



Identification and Quantification of Hazardous Scenarios for Automated Driving

Birte Kramer¹(✉), Christian Neurohr¹(✉), Matthias Bükler², Eckard Böde¹,
Martin Fränzle¹, and Werner Damm¹

¹ OFFIS, Oldenburg, Germany

{kramer,neurohr,boede,franzle,damm}@offis.de

² BTC Embedded Systems, Oldenburg, Germany
matthias.bueker@btc-es.de

Abstract. We present an integrated method for safety assessment of automated driving systems which covers the aspects of functional safety and safety of the intended functionality (SOTIF), including identification and quantification of hazardous scenarios. The proposed method uses and combines established exploration and analytical tools for hazard analysis and risk assessment in the automotive domain, while adding important enhancements to enable their applicability to the uncharted territory of safety analyses for automated driving. The method is tailored to support existing safety processes mandated by the standards ISO 26262 and ISO/PAS 21448 and complements them where necessary. It has been developed in close cooperation with major German automotive manufacturers and suppliers within the PEGASUS project (<https://www.pegasusprojekt.de/en>). Practical evaluation has been carried out by applying the method to the PEGASUS Highway-Chauffeur, a conceptual automated driving function considered as a common reference system within the project.

Keywords: Automated driving · Hazard analysis · Risk assessment · SOTIF · Scenario identification · Environmental triggers

1 Introduction

In order to bring automated driving systems (ADS) [17] to the market, there are several challenges that need to be overcome. One of them is the verification

This study was partially supported and financed by AUDI AG, BMW Group, Continental Teves AG & Co. oHG, Daimler AG, Robert Bosch GmbH, TÜV SVD GmbH and Volkswagen AG within the context of PEGASUS, a project funded by the German Federal Ministry for Economic Affairs and Energy (BMWi). (**P**roject for the **E**stablishment of **G**enerally **A**ccepted quality criteria, tools and methods as well as **S**cenarios and **S**ituations for the release of highly-automated driving functions).

© Springer Nature Switzerland AG 2020

M. Zeller and K. Höfig (Eds.): IMBSA 2020, LNCS 12297, pp. 163–178, 2020.

https://doi.org/10.1007/978-3-030-58920-2_11

and validation of such systems. It has been well-established [11] that a mileage-based approach is infeasible. This is mainly due to the impossibility of driving the vast distance that is required to obtain a statistically valid argument in support of a positive safety statement. The most promising alternative for verification and validation of automated driving is a scenario-based approach where testing is guided by a manageable set of logical scenarios [13] that have been identified as crucial. Recent research projects, such as PEGASUS¹ and ENABLE-S3², explored and pushed this scenario-based approach to testing. A central challenge for this is the systematic identification and quantification of scenarios that are likely to exhibit hazardous behavior of the ADS. While analyzing already existing real-world data (such as accident databases, real world driving, etc.) reveals which scenarios are hazardous for human drivers, the set of hazardous scenarios for automated driving can only be partially obtained from this data prior to the large-scale deployment of ADS. Therefore, a complementary, knowledge-based approach is needed in order to identify and quantify such hazards early on in the development process. This cannot be replaced by massive testing. In Sect. 2 we revisit the challenges for automated driving and explain why existing methods for Hazard Analysis and Risk Assessment (HARA) used in the automotive domain, as suggested by the ISO 26262 [9] and the ISO/PAS 21448 [10], cannot adequately identify hazardous scenarios for automated vehicles. Moreover, we briefly introduce the Highway-Chauffeur, a conceptual ADS with the operational design domain ‘German Highway’, that serves as running example in PEGASUS and throughout this paper. In Sect. 3 we propose a method for the identification of hazardous scenarios for automated driving that combines established methods for HARA with a focus on detecting hazard-triggering environmental conditions. Moreover, we propose a method for estimating the probability of such a scenario in real-world traffic and show how the results can be used for risk assessment, leading to an integrated iterative risk mitigation process. An in-depth review and application of the method have been conducted within the PEGASUS project. More detailed information on the method as well as its application and evaluation targeting the PEGASUS highway-chauffeur have been made public in a German language technical report [5].

2 Problem Characterization and Challenges

Hazardous Scenarios. The ISO 26262 [9] defines a **hazard** as a potential source of harm. In PEGASUS [14], different sources of hazards for automated driving were distinguished, namely hazards arising from (1) the impact of the environment on the ADS, (2) the impact of the ADS on the environment and (3) the interaction between human driver and ADS. Although the methods presented in this paper were developed with a focus on (1), it may be applied to class (2) as well. As Class (3) has a different focus (e.g. HMI concept) it needs different methods to consider the specific problems.

¹ www.pegasusprojekt.de/en.

² www.enable-s3.eu.

According to [19], a **scene** describes a snapshot of the environment, while a scenario describes the temporal development between several scenes in a sequence of scenes. Thus, a **hazardous scenario** can be characterized by adding contextual information to an identified hazard by means of environmental triggers. A comprehensive catalog of such scenarios then enables a test-driven verification approach for automated driving.

Challenges for Automated Driving. Advanced Driving Assistance Systems currently on the market only perform individual sub-tasks of vehicle control, while the residual tasks remain the responsibility of the driver. With the driver acting as a redundant control system, a fail-safe safety concept is sufficient. Safety mechanisms, as addressed in [10], are therefore primarily concerned with avoiding false-positive reactions. With SAE Level ≥ 3 [17], ADS (temporarily) relieve the driver of the driving task. Consequently, the driver is no longer available as a fallback at all times and missing (re-)actions of the ADS's (false negative reactions) play a decisive role for operational safety and thus, requiring a fail-operational (or even fail-silent) safety concept. While hazardous scenarios for human drivers can be reconstructed relatively well from existing data (e.g. accident data bases), the question arises whether the corresponding triggers are the same for ADS. Reflections from metallic objects (e.g. crash barriers) could lead to erroneous recognition of objects by a radar sensor, an unknown accident cause for humans. Extensive databases of observed accidents for automated driving are lacking, so possible hazardous scenarios are not known a priori, rendering safety assessment of automated vehicles particularly challenging. In addition, the criticality of automated driving may be highly discontinuous along the parameter ranges of environmental models: algorithms make discrete decisions, the thresholds of which are only partially known, especially machine learning methods. For example, only a few differences in images that are hardly visible to the human eye can lead to a false object classification [2]. It is therefore dangerous to solely rely on a data-driven approach (i.e. a combination of existing driving data and variation methods) when identifying hazardous scenarios. Although this problem could, in principle, be countered with a complete characterization of all scenarios in the Operational Design Domain (ODD), it is impossible to explicitly describe all relevant scenes and scenarios. Thus, it is inherently difficult to appropriately specify the ADS's behaviour. The intended functionality can therefore neither assumed to be safe nor fully specified (SOTIF). This problem is addressed in [10], but only for advanced driver assistance systems. In order to address the aforementioned issues, it is essential to adapt existing methods for HARA according to the challenges presented by analyzing the safety of ADS. An integrative HARA, which facilitates an iterative feedback loop into the development process, should be performed early in the development of the ADS so that the results can be incorporated appropriately. With this goal in mind, the proposed method for identification and quantification of hazardous scenarios has been developed.

Existing techniques for Hazard Analysis and Risk Assessment. In the proposed method, we use, combine and extend established techniques for hazard analysis and risk assessment. Thus, we briefly summarize them here.

Hazard and Operability Study (HAZOP) was developed and successfully applied in the chemical industry in the 1970s. Starting in the 1990s HAZOP was used in other areas and finally also adapted for the automotive industry. HAZOP is a structured, keyword-based brainstorming approach that investigates significant deviations from the specified behavior in order to identify possible hazards. Optimally, a HAZOP should be performed by a diverse group of specialists from different areas of expertise, who study the system under consideration from various points of view. Selected keywords are applied to process variables and components of the system to investigate deviations from the ideal state. In the process, possible causes, their consequences and potential countermeasures are identified, without making any claims of exhaustiveness. For more details on classical HAZOP we refer to [6]. In this paper we use two different versions of HAZOP (see step (2.1) and (2.2) of Sect. 3) that were specifically adapted for the application to ADS.

Fault Tree Analysis (FTA) is a widely used method to identify fault chains and was originally developed by the U.S. Air Force. The Fault Tree Handbook [20] provides an excellent overview of the method. FTA follows a top-down approach, i.e. starting from a certain event (called Top Level Event), causes for this event are systematically identified down to basic events. These causes are logically entangled using boolean algebra. With quantitative fault trees, minimal sets of triggering events, so-called minimal cut sets, are identified after the fault tree has been created and a probability of occurrence of the Top Level Event is calculated from probabilities of the basic events. Classical quantitative fault trees assume that each Top Level Event is caused by faults of some other component inside the system. This assumption is no longer tenable for automated driving, since every environmental conditions could potentially trigger or propagate faults. Another problem of fault trees are unknown dependencies between events. Therefore, the FTA is often complemented by a Common Cause Analysis [7].

System-Theoretic Process Analysis (STPA) is a relatively new hazard analysis technique that uses STAMP (System-Theoretic Accident Model and Processes, [12]). In this accident model, hazards arise from untreated environmental influences, untreated or uncontrolled defects of components, unsafe interactions between components or from insufficiently coordinated control actions between control loops. In STAMP, safety is understood as an emergent property, which occurs when system components interact in a larger context. This is in contrast to the safety process of ISO 26262 (and other methods such as FTA or FMEA), where only defects and malfunctions of components are considered. Based on this accident model, STPA identifies unsafe control actions and derives safety constraints. The next step is to identify the causal factors for the occurrence of the previously identified unsafe control actions. The causes identified here within the control loop help to refine the previously defined safety constraints into safety requirements.

Related Work. As the presented method aims at identifying and quantifying hazardous scenarios for scenario based testing while deriving safety

requirements for the further development, there is related work for several parts of the method. For example, in [3] an ontology is proposed to automatically generate scenes for the development of automated vehicles. However, this does not generate a classification of hazardous/relevant scenarios. [1] uses STPA to identify unsafe interactions of ADS in the absence of system malfunctions. Thus, this may be used to generate scenarios. However, this is not done there and they do not use methods to quantify scenarios/malfunctions. [18] uses STPA to identify hazards that arise from expectations that other road users have of the ADS (because driving styles of automated vehicles may differ from human driving styles) which corresponds to class (2) of the before-mentioned sources of hazards. [21] uses a scenario based STPA and compares it with more traditional HARA methods. However, they are also only focused on identifying hazards in specific scenarios and do not present an approach on how to generate and quantify them.

The PEGASUS Highway Chauffeur. [15] is an example of a ADS, thus not requiring the driver to monitor the system but needing to be able to take back the driving task within a defined time margin. The ODD and basic functionality of the Highway Chauffeur were defined to include highways in Germany, a speed range of $[0, 130] \frac{\text{km}}{\text{h}}$, lane changes, following in stop & go traffic and emergency braking and collision avoidance. It excludes construction sites, driving onto/exiting the highway and extreme weather conditions.

Contribution. We outline an integrated safety process for ADS that addresses functional safety as well as safety of the intended functionality (SOTIF). Our method is composed of an identification part (Hazard Analysis) and a quantification part (Risk Assessment) for hazardous scenarios. Identification builds on a combination of established methods used for Hazard Analysis, which we adapted to uncover the safety-relevant weaknesses and blind spots of ADS. In the second part, we propose quantification of hazards via probability estimation of the hazard's context. In contrast to the process of ISO 26262, the subsequent risk assessment abandons the factor of 'controllability', which is practically non-existent (or strongly diminished, at least) for automated driving. Moreover, we integrate risk mitigation into the safety process. Lastly, we indicate how regulatory requirements on hazard exposure, can be used to systematically derive requirements on error rates. The method introduces the iterative feedback loops between concept and development phase that are required for establishing a fail-operational safety concept for ADS, while remaining compatible with the ISO 26262 and ISO/PAS 21448. Moreover, the method identifies and quantifies hazardous scenarios for scenario based testing. These can be incorporated into logical scenarios [13]. This enables specific selection of potentially critical scenarios in the testing process and serves to support the argument for a sufficiently complete test space coverage.

3 Method for Identification of Hazardous Scenarios

In the following we describe an iterative method for the identification and quantification of hazardous scenarios for highly automated driving. The method, as

seen in Fig. 1, is divided into two parts. The identification part, described in this section, consists of steps (1)–(5). For more examples of the steps see [5].

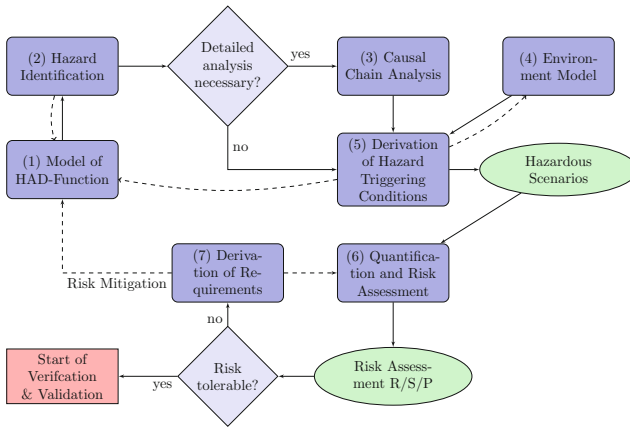


Fig. 1. Overview of the method to identify and quantify hazardous scenarios

Step (1): Modeling of the ADS. specifies the functional architecture and the intended functionality of the ADS. In particular, this needs to include the flow of information between the components of the system (i.e. inputs and outputs of each component). This model serves as the starting point for the method (see Fig. 1) and is iteratively refined in the process as indicated by the dotted lines leading back to step (1). For the Highway Chauffeur this information flow can be seen in Fig. 2 where each arrow represents that information is given to the next component.

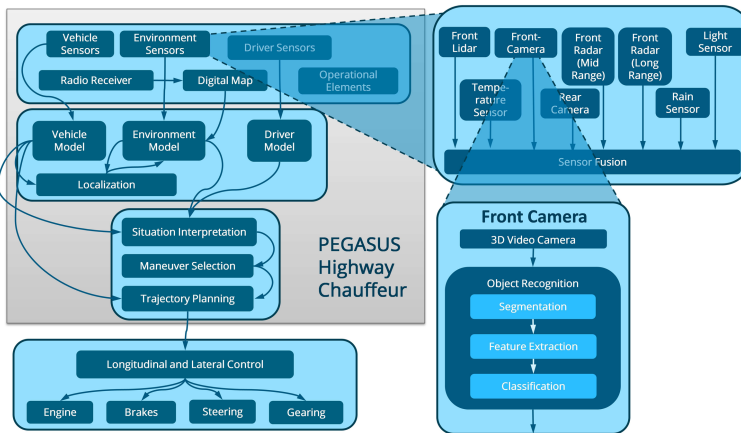


Fig. 2. High-level functional architecture for the PEGASUS highway chauffeur.

Step (2): Hazard Identification aims at identifying hazards related to the ADS, corresponding to causes in the ADS and possible triggers in the environment. The focus lies on hazards that are not caused by random hardware faults, but rather by performance limitations or functional insufficiencies of the ADS in its perception, in the modeling and interpretation of its environment and in the planning of maneuvers and trajectories. In particular, this includes hazards which result from the absence of SOTIF. The hazard identification is split into two substeps (2.1) and (2.2).

Step (2.1): Scenario-based Identification of Hazards on Vehicle-Level.

We start by identifying generic hazards on vehicle-level using a keyword-based, HAZOP-inspired brainstorming approach. We start from a set of basic scenarios and a set of basic maneuvers that are chosen according to the ODD of the ADS. For each combination of basic scenario and basic maneuver, we systematically determine the observable effects of this behavior, potential hazards and additional environmental conditions triggering these hazards. The results of this step are denoted in a modified HAZOP table (see Table 1), which consists of 9 columns. In the 1st column (cln) we denote a unique **ID** for later reference. The **basic scenario** and **basic maneuver** under consideration are entered in the 2nd and 3rd cln respectively. While the set of basic scenarios is highly dependent on the ODD, the basic maneuvers (BM) form a subset of the set of all maneuvers that an (automated) vehicle can perform [16], i.e. $BM(ODD) \subseteq \{\text{start, follow, approach (includes braking), pass, traverse crossover, lane change, turn left/right, turn back, park, safe stop}\}$. In the 4th cln **Correct (if context)** we denote the context in which this vehicle behavior would be considered correct. Now we apply a set of **Keywords**, denoted in the 5th cln, to the respective maneuver under consideration to determine possible **Incorrect Vehicle Behavior (IVB)**, denoted in the 6th cln. We propose the following list of keywords

Table 1. Table (two exemplary rows) for Identification of Hazards on Vehicle-Level.

ID	Basic scenario	Basic maneuver	Correct if (context)	Keyword	Incorrect vehicle behaviour	Observable effect(s) in scenario	Additional scenario conditions (necessaroy for Top Level Event)	Potential top level event
1	Slower turn into path challenger	Decelerate/braking	Front distance < safety distance	no	Necessary breaking maneuver not initiated	Ego continues with constant speed	Challenger with significantly lower speed or critical Time-To-Collision	Front/side collision with challenger
2				less	Braking maneuver not strong enough	Ego does not decelerate to prevent collision	Challenger with significantly lower speed or critical Time-To-Collision	Front/side collision with challenger

- **no, less, more, too early, too late** (classical HAZOP keywords)
- **non-existent, too large, too small, too many, too few, not relevant, physically not possible** (specific keywords for driving assistance systems according to the Sense-Plan-Act paradigm [4])
- **provided in inappropriate context, stopped too soon, provided too long** (STPA-inspired keywords)
- **outdated, misapprehended, inappropriate, falsified, too slow, too fast** (keywords found to be relevant during application of the method)
- **wrong** (generic, only to be applied if no other keyword is applicable)

Based on the IVB, the 7th cln is filled with **Observable Effect(s) in Scenario**. These effects are then used to derive potential hazards at vehicle level (top level events) in the 9th cln **Potential Top Level Event**. If **Additional Scenario Conditions are Necessary for Top Level Event** to happen, this is denoted in the 8th cln. The results of step (2.1) are essentially independent of the concrete implementation and can be used for the development of other ADS within the same ODD.

Step (2.2): Identification of Functional Insufficiencies with Hazardous Effects. Now we systematically apply the keywords from step (2.1) to the triple (Input, Computation, Output) for each functional unit (FU) of the system in order to examine deviations that may lead to hazards. In particular, we analyze the effects of these deviations locally, system-wide and on vehicle-level. Again, the results are documented in a table consisting of 11 columns (Table 2).

Table 2. Table for Identification of Local Failures/Functional Insufficiencies.

Functional Unit	Function			Key-word	Local Failure / Functional Insufficiency	Basic Scenario	System Effect(s) in Scenario	Incorrect Vehicle Behavior	ID(s) of IVB	Possible System Cause(s)	Environmental Condition	Relevant for human driver?
	Input	Computation	Output									
Sensors > Front camera > object recognition	camera image	segmentation	segmented camera image	no	segmented camera image not generated	slower turn into path challenger	challenger not detected by front camera > maneuver planning without information about the challenger	necessary braking maneuver not initiated	1	HW-failure, degradation or design fault	none	no statement
			no segments in camera imaged recognized		s/a	s/a	s/a	s/a	no night vision, lacking sensibility at dark	darkness	likely (human vision also impaired by darkness)	

Using the model of the ADS from step (1), we denote in the 1st cln the considered **Functional Unit** followed by the triple (Input, Computation, Output) in the 2nd cln. Then we check whether this triple, in combination with a **Keyword** (3rd cln) exhibits incorrect behavior that leads to a **Local Failure/Functional Insufficiency** (4th cln). Afterwards, the worst-potential consequences of these are investigated (bottom-up, inductive). This is done separately for each **Basic Scenario** (5th cln). Based on this we derive negative **System Effect(s) in this Scenario** (6th cln) leading to **Incorrect Vehicle Behavior** (IVB) on vehicle

level (7th cln). If the respective IVB was already identified in step (2.1), we denote the corresponding **ID** in the 8th cln. Otherwise, go back to step (2.1) and create a new row in the table for this IVB. Additionally, local causal chains are already identified here. If they exist, we denote **Possible System Cause(s)** (9th cln) as well as **Environmental Trigger(s)** (10th cln). Finally, in the 11th cln, we rate a human driver's ability to cope with the environmental condition. If a human driver is also likely to struggle in this situation, it can be argued that no 'new' cause of risk was identified here. For this, we propose an estimation using an ordinal scale (e.g. no statement, very unlikely, unlikely, likely very likely). Optimally, this estimation is supported by measured real-world data obtained from accident databases (e.g. GIDAS³), Field Operational Tests, Naturalistic Driving Studies, simulator studies or tests on proving grounds.

Step (3): Causal Chain Analysis analyzes environmental conditions that were identified as 'triggering' in step (2.2) thoroughly. While in step (2.2) we merely considered single causes and conditions, the goal here is to identify all combinations of triggering environmental conditions. The necessity (or expendability) of a Causal Chain Analysis must not be based on the above estimates for humans alone, but on expert judgment. It is performed via an extended fault tree analysis focused on identifying design- and specification faults (i.e. systemic faults) that may, in conjunction, expose hazards. These faults are inherent in the system, but usually only lead to actual hazards under additional conditions. These can be other faults in the system, but also environmental conditions.

In order to be able to identify and model them, we extend the classical fault tree analysis [20] by using inhibit-gates to specify environmental conditions (rather than classical events inside the system) as being necessary for the propagation of a fault. Using such an **environment fault tree (EFT)**, environmental conditions can be modeled as basic events that trigger higher-level faults. A hazard H (identified in step (2.1)) constitutes the Top Level Event and therefore the root of the EFT. Additionally, we assume as context the corresponding basic scenario Z . Starting from the Top Level Event H , we create a new node in the tree and connect it to the root (using AND/OR-Gates), for every deviation D from the correct behavior of an internal signal S . The hypothesized causes for these deviations are then subdivided into (random) hardware faults, (systemic) design faults in hardware or software, (systemic) specification faults, i.e. fault in some (sub-)specification, either due to incorrect assumptions (SOTIF) or lacking structural approach. Orthogonal to these classes of faults are so-called propagated faults, i.e. the input of a component is already erroneous; these can be either random or systemic faults. In order to uncover systemic faults, it is particularly useful to compare its functionality (and potential faults) to conventional vehicle operation by human drivers. Additionally, we model every combination of environmental conditions that propagate systemic faults to the underlying FU using inhibit-gates. This process is iterated until we either arrive at the level of

³ German In-Depth Accident Study - www.gidas.org.

perception or the corresponding FU has no further output and thus, there cannot be any more propagated faults. This way of constructing EFTs focuses on identifying systemic faults that arise newly in the context of automated driving and are not already covered by accident databases for conventional vehicles. The first levels of a generic EFT are illustrated in Fig. 3.

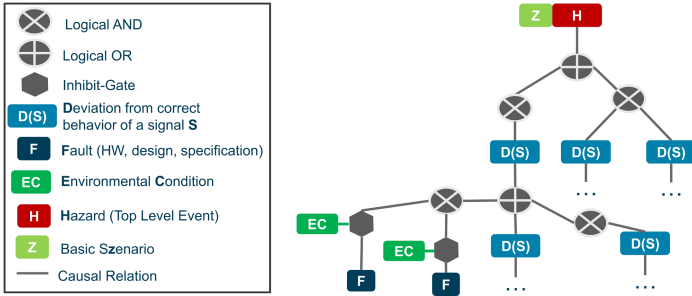


Fig. 3. Generic structure of an environment fault tree (EFT).

The input on the level of perception (i.e. sensory data) is the ADS’s environment. For each type of sensor we can use its characteristics to identify environmental conditions that might cause the faults at the leaves of the EFT, e.g. rain drops, glare or reflections confusing the camera, metal reflections irritating the radar or objects with bad light reflection compromising the lidar.

Step (4): Environment Model builds a model of the environment that is first used for expressing the environmental conditions from Step (3) and later on for the translation into the output scenario specification language. This intermediate step ensures independence from the limitations of a specific scenario modeling language. The environment model has to be built with regard to the ADS and its ODD. Iterative refinements of this model may become necessary later in the process (as indicated by the dashed arrow from (5) back to (4) in Fig. 1). In the context of PEGASUS, we built an exemplary environment model for the ODD ‘German Highway’ which is based on the functional description of the Highway-Chauffeur [15] and the PEGASUS-ontology as described in [3].

Step (5): Derivation of Hazard Triggering Scenario Properties formalizes the previously identified environmental conditions for individual faults such that these are unambiguous and formulated in a language suitable for the description of scenarios. From this we derive properties of scenarios that potentially trigger the corresponding hazard. First, each of the EFTs from step (3) is reduced to those nodes that represent environmental conditions while maintaining the logical structure of the tree. The next step consists of a Common Cause Analysis [7] and expressing the environmental conditions in the reduced tree

using the environment model from step (4). Here, it may be necessary to introduce some extra nodes (using AND-/OR gates) in the tree in case the non-formal descriptions of the environmental conditions contain implicit con-/disjunctions. If the environment model is not or only insufficiently able to represent some EC, we have to go back to step (4), as indicated by the dashed back arrow in Fig. 1, and extend our environment model accordingly. As of now, the identified environmental conditions are still described statically, although the events may have to occur chronologically in order to actually trigger the hazard. Therefore, the formalized environmental conditions are divided into discrete time steps $(\dots, t_{-1}, t_0, t_1, \dots)$ corresponding to scenes $S(t_i)$ such that the relative ordering $S(t_{i-1}) < S(t_i)$ describes a possible evolution of situations in time likely to trigger a hazardous event, resulting in a hazardous scenario $Sc = \{S(t_i)\}_i$. Here, $S(t_0)$ is the starting scene of the scenario, while $\{S(t_i)\}_{i < 0}$ describe previous scenes and $\{S(t_i)\}_{i > 0}$ describe possible evolutions of $S(t_0)$, as illustrated in Fig. 4.

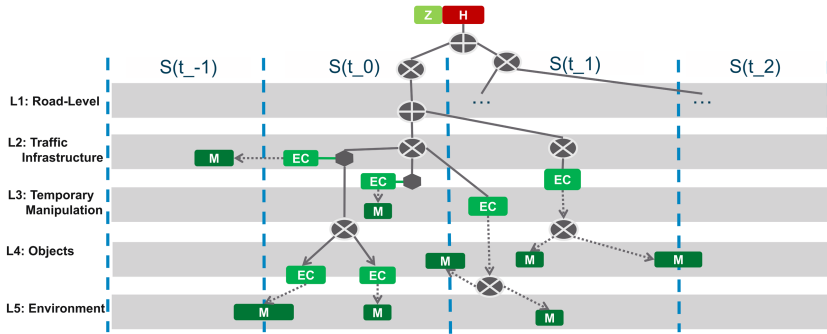


Fig. 4. Introducing chronological ordering of the events.

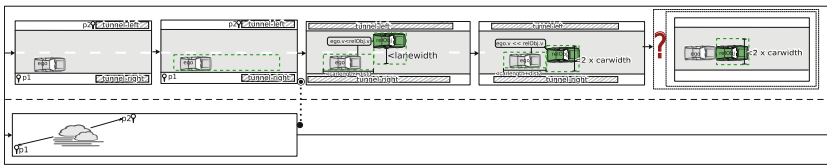


Fig. 5. A hazardous scenario specified as Traffic Sequence Chart.

For each EC, the pertinent time to trigger the hazard (according to the logical structure) must be determined. Here, the environmental conditions may extend over multiple time steps. In a final step, the identified hazardous scenarios should be specified using a sufficiently powerful language for specification of traffic scenarios which allows formal expression of environmental conditions on an

adequate level of abstraction, such as Traffic Sequence Charts (TSCs) [8]. Hazardous scenarios, that are output of the presented method, are more abstract than logical scenarios, i.e. scenarios described as parameter ranges, but more concrete than functional scenarios, i.e. scenarios described using natural language (cf. [13]). Figure 5 depicts a hazardous scenario that was identified during the application of the method to the PEGASUS Highway-Chauffeur, specified as a TSC. In this scenario, the weather is foggy (impacting the Lidar) while the ego-vehicle (gray car) enters a tunnel (bad lighting conditions impacting the cameras). Subsequently, the green car, driving much slower than the ego, challenges the ego to react by turning into its path from the left (therefore, not being in the field of view of ego’s front radar). The final snapshot of the TSC points at the potential hazard.

4 Method for Quantification of Hazardous Scenarios

Based on the results of the method presented in Sect. 3, we now outline how to quantify the identified hazardous scenarios and how the associated risk can be assessed and classified. The proposed method for quantification corresponds to steps (6)–(7) in Fig. 1. We propose to integrate the possibility of iteratively implementing different risk mitigating measures in order to reduce the risk to tolerable levels. Moreover, we sketch how imposing requirements on the probability classification of hazardous scenarios can be used to derive requirements on error rates.

Step (6): Quantification and Risk Assessment. According to the ISO 26262 [9], risk can be described as a function of the probability of occurrence, the controllability and the potential severity. In automated driving the passengers have very limited control over the driving task and controllability only applies to persons outside the ADS. Thus, it is highly questionable whether controllability should be a parameter for risk assessment of ADS. However, assessing the risk associated to a hazard by estimating its probability of occurrence and its severity remains a valid strategy. We propose obtaining an upper bound on the probability of occurrence of a hazardous scenario by quantification of the context, i.e. estimating the exposure of a ADS to the triggering environmental conditions that define the hazardous scenario (as identified in Sect. 3). Let H be a hazard occurring in the context of scenario Z and let c_1, \dots, c_m be the Environmental Conditions (ECs) corresponding to the reduced, formalized EFT from Step (5). We quantify and assess the risk $R(H \cap Z)$ using the following steps:

- (1) Quantify the ECs c_1, \dots, c_m as **probability of occurrence** per hour of driving (exposure), i.e. $e_1 = P(c_1), \dots, e_m = P(c_m) \in [0, 1]$. Optimally, this happens on the basis of data that is representative for the ADS’s ODD. If that is not possible, choose upper bounds $e_1 \geq P(c_1), \dots, e_m \geq P(c_m) \in [0, 1]$ on the basis of exposure catalogues and/or expert judgement.
- (2) For each EC c_i determine the **error rate** $\mu_i \in (0, 1]$, i.e. the probability that the error propagates in the fault tree under the assumptions that c_i occurs.

If to some EC an error rate cannot be associated or it is simply unknown at this point, pessimistically set $\mu_i = 1$.

- (3) Every Minimal Cut Set (i.e. conjunction of triggering ECs) of the reduced, formalized environment fault tree corresponds to a sub-scenario $Z_j(j = 1 \dots n)$ of Z . The probability of the hazard occurring in a subscenario is estimated as $P(H|Z_j) \leq \prod_{i=1, \dots, m: U_i \in Z_j} \mu_i e_i$.
- (4) Under the assumption that Z_1, \dots, Z_n exhaustively cover the scenario Z , we obtain an upper bound B for the probability of occurrence of hazard H in scenario Z via $P(H \cap Z) = P(Z)P(H|Z) \leq P(Z) \sum_{j=1}^n P(H|Z_j) \leq P(Z) \sum_{j=1}^n \prod_{i=1, \dots, m: U_i \in Z_j} \mu_i e_i =: B$.

If B is too large (especially if $B > 1$) it is not a useful bound and the process should be reiterated under use of more accurate values for e_i and μ_i . If $B \leq 1$ is reasonably small, it can be sorted into an a probability class, i.e. $B \in P_k$ for some k where $\bigcup_k P_k = (0, 1]$, according to its order of magnitude, e.g. $E_1 = (0, 10^{-7}] / h$, $E_2 = (10^{-7}, 10^{-5}] / h$, $E_3 = (10^{-5}, 10^{-3}] / h$, $E_4 = (10^{-3}, 1] / h$.

- (5) Estimate the potential severity $S(H \cap Z)$ of hazard H in scenario Z and sort it into S_0, S_1, S_3, S_4 according to the classification in [9, Table B.1].

- (6) Finish the risk assessment by determining whether risk mitigating measures (RMMs) are necessary, using an appropriate table featuring the dimensions ‘probability’ and ‘severity’, see e.g. Fig. 5. In contrast to the automotive safety integrity level (ASIL) assigned to a hazardous event in the ISO 26262 process, this risk assessment indicates

R	E_1	E_2	E_3	E_4
S_0	nM	nM	nM	nM
S_1	nM	nM	nM	M_1
S_2	nM	nM	M_1	M_2
S_3	nM	M_1	M_2	M_3

Fig. 6. Table for Risk Assessment.

whether RMMs have to be implemented or not (nM =no measures); and if RMMs are necessary, how impactful do they need to be ($M_1 < M_2 < M_3$) in order to reduce $R = R(H \cap Z)$ to a tolerable level (Fig. 6).

Step (7): Derivation of Requirements checks whether regulatory guidelines and requirements have been complied with. The existence of such requirements/guidelines is a prerequisite for step (7), because as long as there are no guidelines for automated driving, no requirements for error rates or exposures can be derived. Reducing the risk $R(H \cap Z)$ can be realized by either (i) using tighter bounds for the exposures e_i , (ii) using more exact values for the error rates μ_i , (iii) implementing and verifying risk mitigating measures (RMMs). While changes of type (i) or (ii) require reiteration of step (6) using the updated values, the identification part of the method, i.e. steps (1)–(5), does not have to be repeated. The method indicates this possibility by the dotted arrow from step (7) back to step (6) in Fig. 1. However, this is no longer true for RMMs: they do lead to (far-reaching) changes in the ADS and can trigger a complete reiteration of the method, indicated by the dotted arrow from step (7) back to step (1).

Classes of Risk Mitigating Measures (RMMs). Each RMM can be effective by reducing exposure (E-effective) or severity (S-effective). We distinguish four classes of RMMs: (1) **Functional Safety Measures** according to ISO 26262, such as implementing redundancies, monitors, fault-resistant or reconfigurable systems, can be E-effective (reduction of error rates) or S-effective, (2) **Restriction of the ODD** to exclude hazardous scenarios is E-effective, (3) **Behavioral Safety Measures**, such as a more defensive driving profile for the ADS, can be E-effective (e.g. proactive driving) or S-effective (e.g. keeping larger safety distances, driving more slowly), (4) **External Measures**, such as better traffic infrastructure for ADS or changes in law that aid automated driving. External measures passively mitigate risks (S-effective or E-effective) in the long-run. After implementing a RMM, depending on its class, its effectiveness has to be verified appropriately. Beware that implementing RMMs may lead to crucial changes in the ADS's functional architecture, its ODD, its behavior or its environment and may therefore invalidate the results of steps (1)–(5).

Derivation of Requirements on Error Rates. Under the assumption that there exists a regulatory requirement on the E -classification of a hazardous scenario, stated as $P(H \cap Z) \in E_k$ for some k , it is possible to derive **requirements on the error rates** μ_i such that in the next iteration of step (6) evaluating $P(H \cap Z)$ will fall into probability class. More precisely, the requirement on the E -classification of $P(H \cap Z)$ induces an inequality for every sub-scenario $Z_j (j = 1, \dots, n)$ that can be reformulated as requirements on the error rates. Even if not all of these n equalities can be satisfied, formulating these requirements gives us an idea about the required error rates. Let H be a hazard occurring in the context of scenario Z , let e_1, \dots, e_m be upper bounds for exposure and μ_1, \dots, μ_m the associated error rates from the previous iteration of step (6). We derive conditions on the error rates which ensure $P(H \cap Z) \in E_k$ using:

- (1) Translate the requirement into a worst-case probability $p_{\max} \in (0, 1]$ such that $P(H \cap Z) \leq P(Z) \sum_{j=1}^n P(H|Z_j) \leq p_{\max}$. Here, p_{\max} should be chosen as the upper bound of the interval E_k .
- (2) From each sub-scenario $Z_j (j = 1, \dots, n)$, derive the requirement $\prod_{i=1, \dots, m: U_i \in Z_j} \mu_i e_i = P(H|Z_j) \leq p_{\max}/(n \cdot P(Z))$. These can be aggregated into n requirements on the product of error rates $\prod_{i=1, \dots, m: U_i \in Z_j} \mu_i \leq p_{\max}/(n \cdot P(Z) \cdot \prod_{U_i \in Z_j} e_i)$ for $j = 1, \dots, n$.

In total, we obtain n requirements on products of error rates. In case, the exact number of sub-scenarios n and/or the probability $P(Z)$ are unknown, upper bounds for these values should be approximated.

5 Conclusion

The approach presented in this paper is a first step towards an integrated safety assessment for automated driving systems. On the one hand, it identifies relevant scenarios for scenario based testing while on the other hand deriving

requirements for the further development of the ADS. Application to the PEGASUS Highway-Chauffeur demonstrated usability of the method in practice, more examples can be found in [5]. A full assessment on how well it performs to identify potential hazards in comparison to e.g. STPA or existing standards will need to be evaluated thoroughly and can (as usual with safety analysis methods) only be determined later (e.g. through comparisons of hazard rates in real traffic).

For automation at SAE levels 4/5 and more complex environments (e.g. urban areas), as addressed in the PEGASUS follow-up projects ‘VVMethoden’ and ‘SET Level 4to5’, we aim to extend our method towards analyzing structural criticalities in traffic and thus combining the expert based analysis with a data driven approach. Concerning the specification of hazardous scenarios, we plan on extending Traffic Sequence Charts in order to capture more accurately the critical phenomena and associated causal relations that trigger hazardous behavior.

References

1. Abdulkhaleq, A., Baumeister, M., Böhmert, H., Wagner, S.: Missing no Interaction - using STPA for identifying hazardous interactions of automated driving systems. *Int. J. Saf. Sci.* **02**, 115–124 (2018)
2. Akhtar, N., Mian, A.: Threat of adversarial attacks on deep learning in computer vision: a survey. *IEEE Access* **6**, 14410–14430 (2018)
3. Bagschik, G., Menzel, T., Maurer, M.: Ontology based scene creation for the development of automated vehicles. In: *IEEE Intelligent Vehicles Symposium (IV)*, pp. 1813–1820 (2018)
4. Bagschik, G., Reschka, A., Stolte, T., Maurer, M.: Identification of potential hazardous events for an Unmanned Protective Vehicle. In: *IEEE Intelligent Vehicles Symposium (IV)*, pp. 691–697 (2016)
5. Böde, E., et al.: Identifikation und Quantifizierung von Automationsrisiken für hochautomatisierte Fahrfunktionen. Tech. report, OFFIS e.V. (2019)
6. International Electrotechnical Commission, International Electrotechnical Technical Commission, et al.: Hazard and operability studies (HAZOP Studies)-Application guide. BS IEC 61882 (2001)
7. S.I.S. Committee, et al.: Guidelines and Methods for Conducting the Safety Assessment Process on Civil Airborne System and Equipment. SAE International (1996). <https://doi.org/10.4271/ARP4761>
8. Damm, W., Möhlmann, E., Peikenkamp, T., Rakow, A.: A formal semantics for traffic sequence charts. In: Lohstroh, M., Derler, P., Sirjani, M. (eds.) *Principles of Modeling*. LNCS, vol. 10760, pp. 182–205. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-95246-8_11
9. ISO: ISO 26262:2018: Road vehicles - Functional safety (2018)
10. ISO: ISO/PAS 21448: Road vehicles - Safety of the intended functionality (2019)
11. Kalra, N., Paddock, S.M.: Driving to safety: how many miles of driving would it take to demonstrate autonomous vehicle reliability? *Transp. Res. Part A: Pol. Pract.* **94**, 182–193 (2016)
12. Leveson, N.G.: STAMP: an accident model based on systems theory. In: *Systems Thinking Applied to Safety, Engineering a Safer World* (2012)

13. Menzel, T., Bagschik, G., Maurer, M.: Scenarios for development, test and validation of automated vehicles. In: IEEE Intelligent Vehicles Symposium (IV), pp. 1821–1827. IEEE (2018)
14. PEGASUS: Critical Scenarios for and by the HAD (2017). www.pegasusprojekt.de/files/tmpl/PDF-Symposium/06_Critical-Scenarios-for-and-by-the-HAD.pdf
15. PEGASUS: The Highway Chauffeur (2019). www.pegasusprojekt.de/files/tmpl/Pegasus-Abschlussveranstaltung/04_The_Highway_Chauffeur.pdf
16. Reschka, A.: Fertigkeiten-und Fähigkeitengraphen als Grundlage des sicheren Betriebs von automatisierten Fahrzeugen im öffentlichen Straßenverkehr in städtischer Umgebung. Ph.D. thesis, TU Braunschweig (2017)
17. SAE, T.: Definitions for Terms Related to On-Road Motor Vehicle Automated Driving Systems. J3016, SAE International Standard (2014)
18. Steck, J.: Methodological Approach to Identify Automation Risks of Highly Automated Vehicles Using STPA. Technische Universität München, Masterarbeit (2018)
19. Ulbrich, S., Menzel, T., Reschka, A., Schuldt, F., Maurer, M.: Defining and substantiating the terms scene, situation, and scenario for automated driving. In: 2015 IEEE 18th International Conference on Intelligent Transportation Systems, pp. 982–988. IEEE (2015)
20. Vesely, W.E., Goldberg, F.F., Roberts, N.H., Haasl, D.F.: Fault tree handbook. Tech. report Nuclear Regulatory Commission Washington DC (1981)
21. Yan, F., Tang, T., Yan, H.: Scenario based STPA analysis in Automated Urban Guided Transport system. In: 2016 IEEE International Conference on Intelligent Rail Transportation (ICIRT), pp. 425–431 (2016)