

Is Scenario Generation Ready for SOTIF?

A Systematic Literature Review

Lukas Birkemeyer¹, Christian King², and Ina Schaefer³

Abstract—Scenario-based testing is considered state-of-the-art to verify and validate Advanced Driver Assistance Systems or Automated Driving Systems. Due to the official launch of the SOTIF-standard (ISO 21448), scenario-based testing becomes more and more relevant for releasing those Highly Automated Driving Systems. However, an essential missing detail prevents the practical application of the SOTIF-standard: How to practically generate scenarios for scenario-based testing? In this paper, we perform a Systematic Literature Review to identify techniques that generate scenarios complying with requirements of the SOTIF-standard. We classify existing scenario generation techniques and evaluate the characteristics of generated scenarios wrt. SOTIF requirements. We investigate which details of the real-world are covered by generated scenarios, whether scenarios are specific for a system under test or generic, and whether scenarios are designed to minimize the set of unknown and hazardous scenarios. We conclude that scenarios generated with existing techniques do not comply with requirements implied by the SOTIF-standard; hence, we propose directions for future research.

I. INTRODUCTION

In recent years, the automation of Advanced Driver Assistance System (ADAS) increases aiming for more safety and comfort in road traffic. The gap between highly automated ADAS and Automated Driving System (ADS) seems to disappear. Established methods as provided in ISO 26262 [1] are no longer sufficient for the Verification and Validation (V&V). The standard "Safety of the Intended Functionality" (SOTIF) (ISO 21448) [2] and the UN/ECE regulation R157 [3] require Scenario-based Testing (SBT).

SBT investigates the behavior of a System Under Test (SUT) in a driving scenario. The scenario consists of a sequence of static scenes which are "*snapshot[s] of the environment including the scenery [and] dynamic elements, [...] and the relationships among those entities.*" [2], [4]. The scenery describes lane information, static elements (e.g., traffic lights) and weather conditions. Dynamic elements are road users such as pedestrians or other vehicles. There are different abstraction levels of scenarios [5]: *Logical scenarios* describe the main semantics of a scenario; logical scenarios contain ranges of possible parameters – e.g., a vehicle drives in a city with [20;60] km/h and a pedestrian crosses the street with [2;6] km/h. To perform SBT, logical scenarios need to be concretized into a lower abstraction level (*concrete scenario*) by defining concrete parameter

values [6]. In this paper, we use the term *scenario generation* to describe the process of deriving concrete scenarios from logical scenarios.

The SOTIF-standard was officially published in July 2022; a preliminary version (ISO/PAS 21448) [7] is public since January 2019. Current literature suggesting scenario generation approaches is often motivated by SOTIF but does not discuss the suitability of generated scenarios wrt. the standard. For example, SOTIF requires that the operational domain and foreseeable misuse are covered in the SBT process. SOTIF also requires to focus on scenarios that lead to hazardous behavior of the SUT. Still, SOTIF misses details hindering a proper release of highly automated ADAS and ADS [8]. SOTIF neither specifies techniques to generate scenarios, nor metrics to assess whether generated logical or concrete scenarios (individual or as scenario suite) fulfill the SOTIF requirements. Hence, for testers it is not clear how to design *good* or *better* scenarios for testing; for manufacturers, it is not clear when to release ADS with acceptable residual risk; for legislators, it is unclear which scenario suites are suitable to allow ADS on public streets. Without solving open challenges in scenario generation processes, a proper release of ADS according to the SOTIF-standard is not possible.

Current research activities [9]–[21] present and evaluate approaches to generate scenarios. However, they do not evaluate whether generated scenarios comply with SOTIF requirements. In this paper, we analyze characteristics of scenario generation approaches and pursue the research goal: Identification of approaches that generate scenarios complying with requirements of SOTIF to practically apply scenario-based testing.

In this paper, we perform a Systematic Literature Review (SLR) and make the following contributions:

- We structure existing scenario generation approaches according to their scenario generation technique.
- We analyze which existing scenario generation techniques generate scenarios that comply with SOTIF requirements.
- We propose directions for future work on scenario generation complying with SOTIF requirements.

II. BACKGROUND

In this section, we introduce the SOTIF-standard [2] to verify and validate ADAS/ADS. We also provide fundamental knowledge of a scenario structure that we use in this SLR to make the paper self-contained.

¹Lukas Birkemeyer is with Technical University Braunschweig, Braunschweig, Germany l.birkemeyer@tu-braunschweig.de

²Christian King is with FZI Forschungszentrum Informatik, Karlsruhe, Germany king@fzi.de

³Ina Schaefer is with Karlsruhe Institute of Technology, Karlsruhe, Germany ina.schaefer@kit.edu

A. SOTIF

The "Safety of the Intended Functionality" (SOTIF)-standard (ISO 21448 [2]) complements the established ISO 26262 [1]. The SOTIF-standard proposes SBT to avoid unintended behavior of an ADAS/ADS that is traceable to the interaction of the driving function with the environment. SOTIF classifies scenarios into four classes, deciding whether a scenario is *known* or *unknown* and *hazardous* or *not hazardous*. "Hazardous scenarios are scenarios causing *hazardous behavior*" [2]. Hazardous behavior, in turn, potentially leads to *harm*. The goal of SOTIF is to minimize the set of *unknown* and *hazardous* scenarios. SOTIF processes analyze SUTs for hazards and functional insufficiencies as well as related triggering conditions to define a Verification and Validation (V&V) strategy. Based on the SBT results, SOTIF estimates whether the "*residual risk*" or the "*likelihood of encountering an unknown scenario leading to hazardous behaviour [is] sufficiently small*" [2]. Methods or metrics to determine the size of the hazardous scenario set (i.e. the number and occurrence probability of hazardous scenarios) are not provided; instead, SOTIF suggests 16 methods to evaluate residual risk (cf. Table 11 [2]). However, knowledge regarding completeness wrt. the real-world is necessary for a proper safety argumentation. The SOTIF-standard requires scenarios to fulfill two essential requirements:

Req-1: Generated scenarios have to model the overall ODD and foreseeable misuse scenarios. According to SOTIF, we have to consider Operational Design Domain (ODD)-boundaries and foreseeable misuse in the V&V process [2]. The SOTIF-standard defines an ODD as "*specific conditions under which a given driving automation system is designed to function*" [2]. These conditions include a precise definition of, e.g., the type of road, weather conditions, traffic events. To make sure, that a SUT properly identifies whether it is within or outside of an ODD, we need to test inside and outside scenarios [6]. Wrt. ADAS (and partly ADS) a limited ODD and foreseeable misuse might be sufficient. Regarding full automated driving, SOTIF implicitly requires generated scenarios that cover the real-world.

Req-2: Generated scenarios contribute to minimizing the set of (unknown) hazardous scenarios. The goal of SOTIF activities is to determine the residual risk resulting from unknown scenarios [2]. Minimizing the number of (unknown) hazardous scenarios leads to a lower risk.

B. Structure of a Scenario

To determine covering completeness of generated scenarios, we need to compare scenarios. Bagschik et al. [5] structure, in accordance with SOTIF, a scenario in five levels. The first three levels (*E1–E3*) define the static vehicle environment, while the levels *E4* and *E5* define dynamic elements and weather conditions. Scenario level *E1* describes the trajectory and surface of a road. Scenario level *E2* defines the equipment of the road such as lane marking, traffic lights and signs. The third level *E3* contains temporal adjustments of the scenario levels *E1* and *E2*. These temporal adjustments last longer than one day, containing, e.g., construction sites or

road closures. Scenario level *E4* describes dynamic elements such as vehicles, pedestrians or animals and level *E5* contains weather conditions such as snow or rain.

III. STATE-OF-THE-ART & PROBLEM STATEMENT

Current literature [6], [22]–[24] suggests a binary classification of scenario generation techniques: (1) data-driven and (2) knowledge-based. Data-driven techniques derive scenarios from databases and knowledge-based techniques generate scenarios based on expert knowledge e.g., in the form of traffic regulations. This classification does not cover existing scenario generation approaches that use optimization techniques such as [15], [25]. In these approaches, a fitness function is defined by expert knowledge, but the actual selection of concrete parameters is neither data-driven nor knowledge-based. Ding et al. [26] consider a third scenario generation technique: adversarial scenario generation. This includes optimization techniques but does not cover combinational scenario generation as provided in [16], [17]. Birkemeyer et al. [16] suggest, but do not validate a more fine granular classification of scenario generation techniques that differentiates between (1) random, (2) data-driven, (3) optimization, and (4) combinational scenario generation. In random scenario generation, parameters are randomly selected, while data-driven approaches derive scenarios from databases as in the binary classification above. Optimization techniques iteratively select and evaluate scenario parameters; combinational scenario generation divides scenarios into atomic blocks and combines them systematically.

Current surveys and SLRs focus on SBT to verify and validate ADAS/ADS [24], [27]–[35]. These reviews partially focus on the scenario generation process; However, the scope is not directed to a classification that covers all existing techniques to generate concrete scenarios except [27] identifies a classification similar to [16] and Schütt et al. [35] propose a taxonomy of scenario acquisition categories and their relations. To the best of our knowledge, there is no survey or SLR discussing suitability of scenario generation approaches regarding SOTIF. Thus, this SLR aims to identify and quantitatively justify a fine granular classification of existing scenario generation techniques based on current literature. It also discusses whether existing scenario generation techniques are suitable to generate scenarios that meet requirements implied by SOTIF which is novel.

IV. PLANNING THE LITERATURE REVIEW

Our SLR is structured according to guidelines provided by Kitchenham [36] (cf. Figure 1). In this section, we plan the literature review by deriving research questions from our research goal (IV.A), specifying the search process (IV.B) and defining exclusion criteria (IV.C). The following section V describes the execution of the study. Section VI discusses the findings regarding our research questions.

A. Research Questions

Aiming for general findings, we investigate: **RQ1: Which techniques are used to generate scenarios for scenario-**

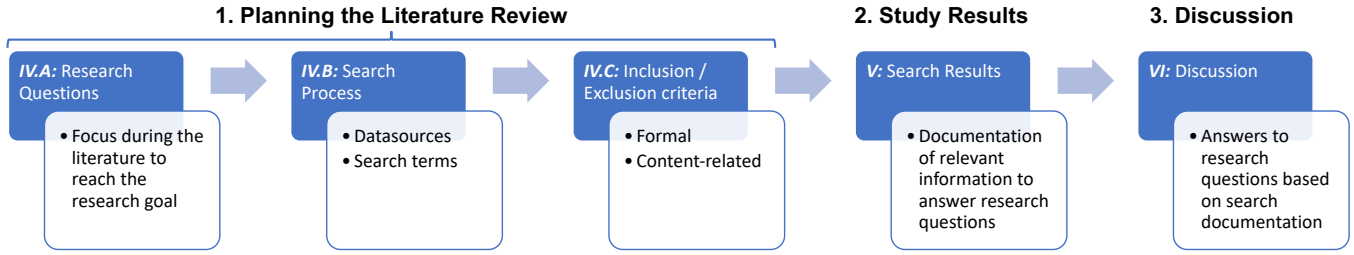


Fig. 1: Guideline to perform a Systematic Literature Review (SLR) according to Kitchenham [36]

based testing? The objective is to cluster existing approaches according to the scenario generation technique that is used.

To reach our research goal – identification of approaches that generate scenarios to practically verify and validate ADAS/ADS according to SOTIF – we define research question **RQ 2: Do existing approaches generate scenarios that comply with the requirements of SOTIF?** We subdivide RQ 2 into the following three aspects.

Completeness in the sense of modeling the real-world is an infeasible objective to reach in SBT [2]. However, completeness of the generated scenarios is essential to calculate residual risks for a proper release of ADAS/ADS. Since we have to consider scenarios that are inside and outside of the SUT’s ODD to avoid foreseeable misuse (cf. Req-1), we are interested in the aspects that are considered by existing scenario generation approaches. We ask: **RQ 2.1: Which details of the real-world are covered by generated scenarios?** The objective is to determine which scenario levels (according to Bagschik et al. [37]) are derived from existing scenario generation techniques. For approaches that address all scenario levels, we identify whether elements of a scenario level are overlooked to potentially point out incomplete coverage in modeling the real-world. Thus, the SOTIF scenario classification and, subsequently, minimizing the set of (unknown) hazardous scenarios (cf. Req-2) significantly depends on the interaction of both, SUT and scenario (including triggering condition). We examine both aspects independently in the research questions RQ 2.2 and RQ 2.3. First, we focus on the correlation of scenario generation and SUT: **RQ 2.2: Are generated scenarios SUT-specific?** The objective is to determine whether system information is required to generate scenarios. Second, we focus on the scenario and potential triggering conditions. We are interested in: **RQ 2.3: Are scenarios designed to trigger hazardous behavior?** The objective is to classify generated scenarios according to the four SOTIF classes. We analyze, whether scenario generation techniques select scenarios that represent an overall scenario space or whether scenarios are explicitly designed to be hazardous. In contrast to RQ 2.1, we do not focus on ODD-completeness.

B. Search process

We systematically identify articles that are relevant for scenario generation by defining a search string and collecting data from search engines by searching for the search string in abstracts (cf. Figure 1, IV.B). We determine terms and

synonyms that enclose the literature of interest. We truncate terms and use placeholders to cover grammatical use cases. We define the search string: *(scenario* OR scenario-based) AND (generat* OR creat* OR select*) AND ("driver assistance" OR "self driving" OR "self-driving" OR ("autonomous" OR "automated")) AND ("driving" OR "car" OR "vehicle")) AND (verif* OR valid* OR test*)*

We collect literature from search engines that are commonly used for reviews of scientific literature. In particular, we focus on *ACM Digital Library*, *IEEE Xplore* and *Google Scholar*. In contrast to the other search engines, *Google Scholar* is not limited to a single library. Moreover, it provides literature that matches the exact search string and related literature. This results in an enormous number of (potentially irrelevant) results. To handle this problem, we start with Google Scholars *most relevant* results and scan abstracts and titles manually. We collect articles that deal with the automotive domain and testing of ADAS/ADS or present simulation tools to generate scenarios until 15 articles in a row do not match these criteria.

C. Exclusion criteria

To make sure, that we only consider relevant articles, we define inclusion/exclusion criteria (cf. Figure 1, IV.C). As a *formal* criterion, we require that relevant articles are peer-reviewed contributions such as conference papers or journal articles ensuring that our study is based on comparable, high-quality articles. Thus, we explicitly exclude magazine articles and extended abstracts. We also require that articles are written in English and publicly available. *Content-related criteria*: We require that articles focus on scenario generation (i.e., the transformation from logical to concrete scenarios) to perform SBT in the automotive context. Another content-related criterion is that a study uses the definition of a scenario according to Ulbrich et al. [4], the SOTIF-standard [2] or similar. We explicitly include articles that present a simulation tool that supports scenario generation and simulation of scenarios, although they do not explicitly present a scenario generation process. This comes with the idea that simulation tools transform discrete input parameters into a continuous physical representation.

V. STUDY RESULTS

We consider articles that are published before 2023 and present, the search results by sharing the numbers of papers that we have found, included and excluded. We substantiate

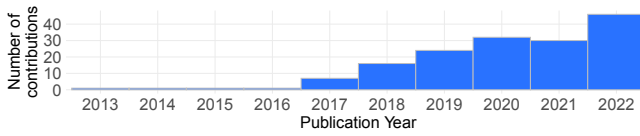


Fig. 2: Number of articles that are published per year.

these results by conducting each step with the four-eye principle of two independent researchers.

Search Results: According to the search process, we collect a total of 892 articles. The search engine of the *ACM Digital Library* provided 122 articles, the search engine of *IEEE Xplore* identified 641 articles and *Google Scholar* provided – combined with manual identification – 129 articles.

Inclusion and Exclusion: We collect the results from the search engines and remove duplicates resulting in a set of 779 articles. We formally exclude 21 articles since they do not comply with our formal criteria. Subsequently, we faithfully check the remaining 758 articles according to content-related criteria; maintaining 159 relevant articles. The excluded articles mainly focus on developing/improving ADAS/ADS or focus on other domains, e.g., aerial vehicles.

Relevant articles: According to the SLR process suggested by Kitchenham [36] (cf. Figure 1), we write a review protocol for each relevant paper. We summarize the paper and note relevant aspects wrt. our research questions. We present all relevant articles and categorizations online¹.

Before we analyze relevant articles wrt. our research questions, we examine their metadata to make sure that our contribution and research questions are novel and significant. First, we consider the publication year of relevant articles. In Figure 2, we plot a histogram showing the number of published articles per year. Between 2017 and 2022, the number of articles per year increases, while the number of articles per year slightly decreases in 2021. However, the increasing trend of published articles relevant for scenario generation and a large number of contributions – even in 2021 – confirm current research interest in this field. Second, we analyze the origin of relevant articles. In Figure 3, we present a histogram of venues that publish relevant articles; publication years are not considered. Venues that publish relevant articles are in the area of Testing, Software Engineering and (automated) Cyber-physical Systems. The main conferences that engage with scenario generation are the International Conference on Intelligent Transportation Systems (ITSC) and the IEEE Intelligent Vehicle Symposium (IV).

VI. DISCUSSION

In this section, we discuss and answer our research questions based on our findings. We say that an *article* proposes one scenario generation *approach*. Regarding RQ 1, we cluster scenario generation *approaches* according to the *technique* that is used. Regarding RQ 2, we focus on the potential and limitations of each *approach* to conclude characteristics of scenario generation *techniques*.

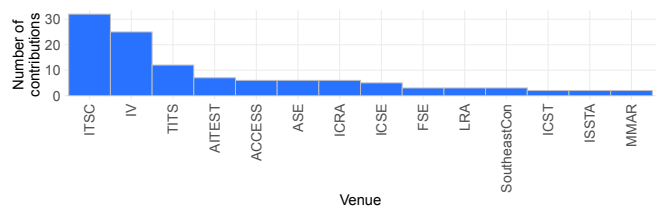


Fig. 3: Number of relevant contributions per venue. For the sake of readability, we only present venues that contribute two or more relevant articles.

RQ 1: Which techniques are used to generate scenarios for SBT? On a high level, we identify five clusters of scenario generation techniques: (1) data-driven, (2) optimization-based, (3) combinational, (4) expert-based, and (5) random.

Data-driven approaches derive scenarios from a database of real-world observation, e.g., by reconstruction [9], [10], applying clustering methods [11], machine learning [12] or statistical analysis [13], [14]. *Optimization-based* scenario generation approaches define a fitness function and generate scenarios to maximize or minimize it. For example, [15] generates scenarios by minimizing the driveable area of the SUT. *Combinational* approaches generate scenarios by systematically selecting and combining (atomic) scenario elements [16], [17]. *Expert-based* scenario generation uses expert knowledge to parameterize a scenario, e.g., by applying traffic rules [18] and considering human behavior [19]. In *random* scenario generation, scenarios are randomly generated, e.g., by randomly selecting parameters [20] or road elements [21].

We assign each scenario generation approach to exactly one technique. However, roughly 10% of scenario generation approaches combine multiple techniques. We assign them to the main technique by identifying their main technique by identifying the technique that mostly impacts the parameter selection while transforming logic into concrete scenarios. We classify approaches that contribute simulation tools to support scenario generation by Computer Aided Design (CAD) as expert-based scenario generation. In Figure 4, we present the distribution of scenario generation techniques used in the relevant articles in this SLR. It is worth noting, that data-driven approaches are the dominating, scenario generation technique. The share of random scenario generation is negligibly small, meaning that scenario generation is predominantly systematic. Except for expert-based scenario generation, current approaches generate scenarios automatically so that they have the potential to handle an extremely large space of possible scenarios.

RQ 2: Do existing scenario generation techniques generate scenarios that fulfill the requirements of the SOTIF-standard? To ensure reliable statements we explicitly avoid statements based on single articles or small clusters. Hence, regarding RQ 2, we do not consider random scenario generation.

RQ 2.1: Which details of the real-world are covered? Although SOTIF requires scenarios that represent the real-

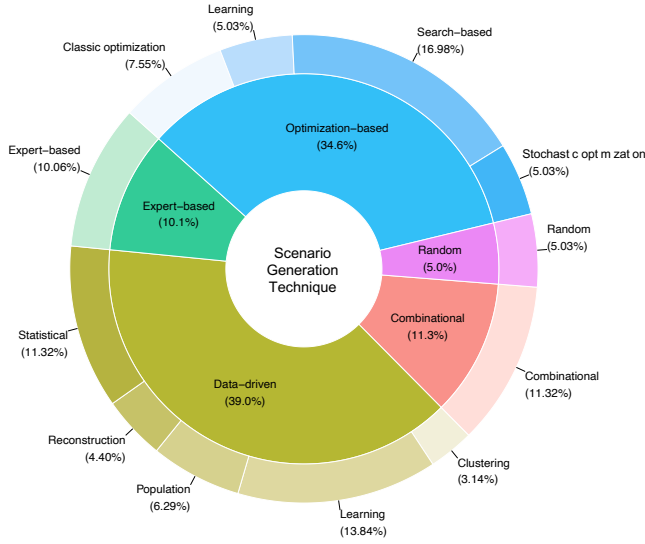


Fig. 4: Existing scenario generation techniques (inner circle) and subdivisions (outer circle), in current literature.

world or a delimited ODD (cf. *Req-1*), SOTIF does not provide a metric to evaluate the completeness of scenario coverage. To evaluate coverage, we use the scenario level structure by Bagschik et al. [37]. If all possible elements of all scenario levels are covered by generated scenarios, we consider the scenario generation approach to be complete in the sense of modeling the real-world. We also determine the number of approaches that contribute to a specific scenario level. We count a contribution if one or more elements of the scenario level are generated and evaluated (e.g., in experiments) or their generation is conceptually discussed. Moreover, we investigate whether a scenario generation approach has the potential to contribute to scenario levels that are not explicitly discussed in the article. Multiple scenario levels per scenario generation approach are possible. In Figure 5, we present the scenario levels to which scenario generation approaches and techniques contribute. We consider the union of all scenario generation approaches (left) and clustered by scenario generation technique (right). The results show that the focus is on scenario level E4; especially, data-driven and optimization-based techniques focus on E4, while combinational and expert-based scenario generation cover all scenario levels approximately equally. On average, scenario level E3 is less covered. The focus of current scenario generation approaches on scenario level E4 might be motivated by the fundamental task of human drivers that contains significant challenges: the interaction with other road users on the path control task [38]. While elements of scenario levels E1 and E2 change slowly and might be part of high definition maps that are deposited in the ADAS/ADS, elements of scenario level E4 change rapidly and are hard to predict, e.g., due to other traffic users. Thus, developing ADAS/ADS that interact with level E4 elements are most challenging and require increased V&V effort, as reflected by current scenario generation approaches.

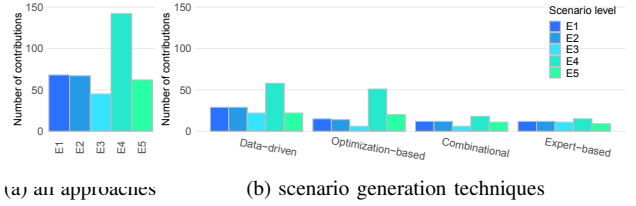


Fig. 5: Covered scenarios levels (E1–E5): (a) for all scenario generation approaches and (b) clustered by scenario generation techniques.

In RQ 2.1, we also analyze the completeness of *individual* scenario generation approaches wrt. the real-world: (1) we analyze, whether all scenario levels are covered; approaches that cover all scenario levels have the potential to be real-world complete; thus, (2) we analyze, whether each scenario level is completely considered. In Figure 6, we show a heatmap that indicates whether a scenario level is addressed in an article. We distinguish between addressing scenario levels with (a) evaluation, (b) concept or (c) potential. Optimization-based and partially also data-driven techniques only focus on E4, while combinational and expert-based scenario generation address all scenario levels. Covering a complete scenario level implies covering a large space of potential elements which is infeasible to assess due to a large number of options. In contrast, it is possible to show incompleteness by finding *missing elements*. We find missing elements for each scenario generation approach that address all five scenario levels. None of the automated approaches (i.e. data-driven, optimization-based, combinational) covers all elements that are probably relevant for a real-world ODD, such as *gravel*, *yellow lane markings*, *defect bulb in a traffic light*, *motorbikes*, or *snow*, etc. Since we determined these missing example elements with expert knowledge, they can in principle be covered in expert-based scenario generation techniques. However, expert-based scenario generation is limited to human knowledge and does not scale for large-scale scenario generation. We argue that limitations of simulation tools might also lead to missing elements of current scenario generation approaches. Regarding *Req-1*, no existing automated scenario generation approach is complete in modeling the real-world and thus, not able to generate scenarios that completely cover it. Covering a delimited ODD is, however, possible.

RQ 2.2: Are generated scenarios SUT-specific? SOTIF requires scenarios to minimize the set of (unknown) hazardous scenarios (cf. *Req-2*). Since this classification is based on both scenario and SUT, we determine, whether scenarios explicitly trigger hazardous behavior of the SUTs by analyzing whether scenario generation requires information of the SUT. As SUT, we consider an overall ADAS or an ADS. Figure 7 presents the share of scenario generation approaches that do (green) or do not (blue) require system information, separated by scenario generation technique. It is remarkable, that optimization-based approaches significantly require system information of the SUT, while data-driven,

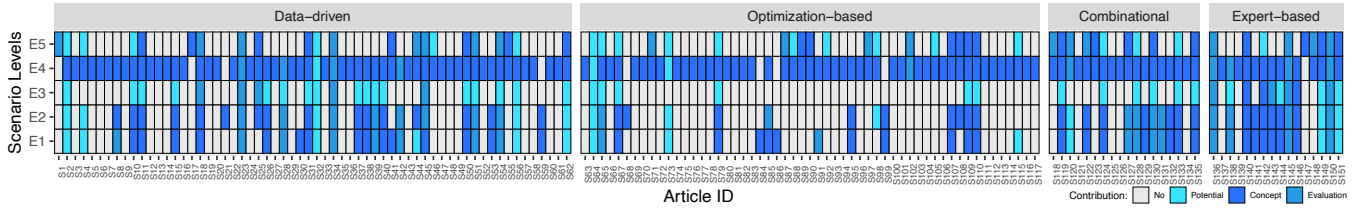


Fig. 6: Scenario levels covered per article for articles S1 - S151; separated by scenario generation technique. The color intensity indicates the focus of an article. The more blue boxes are in a vertical line, the more complete a scenario is covered. The more gray boxes are in a vertical line, the more focused is the article on specific scenario elements.

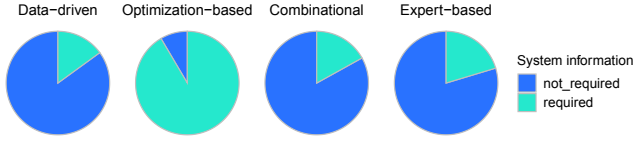


Fig. 7: Share of articles that require system information of the SUT separated by used scenario generation technique.

combinational, and expert-based approaches do not. In the sense of triggering hazardous behavior, approaches that generate SUT-specific scenarios (i.e. wrt. to a specific (type of) SUT), have the potential to outperform approaches that generically generate scenarios. Since SUT-specific scenarios focus on the SUT, they do not holistically represent a scenario space. SUT-specific and generic scenario generation relate to *exploration* vs. *exploitation* of the overall scenario space. Scenario generation that focuses on *exploration* is associated with generic scenario generation discovering *new* (e.g., unconsidered / unknown) areas of scenario spaces. In contrast, *exploitation* is associated with SUT-specific scenario generation, aiming for the *best* (e.g. the most critical / hazardous) scenario. However, exploitation might lead to *local* minima/maxima. Hence, minimizing the area of (unknown) hazardous scenarios (cf. *Req-2*) means carefully balancing exploration vs. exploitation.

RQ 2.3: Are scenarios designed to trigger hazardous behavior? As another aspect regarding *Req-2*, we focus on the SOTIF classification of generated scenarios. SOTIF requires minimizing the set of (unknown) hazardous scenarios; thus, scenario generation approaches that generate hazardous scenarios are highly relevant. We determine which SOTIF scenario classes are generated by existing scenario generation approaches; we consider fully automated driving as SUT. Due to a missing quantifiable definition of hazardous scenarios, we separate *hazardous* and *not hazardous* scenarios with expert knowledge. We consider a scenario as *hazardous* when it explicitly implements potential triggering conditions such as near-miss accidents; otherwise as *not hazardous*. Likewise, an objective definition to distinguish between *known* and *unknown* is missing and depends on the stakeholders' perspective. Each stakeholder (e.g. manufacturer, testing instance, etc.) might have their own database of *known* scenarios. In this study, we consider a scenario as *known* if it is based on real-world observations, part of existing databases

or designed by expert knowledge; otherwise, a scenario is *unknown*. Since the behavior of SUTs is unknown for any kind of *unknown* scenario, we consider *unknown* scenarios as both *hazardous* and *not hazardous* as long as the scenarios do not explicitly implement potential triggering conditions such as near misses. In Figure 8, we present the SOTIF scenario classes that are generated by all existing scenario generation approaches (left) and separated by scenario generation techniques (right). The blue dots indicate, to which SOTIF classes existing scenario generation approaches and techniques contribute. Each SOTIF class is covered by at least one scenario generation approach. Since optimization-based techniques exploit scenario spaces wrt. hazardous scenarios, this technique predominantly leads to hazardous scenarios. Although the logical scenario (at the beginning of the optimization) is known, after the optimization process, the concrete scenario can be both known and unknown. Combinational techniques explore the overall scenario space. Depending on the strategy for combining atomic scenario elements, resulting scenarios are both known/unknown as well as hazardous/not hazardous. Data-driven techniques explore the scenario space by collecting real-world data, e.g., by sensor-equipped vehicles. Reconstruction leads to known scenarios; unknown scenarios are possible by varying parameters. Data-driven techniques generate realistic scenarios, while optimization-based and combinational techniques generate artificial scenarios. It needs to be validated whether artificial scenarios properly represent the real-world. Expert-based techniques focus on known/hazardous scenarios.

VII. THREATS TO VALIDITY

Although an SLR is a sound and established method to determine the state-of-the-art, there are threats to its validity.

Internal Validity: A threat to internal validity is that we misunderstand articles so that we incorrectly classify them wrt. their scenario generation technique, covered scenario levels, required system information, or SOTIF scenario classes. To mitigate this threat, we do not make conclusions based on single or small clusters of articles. Moreover, we double-check article classification by two independent researchers and make it traceable by sharing details online. Another threat is that we collected articles from unsuitable sources. To mitigate this threat, we focus on databases and search engines that are commonly used within technical scientific research. A last aspect that threatens the correctness

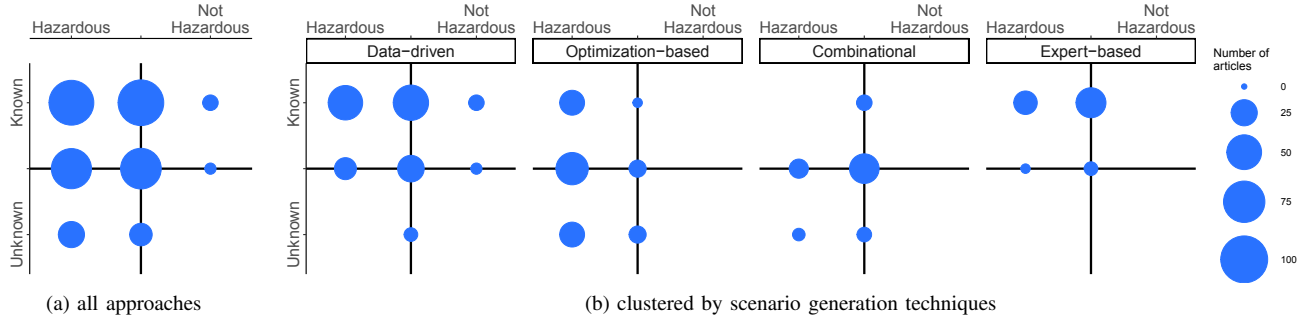


Fig. 8: Scenario classification according to the SOTIF-standard. We present the four scenario classes that are defined in SOTIF. The size and position of the dots indicate the number of approaches that contribute to a specific scenario class.

of our results is that we defined unsuitable search terms to collect literature from databases. To mitigate this threat, we faithfully and iteratively specified the area of interest with expert knowledge from two independent researchers. We determine synonyms and cover arbitrary grammatical forms.

External Validity: A threat to the external validity is that the requirements of SOTIF are an unsuitable scope to discuss existing scenario generation and thus, the results might not be generalizable for V&V of ADAS/ADS. However, we argue that SOTIF is an official standard for V&V of ADAS/ADS and is widely accepted in the automotive domain. We faithfully derived requirements from the latest SOTIF version.

VIII. CONCLUSION & FUTURE RESEARCH

The SOTIF-standard introduces SBT to verify and validate ADAS/ADS. However, SOTIF does not define methods to generate or assess scenarios which hinders the practical applicability of SOTIF. In this article, we analyze existing approaches to generate concrete scenarios and evaluate whether generated scenarios comply with the requirements implied by the SOTIF-standard. We derived five clusters of scenario generation techniques: *data-driven*, *optimization-based*, *combinational*, *expert-based*, and *random*.

Data-driven techniques generate generic and realistic scenarios based on collected data. *Optimization-based* techniques automatically exploit scenario spaces wrt. scenarios that trigger hazardous behavior of the SUT. In contrast to exploiting a hazardous space, *combinational* scenario generation automatically explores the overall scenario space by combining atomic scenario elements. *Expert-based* techniques are not automatic and do not scale for real-world problems. *Random* scenario generation is an automatic, but unsystematic approach which is only marginal in the literature.

Regarding the SOTIF-standard, we need to generate scenarios that cover an ODD and trigger hazardous behavior of the SUT. Existing scenario generation approaches, however, do not fulfill both requirements respectively. First, since complete coverage of the real-world is an infeasible object to reach, we might counteract coverage completeness (cf. *Req-1*) by scenarios that do not cover, but represent an ODD or the real-world. Combinational techniques have the potential to build representative subsets and data-driven scenario generation stands out in generating realistic scenarios.

Second, scenarios need to trigger hazardous behavior of the SUT (cf. *Req-2*), which is the focus of optimization-based scenario generation. However, optimization techniques focus on (local) minima/maxima and do not explore a scenario space; while combinational scenario generation does not explicitly select scenarios that trigger hazardous behavior and data-driven approaches heavily rely on the input-dataset. Hence, as a direction for future work, we strongly suggest combining existing scenario generation techniques. We need to generate scenario suites that carefully balance representativeness (*combinational*), hazardous triggering conditions (*optimization-based*), and realism (*data-driven*). Birkemeyer et al. [39], for example, suggests a concept to integrate optimization techniques in combinational scenario generation and indicate the potential of combining both techniques.

The set of possible scenarios is extremely large and thus, the set of scenarios that represent an ODD or the real-world. Hence, we need to consider scalability during the scenario generation process. Since the scenario generation approaches data-driven, optimization-based and combinational automatically generate scenarios, these techniques have the potential to practically generate scenarios for V&V of ADAS/ADS. Finally, to balance the potential of multiple scenario generation techniques as suggested before, we need to define metrics that assess scenario suites, independently from the scenario generation process. In the sense of V&V, we could apply mutation testing to assess the ability of a scenario suite to detect errors in the SUT [16].

To sum up, scenarios generated with the existing scenario generation approaches do not comply with requirements implied by the SOTIF-standard. To close this gap, we propose the following directions for future research:

- Combining existing scenario generation techniques has potential to generate scenarios that fulfill SOTIF requirements.
- Scenario generation needs to be scalable to generate scenarios suites that represent large-scale ODDs or the real-world.
- A metric that assess scenario suites independently from the scenario generation process is required.

ACKNOWLEDGMENTS

This work has been partially funded by the Ph.D. program "Responsible AI in the Digital Society" funded by the Ministry for Science and Culture of Lower Saxony. This work has been partially supported by the research project SofDCar (19S21002), which is funded by the German Federal Ministry for Economic Affairs and Climate Action.

REFERENCES

- [1] *Road vehicles — Functional safety*, Std. ISO 26262:2018, 2018.
- [2] *Road vehicles — Safety of the intended functionality*, Std. ISO 21448:2022, 2022.
- [3] "UN regulation no 157 – uniform provisions concerning the approval of vehicles with regards to automated lane keeping systems [2021/389]," <http://data.europa.eu/eli/reg/2021/389/oj>, pp. 75–137, Mar 2021.
- [4] S. Ulbrich, T. Menzel, A. Reschka, F. Schuldt, and M. Maurer, "Defining and substantiating the terms scene, situation, and scenario for automated driving," in *IEEE International Conference on Intelligent Transportation Systems (ITSC)*, 2015, pp. 982–988.
- [5] G. Bagschik, T. Menzel, C. Körner, and M. Maurer, "Wissensbasierte szenariengenerierung für betriebsszenarien auf deutschen autobahnen," in *Workshop Fahrerassistenzsysteme und automatisiertes Fahren. Bd. vol. 12*, 2018.
- [6] "Annexes to the commission implementing regulation - c(2022)5402," https://ec.europa.eu/info/law/better-regulation/have-your-say/initiatives/12152-Automated-cars-technical-specifications_en, 2022.
- [7] *Road vehicles — Safety of the intended functionality*, Std. ISO/PAS 21448:2019, 2019.
- [8] L. Birkemeyer, M. Delventhal, I. Schaefer, and F. Schmieder, "Wann fahren wir autonom? eine untersuchung aus technischer und rechtlicher sicht," in *Software Engineering 2022 Workshops*. Gesellschaft für Informatik eV, 2022.
- [9] A. Gambi, T. Huynh, and G. Fraser, "Generating effective test cases for self-driving cars from police reports," in *Proceedings of the 27th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, 2019, pp. 257–267.
- [10] F. Montanari, C. Stadler, J. Sichermann, R. German, and A. Djanatliev, "Maneuver-based resimulation of driving scenarios based on real driving data," in *IEEE Intelligent Vehicles Symposium (IV)*, 2021, pp. 1124–1131.
- [11] F. Hauer, I. Gerostathopoulos, T. Schmidt, and A. Pretschner, "Clustering traffic scenarios using mental models as little as possible," in *IEEE Intelligent Vehicles Symposium (IV)*, 2020, pp. 1007–1012.
- [12] R. Krajewski, T. Moers, A. Meister, and L. Eckstein, "Béziervae: Improved trajectory modeling using variational autoencoders for the safety validation of highly automated vehicles," in *IEEE Intelligent Transportation Systems Conference (ITSC)*, 2019, pp. 3788–3795.
- [13] D. Nalic, A. Eichberger, G. Hanzl, M. Fellendorf, and B. Rogic, "Development of a co-simulation framework for systematic generation of scenarios for testing and validation of automated driving systems," in *IEEE Intelligent Transportation Systems Conference (ITSC)*, 2019, pp. 1895–1901.
- [14] E. de Gelder and J.-P. Paardekoooper, "Assessment of automated driving systems using real-life scenarios," in *IEEE Intelligent Vehicles Symposium (IV)*, 2017, pp. 589–594.
- [15] M. Althoff and S. Lutz, "Automatic generation of safety-critical test scenarios for collision avoidance of road vehicles," in *IEEE Intelligent Vehicles Symposium (IV)*, 2018, pp. 1326–1333.
- [16] L. Birkemeyer, T. Pett, A. Vogelsang, C. Seidl, and I. Schaefer, "Feature-interaction sampling for scenario-based testing of advanced driver assistance systems," in *Proceedings of the 16th International Working Conference on Variability Modelling of Software-Intensive Systems (VaMoS)*, 2022, pp. 1–10.
- [17] E. Rocklage, H. Kraft, A. Karatas, and J. Seewig, "Automated scenario generation for regression testing of autonomous vehicles," in *IEEE International Conference on Intelligent Transportation Systems (ITSC)*, 2017, pp. 476–483.
- [18] I. Paranjape, A. Jawad, Y. Xu, A. Song, and J. Whitehead, "A modular architecture for procedural generation of towns, intersections and scenarios for testing autonomous vehicles," in *IEEE Intelligent Vehicles Symposium (IV)*, 2020, pp. 162–168.
- [19] D. Yang, K. Redmill, and Ü. Özgüner, "A multi-state social force based framework for vehicle-pedestrian interaction in uncontrolled pedestrian crossing scenarios," in *IEEE Intelligent Vehicles Symposium (IV)*, 2020, pp. 1807–1812.
- [20] X. Wang, Y. Dong, S. Xu, H. Peng, F. Wang, and Z. Liu, "Behavioral competence tests for highly automated vehicles," in *IEEE Intelligent Vehicles Symposium (IV)*, 2020, pp. 1323–1329.
- [21] C. Medrano-Berumen and M. I. Akbaş, "Abstract simulation scenario generation for autonomous vehicle verification," in *SoutheastCon. IEEE*, 2019, pp. 1–6.
- [22] T. Menzel, G. Bagschik, L. Isensee, A. Schomburg, and M. Maurer, "From functional to logical scenarios: Detailing a keyword-based scenario description for execution in a simulation environment," in *IEEE Intelligent Vehicles Symposium (IV)*, 2019, pp. 2383–2390.
- [23] J. E. Stellet, M. R. Zofka, J. Schumacher, T. Schamm, F. Niewels, and J. M. Zöllner, "Testing of advanced driver assistance towards automated driving: A survey and taxonomy on existing approaches and open questions," in *IEEE International Conference on Intelligent Transportation Systems (ITSC)*, 2015, pp. 1455–1462.
- [24] D. Nalic, T. Mihalj, M. Bäuml, M. Lehmann, A. Eichberger, and S. Bernsteiner, "Scenario based testing of automated driving systems: A literature survey," in *FISITA web Congress*, vol. 10, 2020.
- [25] B. Gangopadhyay, S. Khastgir, S. Dey, P. Dasgupta, G. Montana, and P. Jennings, "Identification of test cases for automated driving systems using bayesian optimization," in *IEEE Intelligent Transportation Systems Conference (ITSC)*, 2019, pp. 1961–1967.
- [26] W. Ding, C. Xu, M. Arief, H. Lin, B. Li, and D. Zhao, "A survey on safety-critical driving scenario generation—a methodological perspective," *IEEE Transactions on Intelligent Transportation Systems*, 2023.
- [27] Y. Ma, C. Sun, J. Chen, D. Cao, and L. Xiong, "Verification and validation methods for decision-making and planning of automated vehicles: A review," *IEEE Transactions on Intelligent Vehicles*, 2022.
- [28] X. Zhang, J. Tao, K. Tan, M. Torngren, J. M. G. Sanchez, M. R. Ramli, X. Tao, M. Gyllenhammar, F. Wotawa, N. Mohan *et al.*, "Finding critical scenarios for automated driving systems: A systematic mapping study," *IEEE Transactions on Software Engineering*, 2022.
- [29] Z. Zhong, Y. Tang, Y. Zhou, V. d. O. Neves, Y. Liu, and B. Ray, "A survey on scenario-based testing for automated driving systems in high-fidelity simulation," *arXiv preprint arXiv:2112.00964*, 2021.
- [30] A. M. Nascimento, L. F. Vismari, C. B. S. T. Molina, P. S. Cugnasca, J. B. Camargo, J. R. de Almeida, R. Inam, E. Fersman, M. V. Marquezini, and A. Y. Hata, "A systematic literature review about the impact of artificial intelligence on autonomous vehicle safety," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 12, pp. 4928–4946, 2019.
- [31] M. Alawadhi, J. Almazrouie, M. Kamil, and K. A. Khalil, "A systematic literature review of the factors influencing the adoption of autonomous driving," *International Journal of System Assurance Engineering and Management*, vol. 11, no. 6, pp. 1065–1082, 2020.
- [32] P. Jing, H. Hu, F. Zhan, Y. Chen, and Y. Shi, "Agent-based simulation of autonomous vehicles: A systematic literature review," *IEEE Access*, vol. 8, pp. 79 089–79 103, 2020.
- [33] Z. Tahir and R. Alexander, "Coverage based testing for V&V and safety assurance of self-driving autonomous vehicles: A systematic literature review," in *IEEE International Conference On Artificial Intelligence Testing (AITest)*, 2020, pp. 23–30.
- [34] D. Karunakaran, J. S. Berrio, S. Worrall, and E. Nebot, "Challenges of testing highly automated vehicles: A literature review," in *IEEE International Conference on Recent Advances in Systems Science and Engineering (RASSE)*, 2022, pp. 1–8.
- [35] B. Schütt, J. Ransiek, T. Braun, and E. Sax, "1001 ways of scenario generation for testing of self-driving cars: A survey," *arXiv preprint arXiv:2304.10850*, 2023.
- [36] B. Kitchenham, "Procedures for performing systematic reviews," *Keele, UK, Keele University*, vol. 33, no. 2004, pp. 1–26, 2004.
- [37] G. Bagschik, T. Menzel, C. Körner, and M. Maurer, "Wissensbasierte Szenariengenerierung für Betriebsszenarien auf deutschen Autobahnen," in *Workshop Fahrerassistenzsysteme und automatisiertes Fahren. Bd. vol. 12*, 2018, p. 12.
- [38] E. Donges, "Fahrerverhaltensmodelle," in *Handbuch Fahrerassistenzsysteme*. Springer, 2015, pp. 17–26.
- [39] L. Birkemeyer, J. Fuchs, A. Gambi, and I. Schaefer, "SOTIF-compliant scenario generation using semi-concrete scenarios and parameter sampling," in *IEEE International Conference on Intelligent Transportation Systems (ITSC)*, 2023.