different concrete methods towards the safety of ADSs, [40] gives an overview and classification of what information and artefacts that could be considered as evidence when fulfilling different safety standards. There are, nevertheless, some bridges between our paper and [40]. More precisely, the provided taxonomy of [40] relates to the four main categories of our work, which are nicely covered by the four leaf nodes of the *System Lifecycle Plan* category in the taxonomy of [40, Figure 2]. However, the methods covered in our paper are not only discussed in the light of what *Process Information* they provide, but also what *Product Information* that can be supplied.

Similar to our paper, Burton et al. [41] also discuss the importance of a holistic perspective for safety assurance for ADSs. However, the framework presented in [41] provides a complementary view on the problem to the one discussed in our paper. In particular, while our paper focuses on methods for providing safety evidence, Burton et al. address the causes for system complexity and exacerbating factors worsening the consequences of such complexity. For that discussion, Burton et al. include business context, development context and organisational aspects, while we restrict our analysis of technical methods for safety evidence provision.

Complexity of systems such as an ADS has also been acknowledged as a key challenge in [33]. The factors of CPSs complexity, in general, are discussed in [42]. We partly take support from the considerations presented therein, but restrict ourselves to ADSs and methods for safety evidence provision.

In the rest of our paper we draw upon a set of surveys to support our discussions of each of the included methods. While many surveys provide invaluable insight into their own respective domains and scopes, none cover the same holistic perspectives as are covered in our paper. In Sec. VIII-C a summary of which surveys we draw upon in each section of the paper is given.

D. Preliminaries and Definitions

The following subsections introduce the definitions of some central terms used throughout the paper.

1) ADS: An ADS is a system that performs on an SAE automation level 3–5 [43]. This entails that the ADS is completely responsible for the dynamic driving task, at least within a confined operational design domain (applicable for levels 3–4). Without venturing too far down the path of different architectural patterns and proposals for an ADS, it is worth noting that it is common to break down the system into sub-components. To have a common reference frame for further discussions, we consider the breakdown illustrated in Fig. 3. The decision and path planning is considered to be made in the *Decision making* block, that receives as input the perceived surroundings of the vehicle and the available capabilities of the platform. The output path is then used within the vehicle control block and ultimately executed on the vehicle platform. Note that this breakdown does not include monitoring or redundancy aspects, but merely represents the purely functional view of the system.

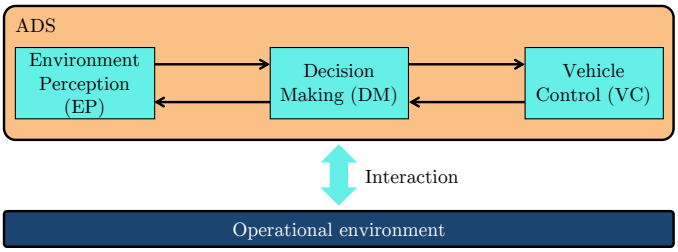

Fig. 3. A common breakdown of subsystems constituting an ADS. The Environment Perception (EP) block provides inputs to the Decision Making (DM) block, which requests a path to be enacted by the Vehicle Control (VC) block.

2) Safety and Risk: Safety is commonly defined as the *absence of unreasonable risk* [4, §3.132], where risk is understood as

$$R(x) = P(x)\ S(x),$$

i.e., the product of the probability $P(x)$ of an event x times the severity or consequence $S(x)$ should the event occur. If we split $S(x)$ into different levels, for example following the severity classification underpinning the Automotive Safety Integrity Levels (ASILs) of ISO 26262 [4], we can understand the requirement of *unreasonable risk* as a quantitative number representing the probability of occurrence of an accident with a given severity.

The acceptable levels of safety can be considered in relation to a positive risk-balance [44], or in relation to the driving performance of human drivers [1, 2, 45]. Junietz et al. [46]

provide an overview of quantitative risk levels from different industries and discuss these in relation to ADSs. Further, Warg et al. [47] propose the concept of a Quantitative Risk Norm (QRN), collecting quantitative safety requirements for an ADS. For the purpose of the discussions in our paper, we assume that quantitative requirements differentiating reasonable and unreasonable risks are present. It should be noted that quantitative risk metrics are a research topic on its own, but considered out of scope for our paper. Furthermore, we broadly interpret the term safety such as to encompass not only functional safety (e.g. ISO26262) but also Safety Of The Intended Functionality (SOTIF) [34].

3) Dependability: In a wider context, safety is just one attribute of the system's *dependability* [48], along with: *Availability*, *Reliability*, *Confidentiality*, *Integrity*, and *Maintainability*. While all dependability aspects are not considered in our paper we note that availability, reliability, and maintainability are attributes tightly linked to safety.

4) Fault Tolerance: In [48], four means to achieve a dependable system is detailed: fault prevention, fault tolerance, fault removal and fault forecasting. For the continued discussion of our paper we will consider fault-tolerance in particular. Such methods focus on increasing the reliability of the system, and in particular the ability of the system to tolerate certain types and frequency of faults. Following the dependability terminology of Avizienis et al. [48], a fault in the system (e.g. a software bug or inherent performance limitation of the system) might cause an error (an incorrect state in e.g. a software variable) which, in turn, might lead to a failure (at some level of the system, then with a potential continued fault propagation). The faults considered do not necessarily result in safety critical failures.

Further, for the discussions of our paper, we define a **Fault-Containment Unit** (FCU) as a unit, within which a fault is being contained [49, p. 155]. Such a unit (i.e. subsystem) should exhibit a defined failure at the boundary to its environment, and have its own software and hardware to contain the direct effects of an internal fault. Clearly the value of FCUs are higher if the separated units are ensured to fail independently.

5) Safety/Assurance Case: A **safety case**, an important concept for safety argumentation, is defined as *"a structured argument, supported by a body of evidence that provides a compelling, comprehensible and valid case that a system is, or will be, adequately safe [...]"*, see [50]. In the context of functional safety in the automotive domain, one could instead consider the definition of ISO 26262, defining it as *"[...] argument that functional safety is achieved for items, or elements, and satisfied by evidence compiled from work products of activities during development"* [4]. Yet another view of a safety case is to require the safety arguments (i.e. safety case) to provide justified belief in the safety of the system (e.g. [29]). Close to the latter, we instead refer to providing sufficient and appropriate evidence to support the safety claims of the system. For the sake of our discussions, the first definition would also be appropriate, particularly as it is gives a broader scope compared to the definition of ISO 26262, not limiting the evidence to be rendered during the development stages only.

Similar to a safety case an **assurance case** instead considers any requirement placed on the system, including dependability, safety and quality.

6) Design-time Activities: Traditionally, the activity of compiling safety evidence has been carried out and completed before the deployment of the system. In such context, all needed evidence to support the safety claim is collected throughout the specification, analysis, design, development, verification and validation of the system. For example, in the V-model, depicted in Fig. 4: the system is specified (Item Definition); a Hazard Analysis and Risk Assessment (HARA) conducted; the Functional and Technical Safety Concepts (FSC/TSCs) devised; and requirements further refined. The system is then designed, implemented, verified and validated. Following this process, which for example is prescribed by ISO 26262 [4], has proven to yield sufficient safety to most automotive Electrical and Electronic (E/E) systems operating today.

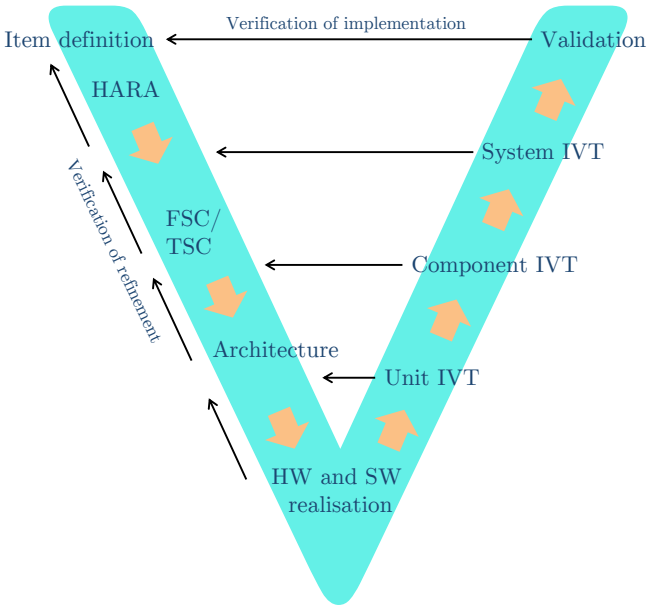

Fig. 4. The V-model as considered in the ISO 26262 [4] standard. The left-hand-side depicts the specification, design and implementation process, while the right shows the Integration, Verification and Testing (IVT) of the corresponding abstraction levels of the system.

III. The Challenges for Safety Evidence Provision

There are several traditional safety processes and concepts that have proven highly valuable and useful in the past, but that do not suffice in providing safety evidence for ADSs. This is partly due to (a) – (c) and (i) – (iv) discussed in Sec. I. Below we reformulate and elaborate on these aspects, presented as a list of eight challenges for providing safety evidence for ADSs:

Uncertainties:

(C1) **Uncertainties** associated with the **operational environment** of the ADS,

(C2) **Uncertainties** originating from the **interaction** of the ADS with other traffic participants,

Behavioural and structural complexity:

(C3) **ADS's responsibility** for all strategic, tactical and operational decisions of the driving task,

(C4) **Complex set of interwoven functions** and sub-systems required to realise an ADS,

(C5) **Self-adaptation capabilities** of the ADS, in particular, to cope with (temporary) degradations of the system,

Dependability requirements:

(C6) **High dependability requirements** imposed on the system, originating from comparison with human performance, highlighting the contribution of corner and edge cases to the overall safety,

AI and ML components

(C7) **Validation of (black box) components** relying on Artificial Intelligence (AI) and Machine Learning (ML),

Agile development and continuous deployment

(C8) **Frequent releases and continuous learning**, due to a shift from a sequential development process into an iterative and agile one including software upgrades, requiring reduction of safety/assurance case compilation efforts (or re-certification of the system).

While most of the challenges are pertinent also to safety critical systems in general (namely challenges $\{(\text{CI})\}_{\text{I}\neq 3}$), challenge (C3) stands out as a challenge particular to an ADS. Further, the challenges (C1) – (C8) collectively distinguish a representative view of the challenge for providing safety evidence for a class of highly automated CPS, acting in open environments, such as an ADS, compared to other safety critical systems.

IV. Design-Time Design Techniques

The way a system is specified, analysed, designed and implemented greatly contributes to safety evidence provision. In the following subsections we discuss some common design techniques and methods, as well as their limitations in relation to challenges (C1) – (C8) listed above. The reader can also refer to Fig. 2 for an articulation of the different methods and the corresponding domain areas.

A. Operational Design Domain (ODD)

According to [43], an ODD is defined as *"Operating conditions under which a given driving automation system or feature thereof is specifically designed to function[...]"*. It therefore provides the scope for the design intent of the system, and delimits the design-time activities. Specifically, it provides the context for the Hazard Analysis and Risk Assessment (HARA) and the conditions to consider when verifying and validating the ADS. In [51], the concept of the ODD is further elaborated in relation to its ability to support the safety argumentation of an ADS. [51] also presents four generic strategies able to ensure that the ADS remains inside its ODD while operating: inherent in ADS feature definition; checking mission when accepting strategic task; statistically defined spatial and temporal triggering conditions; and run-time measurable triggering conditions related to operating conditions. The first three strategies pertain to the relationship between the strategic mission and how it is accepted by the ADS. The third additionally, together with the fourth strategy, imply the need for run-time monitors of relevant triggering conditions. Note that the former aspect is tightly linked to the concept of Minimal Risk Condition (MRC) and Restricted Operational Domains (RODs), which are further discussed later in this paper in Sec. VII-A.

While the ODD defines the design intent and the scope of the V&V activities, thus specifying what can be called as the problem domain, it is difficult to ensure that this design intent adequately captures all operational uncertainties of challenges (C1) and (C2). However, an appropriate specification of the problem domain, provided by the ODD, can simultaneously help avoid certain aspects of the same challenges by explicitly avoiding or limiting certain uncertainties. Furthermore, it helps alleviate some of the difficulties related to challenges (C3), (C4) and (C6). As the ODD explicitly define the operating conditions, for which the ADS is designed and verified, it can be matched to the operating conditions required by the intended real world use cases, as suggested in [51]. Consequently, the use of the ODD could help facilitate incremental improvements (part of challenge (C8)) as well as to cope with restricted capabilities of the ADS (formulated in challenge (C5)). The latter aspect, referring once again to the concept of RODs, is later discussed in Sec. VII-A. One major challenge in the use of an ODD is how to ensure that the information is distributed and appropriately manifested throughout the system and the development cycle, in order to strengthen the evidence for completeness of the V&V activities and assurance of the system. Rather than addressing this challenge, many research works have focused on monitoring of the functional boundaries of the ADS, i.e. the limits within which the function is intended to operate (as defined by e.g. its ODD), e.g. [52, 53]; the definition of an ODD for the ADS, e.g. [54]; and what dimensions to consider in such a definition, e.g. [55, 56, 57]. Recently, BSI [57] published a set of considerations and taxonomies for the construction of an ODD. When it comes to relying on ML-based AI components (related to challenge (C7)), the ODD might ameliorate some of the issues by concretely defining what is inside the operational domain and what is not. This entails giving a formalised means for out-of-distribution detection, a concept elaborated upon later in Sec. VI-B.

B. Hazard Analysis and Risk Assessment

When combined, the challenges presented in Sec. III make the process arguments of ISO 26262 insufficient [4]. The Hazard Analysis and Risk Assessment (HARA) is traditionally made through a manual effort, where all hazards and the associated risks are identified. Regarding (future) ADS, this is, however, no longer tractable considering challenges (C1) – (C4). Indeed, challenges (C1), (C2) and (C4) [58] make the enumeration of *all* hazards difficult (if not impossible). Furthermore, challenge (C3) highlights the ADS responsibility,

where the autonomous system should have the ability to mitigate hazards it might face even before they occur, therefore impacting the applicable hazards as well as the associated risks. The complexity of the ADS (related to challenge (C4)) also impose an obstacle to the safety-goal breakdown following the HARA activities.

The effectiveness of a safety or risk analysis technique is not clearly quantified, as discussed in e.g. [59, 60], also implicitly suggesting a correlation between the results of the analysis with the availability of experts with appropriate domain knowledge. For a novel system, such as the ADS, it is obviously difficult to gather such a collection of experts. Further, even for relatively simple systems, such as an Autonomous Emergency Braking (AEB) system, the two different hazard analysis techniques, System Theoretic Process Analysis (STPA) [61] and Failure Modes and Effects Analysis (FMEA) [62], have been shown to be insufficient for identifying all hazards [63].

There has, however, been several suggestions on how to bridge these gaps. For instance, Kramer et al. [64] suggest a method for iterative and data-driven identification of hazards for ADSs. This said, such a method still falls short with respect to achieving completeness of the set of hazards. In another work, Khastgir et al. [65] suggest a run-time alteration of the Automotive Safety Integrity Levels (ASILs) associated with each hazard, in the light of the tactical decisions made by the ADS, providing also as a method to guide those tactical decisions. This method relies on a high integrity *hazard detection system*, and consequently it does not address the completeness of the hazards. In [47], a tailoring of the HARA process is suggested by using a Quantitative Risk Norm (QRN) with consequence classes, and thus relieving some of the burden of achieving a complete enumeration of hazards. The question of how to collect ”sufficient” evidence to support the ADS’s fulfilment of such a QRN, however, is still under debate.

For the purpose of the HARA, ML-based components (related to challenge (C7)) could be considered as any other subsystem [66]. However, some particular challenges arise in relation to classification accuracy and adversarial attacks, and that is why Salay et al. [66] suggest an alternative analysis method called Classification Failure Mode Effects Analysis (CFMEA).

As the current HARA process relies on manual intervention, it would be challenging to match with high cadence releases within an agile development process and the impact from incorporation of new evidence through continuous learning is also unclear (i.e. challenge (C8)). However, the process itself does not need to be manual, but a solution to overcome that is yet to be defined. As for challenge (C6), regarding the high dependability requirements, it is currently difficult to assess how the HARA will be able to ameliorate this aspect considering the diversity of relevant events to consider. After the first successful deployment of an ADS, and once sufficient operational data becomes available, this might, however, no longer present itself as a challenge, as the completeness of scenarios and events underpinning the HARA could then be deduced from the collected data itself, e.g. following the approach of [64].

C. (Qualitative) Process Arguments

The outcome of the HARA and FSC/TSC steps in the V-model, e.g. of ISO 26262 [4] as depicted in Fig. 4, is a set of ASIL requirements allocated to a collection of subsystems or components. In the ISO 26262 [4], a set of requirements including qualitative process arguments are prescribed to ensure the fulfilment of the ASILs. Even though such qualitative processes seem to jointly work for less complex systems, the exact quantitative contributions from each risk reduction method are not fully known. In fact, this holds true for the entire safety case approach, for which it is hard to prove its overall effectiveness and quantitative contributions to safety, as discussed in [60]. Regarding challenge (C4), as the complexity of the system grows, so does the number of process arguments. Hence, if all proposed processes leave a shard of residual risk, this might eventually amount to a considerable contribution when compiling the entire complex ADS. One way to circumvent this would be to transition from a focus on *qualitative* processes into a focus on *quantitative* ones. If one is to consider safety in a quantitative sense, however, there is also the need for the top-level claims to be prescribed as quantitative targets, for example according to a QRN [47]. Similar to the HARA, process arguments struggle with encapsulating the uncertainties imposed on the ADS through challenges (C1) and (C2). Further, challenges (C3) – (C6) could, in principle, be supported by process arguments, though the quantitative contributions would then need to be better understood.

Traditional development processes also do not suffice for tackling challenge (C7) related to integration of ML-based components, even if some steps towards this direction have been done recently, as surveyed in Rabe et al. [67]. Notably, a W-model for learning assurance is suggested in [68, Fig. 6.1, p. 43], which might provide a stepping stone towards a design process for learning-based components. Assurance of Machine Learning for use in Autonomous Systems (AMLAS) [20] also provides guidance for how to incorporate ML-based components, by providing safety case patterns and adjoined processes for systematic integration of safety when developing such components. Nevertheless, how to merge process arguments and traditional development processes with agile development (so to address challenge (C8)), especially in relation to safety-critical systems, is still an open challenge [69].

D. Contract-Based Design

Similar to the Hoare triple [70], Contract-Based Design (CBD) suggests expressing the interactions between elements (systems and components) in terms of contracts, expressing the *preconditions* that each element expects and, under which, the element can provide the *postconditions*. Given a suitable formalism, these contracts can be implemented and monitored at compilation, configuration or execution time, where a failed assertion of the contract would result in an exception. In [71], Benveniste et al. provide a formalisation of *Assume-Guarantee* (A/G) contracts for system design, describing the preconditions (assumes) and postconditions (guarantees) for the system

elements. A simple example of such contracts, allocated on component level, is depicted in Fig. 5.

CBD for safety critical properties, also termed safety contracts (i.e. contracts to encode safety-critical properties of the system), has been proposed, e.g. [72, 73]. Also highly configurable systems have been considered, where in [74] the use of CBD to assure an entire product-line is explored. A/G-like contracts is notably also the approach of Digital Dependability Identities (DDIs) [27] and ConSerts [25], which we discuss further in Sec. VII-B. Furthermore, Warg et al. [75] suggest using contracts in all abstraction levels of the ADS in order to achieve a *continuous assurance case*, thus mitigating challenge (C8). The interested reader is referred to [76], where these methods are discussed more in detail with respect to their potential contributions to safety assurance in the scope of ADSs.

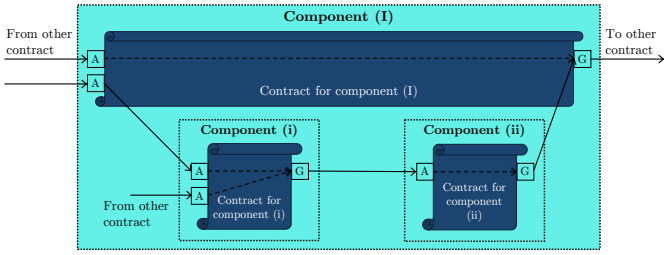

Fig. 5. A simplified example of contracts allocated to components (I), (i) and (ii). If the *assumes* of the components are fulfilled they can *guarantee* the output.

As CBD provides a clear interface between system elements, it effectively supports modularity of the elements and consequently ameliorates challenge (C8), pertaining to frequent releases and continuous learning. However, the complexity of the ADS (formulated in challenge (C4)) raises the question of the scalability of the approach for highly interwoven functionalities and complex systems. Further, the defined contracts require formalised specification of the assumes and guarantees, which, considering challenges (C1) and (C2), might be difficult to achieve on the boundary of the ADS towards its environment. When using machine learning-based black-box components (e.g., challenge (C7)), the difficulty of applying contracts is even greater, as small perturbations to the inputs might lead to large perturbations in the output [77], which requires highly dependable systems for monitoring and out-of-distribution detection. Finally, encoding formalised contracts for the tactical decisions of the ADS (related to challenge (C3)), such that they fulfil the safety requirements, will be a considerable challenge.

E. Supervisor Architectures

The architecture of an ADS is crucial for reaching dependability targets (e.g. [78]). The generic and general layout presented in Fig. 3 provides a basis for common functionalities that need to be realised in an ADS. This generic view can also be deduced from more complex approaches, such as the ones analysed in [79] or the functional architecture proposed in [44, p. 68, Figure 27]. Common to the discussion of ADS architectures seem to be the usage of monitoring or surveillance capabilities for reliability [44, 78, 79, 80, 81, 82, 83]. Such capabilities are further discussed in this section. Discussions on particular metrics and ways to assess the risk incurred by the system during run-time are, however, deferred to Sec. VI. Similarly, detailed discussions on how detected limitations and faults should be handled through degradations are postponed for Sec. VII.

The ISO 42010 standard [84] defines "architecture" as *"fundamental concepts or properties of a system in its environment embodied in its elements, relationships, and in the principles of its design and evolution"*. Architecture design involves deciding on these principles and the mapping of behaviours to the components and their organisation, with the purpose to meet requirements posed on the system and dealing with the involved trade-offs [85].

For an ADS, architectural design can be seen to have the goal to realise sufficient dependability while reaching performance, cost and scalability targets, subject to further constraints such as energy consumption, space, etc. (recall Fig. 4 for the position of the architecture design as part of the development process).

An ADS (its functions, software and hardware), will be subject to failures and unintended behaviours in several ways (related to challenges (C1), (C2) and (C5)), relating to functional safety (faults in HW and SW), safety of the intended functionality (performance limitations and an incomplete understanding of the environment and how it interacts with the ADS). Moreover, the high dependability requirements (i.e. challenge (C6)), and the fact that an ADS in general does not have a fail-safe state, see e.g. [78, 81], leads to the question what an appropriate highly dependable, yet cost-effective architecture looks like.

Key elements part of ADS architecture design, include the considerations of relevant fault models (or fault hypothesis), suitable patterns, where to deploy error detection mechanisms, how to contain failures and how detected errors should be handled. It appears generally accepted that traditional fault-tolerance concepts such as triple modular redundancy (as applied in flight controls [86]), would be too expensive and not able to deal with common cause failures [78, 81]. The appropriate level of redundancy and diversity required for cost-effective designs remains an open issue, receiving increasing attention in both industry and academia, with many current proposals. Further open research challenges include how environment perception sensors could potentially be shared between different channels of the system (for cost-efficiency) and the level of independence of such channels (relating to potential common-mode failures) [81].

A common solution for realising a supervisory architecture is using a nominal and a supervisory/safety/fallback/high assurance channel [78, 81, 83, 88], an example of which is shown in Fig. 6. The idea is that a high-performance system (possibly with low dependability) is monitored (by a high dependability component) and the control is, when necessary, handed over to a supervisory channel (also of high dependability). For this solution to support an ultra-dependable system [33, p. 2], i.e. fulfilling challenge (C6), Kopetz [78] stresses the importance of each subsystem to be its own

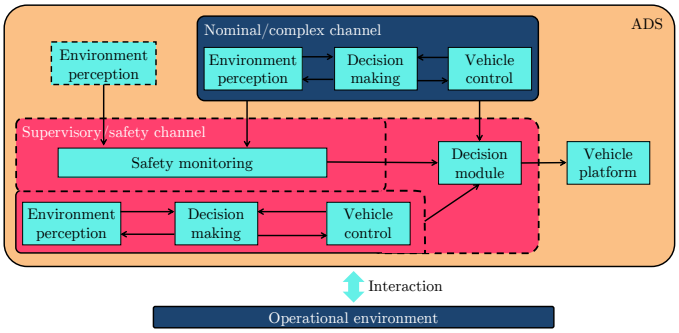

Fig. 6. Version of the simplex architecture [87] in the ADS context. A nominal channel and a safety channel run in parallel, a monitor of the nominal channel and a decision module is tasked with switching between the two based on the input from the monitor. Both the monitor and the decision module could optionally be allocated inside the safety channel, as suggested in [81], or as a separate components, as recommended in [78].

Fault Containment Unit (FCU). Monitoring and deciding when to switch to the supervisory channel, represent an essential ingredient in such solutions [44, 78, 79, 80, 81, 82, 83].

Detecting failures is non-trivial given the inherent uncertainty and risk in ADS operation (again, related to challenges (C1) and (C2)). Deriving supervision mechanisms (constraints, rules, etc.) for detecting failures represents an important area, (see [89, 90]), needing more research. The definition of such mechanisms, and specifications thereof, face some of the same challenges as those enumerated later in relation to the verification and validation methods of Sec. V, especially when deploying formal techniques [88] for realising the supervision. In particular, formally capturing conditions and scenarios from challenges (C1) – (C3), and ensuring fulfilment to the high dependability required in challenge (C6) pose an obstacle. Further, challenge (C4) makes it difficult to anticipate all possible system states.

While the vast number of possible degradations (related to challenge (C5)) pose a challenge for the implementation of a supervisor, supervision is also the only solution allowing for appropriate adaptation accounting for the degradation.

However, despite these limitations, surveillance architectures still help ameliorating challenges (C1) – (C3), and are integral in handling challenge (C5), i.e. system degradations. Further, by deploying anomaly and Out-Of-Distribution (OOD) detection (as discussed further in Sec. VI-B), ML-based components (related to challenge (C7)) can be effectively supervised. Lastly, appropriate supervision capabilities might ease the requirements on the assurance efforts done before deployment of the system, and thus ameliorate challenge (C8) by reducing time-to-market for each ADS version.

As a final note on supervisory architectures, Jha et al. [21] give an alternative approach for supervising the environment perception block using *predictive processing*, where the quantities monitored are the deviations of each sensor measurement with respect to the constructed (internal) world model. The approach is still conceptual and a concrete implementation is yet to be provided, but it seems promising as such a system directly incorporates redundancy in the sensor processing. This, they argue, could also provide a way forward to trust and rely on black box algorithms, such as ML-based components [21], i.e. addressing challenge (C7).

V. Verification and Validation Methods

Verification and Validation (V&V) is an integral part in providing evidence of the safety of a system, not the least with respect to the system's fulfilment of its specifications. Riedmaier et al. [10] give a comprehensive overview of existing methods for V&V of ADSs, focusing on methods using scenarios in the assessment, but also give an overview of safety-assessment approaches in general. Complementing this work, Wishart et al. [91] also present a comprehensive list of current V&V activities within industry and academia.

The presupposition for the V&V is the existence of a system (or at least an implementation on some abstraction level, e.g. a sub-component) and a specification that the system is expected to fulfil. In terms of supplying evidence for the assurance case of the system, there are four general caveats regarding the V&V activities:

- The completeness of the specification vis-à-vis the real operational use case,
- The verification might not exhaustively verify the system with respect to the specification,
- The scalability of the method to enable coverage of verification of the complete ADS system with respect to its specification, and
- The complexity of the ADS will also propagate to the V&V methods and tools, implying enormous efforts to develop and ensure that these can be trusted, referred to as "tool qualification" in functional safety standards such as ISO26262.

The impact of these four caveats on the remaining residual risk, after the complete V&V processes/activities, is yet an open research question and one which, in the light of challenges (C1) – (C4) and (C6), impacts the overall assurance of the ADS.

In the sequel, we explore some of the approaches to V&V, as illustrated in Fig. 2, and discuss their limitations in the light of supplying evidence for the safety assurance of an ADS.

A. Field Operational Tests

In order to test a system in its real operational conditions, one can make use of Field Operational Tests (FOTs). This is arguably the validation method that gets the system closest to the real operating conditions, therefore capturing the uncertainties related to challenge (C1) most accurately. Systems that do not provide tactical decisions, such as for example Autonomous Emergency Brake (AEB) or Lane Keeping Assist (LKA), or (sub)systems merely providing input to tactical decisions, such as the perception system, are possible to evaluate in *open loop*. This entails running the system passively in a vehicle (but not intervening), that is otherwise manoeuvred by a human driver (or another system). This approach is sometimes also referred to as *shadow-mode testing*. From an ADS perspective, while this could alleviate some validation gaps (corresponding to challenges (C1), (C4), (C5), and (C7)), understanding the behaviour of the full system requires the evaluation of the

system in *closed loop*. By allowing the actions of the ADS (formulated in challenge (C3)) to be enacted one can also measure the ADS's interactions with its environment (corresponding to challenge (C2)). From a safety perspective, closed loop verification is difficult, as it might be dangerous to rely on safety drivers as backup due to human performance issues, such as automation complacency [37, 38]. Furthermore, it is also costly and difficult to achieve such testing at the scale required to provide sufficient evidence in relation to the high dependability requirements prescribed for an ADS [1, 2, 3], therefore failing to address challenge (C6). The feasibility of such a V&V method is also questionable considering the higher cadence releases resulting from an agile development process, or the sought continuous learning cycle of the system, pertaining to challenge (C8). Collecting operational data from the field, as discussed later in Sec. V-C, is closely related to FOTs, and further to the supervisory architectures discussed in Sec. IV-E. Such operational data supports the development of an ADS in several different ways, in particular for:

- characterising the Operational Design Domain (ODD) of the system [51], and consequently the specification of the system,
- extrapolating the performance of the ADS through, for example, the use of Extreme Value Theory (EVT) models [11] (as discussed in Sec. V-B),
- supplying operational data to be consumed for consecutive releases of the ADS [13] (as discussed in Sec. V-C, and
- serving as a baseline data set for scenario-based testing and evaluation [6, 7, 92] (discussed in Sec. V-D).

A non-exhaustive list of large-scale FOTs is provided by Batsch et al. [93, p. 4, Table 1].

B. Extreme Value Theory

Extreme Value Theory (EVT) focus on modelling the tails of a probability distribution by considering the "extreme" events present in data (potentially provided by FOTs). In the context of validation of ADSs, one could consider different types of threat measures for the purpose of providing validation evidence of the ADS. These threat measures are used as a proxy for estimating the risk of the operational situation faced by the ADS. Based on field data, such threat measures can be calculated and the extreme events modelled through EVT artefacts. Åsljung et al. [11] present an EVT analysis of field data for the two threat measures: Break Threat Number (BTN) and Time-to-collision (TTC). The proposed method does not require detailed models of the system itself and its operational environment, therefore alleviating challenges (C1) and (C4) – (C5), which is a prerequisite for the other V&V methods discussed in this section. Furthermore, EVT approaches also alleviate challenge (C6) by extrapolating the ADS's performance from the data available, e.g. from FOTs, therefore allowing for inference on the integrity of the system, beyond the data collected. Since the system operates in its true environment, the validity of the EVT approach is high with respect to the data collected. However, the results are dependent on the threat measure used [94], which might impact the validity of the results provided with respect to the ADS's actual failure rates in real traffic. It should be noted that this could impose a major limitation to the usefulness of EVT if an appropriately predictive threat/risk metric cannot be selected. It is also paramount that the data collected through the FOTs is representative of the actual operating conditions of the ADS. Thus, EVT faces the same challenges as FOTs with respect to ensuring safety while testing the system in closed-loop (i.e. to assess challenges (C2) and (C3)). In terms of challenge (C8), pertaining to agile development and accommodating continuous learning, the EVT approach helps ameliorating parts of these challenges by leveraging collected data to infer the systems performance level beyond the operational hours used to collect it. However, the reliance on data is detriment to the method's ability to support frequent releases.

C. Operational Data Collection

Despite the best efforts to design and develop the ADS, it will nevertheless be essential to monitor the ADS in operation to ensure its safe operation. Not the least, to mitigate uncertainties in relation to the ADS's interactions, as per challenge (C2), and its operational environment, regarding challenge (C1). For the purposes of our discussions we can distinguish between three types of monitoring capabilities:

(i) monitors for the collection of operational data (e.g. similar to gather data from FOTs, see Sec. V-A above),
(ii) monitors for assurance reasons and inhibition/recall of the system, contributing to the containment of the residual risk and thus being part of the V&V approach for releasing the system in the first place, and
(iii) run-time supervisors to ensure that the ADS is operating safely according to its ODD and according to its current operational capabilities in relation to the requirements of the current strategic mission.

The first two (i) and (ii), are tightly linked to the V&V of the system. While (i) refers to the general case of data collection for (off-line) modelling, (ii) refers the monitoring of specific indicators for the purpose of assurance processes carried out centrally, i.e. not in the vehicle itself. Both these aspects are further elaborated in this section. The last point (iii), however, is discussed more in detail in Sec. VI, as it provides input to the operational decisions in run-time and is therefore better suited for discussion in the context of run-time risk assessment.

In [76], several of the assurance approaches discussed require (ii) monitoring of the system, its operational contexts and/or some Key Performance Indicators (KPIs) [22, 23, 24]. Concrete measures and metrics underpinning such KPIs are elaborated upon in Sec. VI-A. In the dynamic safety case concept of Denney et al. [22], it is acknowledged that the monitoring is to be done both inter-mission, corresponding to our second category above (ii), as well as intra-mission, corresponding to (iii).

Asaadi et al. [23] suggest to monitor certain indicators that can be analysed to identify trends and shifts, and trigger an update of the assurance case. Similarly, in the UL 4600 [95] standard, it is suggested to monitor Safety Performance Indicators (SPIs) of the system. These leading measures (should)

give predictive indication for when the system might operate unsafely and spur appropriate actions to mitigate the risks. The main purpose of monitoring such indicators is therefore to contain the residual risk related to challenges (C1) – (C5). This type of indicator monitoring could be further enhanced by the use of Extreme Value Theory modelling of the SPIs/KPIs, as suggested in [96], helping to compare sparse data to the high dependability requirements of the system (challenge (C6)).

The first type of monitoring (i), on the other hand, particularly supports increased release cadence as well as promoting a learning cycle (related to challenge (C8)) by improving the basis on which models, analysis and design are founded. Continuously adding new data from operations also provides basis for retroactively fulfilling a statistical proof of the high dependability requirements, pertaining to challenge (C6). Further, systems for retroactive in-vehicle assessment could also be used for validation purposes (see e.g. [97, 98, 99]), notably for ML-based components of the perception system, and thus reduce the gap of challenge (C7). Incorporating this type of operational data is also beneficial for capturing changes to the traffic (behaviour) due to increased penetration of technologies such as ADSs.

Even though collecting and incorporating operational data into the development process helps ameliorating all of the challenges of Sec. III it should be noted that it might be difficult to ensure the applicability of the data collected. For example, it might be hard to ascertain that the data collected with a previous version of the ADS is also useful for the next generation of the system. Thus, frequent releases of challenge (C8) might in fact limit the usability of operational data.

There are however some challenges for collecting such operational data from monitoring activities (i) and (ii):

- potential lack of computational resources for run-time evaluation,
- limited transmission bandwidth, requiring a careful selection and curation of the data, and
- limited predictive power of the KPI/SPIs resulting in limited risk reduction, related to monitoring activity (ii).

D. Scenario-Based Verification and Validation Methods

Most situations occurring during traffic are relatively mundane and do not, consequently, bring value to the testing of an ADS. Hence, FOTs are particularly exposed to this phenomenon. However, when doing simulations or directed testing, one can rely on scenario-based techniques as a means to expose the ADS to more relevant test cases. Scenarios can be generated from real data, as suggested in, e.g. [5, 6, 7, 100, 101], synthesised based on models, or through expert knowledge, e.g. by using an ontology [102, 103]. In all scenario generation approaches, the goal is to find relevant scenarios to challenge the ADS. Irrespective of the approach deployed, it is difficult to generate scenarios to capture all uncertainties of the ADS's environment and its interactions with other road users (outlined in challenges (C1) and (C2)). In fact, these aspects result in an infinite scenario space. Further, the high dependability requirements of the ADS (challenge (C6)) mean that rare scenarios, corner and edge cases will have an impact on the assurance of the ADS. However, capturing all relevant rare scenarios is close to an impossible task, as it would either require huge amounts of driving hours [1, 2], or be, if generated from expert knowledge, non relevant and result in worst-case assumptions. An illustration of the scenario space is shown in Fig. 7, where three different types of scenarios are shown. The scenarios applicable for the ADS given its intended operational domain, A, the scenarios generated for V&V, G, and the scenarios that would lead to safety critical failures, C. The system could only be completely assured if A is fully contained in G, and, in particular, if the intersection between C and A is inside G. Any excessive scenarios generated that are not applicable for the operational design domain (i.e. $G \cap \bar{A}$) could potentially lead to loss of performance due to negative test outcomes.

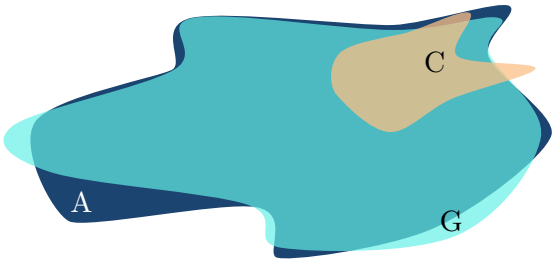

Fig. 7. Illustration of the "scenario space". A corresponding to all possible scenarios in the intended operational design domain of the ADS, G the identified scenarios, and C the safety critical scenarios for the system.

When testing black-box components such as neural networks or other ML-based components, as per challenge (C7), the vastness of the scenario space becomes a problem. Indeed, it is challenging to decide on an appropriate granularity for testing across the scenario space, as the validity of interpolation of the results is unclear [77]. Moreover, placing all tactical responsibility on the ADS (i.e. challenge (C3)) makes the use of scenario-based testing difficult, since the ADS might take actions to avoid the initial state of the scenario altogether, rendering the testing results irrelevant. The complexity of the system and its ability to handle degradations (challenges (C4) and (C5) respectively) could, on the other hand, be efficiently validated through scenario-based methods, as they scale according to the testing environments used. Further, the use of simulations helps executing testing and verification quicker than real-time, thus supporting high release cadences (challenge (C8)), as V&V evidence can be collected in a shorter time period. Additionally, this aspect enables efficient testing of large changes to the system. This said, validation of the models and tools used for simulation itself still impose significant challenges. Riedmaier et al. [10] give a comprehensive survey of scenario-based approaches and present a taxonomy within which the surveyed papers are situated. Similar to the note by Dijkstra et al. in [104, p. 6] regarding software testing, Riedmaier et al. point out that, despite the value and usefulness of scenario-based approaches, the scenarios can only provide evidence for falsification of the system, or provide a means to construct test cases. Ensuring completeness of the scenario space itself has not been widely

addressed despite the activeness in the field. Several studies have, however, focused on the coverage of the scenario space through testing, as pointed out in e.g. [9, 10, 93, 105], but how well the scenario space represents the real world operational conditions has not been shown. Thus, a quantitative measure of the residual risk after scenario-based assessment is still missing. Scenario-based methods do, however, provide an efficient means for falsification of the ADS. Especially, if coupled with search strategies rewarding critical scenarios, such as, for example, importance sampling [106] or sub-set simulation [107]. For the interested reader, Zhang et al. [9] provide a systematic mapping of methodologies for critical scenario identification.

From the discussion above it can be concluded that:

- when considering black-box components (challenge (C7)) it is difficult to judge the coverage level provided by the assessment,
- the ADS, being responsible for the tactical decisions (challenge (C3)), might avoid the initial state of the scenario altogether, rendering the testing thereof superfluous, and
- ensuring completeness of the scenario space with respect to the real world operational conditions is difficult, but nevertheless crucial to ensure that rare and relevant scenarios, as derived from the high dependability requirements (challenge (C6)), are not omitted in the assessment.

E. Formal Methods

Formal methods provide a means to perform verification of the system, taking as inputs a specification of intended behaviours as well as models of the system and its operational environment. Relying on models of the system as well as of its operational context such methods verify the system's fulfilment of its specification. Riedmaier et al. [10] give an overview of existing methods for safety verification of ADSs [10, p. 12–13] and distinguish between three different branches within formal verification: theorem proving, reachability analysis and correct-by-construction synthesis. Furthermore, Riedmaier et al. [10] also provide a characterisation of the methods and an evaluation based on several criteria. In [108], a complementary classification of automatic formal methods for automotive systems, that provides some guarantee of quality, is presented and includes: abstract interpretation, model checking and deductive methods. Abstract interpretation methods assume an approximation of the system in order to support the verification, whereas deductive methods correspond to the theorem proving category of [10].

While formal methods can assess the system's fulfilment of its specification, it might nevertheless be tedious to transfer the results from one assessment of one part of the system to other parts, unless the design of the system is modular, suggesting the usefulness of a contract-based design of the system. Further, the dependency on models for each (sub)system under assessment and models of its intended operational context constitute a challenge for conducting formal verification with respect to black-box models such as neural networks or other ML-based components [10, Figure 8].

It is worth noting that correct-by-construction synthesis could be considered as a design technique rather than a V&V method. However, the challenges faced are the same as for the other methods within formal verification domain, and this is why it is discussed in this section rather than within Sec. IV.

Despite the merits and advantages of formal methods, one can identify several potential difficulties and limitations with respect to the safety assurance of an ADS, as detailed below. Indeed, it may be difficult to:

- construct a complete specification with respect to the real operating environment of the ADS, related to challenges (C1) – (C3), which may also be exacerbated by challenge (C6),
- scale such methods to cover the entirety of the ADS, corresponding to challenge (C4),
- ensure validity of the verification with respect to the specification when using AI/ML-based components, corresponding to challenge (C7), and
- ensure the correctness of the models and parameter values used in conducting the verification, which again stem from challenges (C1) – (C3).

On the other hand, the successful application of formal verification methods would provide an efficient means to ensure the safety of the ADS and thus support high release cadence as well as continuous learning (i.e. challenge (C8)). Formal methods might also help analysing and understanding challenge (C5), i.e. the capabilities of degraded modes of the system, and how the system can safely adapt to cope with such changes. Within a well defined setting, and for a limited component, subsystem or specification, it should be noted that formal methods are both suitable as well as useful. However, as noted in [109], any evidence supplied from formal methods should ideally be accompanied by the applicability of that evidence, as well as the formal method/tool used, whenever incorporated into the assurance case.

Formal Rules for Driving Behaviour: As a means to provide a specification for formal methods, it is convenient to stipulate (formal) rules regarding how the ADS should behave in order to ensure that the system is never at fault in case of an accident. Examples of approaches tackling such problems are, for instance: Mobileye/Intel's "Responsibility Sensitive Safety" (RSS) [14], Nvidia's "Safety Force Field" [15], the "rulebooks" approach taken by nuTonomy [110] and Arechiga [111]. Arechiga [111] propose a specification in signal temporal logic for safe ADS, while the rulebooks approach in [110] refines and elaborates the use of rules to guide the behaviours of the ADS. While [14] and [15] define interaction rules based on mathematical formulas, the rulebooks approach [110] can handle the priority amongst several (potentially conflicting) rules. This capability differentiates the rulebooks approach [110] from the more simplistic approaches of [14] and [15]. One could further consider using worst-case assumptions, for which closed-form solutions of the driving behaviour might be attainable, to elicit a specification of the driving behaviours. An example of such, in relation to a collision avoidance setup, is given in [112].

To summarise, there are four key limitations to keep in mind when using (formal) rules or specifications, as detailed bellow:

- the methods assume that other traffic participants follow (the same) set of rules, which might not be the case with human drivers,
- the methods (implicitly) define who is to blame for an accident. While human drivers tend to naturally help out and collectively avoid accidents, rigidly following a set of rules might instead inhibit such collaborative avoidance by the ADS, therefore increasing the overall number of accidents,
- the approaches rely on assumptions on the parameters used in the models and rules. For instance, RSS [14] implicitly relies on assumptions regarding the vehicle's braking capabilities, as well as those of the surrounding vehicles [113]. Ensuring that these assumptions are correct in all operational conditions of the ADS is central to safety, as a mismatch could yield safety issues, as discussed in [113], and
- accurately estimating the system's parameters (as well as those of the operating environment) is difficult and one is often left with making worst-case assumptions, which could yield a system that is unable to operate due to an overly pessimistic view of the system's capabilities [35].

VI. Run-Time Risk Assessment

Despite the existing design techniques and methods for verification and validation purposes, covered in Sections. IV and V respectively, challenges (C1) – (C8) also warrant efforts for upholding safety of the system during operations. Within that scope, the first aspect, pertaining to risk assessment, is discussed within this section, while the second aspect, regarding what to do about it, is covered in Sec. VII. These two aspects are closely related, such that the output of the risk assessment (discussed in this section) is consumed and guides the adaptation (discussed in Sec. VII). Further, the available adaptations of the system impact and determine what metrics and quantities that should be monitored and assessed during operations. The monitoring and degradation capabilities of the system are themselves inherent of the architecture deployed, which is discussed in Sec. IV-E.

As discussed in Sec. V-C, run-time monitoring of an ADS can have different purposes. The collection of operational data also highlights the importance of appropriate KPIs. In the following section, we focus on the aspects of risk assessment related to run-time (safety) supervision, effectively providing input to, as well as ensure the safety of, the (tactical) decision making of the ADS. The adaptation of the system due to such tactical decisions is in turn discussed in Sec. VI. There are three reasons for having run-time monitoring, to:

(i) Ensure safe tactical decisions despite internal errors or (unexpected) changes to the operating environment, while in the ODD,
(ii) Cope with (more permanent) system degradations, and
(iii) Avoid leaving the ODD.

These three types of monitoring are all related to the system's fault tolerance, where the focus is to identify errors and faults and to establish appropriate counter-action measures, in order to avoid safety critical failures. One way of viewing this problem is through partitioning the operational space into safe, warning and catastrophic states [88, Fig. 1]. With respect to the different considerations above (i) – (iii), the states that one needs to avoid are slightly different. Regarding (i), the focus concerns the catastrophic states related to (temporary) errors in any one of the ADS's sub-systems or (unexpected) changes to the operating environment that might result, if not mitigated appropriately, in an accident or a requirement violation. Such states can be reached due to an erroneous perception of the world by the Environment Perception (EP) block (see Fig. 3), an erroneous plan by the Decision Making (DM) block, or unsuitable path following by the Vehicle Control (VC) block, all of which might lead to a violation of a safety requirement.

As for (ii), the catastrophic states are similar to those of (i), but pertaining to challenge (C5) associated with more permanent or larger degradations, such as, for example, the (permanent) loss of a sensor, reduced braking capabilities or limited computational resources. This type of hazards might also result in an inability of the ADS to safely fulfil its strategic mission. One might consider using Restricted Operational Domains (RODs) as a means to analyse and cope with situations of system degradation, as elaborated on in Sec. VII-A. The capabilities of the ADS to monitor its own system performance is largely dependent on the architecture as well as what requirements are imposed on each of the subsystems and components. Architectural considerations have already been covered in Sec. IV-E. Note that more detailed discussions on component requirement supporting internal monitoring is left for future work.

Lastly, the purpose of (iii) is, on the other hand, to avoid the "catastrophic" state of operating outside of the ODD, which can be mitigated by employing the ODD-strategies given in [51] and, where appropriate, transitioning into a Minimal Risk Condition (MRC). In [51], some considerations for defining a suitable set of quantities and trigger conditions to support such ODD exit strategies are given.

To give a broader context to metrics for run-time monitoring we start by discussing threat assessment techniques used for Advanced Driver Assistance Systems (ADASs), in Sec. VI-A. This is followed by a section on OOD detection, Sec. VI-B and a section revolving around dynamic risk assessment, Sec. VI-C.

It should be noted that run-time verification [114] would provide a means for run-time assessment of the system, especially in relation to the fulfilment of the specification by the present system configuration. Such methods face similar challenges as formal methods, already discussed in V-E, and will thus not be dedicated a separate subsection here.

A. Threat Assessment Techniques

Within the domain of ADASs, assessing the collision threat is an integral part of being able to trigger appropriate corrective measures that are able to avoid collisions through driver support functions. The role of such threat metrics in relation to validation and assurance of an ADS has already been discussed earlier in this paper, namely in sections Sec. V-B and Sec. V-C. There are several recent overviews on the

literature focusing on threat metrics and Threat Assessment (TA). For instance, a comprehensive analysis of different metrics for collision avoidance has recently been provided in [115, Table 3.1, pages 44–48]. In addition to listing the metrics, Feth also provide a short description about which situation is targeted, what assumption are made in terms of prediction models and towards what such metrics are aimed at. Complementary to that work, Dahl et al. [116] give a detailed literature review on available TA techniques for collision avoidance. In [116, TABLE I, p. 9], the reviewed literature is positioned with respect to one of the five identified TA areas, as well as which automotive-related application the reference considers. Further, also Chia et al. [117] provide a set of risk assessment methodologies, some of which are acknowledged to be supporting run-time assessment (see last column of [117, TABLE III, p. 7]). Lefèvre et al. [118] also present a survey on motion prediction and risk assessment for intelligent vehicles. In [118], the authors divide the methods into three categories: physics-based, manoeuvre-based and interaction-aware motion models. It is concluded that while the latter is the most refined, it faces issues with computational complexity due to the high number of considerations, consequently inhibiting run-time applications (at least in 2014 when the survey was conducted). This obstacle in particular is addressed in [119], where a new risk assessment methodology, merging a network-level collision estimate with an estimate on vehicle level, is given. More precisely, this approach simultaneously integrates a dynamic Bayesian network and interaction-aware motion models [119].

Naturally, these different threat metrics come with their own set of assumptions and limitations. In the sequel, we will nevertheless try to assess their collective ability to alleviate the challenges of Sec. III.

The intention of the threat metrics is to assess the current threat or risk faced by the system during its operations. However, accurate modelling and assessment incorporating the uncertainties of challenges (C1) and (C2) into the metrics remain difficult. The metrics themselves provide support for the tactical decisions of the ADS (related to challenge (C3)) and also give a quantitative assessment of the risks irrespective of the complexity of the system, i.e. bypassing challenge (C4). This latter aspect is further discussed in Sec. V with respect to operational data and EVT modelling.

The focus of most TA metrics is to assess the risk imposed on the vehicle from its external environment. However, integral to these assessments are the capabilities of the ego vehicle. For example, the BTN or TTC [11] both incorporate the vehicle's braking capability. Consequently, the TA could partially support the understanding of the necessary adaptation needed to cope with system degradations related to challenge (C5). If coupled with EVT modelling, TA could provide a way to ameliorate the high dependability requirements of challenge (C6), but accurately considering the probabilities of rare events might simultaneously reduce the accuracy of the TA and thus suggesting that these same requirements might pose an obstacle to TA. If the calculation of the metric is reliant on AI/ML-based component, the TA will also be difficult to validate, corresponding to challenge (C7). However, if that is not the case, the use of metrics might provide a means to validate the AI/ML-based components in the system based on operational data, as discussed in Sec. V-C. The frequency of releases of the ADS (challenge (C8)) will not constitute an obstacle for deploying an appropriate TA. Furthermore, such TA techniques would also not be of any particular help to improve the release cadence of the ADS, for example by providing complementary assurance evidence. Continuous data gathering and analysis of the trends of the resulting TAs could however support a learning cycle, also related to challenge (C8).

B. Out-of-Distribution Detection

In order to integrate and trust AI/ML-based components, their ability and performance naturally needs to be ensured. It is difficult, however, to assess the ability of neural network-based algorithms to extrapolate to unseen samples, as small changes in input might drastically alter the results from neural network-based algorithms [77]. Further, the estimated accuracy and performance of such components are measured based on a validation set concerning the intended operational domain. Thus, to be able to rely on such performance estimates, it is paramount to know that AI/ML-based components operate on samples from the same distribution that it has been trained on. For that purpose, anomaly or Out-Of-Distribution (OOD) detection approaches can be used [120, 121, 122]. Alternatively, one can strive for a network that directly *rejects* unrelated open set inputs [123], whereby outputs are produced only if inputs, from within some defined set, are provided to the network. This last proposal could be seen as a means for a network to operate under a contract, requiring the inputs to come from the specified set.

OOD detection is believed to be helpful in increasing the reliability of AI/ML-based components and consequently address challenge (C7). As the components are ensured to operate within the set of samples known to the algorithm, one can also rely on the validation results provided (that are based on samples from the same set). The environment uncertainties faced by the ADS, formulated within challenge (C1), might be partly mitigated by the use of OOD detection whereas the uncertainties originating from the interactions, concerning challenge (C2), are unrelated to the use of an OOD detection method. The challenges of behavioural and structural complexity, challenges (C3) – (C5), are also not applicable. Even though OOD detection methods might support the validation of AI/ML-based components, ensuring sufficient integrity of the OOD detection itself will be challenging and, as a consequence, the high dependability requirements, formulated within challenge (C6), will present obstacles. As OOD detection methods will have to be trained on the same data as the AI/ML-based components nothing does per se hinder frequent development. In some sense updating the OOD detection alongside the AI/ML-based components might even be seen as supporting a learning cycle of the system.

C. Dynamic Risk Assessment

As an alternative to completely assure safety of the ADS's tactical decisions in design-time, where one needs to resort

to employing worst-case assumptions or hard limits on operational parameters, one could instead allow the system to dynamically adapt according to the current situation it faces. In practise, such a system would rely on creating situation awareness, according to which the ADS can modulate or adapt its behaviour [124]. Through this adaptation, the ADS could achieve improved performance, while ensuring safety [30]. Situation awareness is constructed based on the perceived surroundings of the ADS, prediction models of how the current state will evolve [124], as well as knowledge of the capabilities of the own system [125].

If such situation awareness is used for adapting the behaviour of the ADS, one could view the action space of the ADS being restricted by: the system's capabilities, including the uncertainties of the perception system, and the surrounding environment, as exemplified in Fig. 8. This could be compared to the admissible action space from the system's representation of itself and its environment, as discussed in [125, notably Fig. 2]. Note that the time evolution of the scene is not included in this illustration, for simplicity.

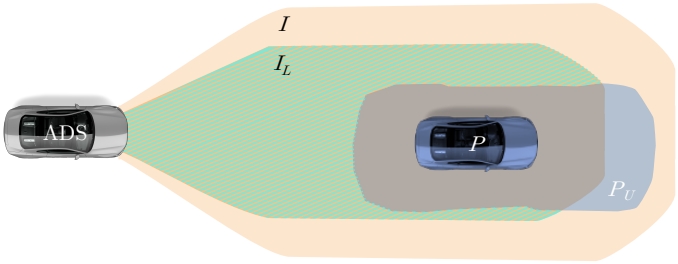

Fig. 8. The action space (striped region) is limited by the internal capabilities of the ADS, I_L, and the appropriately accounting for uncertainties, P_U, around the perceived object, P. I exemplifies the full possible actions but without accounting for uncertainties in the estimate of the internal capabilities.

Intention prediction is the common name of the task concerning the prediction of the movements of other traffic participants, for which Brown et al. [126] present a taxonomy. The proposed taxonomy is constructed around four core tasks: state estimation, intention estimation, trait estimation, and motion prediction. Furthermore, it is also acknowledged that risk estimation constitute an auxiliary task of the modelling. Commonly speaking, these types of models rely heavily on ML-based algorithms, and could provide a means for improving the situation awareness.

Dynamic Risk Assessment (DRA) relies on situation awareness to support run-time decision-making of an ADS, e.g. [31, 124, 127, 128]. In addition to situation awareness, DRA capabilities need to connect a given situation to the safety requirements of the system, or at least to some kind of (quantitative) risk measurement. Similar to the surveillance methods discussed in Sec. IV-E and the TA metrics discussed in Sec. VI-A, Reich and Trapp [31] and Feth et al. [128] suggest using risk metrics as a proxy for deducing the current (dynamic) risk.

In [128], DRA is done by three components, one for each integrity level (low, mid and high), corresponding to the integrity levels with which each (sub)system has been developed, i.e. by following ISO 26262. In the approach of [128], also the Environment Perception (EP) block (see Fig. 3) reports with respect to these three integrity intervals (in-line with the proposal of [129]) and each of the DRA components consume the appropriate EP outputs corresponding its integrity level.

Dynamic behaviour risk assessment is further elaborated upon in the thesis [115] by Feth, where the connection between safety supervision and such a DRA method is also explored.

When approaching a solution to DRA, the more factors and parameters included, the more refined model for situation awareness can be achievable. However, the more factors included, the larger the need for data to determine the models underpinning such a situation awareness. Thus, we face the challenge of state space explosion due to the uncertainties of the operational space of the ADS (related to the challenges (C1) and (C2)), as discussed in Sec. V in terms of the V&V methodologies. However, the ability for a DRA method to handle uncertainties in run-time, formulated within the challenges (C1) and (C2), is completely dependent on the models used.

The flexibility of DRA seem to lend itself well to address challenge (C8), where, for example, the models underpinning the DRA can be easily updated provided that more operational data becomes available. However, when trying to achieve human-like performance (challenge (C6)), the question is how to show the reliability of such methods, especially if such needs to be done before the first deployment. Further, given that much of the perception of an ADS and the subsequent construction of its situation awareness are reliant on ML-based algorithms, the question is also how to connect the outputs thereof to the risk estimates of the DRA, an example related to challenge (C7). This aspect is especially prominent for the intention prediction task. The complexity of the system (formulated in challenge (C4)) can somewhat be circumvented by using DRA, as the down-stream decision-making can be done in run-time based on the outputs of the relatively less complex DRA system. However, how well these methods are able to accommodate degradations (challenge (C5)) is still an open question. Finally, even though an accurate risk assessment at the present time is available through DRA, how to account for the subsequent impact of the tactical decisions of the ADS (i.e. challenge (C3)) has not yet been discussed in the literature.

VII. Run-Time (Self) Adaptation

Having presented different risk assessment techniques in the previous section, we now focus on the task of adapting the ADS's behaviour based on such situation assessment.

One definition of *self-adaptation* considered in this paper determines that the system adapts its behaviour to its environment and context [130, p. 49]. For an ADS, this adaptation can be viewed at different levels. The ADS is explicitly designed to be self-adaptive in the sense of avoiding collisions with other (dynamic) objects, as well as to follow the road, etc., effectively integrating monitoring aspect (i) discussed in Sec. VI. However, the system can adapt the way these objectives are fulfilled and it can possibly also adapt its available capabilities and features, formulated as challenge

(C5). The latter aspect corresponds to monitoring aspects (ii) and (iii) discussed in Sec. VI. For the sake of the clarity of the following discussion, let us distinguish between three different notions of self-adaptiveness of an ADS:

(a) Adaptation to changed user requirements or to changes in the operational context in terms of services, features, capabilities, or inter-operational systems (c.f. monitoring aspects (ii) and (iii) in Sec. VI),
(b) Operational adaptation enabling the fulfilment of certain (safety) objectives (c.f. monitoring aspect (i) from Sec. VI), and
(c) Adaptation of the (safety) objectives and/or adaptation of the constraints for the operations of the system.

The focus of (a) is to elucidate when the user requests changes to the mission, or when system-level changes or degradations result in the need of an adaptation. (b), on the other hand, regards the abilities of the ADS to avoid obstacles in its environment as well as to account for intentions and predicted behaviours of other traffic participants. Thirdly, (c) refers to changing admissible risk levels in the light of the present operating conditions. Each of these adaptation types are discussed in the following subsections.

The first type of adaptation (a) aims at accounting for changes in user requirements, active services, features, and capabilities. To cope with severe system degradations or to avoid leaving the ODD, a common strategy is to transition into a Minimal Risk Condition (MRC). However, simply stopping the ADS upon each and every (small) change to the system's capabilities might not be feasible nor desirable. Hence, the concept of Restricted Operational Domains (RODs) have been proposed in the literature [131]. Such degradation strategies are further discussed in Sec. VII-A below.

As the possible space of all configurations, in relation to adaptation type (a), could be vast, it has been proposed to shift the certification of the specific configuration of the system from design-time to run-time, called run-time certification [25]. As a consequence, this poses an alternative means of adaption according to adaptation type (a). Similar to contract-based design, run-time certification relies on defined contracts (i.e. certificates) for each possible configuration of the system. This set of contracts are defined in design-time, but evaluated (certified) during run-time, in the light of available run-time evidence. This restricts the certification activity to only consider one specific system configuration at a time. Run-time certification is discussed in Sec. VII-B.

Solutions to adaptation type (b) are largely provided through the different methods discussed in Sec. IV, and are also the focus of the V&V methods discussed in Sec. V. However, to show that such solutions yield a safe ADS, while considering all operational uncertainties (i.e. challenges (C1) and (C2)), there is a need to make certain assumptions, often worst-case assumptions, which encapsulate all (statistically) relevant operational situations. These worst-case assumptions could yield a safe but oftentimes unnecessarily conservative system. To circumvent this, one can monitor the operational environment, as discussed in Sec. VI, and adapt the (worst-case) assumptions subject to the present operational situation of the ADS. This effectively corresponds to adaptation type (c). Thus, the specific solution provided for adaptation type (b) is deferred to run-time by adapting with respect to dynamic objectives or constraints, i.e. according to adaptation type (c). This is explored in [26], where a framework for Dynamic Safety Management (DSM) is proposed, and further discussed in Sec. VII-C.

Precautionary safety is yet another approach for ensuring the fulfilment of the safety requirements of the ADS. Here, the driving policy of the ADS is derived in such a way as to ensure the fulfilment of quantitative safety goals, based on the estimated exposure levels to certain events as well as the capabilities of the sensor and actuator platform [12, 13]. Adapting such a policy, subject to the current operational conditions of the ADS, would correspond to adaptation type (c). Such concepts are discussed in more detail in Sec. VII-D.

A. Degradation Strategies

It is central for an ADS to appropriately handle degradations and avoid leaving the expected ODD, related the two last monitoring types (ii) and (iii), discussed in Sec. VI. If faced with a severe system degradation, or when approaching an ODD exit, the ADS can resort to transitioning into a Minimal Risk Condition (MRC). The MRC is a *"stable stopped condition at a position with an acceptable risk [...] The ADS is brought to this state by the user or the system itself, by performing the Dynamic Driving Task Fall-Back (DDT-FB), when a given trip cannot or should not be completed"* [132, p. 2]. It alleviates the risk of an ODD exit [51] as well as avoids operating the system while facing severe degradations that inhibit the fulfilment of the original, user-defined strategic mission [132].

To avoid abandoning the strategic mission of the ADS upon any given system degradation, Colwell et al. [131] suggest using a Restricted Operational Domain (ROD), which encodes the operational domain of the "new" system after the degradation. The ROD could thus effectively help determining whether it is feasible to safely fulfil the strategic mission, despite the system degradation, or if the mission should be abandoned in favour of an MRC. The relationship between the MRC and the ROD is elaborated upon in [132], where the contribution to the safety assurance of such concepts is also discussed.

Fu et al. [133] present a distributed safety mechanism concept, that provides multiple layers of monitoring and enables degradation policies for the ADS. Degradation strategies may range from a reduced driving envelope (e.g. corresponding to a ROD), all the way through to a worst case, immediate stop (corresponding to a highly restrictive MRC). It is worth mentioning that any degradation strategy will require sufficient internal capabilities of the ADS, as well as architectural support (e.g. sensors and computing to actually carry out the required manoeuvre).

An MRC effectively ameliorates the impact of foreseeable changes of the uncertainties related to challenges (C1) and (C2). That is, when it is possible to assess that the operational context suggests uncertainties outside those specified in the ODD, for example, the MRC could be invoked to avoid the associated risks. However, in some cases, such shifts in the uncertainty might not be detected early enough as to

let the ADS avoid an accident. The decision responsibility of the ADS, related to challenge (C3), is slightly clarified through the decision hierarchy proposed in [132]. However, the use of degradation strategies does not ameliorate nor support challenge (C3) as such. The self-adaptation capabilities of the ADS, related to challenge (C5), are highly reliant on appropriate MRCs, and the performance and utility of the system can significantly be improved through the use of RODs. The complexity of the ADS, pertaining to challenge (C4), makes the use of a degradation strategy, such as the ROD, more difficult due to the large number of parameters to be considered. The degradation strategies might help mitigate faults or errors in the system and avoid catastrophic outcomes, consequently supporting the achievement of high dependability requirements, related to challenge (C6). If coupled with anomaly detection methods for AI/ML-based components, such as the out-of-distribution detection methods discussed in Sec. VI-B, the degradation strategies might support the safe handling of situations problematic to AI/ML-based components. In other words, the use of degradation strategies provides a partial support for solving challenge (C7). As for agile development processes/methodologies, related to challenge (C8), the use of both MRCs and RODs would require significant efforts and analyses before being deployed, which might inhibit frequent releases of the software, as most of such analysis could likely not be completely automated and would therefore be time-consuming.

B. Run-time Certification

Rather than anticipating and analysing each possible configuration of a system at design-time, the idea behind run-time certification is shifting some of the assurance (or certification) aspects from design-time to run-time, where more evidence of the system's actual operational context is available. Rushby suggested this in 2007, through what the author calls *Just-in-Time Certification*. This concept has been further developed and, in the context of open adaptive systems, Schneider and Trapp [25] propose conditional safety certificates (ConSerts) for this purpose. In essence, the ConSerts represent a contract with *demands* (cf. *assumes* of contract-based design, Sec. IV-D) under which the subsystem is *guaranteeing* the supply of a specific output. The ConSerts of a (run-time configured) system are evaluated in the light of the available run-time evidence, in order to assess the applicability of a particular configuration. If one configuration is found to be invalid, Schneider and Trapp [25] propose to continue evaluating the next available configuration of the system, suggesting the existence of some hierarchy of system configurations. As such, ConSerts provides a potential way of managing system degradations (related to challenge (C5)). Further, Digital Dependability Identities (DDIs) [27] have been proposed to formalise the information exchange within a systems of systems setup, and to support run-time certification in the context of systems of systems [28, 29]. DDIs or ConSerts could, in practise, be used as a means to facilitate safety supervision, given formalisable properties of the system, while also accommodating for configurability and systems of systems facets. The formalisation required for this approach does, however, face some of the same obstacles as those discussed in relation to formal methods, see Sec. V-E. However, with an appropriate granularity of the system configuration and of the limitations to the factors modelled, the impact of such limitations might be reduced. Nevertheless, this remains an aspect to be shown.

A dynamic measure of the operational environment, provided e.g. by DRA (discussed in Sec. VI-C), could be matched to the assumes of the contracts, corresponding to the run-time certification concept discussed before. Consequently, the two approaches could likely support each other in the construction of a safe and performant ADS.

Even though run-time certification mitigates the state space explosion related to challenges (C1), (C2) and (C3), the approach faces the same issues as contract-based design. Is it possible to create contracts (e.g. ConSerts) that adequately capture the uncertainties related to challenges (C1) and (C2), and the flexibility emanating from challenge (C3)? Similarly, the scalability of run-time certification in the light of challenges (C3) and (C4), remains to be shown. However, by formalising the interfaces between subsystems and components of the ADS, such methods can effectively provide a means to support high cadence releases and continuous learning, i.e. challenge (C8), similar to contract-based design. Lastly, given appropriate measures and monitors for anomaly and OOD detection, run-time certification approaches might help growing the trust in ML-based components (challenge (C7)), as the usage of such components can be adapted given the fulfilment of their respective demands.

C. Dynamic Safety Management

To circumvent the need for a static safety analysis at design time, Trapp and Weiss [30] propose the framework of Dynamic Safety Management (DSM), which allows the system to "self-optimise its performance during run-time" [26, p. 1]. DSM presumes access to run-time information providing a contextual- as well as self-awareness. Considerations, upon which, the more recent development of DRA methods discussed in Sec. VI-C has been founded. This run-time information is used to derive what is called *safety awareness*, and allows the system to reason about the current risk and adapt its behaviour accordingly. Notably, Trapp et al. [26] explores this idea by proposing a dynamic risk analysis, where the quantification of the HARA is done in run-time based on such safety awareness. This quantification is proposed to subsequently ensure that the configurations of the system is valid and safe. Developed in parallel, Khastgir et al. [65] also suggest to dynamically update the parameters of the HARA, in run-time, based on the current operational situations. However, [65] suggests that the update to the HARA should imply changes to the driving behaviour of the ADS, by restricting or relaxing the integrity requirements for the system to solve the current operational situation.

Also, Calinescu et al. [24] suggest allocating some of the assurance tasks to run-time, by dynamically generating the assurance case throughout both design-time as well as run-time. This run-time assurance generation is predominantly

dependent on formal methods and model checking [24], assuming formalisable system models and requirements.

While [30] suggest DSM as a means to optimise the system (configuration) according to the current safety awareness, and [65] propose to alter the behaviour of the ADS implicitly by evaluating the HARA in run-time, one can also consider adapting the (tactical) behaviour of the ADS according to the dynamically assessed risk (e.g. from DRA). This latter concept is explored in [115, 127] and is also hinted at in [135, 136, 137].

Deferring the assurance of the tactical decisions to run-time is proposed in order to ameliorate the effects of the operational uncertainties related to challenges (C1) and (C2). The considerable complexity of the required risk assessment techniques to support DSM seem to be exacerbated by the ADS's responsibility for tactical decisions, i.e. challenge (C3). However, the impact from the complexity of the ADS itself (challenge (C4)) could be ameliorated, as already argued in Sec. VI-C for DRA, by relying on a relatively less complex system for DRA/DSM. Degradation capabilities (related to challenge (C5)) would not only be solved through DSM but could further support a more elaborate handling of any type of degradation, effectively providing understanding for the RODs of each degradation of the system. However, this presumes good self-awareness capabilities of the system.

By being highly dependent on the DRA methods, the lack of proof of reliability on the part of the DRA approaches is also inherited by the DSM, making it difficult to assure the reliability of the resulting actions of the ADS. Consequently, it might be difficult to quantify and assure the DSM method before deployment, at least when comparing to the high dependability requirements related to challenge (C6). This would be exacerbated if one is reliant on AI/ML-algorithms for the implementation of either the DRA or the DSM. In which case, the validation of such components, i.e. related to challenge (C7), would impose an obstacle. However, the DSM might also provide a means to rely on AI/ML-components for path planning, as the risk of each generated path could effectively be assessed through DRA, see e.g. [135].

Assuming that the models underpinning the DRA and DSM are updated based on collected operational data, they would promote a learning cycle of the system (corresponding to challenge (C8)). Further, having assured a method for DSM would also support the frequent changes of other components in the system (the first aspect of challenge (C8)), as suggested in [136].

D. Precautionary Safety

The concept of precautionary safety policy was first introduced by Rodrigues de Campos et al. [12] with the purpose of achieving an improved ADS performance while ensuring the fulfilment of ambitious quantitative safety requirements, prescribed as a Quantitative Risk Norm (QRN) [47]. The proposed methodology accounts for the system's emergency response capabilities, sensing performance and the exposure levels to different adverse events in order to enable the derivation of an appropriate driving policy, with which the ADS is able to fulfil the prescribed safety requirements. The fulfilment of the quantitative requirements is shown in a statistical way, rather than proving the fulfilment of safety requirements based on worst-case assumptions. In [13], this methodology is elaborated upon to include more complex perception error rates and a process of rate estimation, both for perception error rates as well as for the arrival rate of an adverse event. These aspects led to a probabilistic approach for coping with random errors or degradations of the system (related to challenge (C5)).

While the precautionary safety methodology alleviates some of the restrictions of a worst-case design-time assumption with respect to the operational uncertainties of the ADS (challenges (C1) and (C2)), the scalability aspects has not been exhaustively addressed. Thus, the ability for this methodology to overcome challenge (C4) is still an open question. The approach of [13] suggests that challenges (C6) and (C7) could be overcome, but at the (initial) expense of a reduced performance of the system. Furthermore, placing the tactical responsibility onto the ADS (challenge (C3)) might in turn impact the event exposure rates, which are the central tenant of the methodology. Consequently, it remains unclear how well such methodology could work for releases of a specific ADS (version) without considerable closed-loop data from that specific version of the system. How a design methodology based on precautionary principles can help support agile development and frequent releases of challenge (C8), also remains to be seen. However, the incorporation of operational data, as suggested in [13], suggests that this methodology could support the continuous learning aspect related to challenge (C8).

The notion of precautionary safety could also be merged with a framework for DRA in order to achieve a dynamic adaptation of the policy, based on the current risk levels of the ADS, which could help to achieve an even more performant system. This is partly exemplified in [127], but with the main difference that the requirements on the ADS are not posted as quantitative elements. However, the risks are dynamically estimated in [127], including a probabilistic formulation of the uncertainties, which is the same as suggested in [12] and [13]. For example, the jay-walking avoidance use case analysed by Rodrigues de Campos et al. presents a very crude way of DRA, where the considered two different road types are associated with different exposure levels. Thus, given the knowledge about which road type the ADS is operating on, it is possible to adapt the driving policy in order to ensure the fulfilment of the safety requirements.

VIII. Results and Discussion

TABLE I summarises the ability of each of the discussed methods to overcome the eight challenges (C1) – (C8). The table is a result of a qualitative assessment made by the authors, resulting in a classification (identified by letters) that indicates how each of the the surveyed methods responds to the identified challenges, as detailed in the caption of TABLE I. Each assigned letter is motivated in the section where the respective method has been discussed. The classification can be structured into three main groups:

	Section reference	Method	Challenges							
			Uncertainties		Behavioural and structural complexity			Dependability requirements	AI and ML components	Agile development
			(C1)	(C2)	(C3)	(C4)	(C5)	(C6)	(C7)	(C8)
Design Techniques	IV-A	Operational design domain	**C**	**C**	**A**	**A**	**A**	**A**	**A**	**A**
	IV-B	Hazard and risk assessment	**C**	**C**	**C**	**O**	**O**	**O**	**N**	**U**
	IV-C	Process arguments	**C**	**C**	**O**	**O**	**O**	**O**	**U**	**O**
	IV-D	Contract-based design	**C**	**C**	**C**	**O**	**A**	**N**	**O**	**S**
	IV-E	Supervisor architectures	**A**	**A**	**A**	**O**	**S**	**A**	**A**	**A**
Verification and validation methods	V-A	Field operational tests	**A**	**O**	**O**	**A**	**A**	**C**	**A**	**C**
	V-B	Extreme value theory	**A**	**O**	**O**	**A**	**A**	**A**	**A**	**N**
	V-C	Operational data collection	**A**	**A**	**A**	**A**	**A**	**A**	**A**	**A**
	V-D	Scenario-based V&V methods	**C**	**C**	**C**	**A**	**A**	**O**	**C**	**A**
	V-E	Formal methods	**C**	**C**	**C**	**O**	**A**	**O**	**O**	**A**
Run-time assessment	VI-A	Threat assessment	**O**	**O**	**A**	**A**	**A**	**A**	**A**	**A**
	VI-B	Out-of-distribution detection	**A**	**N**	**N**	**N**	**N**	**O**	**A**	**A**
	VI-C	Dynamic risk assessment	**A**	**A**	**U**	**A**	**U**	**U**	**U**	**A**
Run-time adaptation	VII-A	Degradation strategies	**A**	**A**	**N**	**O**	**S**	**A**	**A**	**O**
	VII-B	Run-time certification	**O**	**O**	**O**	**O**	**S**	**N**	**A**	**S**
	VII-C	Dynamic safety management	**A**	**A**	**O**	**A**	**S**	**O**	**O**	**A**
	VII-D	Precautionary safety	**A**	**A**	**U**	**U**	**A**	**S**	**S**	**U**

TABLE I

CLASSIFICATION OF HOW THE IDENTIFIED CHALLENGES ARE ADDRESSED BY THE DIFFERENT METHODS DISCUSSED. LEGEND: **S**={PROVIDES SOLUTION}, **A**={AMELIORATED BY/SUPPORTS SOLVING}, **U**={UNCERTAIN/UNCLEAR, MORE WORK IS NEEDED}, **N**={NEUTRAL, NEITHER SUPPORTS NOR IS AFFECTED BY}, **O**={OBSTACLE, MINOR CHALLENGE}, AND **C**={CHALLENGE, FUNDAMENTAL TO METHOD}.

- the ones indicating a *positive* contribution (**S** and **A**),
- the ones indicating a *neutral* contribution (**U** and **N**), and
- the ones indicating that the particular challenge is *difficult* to, or not tackled by, the method (**O** and **C**).

Methods promising a solution to a particular challenge are annotated with an **S**, referring to a *Solution*. For example, completely adopting contract-based design promises to solve challenge (C8). Whether this is feasible considering the remaining seven challenges is, of course, questionable. Nevertheless, the notation **S** indicates some clear advantages of this solution with respect to those challenges. An **A**, referring to an *Amelioration* of the challenge, is reserved for methods that support or partly solve the given challenge. For instance, operational data ameliorates all discussed aspects in one way or another, but it is not sufficient on its own to provide a complete solution to any of the challenges. Cells annotated with a **U** indicate that the authors are *unable* to deduce the method's applicability to solve a given challenge. This suggests the need for future work to answer that question. An **N**, referring to *Neutral*, implies that the method is deemed indifferent with respect to the particular challenge, since it neither ameliorates nor exacerbates the challenge. Challenges that impose an *obstacle* for the methods to provide valuable assurance evidence are annotated with an **O**. Finally, a **C** denotes challenges that are fundamental to the method and that, despite continued efforts and future work, will likely remain troublesome. While a **C** is used to indicate the fundamental limitation to the method with respect to a given challenge, an **O** suggests that future work might provide solutions to overcome the obstacles currently present.

The novelty of a system such as an ADS, as well as the lack of best practises and sufficient data affect all methods and techniques discussed. The proposed classification may come to change as new best practises evolve and, especially, when more data is gathered. Particularly, the challenges pertaining

to uncertainties (i.e. challenges (C1) and (C2)) might not be as daunting given the existence of billions of miles of operational data. Thus, the assessment of TABLE I reveals a snapshot at this point in time, of which of the challenges that currently impose issues for the discussed methods.

Another important aspect, only covered implicitly in the discussions, is the question of residual risk, i.e. what is the quantitative risk that each method is not able to (sufficiently) capture. This is, for example, related to the question of tool/process qualification and also partly captured by the challenge of ensuring the integrity of each method, thus related to the methods' abilities to solve challenge (C6).

A. Addressing the Challenges

Below we analyse, in more detail, the collected results of TABLE I in order to deduce which methods suggest solutions to each of the challenges. Notably, each challenge has a method which seems to at least ameliorate the challenge, albeit not solve it completely.

In general, many of the challenges are (at least to some extend) supported by the collection of more operational data together with the use of DRA and DSM.

Challenges (C1) and (C2), pertaining to the uncertainties imposed on the ADS, seem promising to address by shifting at least parts of the assurance provision into run-time. This can be achieved through monitoring of the operations of the system, using a run-time monitor or a supervisor. Ideally, such monitoring is coupled with DSM or a precautionary driving policy to optimise available performance of the ADS.

Challenge (C3), the tactical responsibility, might be possible to address through DRA, but more work remains for a conclusive statement. However, the challenge can be partly ameliorated on the specification side, through the use of the ODD. Further, it can be mitigated through appropriate supervisor architectures and operational data collection.

Coping with challenge (C4), the complexity of the ADS, can be supported by DSM, together with DRA, as well as through confinement using the ODD. Further, it can be assessed through the collection of operational data (through FOTs, operational data collection and by extension EVT) as well as through scenario-based methods.

Handling degradations of the system (challenge (C5)), requires supervision, but can also be solved through appropriate degradation strategies, run-time certification and DSM. All of the V&V techniques also provide means to assess the validity of such degradations, albeit under the assumption of explicit analysis of each subsystem in relation to its capabilities, performance and ROD.

The high dependability requirements (challenge (C6)), are difficult to ensure through the V&V methods presently available, even though operational data collection and EVT would provide some support. These requirements seem best addressed by supervision methods and architectural patterns coupled with appropriate degradation strategies. However, the probabilistic formulation for precautionary safety also suggests a solution.

As for incorporating and growing trust in AI and ML-based components (challenge (C7)), we also seem to have a solution in the precautionary safety approach, however, operational data collection, supervision architectures as well as FOTs, coupled with EVT, also provide ameliorating solutions, as does the use of OOD detection methods.

Lastly, many of the discussed methods could to be compatible with agile development and frequent releases, related to challenge (C8). Most notably, the contract-based techniques, such as contract-based design and run-time certification, lend themselves particularly well for this purpose, but with the caveat of scalability and the ability to compose contracts in the light of challenges (C1) – (C3).

B. Identified Research Gaps

Supported by TABLE I and our discussions from Sections. IV – VII, we conclude five categories of research gaps, as given in the sub-sections below. The gaps are derived by consulting TABLE I and identifying the challenges for each method that provide an obstacle (**O**), a fundamental challenge (**C**) or where the assessment is yet uncertain/unclear (**U**). While the fundamental challenges (**C**) might not directly warrant further development of the method itself, they could still leave a missing piece for the safety assurance of the ADS and are hence included in the derivation of the research gaps. The derived individual gaps are subsequently gathered into similar themes forming the five categories.

1) Completeness of provided safety evidence:

- How to ensure that the confinement to the design made through the ODD (IV-A) is appropriate with respect to the uncertainties of challenges (C1) and (C2)?
- How to amend or tailor the process for HARA (IV-B) to ensure completeness of the provided hazards with respect to the operational uncertainties ((C1) and (C2)) and the fact that the ADS is responsible for the tactical decisions (challenge (C3))?
- What are the implications and potential remedies for contract-based design (IV-D) and formal methods (V-E) if considerations from the tactical responsibility and operational uncertainties of the ADS (corresponding to challenges (C1) – (C3)) cannot be adequately formalised?
- How to mitigate the impact from a mismatch between the real operational uncertainties of the ADS (challenges (C1) and (C2)) and the considered scenario space for scenario-based V&V (V-D)?

2) Improvements, analyses, and automation of methods:

- How to automate HARA (IV-B) to support challenge (C8), with frequent releases and continuous learning?
- What are the quantitative contributions from current (safety) design and development processes (IV-C), especially considering the operational uncertainties of the ADS, i.e. challenges (C1) and (C2), but also challenges (C3) – (C6)?
- What are appropriate leading metrics for (safety) operational data collection (V-C) of an ADS, in particular to capture the operational uncertainties ((C1) and (C2)), the tactical responsibilities (C3) in relation to the high dependability requirements of challenge (C6)?

- How to derive realistic and statistically probable (albeit rare) scenarios (V-D) corresponding to challenge (C6), the high dependability requirements of an ADS?
- How to ensure that tested scenarios are relevant considering the ability of the ADS to avoid the situation leading up to the scenario through its tactical decision, i.e. challenge (C3)?
- What are appropriate metrics for threat assessment (VI-A) to appropriately capture the uncertainties of challenges (C1) and (C2), especially considering rareness of events (related to challenge (C6))?
- How to assure the integrity of run-time methods: OOD detection methods (VI-B), DRA (VI-C), run-time certification (VII-B) and DSM (VII-C) in the light of the high dependability requirements on the ADS (challenge (C6))?
- How well does DRA (VI-C) accommodate degradations of the system (challenge (C5))?
- How to construct run-time contracts (VII-B) to appropriately capture the uncertainties present in run-time (i.e. challenges (C1) and (C2))?

3) Collecting closed loop data and handling the responsibility of tactical decision allocated to the ADS:

- How to safely collect (large quantities of) closed loop data (supporting the fulfilment of the high dependability requirements of challenge (C6)) from FOTs (V-A), and thereby support EVT (V-B) and provide input to precautionary safety (VII-D), and how to account for the tactical responsibility of challenge (C3)?
- How does the tactical decision responsibility of the ADS (challenge (C3)) impact the DRA (VI-C), DSM (VII-C) and precautionary safety (VII-D) methods?

4) Coping with AI/ML-based components: This category of gaps corresponds to the column with challenge (C7) of TABLE I.

- What are appropriate design and development processes (IV-C) to incorporate and rely on AI/ML-based components?
- What is the impact on scenario-based V&V considering the non-interpolatable results when testing AI/ML-based components?
- How to ensure validity when using formal methods (V-E) for such components, especially in relation to the high dependability requirements of challenge (C6)?
- How to compose contracts for AI/ML-based components, both for contract-based design (IV-D) as well as run-time certification (VII-B)?
- How to derive quantitative risk measures from such components for the use in DRA (VI-C) and in turn for DSM (VII-C), while also ensuring dependability of the resulting outputs?

5) Scalability of method and patterns:

- How do contract-based design (IV-D), supervisor architectures (IV-E), formal methods (V-E), run-time certification (VII-B), DRA (VI-C), degradation strategies (VII-A), DSM (VII-C) and precautionary safety (VII-D) scale when applied on a complex system such as the ADS (c.f. challenge (C4))?
- How to best leverage FOTs (V-A) for providing safety evidence of the system in relation to an agile development process (challenge (C8)) and considering the high dependability requirements of challenge (C6)?

C. Threats to Validity

This paper presents a holistic perspective on safety evidence provision for an ADS and discuss methods related thereto. For each of the methods discussed we draw upon a selection of papers to support the view presented. Due to the diversity of topics included in this paper, and the novelty and specificity of the application (ADS), we eventually discarded explicit systematic searches and methods, due to the vastness of publications found through such an approach. Thus, two key validity concerns for this work are: the lack of reproducability, and the lack of proof for completeness and exhaustiveness. The work of trying to systematise the search did however yield a solid basis for further work, both in terms of a comprehensive list of related work as well as resulting in the mindmap of Fig. 2 providing structure to both the work with the survey as well as for this paper.

Where applicable, we rely on literature surveys conducted in the field to provide an overview of the respective areas. [67] gives an overview of design processes for safety of AI/ML-based components and [59] discuss qualitative processes in relation to safety in general, giving support to the section on qualitative process arguments of Sec. IV-C. For supervisor architectures, we draw upon the results of [79, 82]. [10, 91] give an overview of V&V methods for ADSs, guiding the discussions of Sec. V. The connection between operational data and assurance methodologies is given in [76]. [8, 9, 10, 105] provide insights into scenario-based V&V methods, whereas we draw upon the works in [10, 108] for the section on formal methods. As for the run-time risk assessment methods, discussed in Sec. VI: threat assessment techniques are discussed taking support from [115, 116, 117, 118], whereas the survey on open adaptive systems and run-time certification of [138] helps us in the discussions on run-time certification as well as dynamic risk assessment.

Based on these surveys we implicitly inherit completeness with respect to at least those sub-topics. For other topics, we have relied on snowballing [139] starting from one or two prominent papers on the topic. We have naturally also drawn upon the complementary expertise of the co-authors for the covered areas. Taken together we believe that we have presented an appropriate and representative view of each of the discussed methods. Despite a possible lack of completeness and exhaustiveness of this review, we believe that it provides a representative and useful overview of the current challenges present for safety evidence provision for ADSs. We further believe that potential work overlooked would not have a significant impact on the assessments of TABLE I nor the derived research gaps above.

D. Future Work

In this paper the methods are discussed in isolation, whereby their interaction and interplay have not been analysed in-depth.

How well the methods work together, and how they can be combined to address the challenges presented in this paper have only been briefly discussed herein and are also suggested for future work. Further, an analysis of which assumptions, models and uncertainties that each method is imposing, consuming, and mitigating/leaving would be a reasonable next step for further work. Finally, to understand the holistic safety perspective on that level of detail, it will be paramount to also include assurance methodologies, focusing on the organisation and traceability of the arguments and evidence supporting the assurance case, in such an analysis.

IX. Conclusion

In this paper we identify eight challenges pertaining to the safety evidence provision for ADSs. Furthermore, we analyse methods of the state-of-the-art in relation to these eight challenges, thereby providing a holistic perspective of the current progress of safety evidence provisioning for ADSs. The results of the discussion are summarised in TABLE I, where the ability of each method to mitigate the challenges is given. Additionally, the challenges especially onerous, with respect to each method, are highlighted. Supported by these results, a list of research gaps are identified, grouped into five major themes: VIII-B1 completeness of provided safety evidence, VIII-B2 improvements and analysis needs, VIII-B3 safely collecting closed loop data and accounting for tactical responsibility on the part of the ADS, VIII-B4 coping with AI/ML-based components, and VIII-B5 the scalability of the approaches with respect to the complexity of the ADS.

We conclude that the existing methods provide a good base for safety evidence provision, but there are several challenges remaining when considering the complexity and novelty of an ADS. Several methods need to come together to bridge this gap. As a next step, we propose to include assurance concepts (i.e. how to organise, trace and present the assurance arguments and evidence into an assurance case) in the analysis as well as expand such analysis to include assumptions and models deployed by each method. Including these aspects will help elucidate the interplay between the methods. Finally, we suggest to analyse how and where, throughout the assurance life-cycle of the ADS, that uncertainties originate and are mitigated in relation to the analysed methods.

Acknowledgement

The authors would like to thank Mattias Brännström (formerly at Zenseact, currently at Waymo) for initial discussions inspiring this work; to Christine Räisänen (Chalmers) for her excellent feedback and guidance through the course *Writing up for publication*, resulting in clear improvements to the text's readability and focus; to Jonas Krook (Zenseact) for a thorough review and valuable feedback; to Fredrik Warg (RISE) for insightful comments and feedback during the writing process; and to Yuvaraj Selvaraj (Zenseact) for feedback on the section on formal methods.

References

[1] N. Kalra and S. M. Paddock, "Driving to safety: How many miles of driving would it take to demonstrate autonomous vehicle reliability?" *Transportation Research Part A: Policy and Practice*, vol. 94, pp. 182–193, 2016.

[2] R. A. Young, "Automated driving system safety: Miles for 95% confidence in "vision zero"," *SAE Int. Journal of Advances and Current Practices in Mobility*, vol. 2, no. 2020-01-1205, pp. 3454–3480, 2020.

[3] X. Zhao, K. Salako, L. Strigini, V. Robu, and D. Flynn, "Assessing safety-critical systems from operational testing: A study on autonomous vehicles," *Information and Software Technology*, vol. 128, p. 106393, 2020.

[4] ISO, "ISO 26262:2018 Road vehicles – Functional safety," 2018.

[5] A. Pütz, A. Zlocki, J. Bock, and L. Eckstein, "System validation of highly automated vehicles with a database of relevant traffic scenarios," *Situations*, vol. 1, p. E5, 2017.

[6] H. Elrofai, J.-P. Paardekooper, E. de Gelder, S. Kalisvaart, and O. Op den Camp, "Streetwise scenario-based safety validation of connected and automated driving," TNO, Tech. Rep., July 2018.

[7] E. De Gelder, J. Manders, C. Grappiolo, J.-P. Paardekooper, O. Op den Camp, and B. De Schutter, "Real-world scenario mining for the assessment of automated vehicles," in *Int. Conf. on Intelligent Transportation Systems (ITSC)*. IEEE, 2020.

[8] C. Neurohr, L. Westhofen, T. Henning, T. de Graaff, E. Möhlmann, and E. Böde, "Fundamental considerations around scenario-based testing for automated driving," in *Intelligent Vehicles Symposium (IV)*. IEEE, 2020.

[9] X. Zhang, J. Tao, K. Tan, M. Törngren, J. M. Gaspar Sanchez, M. R. Ramli, X. Tao, M. Gyllenhammar, F. Wotawa, N. Mohan, M. Nica, and H. Felbinger, "Finding critical scenarios for automated driving systems: A systematic mapping study," *IEEE Transactions on Software Engineering*, pp. 1–1, 2022.

[10] S. Riedmaier, T. Ponn, D. Ludwig, B. Schick, and F. Diermeyer, "Survey on scenario-based safety assessment of automated vehicles," *IEEE access*, vol. 8, pp. 87 456–87 477, 2020.

[11] D. Åsljung, J. Nilsson, and J. Fredriksson, "Using extreme value theory for vehicle level safety validation and implications for autonomous vehicles," *IEEE Transactions on Intelligent Vehicles*, vol. 2, no. 4, pp. 288–297, 2017.

[12] G. Rodrigues de Campos, R. Kianfar, and M. Brännström, "Precautionary safety for autonomous driving systems: Adapting driving policies to satisfy quantitative risk norms," in *Int. Conf. on Intelligent Transportation Systems (ITSC)*. IEEE, 2021.

[13] M. Gyllenhammar, G. Rodrigues de Campos, F. Sandblom, M. Törngren, and H. Sivencrona, "Uncertainty aware data driven precautionary safety for automated driving systems considering perception failures and

event exposure," in *Intelligent Vehicles Symposium (IV)*. IEEE, 2022.

[14] S. Shalev-Shwartz, S. Shammah, and A. Shashua, "On a formal model of safe and scalable self-driving cars," *arXiv preprint arXiv:1708.06374*, 2017.

[15] D. Nistér, H.-L. Lee, J. Ng, and Y. Wang, "The safety force field," *NVIDIA White Paper*, 2019.

[16] C. Pek, M. Koschi, and M. Althoff, "An online verification framework for motion planning of self-driving vehicles with safety guarantees," in *AAET-Automatisiertes und vernetztes Fahren*, 2019.

[17] R. Salay, R. Queiroz, and K. Czarnecki, "An analysis of iso 26262: Using machine learning safely in automotive software," *arXiv preprint arXiv:1709.02435*, 2017.

[18] J. Henriksson, M. Borg, and C. Englund, "Automotive safety and machine learning: Initial results from a study on how to adapt the iso 26262 safety standard," in *Int. Workshop on Software Engineering for AI in Autonomous Systems (SEFAIAS)*. IEEE/ACM, 2018.

[19] E. Asaadi, E. Denney, and G. Pai, "Towards quantification of assurance for learning-enabled components," in *European Dependable Computing Conf. (EDCC)*. IEEE, 2019.

[20] R. Hawkins, C. Paterson, C. Picardi, Y. Jia, R. Calinescu, and I. Habli, "Guidance on the assurance of machine learning in autonomous systems (amlas)," *arXiv preprint arXiv:2102.01564*, 2021.

[21] S. Jha, J. Rushby, and N. Shankar, "Model-centered assurance for autonomous systems," in *Int. Conf. on Computer Safety, Reliability, and Security (SAFECOMP)*. Springer, 2020.

[22] E. Denney, G. Pai, and I. Habli, "Dynamic safety cases for through-life safety assurance," in *Int. Conf. on Software Engineering*, vol. 2. IEEE/ACM, 2015.

[23] E. Asaadi, E. Denney, J. Menzies, G. J. Pai, and D. Petroff, "Dynamic assurance cases: A pathway to trusted autonomy," *IEEE Computer*, vol. 53, no. 12, pp. 35–46, 2020.

[24] R. Calinescu, D. Weyns, S. Gerasimou, M. U. Iftikhar, I. Habli, and T. Kelly, "Engineering trustworthy self-adaptive software with dynamic assurance cases," *IEEE Transactions on Software Engineering*, vol. 44, no. 11, pp. 1039–1069, 2017.

[25] D. Schneider and M. Trapp, "Conditional safety certification of open adaptive systems," *ACM Transactions on Autonomous and Adaptive Systems (TAAS)*, vol. 8, no. 2, pp. 1–20, 2013.

[26] M. Trapp, D. Schneider, and G. Weiss, "Towards safety-awareness and dynamic safety management," in *European Dependable Computing Conf. (EDCC)*. IEEE, 2018.

[27] D. Schneider, M. Trapp, Y. Papadopoulos, E. Armengaud, M. Zeller, and K. Höfig, "WAP: digital dependability identities," in *Int. Symposium on Software Reliability Engineering (ISSRE)*. IEEE, 2015.

[28] J. Reich, M. Zeller, and D. Schneider, "Automated evidence analysis of safety arguments using digital dependability identities," in *Int. Conf. on Computer Safety, Reliability, and Security (SAFECOMP)*. Springer, 2019.

[29] J. Reich, D. Schneider, I. Sorokos, Y. Papadopoulos, T. Kelly, R. Wei, E. Armengaud, and C. Kaypmaz, "Engineering of runtime safety monitors for cyber-physical systems with digital dependability identities," in *Int. Conf. on Computer Safety, Reliability, and Security (SAFECOMP)*. Springer, 2020.

[30] M. Trapp and G. Weiss, "Towards dynamic safety management for autonomous systems," in *Safety-Critical Systems Symposium*, 2019.

[31] J. Reich and M. Trapp, "Sinadra: towards a framework for assurable situation-aware dynamic risk assessment of autonomous vehicles," in *European Dependable Computing Conf. (EDCC)*. IEEE, 2020.

[32] E. R. Griffor, C. Greer, D. A. Wollman, M. J. Burns *et al.*, "Framework for cyber-physical systems: Volume 1, overview," 2017. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.1500-201.pdf

[33] P. Koopman and M. Wagner, "Autonomous vehicle safety: An interdisciplinary challenge," *IEEE Intelligent Transportation Systems Magazine*, vol. 9, no. 1, pp. 90–96, 2017.

[34] ISO, "ISO/PAS 21448:2019 Road vehicles - Safety of the intended functionality," 2019.

[35] A. Shetty, M. Yu, A. Kurzhanskiy, O. Grembek, H. Tavafoghi, and P. Varaiya, "Safety challenges for autonomous vehicles in the absence of connectivity," *Transportation Research Part C: Emerging Technologies*, vol. 128, p. 103133, 2021.

[36] F. Warg, S. Ursing, M. Kaalhus, and R. Wiik, "Towards safety analysis of interactions between human users and automated driving systems," in *European Congress on Embedded Real Time Software and Systems*, 2020.

[37] NTSB, "Collision between vehicle controlled by developmental automated driving system and pedestrian," *National Transportation Safety Board, Washington, DC, USA, Technical Report HAR-19-03*, 2019.

[38] T. W. Victor, E. Tivesten, P. Gustavsson, J. Johansson, F. Sangberg, and M. Ljung Aust, "Automation expectation mismatch: incorrect prediction despite eyes on threat and hands on wheel," *Human factors*, vol. 60, no. 8, pp. 1095–1116, 2018.

[39] Sven E. Hammarberg, " BOEING 737 MAX – The failure of a safety culture," in *Scandinavian Conf. on Systems and Software Safety*, 2021. [Online]. Available: https://www.saferresearch.com/sites/safer.cloud.chalmers.se/files/2021-11/1%20Boeing%20737%20MAX%2C%20Sven%20E.%20Hammarberg.pdf

[40] S. Nair, J. L. De La Vara, M. Sabetzadeh, and L. Briand, "An extended systematic literature review on provision of evidence for safety certification," *Information and Software Technology*, vol. 56, no. 7, pp. 689–717, 2014.

[41] S. Burton, J. A. McDermid, P. Garnett, and R. Weaver, "Safety, complexity, and automated driving: Holistic perspectives on safety assurance," *Computer*, vol. 54, no. 8, pp. 22–32, 2021.

[42] M. Törngren and U. Sellgren, "Complexity challenges in development of cyber-physical systems," in *Principles of Modeling*. Springer, 2018, pp. 478–503.

[43] SAE, "SAE J3016:202104 - SURFACE VEHICLE RECOMMENDED PRACTICE - Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles," 2021.

[44] M. Wood, P. Robbel, M. Maass, R. D. Tebbens, M. Meijs, M. Harb, and P. Schlicht, "Safety first for automated driving," *Aptiv, Audi, BMW, Baidu, Continental Teves, Daimler, FCA, HERE, Infineon Technologies, Intel, Volkswagen*, 2019.

[45] M. Lindman, I. Isaksson-Hellman, and J. Strandroth, "Basic numbers needed to understand the traffic safety effect of automated cars," in *Int. Research Council on Biomechanics of Injury (IRCOBI) Conf.*, 2017.

[46] P. Junietz, U. Steininger, and H. Winner, "Macroscopic safety requirements for highly automated driving," *Transportation research record*, vol. 2673, no. 3, pp. 1–10, 2019.

[47] F. Warg, M. Skoglund, A. Thorsén, R. Johansson, M. Brännström, M. Gyllenhammar, and M. Sanfridson, "The quantitative risk norm – a proposed tailoring of HARA for ADS," in *Int. Conf. on Dependable Systems and Networks Workshops (DSN-W)*. IEEE/IFIP, 2020.

[48] A. Avizienis, J.-C. Laprie, B. Randell, and C. Landwehr, "Basic concepts and taxonomy of dependable and secure computing," *IEEE transactions on dependable and secure computing*, vol. 1, no. 1, pp. 11–33, 2004.

[49] H. Kopetz, *Real Time Systems—Design Principles for Distributed Embedded Applications*. Second Edition, Springer Verlag, 2012.

[50] U.K. Ministry of Defence, Defence Standard 00-56, "Safety Management Requirements for Defence Systems," London, UK. 2007.2.61.

[51] M. Gyllenhammar, R. Johansson, F. Warg, D. Chen, H.-M. Heyn, M. Sanfridson, J. Söderberg, A. Thorsén, and S. Ursing, "Towards an operational design domain that supports the safety argumentation of an automated driving system," in *European Congress on Embedded Real Time Systems*, 2020.

[52] M. Hörwick and K.-H. Siedersberger, "Strategy and architecture of a safety concept for fully automatic and autonomous driving assistance systems," in *Intelligent Vehicles Symposium (IV)*. IEEE, 2010.

[53] D. Wittmann, C. Wang, and M. Lienkamp, "Definition and identification of system boundaries of highly automated driving," in *7. Tagung Fahrerassistenz*, 2015.

[54] C. W. Lee, N. Nayeer, D. E. Garcia, A. Agrawal, and B. Liu, "Identifying the operational design domain for an automated driving system through assessed risk," in *Intelligent Vehicles Symposium (IV)*. IEEE, 2020.

[55] K. Czarnecki, "Operational world model ontology for automated driving systems–part 1: Road structure," *Waterloo Intelligent Systems Engineering Lab (WISE) Report*, 2018.

[56] ——, "Operational world model ontology for automated driving systems–part 2: Road users, animals, other obstacles, and environmental conditions,"," *Waterloo Intelligent Systems Engineering Lab (WISE) Report, University of Waterloo*, 2018.

[57] BSI, "PAS 1883:2020 Operational Design Domain (ODD) taxonomy for an automated driving system (ADS) - Specification," 2020.

[58] H. Martin, K. Tschabuschnig, O. Bridal, and D. Watzenig, "Functional safety of automated driving systems: Does iso 26262 meet the challenges?" in *Automated Driving*. Springer, 2017, pp. 387–416.

[59] I. Habli, "Safety standards: Chronic challenges and emerging principles," *Handbook of Safety Principles*, pp. 732–746, 2017.

[60] I. Habli, R. Alexander, and R. D. Hawkins, "Safety cases: An impending crisis?" in *Safety-Critical Systems Symposium*. York, 2021.

[61] N. G. Leveson, *Engineering a safer world: Systems thinking applied to safety*. The MIT Press, 2016.

[62] IEC, "60812: 2018–Failure modes and effects analysis (FMEA and FMECA)," 2018.

[63] S. M. Sulaman, A. Beer, M. Felderer, and M. Höst, "Comparison of the FMEA and STPA safety analysis methods–a case study," *Software quality journal*, vol. 27, no. 1, pp. 349–387, 2019.

[64] B. Kramer, C. Neurohr, M. Büker, E. Böde, M. Fränzle, and W. Damm, "Identification and quantification of hazardous scenarios for automated driving," in *Int. Symposium on Model-Based Safety and Assessment*. Springer, 2020.

[65] S. Khastgir, H. Sivencrona, G. Dhadyalla, P. Billing, S. Birrell, and P. Jennings, "Introducing ASIL inspired dynamic tactical safety decision framework for automated vehicles," in *Int. Conf. on Intelligent Transportation Systems (ITSC)*. IEEE, 2017.

[66] R. Salay, M. Angus, and K. Czarnecki, "A safety analysis method for perceptual components in automated driving," in *Int. Symposium on Software Reliability Engineering (ISSRE)*. IEEE, 2019.

[67] M. Rabe, S. Milz, and P. Mader, "Development methodologies for safety critical machine learning applications in the automotive domain: A survey," in *Conf. on Computer Vision and Pattern Recognition*. IEEE/CVF, 2021.

[68] EASA AI Task Force and Daedalean AG, "Concepts of Design Assurance for Neural Networks (CoDANN)," 2020. [Online]. Available: https://www.easa.europa.eu/document-library/general-publications/concepts-design-assurance-neural-networks-codann

[69] J.-P. Steghöfer, E. Knauss, J. Horkoff, and R. Wohlrab, "Challenges of scaled agile for safety-critical systems," in *Int. Conf. on Product-Focused Software Process Improvement*. Springer, 2019.

[70] C. A. R. Hoare, "An axiomatic basis for computer programming," *Communications of the ACM*, vol. 12, no. 10, pp. 576–580, 1969.

[71] A. Benveniste, B. Caillaud, D. Nickovic, R. Passerone, J.-B. Raclet, P. Reinkemeier, A. Sangiovanni-Vincentelli, W. Damm, T. Henzinger, and K. G. Larsen,

"Contracts for system design," Ph.D. dissertation, Inria, Rapport de recherche RR-8147, 2012.

[72] I. Bate, R. Hawkins, and J. McDermid, "A contract-based approach to designing safe systems," in *8th Australian workshop on Safety critical systems and software (SCS) - Volume 33*, 2003, pp. 25–36.

[73] I. Sljivo, B. Gallina, J. Carlson, and H. Hansson, "Strong and weak contract formalism for third-party component reuse," in *Int. Symposium on Software Reliability Engineering Workshops (ISSREW)*. IEEE, 2013.

[74] D. Nešić, M. Nyberg, and B. Gallina, "Product-line assurance cases from contract-based design," *Journal of Systems and Software*, vol. 176, p. 110922, 2021.

[75] F. Warg, H. Blom, J. Borg, and R. Johansson, "Continuous deployment for dependable systems with continuous assurance cases," in *Int. Symposium on Software Reliability Engineering Workshops (ISSREW)*. IEEE, 2019.

[76] M. Gyllenhammar, C. Bergenhem, and F. Warg, "ADS Safety Assurance–Future Directions," in *Int. Workshop on Critical Automotive Applications: Robustness & Safety (CARS)*, 2021.

[77] K. Eykholt, I. Evtimov, E. Fernandes, B. Li, A. Rahmati, C. Xiao, A. Prakash, T. Kohno, and D. Song, "Robust physical-world attacks on deep learning visual classification," in *Conf. on Computer Vision and Pattern Recognition*. IEEE, 2018.

[78] H. Kopetz, "An architecture for safe driving automation," in *Insights*. The Autonomous, 2021.

[79] Ö. Ş. Taş, F. Kuhnt, J. M. Zöllner, and C. Stiller, "Functional system architectures towards fully automated driving," in *Intelligent Vehicles Symposium (IV)*. IEEE, 2016.

[80] A. Mehmed, M. Antlanger, and W. Steiner, "The monitor as key architecture element for safe self-driving cars," in *IEEE-IFIP Int. Conf. on Dependable Systems and Networks-Supplemental Volume (DSN-S)*. IEEE, 2020.

[81] M. Törngren, X. Zhang, N. Mohan, M. Becker, L. Svensson, X. Tao, D.-J. Chen, and J. Westman, "Architecting safety supervisors for high levels of automated driving," in *Int. Conf. on Intelligent Transportation Systems (ITSC)*. IEEE, 2018.

[82] A. Reschka, J. R. Böhmer, T. Nothdurft, P. Hecker, B. Lichte, and M. Maurer, "A surveillance and safety system based on performance criteria and functional degradation for an autonomous vehicle," in *Int. Conf. on Intelligent Transportation Systems (ITSC)*. IEEE, 2012.

[83] Phil Koopman, "Safety Requirements," in *Carnegie Mellon University – 18-642: Embedded Software Engineering*, 2020. [Online]. Available: https://users.ece.cmu.edu/~koopman/lectures/ece642/31_SafetyRequirements.pdf

[84] ISO/IEC/IEEE, "ISO/IEC/IEEE 42010:2011 Systems and software engineering – Architecture description," 2011.

[85] D. W. Oliver, T. P. Kelliher, and J. G. J. Keegan, *Engineering Complex Systems with Models and Objects*. McGraw-Hill, 1997.

[86] Y. C. Yeh, "Design considerations in boeing 777 fly-by-wire computers," in *Int. High-Assurance Systems Engineering Symposium (Cat. No. 98EX231)*. IEEE, 1998.

[87] L. Sha, "Using simplicity to control complexity," *IEEE Software*, vol. 18, no. 4, pp. 20–28, 2001.

[88] M. Machin, J. Guiochet, H. Waeselynck, J.-P. Blanquart, M. Roy, and L. Masson, "Smof: A safety monitoring framework for autonomous systems," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 48, no. 5, pp. 702–715, 2016.

[89] L. Masson, J. Guiochet, H. Waeselynck, K. Cabrera, S. Cassel, and M. Törngren, "Tuning permissiveness of active safety monitors for autonomous systems," in *NASA Formal Methods Symposium*. Springer, 2018.

[90] A. Desai, S. Ghosh, S. A. Seshia, N. Shankar, and A. Tiwari, "Soter: A runtime assurance framework for programming safe robotics systems," in *Int. Conf. on Dependable Systems and Networks (DSN)*. IEEE/IFIP, 2019.

[91] J. Wishart, S. Como, U. Forgione, J. Weast *et al.*, "Literature review of verification and validation activities of automated driving systems," *SAE Int. J. Connect. Automat. Veh*, vol. 3, no. 4, pp. 267–323, 2020.

[92] A. Zlocki, L. Eckstein, and F. Fahrenkrog, "Evaluation and sign-off methodology for automated vehicle systems based on relevant driving situations," *Transportation Research Record*, vol. 2489, no. 1, pp. 123–129, 2015.

[93] F. Batsch, S. Kanarachos, M. Cheah, R. Ponticelli, and M. Blundell, "A taxonomy of validation strategies to ensure the safe operation of highly automated vehicles," *Journal of Intelligent Transportation Systems*, vol. 26, no. 1, pp. 14–33, 2021.

[94] D. Åsljung, J. Nilsson, and J. Fredriksson, "Comparing collision threat measures for verification of autonomous vehicles using extreme value theory," *IFAC-PapersOnLine*, vol. 49, no. 15, pp. 57–62, 2016.

[95] Underwriters Laboratories, "4600: Standard for Evaluation of Autonomous Products," 2020-04-01.

[96] D. Åsljung, C. Zandén, and J. Fredriksson, "A Risk Reducing Fleet Monitor for Automated Vehicles Based on Extreme Value Theory," *techrxiv*, 5 2022.

[97] M. Gyllenhammar and C. Zandén, "Performance monitoring and evaluation of a vehicle adas or autonomous driving feature," Apr. 22 2021, US Patent App. 17/075,181.

[98] M. Gyllenhammar, C. Zandén, M. K. Vakilzadeh, and A. Falkovén, "Perception performance evaluation of a vehicle adas or ads," Jul. 22 2021, US Patent App. 17/154,202.

[99] M. Gyllenhammar, C. Zandén, and M. K. Vakilzadeh, "Assessment of a vehicle control system," Dec. 23 2021, US Patent App. 17/337,121.

[100] M. Gyllenhammar, C. Zandén, and M. Törngren, "Defining fundamental vehicle actions for the develop-

ment of automated driving systems," in *World Congress (WCX)*. SAE, 2020.

[101] H. Nakamura, H. Muslim, R. Kato, S. Préfontaine-Watanabe, H. Nakamura, H. Kaneko, H. Imanaga, J. Antona-Makoshi, S. Kitajima, N. Uchida *et al.*, "Defining reasonably foreseeable parameter ranges using real-world traffic data for scenario-based safety assessment of automated vehicles," *IEEE Access*, vol. 10, pp. 37 743–37 760, 2022.

[102] G. Bagschik, T. Menzel, and M. Maurer, "Ontology based scene creation for the development of automated vehicles," in *Intelligent Vehicles Symposium (IV)*. IEEE, 2018.

[103] S. Geyer, M. Baltzer, B. Franz, S. Hakuli, M. Kauer, M. Kienle, S. Meier, T. Weißgerber, K. Bengler, R. Bruder *et al.*, "Concept and development of a unified ontology for generating test and use-case catalogues for assisted and automated vehicle guidance," *Intelligent Transport Systems*, vol. 8, no. 3, pp. 183–189, 2014.

[104] E. W. Dijkstra *et al.*, "Notes on structured programming," 1970.

[105] Z. Tahir and R. Alexander, "Coverage based testing for V&V and Safety Assurance of Self-driving Autonomous Vehicles: A Systematic Literature Review," in *Int. Conf. On Artificial Intelligence Testing (AITest)*. IEEE, 2020.

[106] D. Zhao, H. Lam, H. Peng, S. Bao, D. J. LeBlanc, K. Nobukawa, and C. S. Pan, "Accelerated evaluation of automated vehicles safety in lane-change scenarios based on importance sampling techniques," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 3, pp. 595–607, 2016.

[107] S. Zhang, H. Peng, D. Zhao, and H. E. Tseng, "Accelerated evaluation of autonomous vehicles in the lane change scenario based on subset simulation technique," in *Int. Conf. on Intelligent Transportation Systems (ITSC)*. IEEE, 2018.

[108] V. Todorov, F. Boulanger, and S. Taha, "Formal verification of automotive embedded software," in *Conf. on Formal Methods in Software Engineering*, 2018.

[109] E. Denney and G. Pai, "Evidence arguments for using formal methods in software certification," in *Int. Symposium on Software Reliability Engineering Workshops (ISSREW)*. IEEE, 2013.

[110] A. Censi, K. Slutsky, T. Wongpiromsarn, D. Yershov, S. Pendleton, J. Fu, and E. Frazzoli, "Liability, ethics, and culture-aware behavior specification using rulebooks," in *Int. Conf. on Robotics and Automation (ICRA)*. IEEE, 2019.

[111] N. Arechiga, "Specifying safety of autonomous vehicles in signal temporal logic," in *Intelligent Vehicles Symposium (IV)*. IEEE, 2019.

[112] J. Nilsson, A. C. Ödblom, and J. Fredriksson, "Worst-case analysis of automotive collision avoidance systems," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 4, pp. 1899–1911, 2015.

[113] P. Koopman, B. Osyk, and J. Weast, "Autonomous vehicles meet the physical world: RSS, variability, uncertainty, and proving safety," in *Int. Conf. on Computer Safety, Reliability, and Security (SAFECOMP)*. Springer, 2019.

[114] M. Leucker and C. Schallhart, "A brief account of runtime verification," *The Journal of Logic and Algebraic Programming*, vol. 78, no. 5, pp. 293–303, 2009.

[115] P. Feth, "Dynamic behavior risk assessment for autonomous systems," Ph.D. dissertation, Fraunhofer Verlag, 2020.

[116] J. Dahl, G. R. de Campos, C. Olsson, and J. Fredriksson, "Collision avoidance: A literature review on threat-assessment techniques," *IEEE Transactions on Intelligent Vehicles*, vol. 4, no. 1, pp. 101–113, 2018.

[117] W. M. D. Chia, S. L. Keoh, C. Goh, and C. Johnson, "Risk assessment methodologies for autonomous driving: A survey," *IEEE Transactions on Intelligent Transportation Systems*, 2022.

[118] S. Lefèvre, D. Vasquez, and C. Laugier, "A survey on motion prediction and risk assessment for intelligent vehicles," *ROBOMECH journal*, vol. 1, no. 1, pp. 1–14, 2014.

[119] C. Katrakazas, M. Quddus, and W.-H. Chen, "A new integrated collision risk assessment methodology for autonomous vehicles," *Accident Analysis & Prevention*, vol. 127, pp. 61–79, 2019.

[120] D. Hendrycks and K. Gimpel, "A baseline for detecting misclassified and out-of-distribution examples in neural networks," *arXiv preprint arXiv:1610.02136*, 2016.

[121] S. Liang, Y. Li, and R. Srikant, "Enhancing the reliability of out-of-distribution image detection in neural networks," *arXiv preprint arXiv:1706.02690*, 2017.

[122] W. Liu, X. Wang, J. Owens, and Y. Li, "Energy-based out-of-distribution detection," *Advances in Neural Information Processing Systems*, vol. 33, pp. 21 464–21 475, 2020.

[123] A. Bendale and T. E. Boult, "Towards open set deep networks," in *Conf. on computer vision and pattern recognition*. IEEE, 2016.

[124] A. Wardziński, "Safety assurance strategies for autonomous vehicles," in *Int. Conf. on Computer Safety, Reliability, and Security (SAFECOMP)*. Springer, 2008.

[125] M. Nolte, I. Jatzkowski, S. Ernst, and M. Maurer, "Supporting safe decision making through holistic system-level representations & monitoring–a summary and taxonomy of self-representation concepts for automated vehicles," *arXiv preprint arXiv:2007.13807*, 2020.

[126] K. Brown, K. Driggs-Campbell, and M. J. Kochenderfer, "A taxonomy and review of algorithms for modeling and predicting human driver behavior," *arXiv preprint arXiv:2006.08832*, 2020.

[127] J. Reich, M. Wellstein, I. Sorokos, F. Oboril, and K.-U. Scholl, "Towards a software component to perform situation-aware dynamic risk assessment for autonomous vehicles," in *European Dependable Computing Conf. (EDCC)*. Springer, 2021.

[128] P. Feth, R. Adler, and D. Schneider, "A context-aware, confidence-disclosing and fail-operational dy-

namic risk assessment architecture," in *European Dependable Computing Conf. (EDCC)*. IEEE, 2018.
[129] R. Johansson and J. Nilsson, "The need for an environment perception block to address all asil levels simultaneously," in *Intelligent Vehicles Symposium (IV)*. IEEE, 2016.
[130] Y. Brun, G. D. Marzo Serugendo, C. Gacek, H. Giese, H. Kienle, M. Litoiu, H. Müller, M. Pezzè, and M. Shaw, "Engineering self-adaptive systems through feedback loops," in *Software engineering for self-adaptive systems*. Springer, 2009, pp. 48–70.
[131] I. Colwell, B. Phan, S. Saleem, R. Salay, and K. Czarnecki, "An automated vehicle safety concept based on runtime restriction of the operational design domain," in *Intelligent Vehicles Symposium (IV)*. IEEE, 2018.
[132] M. Gyllenhammar, M. Brännström, R. Johansson, F. Sandblom, S. Ursing, and F. Warg, "Minimal risk condition for safety assurance of automated driving systems," in *Int. Workshop on Critical Automotive Applications: Robustness & Safety (CARS)*, 2021.
[133] Y. Fu, A. Terechko, J. F. Groote, and A. K. Saberi, "A formally verified fail-operational safety concept for automated driving," *SAE International Journal of Connected and Automated Vehicles*, vol. 5, no. 1, pp. 7–21, jan 2022.
[134] J. Rushby, "Just-in-time certification," in *IEEE Int. Conf. on Engineering Complex Computer Systems (ICECCS)*. IEEE, 2007.
[135] M. Gyllenhammar and H. Sivencrona, "Path planning in autonomous driving environments," Mar. 24 2022, US Patent App. 17/477,906.
[136] ——, "Risk estimation in autonomous driving environments," Mar. 24 2022, US Patent App. 17/477,920.
[137] ——, "Scenario identification in autonomous driving environments," Mar. 24 2022, US Patent App. 17/477,943.
[138] M. Trapp and D. Schneider, "Safety assurance of open adaptive systems – a survey," in *Models@Run.time*. Springer, 2014, pp. 279–318.
[139] C. Wohlin, "Guidelines for snowballing in systematic literature studies and a replication in software engineering," in *Int. Conf. on evaluation and assessment in software engineering*, 2014.

Magnus Gyllenhammar pursues a PhD at KTH Royal Institute of Technology as part of his employment at Zenseact, Gothenburg, Sweden. His research focuses on finding efficient strategies for safety argumentation of ADSs, especially focusing on precautionary safety and dynamic risk assessment in relation to the fulfilment of a quantitative risk norm. He received his MSc. in Engineering Physics, major in Complex Adaptive System, from Chalmers University of Technology, in 2016. In 2018, he joined Zenseact (then Zenuity) and has since worked on creating and realising data-driven strategies for verification and safety argumentation of ADSs.

Gabriel Rodrigues de Campos received his Ph.D. in Automatic Control in 2012 from Grenoble University/Grenoble INP, France. He is currently a researcher with Zenseact in Gothenburg, Sweden. Prior to joining Zenseact, he was a postdoctoral fellow with the Department of Signals and Systems, Chalmers University of Technology, Sweden and the DEIB, Politecnico di Milano, Italy. His research interests include cooperative and distributed control, safety assurance, and threat-assessment and decision-making techniques.

Martin Törngren has an engineering background in Mechatronics. After starting a company in the mid 90s, specializing in advanced tools for developers of embedded control systems, he embarked on an academic career, becoming a Professor in Embedded Control Systems at KTH in 2002. His core research interests are in cyber-physical systems design methodology including architecting, safety, and model-based engineering. Networking, multidisciplinary research and industrial collaboration have been characteristic throughout his career. He is the initiator of the Innovative Centre for Embedded Systems (www.ices.kth.se), launched in 2008, and the initiator and director of the TECoSA research center on Trustworthy Edge Computing Systems and Applications at KTH (www.tecosa.center.kth.se).