VALIDATION AND PLAUSIBILIZATION OF X-IN-THE-LOOP TESTS FOR DRIVING AUTOMATION

Felix Reisgys
Johannes Plaum
Dr. Andreas Schwarzhaupt
Daimler Truck AG
Germany
Prof. Dr.-Ing. Eric Sax
Institut für Technik der Informationsverarbeitung, Karlsruhe Institute of Technology
Germany

Paper Number 23-0064

ABSTRACT

Virtual X-in-the-Loop (XiL) environments are gaining significant importance in the test of Advanced Driver Assistance Systems (ADAS) and Automated Driving Systems (ADS). In order to derive reliable test results, credibility of XiL environments must be evaluated using suitable methods for XiL validation. This typically involves a back-to-back comparison to reference proving ground (PG) tests. Due to uncertainties inherent to PG and XiL, this validation requires the analysis of multiple test executions. Since this may not always be feasible with limited data availability, we define plausibilization as a preliminary step towards validation, comparing two single test executions in PG and XiL. A plausibilization method is presented, combining the evaluation of pass/fail criteria (PFC) and scenario distance measures. Finally, the application of the method in an ADAS series development project by evaluating three example Software-in-the-Loop (SiL) scenarios confirms that this is a reasonable plausibilization approach. Furthermore, it is shown that the method can be adjusted in a flexible way to meet requirements from different automation levels, systems or scenarios.

INTRODUCTION

In recent years, the scope of driving automation has increased significantly to improve vehicle safety, driver comfort and efficiency. Given the growing capabilities of both ADAS and ADS, the required testing depth and width increases as well. Current test strategies will not be able to cover future testing demands for safety validation as they rely heavily on real-world testing [1]. It is therefore necessary to make use of alternative testing methods. A promising approach pursued in research and industry is scenario-based testing in combination with XiL environments [2]. While XiL is a key enabler for efficient scaling of test volume, the credibility of XiL environments has a major impact on the potential to replace or supplement real-world testing [3]. In this paper, we discuss the validation of XiL environments and the role of uncertainties in test environments in this process. Furthermore, we introduce “plausibilization” as an initial step in a validation process. Finally, a plausibilization method is presented and applied to an example dataset from an ADAS series development project.

STATE OF THE ART

Advanced Driver Assistance Systems

Advanced Driver Assistance Systems (ADAS) process environment sensor data to support the driver in longitudinal and/or lateral vehicle control. Even if an ADAS is active, the driver retains full responsibility for vehicle control and can always override the system. [4] In this paper, two specific ADAS are considered:

An **Advanced Emergency Braking System (AEBS)** tracks objects in front of the ego vehicle and triggers driver warnings and brake interventions to avoid or mitigate collisions. AEBS are mandatory for new vehicles in the European Union since 2015. [5] While the minimum legal requirement is a speed reduction of 20 km/h [6], state-of-the-art AEBS are able to avoid collisions up to 80 km/h ego speed on stationary objects[1]. According to the SAE

[1] https://www.adac.de/rund-ums-fahrzeug/ausstattung-technik-zubehoer/assistenzsysteme/lkw-notbremsassistenten/

automation levels, an AEBS system is classified as a SAE level 0 system, as it executes a braking action only temporarily in case of a hazard.

Similar to an AEBS, a **Blind Spot Information System (BSIS)** also tracks objects, however in a lateral zone next to the ego vehicle. The main objective of this system is to support the driver in turning scenarios as especially pedestrians and cyclists may be difficult to see for the driver due to the obstructed field-of-view from the truck cabin. Whenever there is an object close to the ego vehicle, a BSIS information is issued. In case of a potential collision during a turning maneuver, the BSIS issues a warning to the driver. [7] The so-called **Active Sideguard Assist (ASGA)** is a system by Daimler Truck, which additionally triggers a braking intervention in parallel to a BSIS warning [8]. BSIS and ASGA can also be classified as SAE level 0.

Metrics and Pass/fail Criteria

Metrics and Pass/fail criteria (PFC) are used to evaluate the behavior of an ADAS or ADS. While we consider metrics as continuous values, a pass/fail criterion yields a binary information [9]. **Performance metrics** cover aspects such as driver comfort or fuel efficiency [10, 11]. For safety-related analysis, a variety of **criticality metrics** exists [12] to evaluate "the combined risk of the involved actors when the traffic situation is continued". [13] An example criticality metric example is Time-to-Collision (TTC) [14].

X-in-the-Loop Testing

In X-in-the-Loop (XiL) testing, different representations of the System under Test (SUT) can be tested in closed loop simulation environments [15]. Depending on SUT representation, different XiL variants such as the following exist:

- Model-in-the-Loop (MiL): A software model is tested in a virtual environment.
- Software-in-the-Loop (SiL): Real software code is tested in a virtual environment.
- Hardware-in-the-Loop (HiL): Real software code is integrated into target hardware and tested in a virtual environment.

Combining XiL and scenario-based testing enables the continuity of test cases across different XiL environments. Testing different representations of the SUT aims to reduce the share of real-world testing in order to cover the growing test width and depth for higher automation levels. [2] Nevertheless, real-world testing environments such as proving ground and field operational testing continue to play an important role for the system release [16] and serve as benchmark for the validation and plausibilization of XiL environments.

Both XiL and PG testing are prone to **uncertainties** and **errors** [3, 17]. While errors represent an explicit deviation of system behavior from a reference behavior, uncertainties relate to the possible and/or expected errors of a system [18]. Consequently, an error can be regarded as a result of an uncertainty. A distinction can be made regarding the type of uncertainties. **Aleatory** uncertainties are inherent to a system and occur stochastically. They cannot be reduced, however it is possible to quantify them with statistical models [18, 19]. In contrast, **epistemic** uncertainties are theoretically reducible as they result from a lack of knowledge or modeling inaccuracy [18, 19]. In general, both types of uncertainties can occur in XiL and PG tests [3].

If XiL test results are used in an overall test statement, **XiL validation** is a crucial contribution to substantiate credibility of the XiL environment. In contrast to validation of an SUT, XiL validation evaluates the behavior of the test environment only. While there exist various approaches to validate simulation models of XiL subsystems such as sensor models [20, 21] or vehicle models [22, 23], the focus of this paper shall be the overall validation of a full XiL environment. This can be achieved by comparing XiL results to a real-world reference, such as PG tests or other real-world driving data. Relevant inputs are trajectory data [3, 24–26] or criticality metrics [3, 24, 26]. It is possible to directly compare time rows as well as features derived from time rows [3, 25]. Another option is to evaluate maneuver similarity [26]. Furthermore, by using statistical means such as standard deviations [27] or tolerance intervals [3], it is possible to cover uncertainties in both XiL and PG.

Scenario-based Testing

Scenario-based testing provides a structured approach to describe real-world traffic through scenarios that form the basis for the test and release process. A **scenario** is defined as the temporal development between several scenes, similar to a storyline [28]. Each **scene** describes a snapshot of static elements and dynamic actors that

form the ego vehicle's environment. Three abstraction levels of scenarios were initially introduced by [29] and used in in the PEGASUS project. They have been extended by a fourth level of abstract scenarios in the VVM project [13]. While the abstraction level decreases from functional to concrete scenarios, the number of potential scenarios to be tested increases:

- **Functional scenarios:** A human readable description of a scenario, concentrating on the behavior and relation of included actors. This may include a visualization.
- **Abstract scenarios:** A formalized, machine readable description, including declarative descriptions such as constraints.
- **Logical scenarios:** A parameterized scenario representation, including parameter ranges or distributions as basis for a parameter variation.
- **Concrete scenarios:** A scenario derived from a logical scenario by selecting a specific parameter set, e.g. start velocities and distances.

Bock et al. [30] propose a **6-layer model** for structured traffic and environment description of scenarios that is based on the previous work in the PEGASUS project by [31, 32]. Scholtes et.al. adapt the 6-layer model to allow the separation of spatial (L1-L3) and temporal (L4-L6) scenario elements [33] and to represent the urban operational design domain that is considered in the VVM project:

1. Road Network and Traffic Guidance Objects, e.g. roads, sidewalks, traffic lights.
2. Roadside Structures, e.g. buildings, guardrails, street lamps.
3. Temporary Modifications of L1 and L2, e.g. temporary signs, covered markings.
4. Dynamic Objects, e.g. vehicles, pedestrians, animals.
5. Environmental Conditions, e.g. lighting, wind, road surface condition.
6. Digital Information, e.g. V2X messages, traffic light states.

This paper focuses on layer 4 and SUT behavior in relation to other traffic participants.

In analogy to requirements-based testing, [31] defines a logical **test case** as a logical scenario including PFC and requirements for test execution. A concrete test case is derived from a logical test case by specifying the scenario parameters. PFC for test cases may use performance and criticality metrics including a target or threshold value.

CONSIDERATIONS FOR PLAUSIBILZATION OF XIL TESTS

Validation and Plausibilization

While validation is a term commonly used when it comes to actual product development and testing, the validation of XiL environments relates to the test environment itself. Since a XiL environment is composed of multiple simulation models [34], the definition of validation is based on the definition for simulation models by Schlesinger:

> "[**Validation** is the] substantiation that a [simulation] model [or XiL environment] within its domain of applicability possesses a satisfactory range of accuracy consistent with the intended application of the model [or environment]." [35]

A simulation model is an implementation of a conceptual model as illustrated in Figure 1. Prior to validation, the steps qualification and verification are to be applied. We use the same reference to define qualification:

> "[**Qualification** is the] determination of adequacy of the conceptual model to provide an acceptable level of agreement for the domain of intended application." [35]

For verification we use the same reference with adjustments in accordance to Sargent [36]:

> "[**Verification** is the] substantiation that a [simulation] model [is a correct implementation of a conceptional model and] represents [the] conceptual model within specified limits of accuracy." [35]

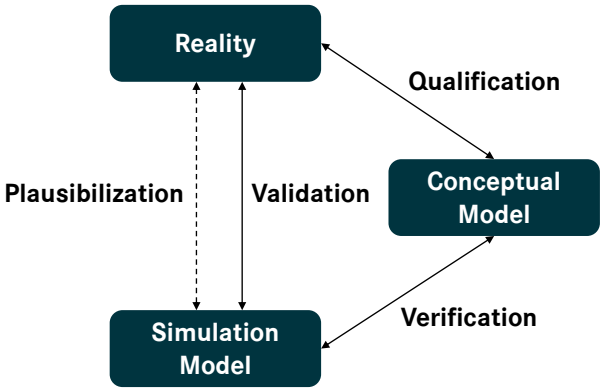

Figure 1. Relation between qualification, validation, verification of simulation models ([35], extended by "Plausibilization").

For the scope of this publication, we only discuss validation and therefore assume that qualification and verification have been performed successfully for all simulation models used in a XiL environment.

Since the intended application of a XiL environment is to reduce real-world tests by generating evidence on real-world behavior of the SUT, existing uncertainties [37] need to be taken into account when it comes to XiL validation. Consequently, XiL validation needs to cover statistical aspects. In the context of scenario-based testing, this requires multiple samples of reference scenarios of both XiL and real-world testing. We therefore extend the definition of XiL validation above for scenario-based testing by the following requirement:

> In the context of scenario-based testing, proving a satisfactory range of accuracy must consider uncertainty of test results and system behavior. This requires data from multiple back-to-back test executions of both real-world and XiL test.

While execution of multiple real-world or XiL tests may be feasible for some applications, there are applications where only a single real-world and XiL scenario execution can be tested and analyzed, e.g. due to economic or practical reasons. Since the term validation as previously specified would not be applicable, we introduce plausibilization as a preliminary step towards validation:

> **Plausibilization** in the context of scenario-based testing is the substantiation that a XiL environment possesses a satisfactory range of accuracy proven for one back-to-back test execution in real world and XiL. It does not consider uncertainty of test results or system behavior.

Validation implicitly requires a successful plausibilization. The rest of this paper will focus on XiL plausibilization in the context of scenario-based testing, without specifying subsequent steps for XiL validation.

Pass/fail Criteria and Test Result

As a prerequisite to the following sections, we further introduce PFC as a binary output of a test evaluation:

$$\mathrm{PFC} = \mathrm{f}_1(\mathrm{x}_1^{\mathrm{out}}, \dots, \mathrm{x}_\mathrm{n}^{\mathrm{out}}) \qquad \text{Equation (1)}$$

With x^{out} representing an output of the test execution, e.g. trajectory data or criticality metrics. PFC are defined based on expert knowledge depending on the SUT and scenario to be tested. In addition, we formally introduce a test result T as the aggregation of all PFC:

$$\mathrm{T} = [\mathrm{PFC}_1, \dots, \mathrm{PFC}_\mathrm{m}] \qquad \text{Equation (2)}$$

A METHOD FOR PLAUSIBILIZATION OF XIL TESTS

Overview

As already discussed in the previous section, plausibilization compares a XiL test execution data to a real-world test execution, represented by PG. Each test execution shall be called a "sample", yielding the terms "XiL sample"

and “PG sample”. Plausibilization is successful if the samples (S^{XiL}, S^{PG}) of both test environments are classified as equivalent. Equivalence ($E = 1$) requires the following two criteria:

1. The test results of both test environments are identical.

$$E_1 = \begin{cases} 1, & T^{XiL} = T^{PG} \\ 0, & \text{otherwise} \end{cases} \qquad \text{Equation (3)}$$

2. The scenario trajectories which lead to the test result are equivalent. This is evaluated using k scenario distance measures d_k as follows:

$$E_2 = \begin{cases} 1, & d_k(S^{XiL}, S^{PG}) < d_{k,max} \forall k \\ 0, & \text{otherwise} \end{cases} \qquad \text{Equation (4)}$$

Equivalence is achieved if both criteria are met:

$$E = \begin{cases} 1, & E_1 \wedge E_2 \\ 0, & \text{otherwise} \end{cases} \qquad \text{Equation (5)}$$

It is assumed that all data required to compute both PFC and scenario distance measures is available, including ego and object trajectories. If there are limitations in data availability in at least one test environment, the method may still be applicable if PFC or scenario distance measures are adapted.

Pass/fail Criteria

PFC are defined and selected based on expert knowledge, given the logical scenario and the SUT. If applicable, PFC are designed in a way that yields “1” for a desired behavior and “0” for undesired behavior. For this publication, we introduce the following PFC:

No collision: Indicates that no collision of the ego vehicle and another dynamic object such as vehicle or pedestrian has occurred in the scenario:

$$noColl = \begin{cases} 1, & \text{no collision occurred} \\ 0, & \text{collision occurred} \end{cases} \qquad \text{Equation (6)}$$

TTC threshold: Indicates that the minimal TTC has not undercut a specific threshold in the scenario. The threshold needs to be selected based on the logical scenario and the SUT.

$$ttcTh = \begin{cases} 1, & TTC(t) \geq TTC_{min} \ \forall t \\ 0, & \text{otherwise} \end{cases} \qquad \text{Equation (7)}$$

AEBS warning: Indicates if the AEBS of the ego vehicle has triggered a warning to the driver in the scenario.

$$aebsW = \begin{cases} 1, & \text{warning triggered} \\ 0, & \text{otherwise} \end{cases} \qquad \text{Equation (8)}$$

AEBS partial braking: Indicates if the AEBS of the ego vehicle has triggered a partial braking in the scenario.

$$aebsPB = \begin{cases} 1, & \text{partial braking triggered} \\ 0, & \text{otherwise} \end{cases} \qquad \text{Equation (9)}$$

AEBS full braking: Indicates if the AEBS of the ego vehicle has triggered a full braking in the scenario.

$$aebsFB = \begin{cases} 1, & \text{full braking triggered} \\ 0, & \text{otherwise} \end{cases} \qquad \text{Equation (10)}$$

Blind spot information: Indicates if the BSIS of the ego vehicle has triggered a blind spot information to the driver in the scenario.

$$bsisI = \begin{cases} 1, & \text{information triggered} \\ 0, & \text{otherwise} \end{cases} \qquad \text{Equation (11)}$$

Blind spot warning: Indicates if the BSIS of the ego vehicle has triggered a blind spot warning to the driver in the scenario.

$$\text{bsisW} = \begin{cases} 1, & \text{warning triggered} \\ 0, & \text{otherwise} \end{cases} \qquad \text{Equation (12)}$$

ASGA braking: Indicates if the ASGA of the ego vehicle has triggered a blind spot braking in the scenario.

$$\text{asgaB} = \begin{cases} 1, & \text{braking triggered} \\ 0, & \text{otherwise} \end{cases} \qquad \text{Equation (13)}$$

Scenario Distance Measures

The general idea of scenario distance measures is to quantify the dissimilarity of two concrete scenarios based on the data generated from their execution [38]. Though being deployed in multiple applications, scenario distance measures have rarely been named explicitly in publications. We focus on those scenario distance measures that can be computed from recorded trajectory data [39, 40] . This requires a definition of three coordinate systems as follows (see Figure 2):

Inertial coordinate system: A coordinate system bound to a static reference point and without any rotation. For the sake of simplicity, the coordinate system is initialized with its x-axis coinciding with the ego vehicles' x-axis at the beginning of a scenario. Variables using this coordinate system are indicated by index "i".

Ego coordinate system: A coordinate system bound to a static reference point and rotating with the ego vehicle. Consequently, both x-axes of coordinate system and ego vehicle coincide at any time. Index "e" indicates this coordinate system.

Object coordinate system: A coordinate system bound to the front of the ego vehicle and rotating with the ego vehicle. Hence, the coordinate system moves with the ego vehicle and has coinciding x-axes with the ego vehicle at any time. This coordinate system is indicated by index "o".

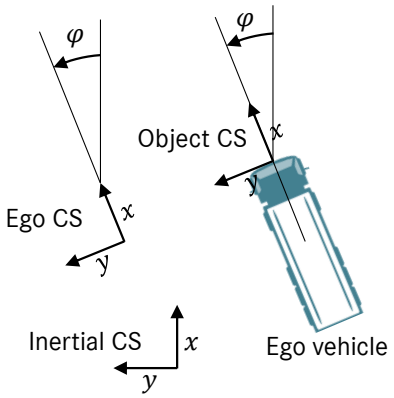

Figure 2. Overview of coordinate systems used.

The following trajectory data are processes for the scenario distance measures:

- Ego trajectory in inertial coordinate system: $\mathrm{r}^{\mathrm{i}}_{\mathrm{ego}} = [\mathrm{x}^{\mathrm{i}}, \mathrm{y}^{\mathrm{i}}]$
- Ego longitudinal velocity in ego coordinate system: $v^{e}_{x,ego}$
- Ego yaw angle in inertial coordinate system: φ^{i}_{ego}
- Object relative trajectory in object coordinate system: $r^{o}_{q} = [x^{o}_{q}, y^{o}_{q}]$

The initial step to compute the scenario distance measures is the application of Dynamic Time Warping (DTW) [41] on the ego trajectory data and the extraction of the respective DTW path. This eliminates time dependency. Given two time series $A = [a_1, \dots, a_n]$ and $B = [b_1, \dots, b_m]$, DTW assigns for each element a corresponding element of the other time series: $[a_u, b_v]$. The vectors $u = [u_1, \dots, u_p]$ and $v = [v_1, \dots, v_p]$ represent the DTW path.

The DTW path is adjusted in the following way: In order to reduce the number of elements which are assigned multiple times, only the elements of the time series containing fewer elements are assigned to the elements of the time series with more elements. Hence, there is no change in the time series containing more elements. Given $n \leq m$, this yields an alternative assignment $[a_{\hat{u}}, b_{\hat{v}}]$ with $\hat{u} = [\hat{u}_1, \dots, \hat{u}_m]$ and $\hat{v} = [1, \dots, m]$. In case multiple elements $u_{p,1}, \dots, u_{p,k}$ of A are assigned to an element of B, the last element $\hat{u}_p = u_{p,k}$ is selected. Furthermore, if $m < n$, $\hat{u}$ and $\hat{v}$ are determined vice versa.

Using the DTW path, we define three scenario distance measures:

Scenario Distance Measure 1 uses the ego vehicle trajectory and the relative trajectory of an object:

$$\mathrm{d}_1(\mathrm{S}^{\mathrm{XiL}}, \mathrm{S}^{\mathrm{PG}}) = \max_{\mathrm{j}} 0.5 \cdot \left(\mathrm{g}\left(\mathrm{r}^{\mathrm{i,XiL}}_{\mathrm{ego},\hat{\mathrm{u}}_\mathrm{j}}, \mathrm{r}^{\mathrm{i,PG}}_{\mathrm{ego},\hat{\mathrm{v}}_\mathrm{j}}\right) + \mathrm{g}\left(\mathrm{r}^{\mathrm{o,XiL}}_{\mathrm{q},\hat{\mathrm{u}}_\mathrm{j}}, \mathrm{r}^{\mathrm{o,PG}}_{\mathrm{q},\hat{\mathrm{v}}_\mathrm{j}}\right)\right) \qquad \text{Equation (14)}$$

$$\mathrm{g}(\mathrm{x}, \mathrm{y}) = \min(|\mathrm{x} - \mathrm{y}|, \mathrm{g}_{\mathrm{th}}) \qquad \text{Equation (15)}$$

The parameter g_{th} defines a threshold for the maximum Euclidean distance of x and y.

Scenario Distance Measure 2 analyzes the ego vehicle longitudinal velocity:

$$\mathrm{d}_2(\mathrm{S}^{\mathrm{XiL}}, \mathrm{S}^{\mathrm{PG}}) = \frac{1}{\max(\mathrm{n}, \mathrm{m})} \sum_{\mathrm{j}=1}^{\max(\mathrm{n},\mathrm{m})} \mathrm{g}(\mathrm{v}^{\mathrm{e,XiL}}_{\mathrm{x,ego},\hat{\mathrm{u}}_\mathrm{j}}, \mathrm{v}^{\mathrm{e,PG}}_{\mathrm{x,ego},\hat{\mathrm{v}}_\mathrm{j}}) \qquad \text{Equation (16)}$$

Scenario Distance Measure 3 considers the ego vehicle yaw angle:

$$\mathrm{d}_3(\mathrm{S}^{\mathrm{XiL}}, \mathrm{S}^{\mathrm{PG}}) = \frac{1}{\max(\mathrm{n}, \mathrm{m})} \sum_{\mathrm{j}=1}^{\max(\mathrm{n},\mathrm{m})} \mathrm{g}(\varphi^{\mathrm{i,XiL}}_{\mathrm{ego},\hat{\mathrm{u}}_\mathrm{j}}, \varphi^{\mathrm{i,PG}}_{\mathrm{ego},\hat{\mathrm{v}}_\mathrm{j}}) \qquad \text{Equation (17)}$$

If there are two or more dynamic objects in the scenario, they have to be assigned to each other. Since in this publication only scenarios with one other object are considered, there is no further assignment algorithm necessary.

Scenario Distance Measure Threshold Determination

In order to apply Equation (4), it is necessary to define suitable scenario distance thresholds which mark equivalence of two samples. As there is no standardized process for this, we propose and apply a new method for this.

As previously mentioned, uncertainties are inherent to both XiL and PG tests. Consequently, test results may differ if an identical concrete scenario is executed repeatedly. Each sample is assigned to a respective test result, yielding groups of samples. For each group, scenario distance measures to samples within the group are computed. It is assumed that there is at least one group of samples where 95 % (Coverage C) of the scenario distance measure population falls below the threshold that indicates equivalence. Since the number of samples in each group is a finite number, a one-sided tolerance interval [42] is used to compute the respective thresholds for each group:

$$\mathrm{P}\left(\mathrm{F}\left(\mathrm{d}_{\mathrm{k},\mathrm{T_i},\max}\right) \geq \mathrm{C}\right) \geq 1 - \alpha \qquad \text{Equation (18)}$$

The parameter α represents the confidence and is set to 95 %. This method is applied to each test result/sample group occurring, given that there exist at least three samples in the respective group. As a next step, the minimum threshold value of all groups is selected:

$$\mathrm{d}_{\mathrm{k,max}} \min_{\mathrm{T_i}} \mathrm{d}_{\mathrm{k},\mathrm{T_i},\max} \qquad \text{Equation (19)}$$

APPLICATION OF XIL PLAUSIBILIZATION

Reference SUT and Scenarios

The plausibilization method presented is applied to the software test of a series AUTOSAR-based [43] Electronic Control Unit (ECU) for execution of ADAS algorithms, including those for AEBS, BSIS and ASGA. Environment sensors provide the required information to the SUT, while respective actuator controllers are deployed for vehicle control.

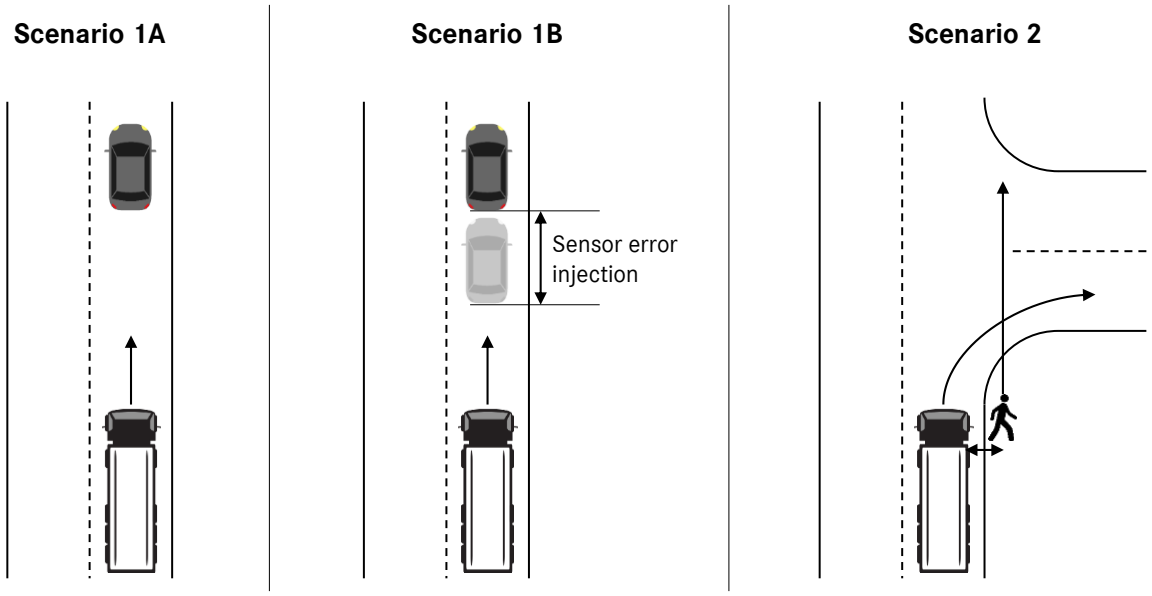

Figure 3. Overview of reference scenarios.

For evaluation, three concrete scenarios are instantiated from the logical scenario illustrated in Figure 3 and described below:

Scenario 1A: The ego vehicle drives with highway speed towards a stationary passenger car.

Scenario 1B: Identical to Scenario 1A, except to an additional sensor error injection for the relative distance of the preceding passenger car. This scenario is only performed in XiL tests.

Scenario 2: The ego vehicle drives with city speed and turns right. A pedestrian walks next to the right side of the ego vehicle, in the same direction as the ego vehicle drives prior to the turn maneuver. Due to the turning maneuver of the ego vehicle, the trajectories of ego vehicle and pedestrian intersect.

While scenarios 1A and 1B are intended to trigger an AEBS reaction, scenario 2 is supposed to lead to a BSIS and ASGA reaction. Based on the SUT, the following PFC are selected for each scenario:

Scenario 1A/1B: [noColl, ttcTh, aebsW, aebsPB, aebsFB]

Scenario 2: [noColl, bsisI, bsisW, asgaB]

XiL Environment and Proving Ground

A SiL environment is used as XiL representation, including simulation models for vehicle, other ECUs, driver, environment (objects etc.) and sensors. The SUT representation is fully virtual as well and contains the real SUT series application code, while base software is modeled in a simplified way. Scenarios are defined using a combination of Open Drive [44] and proprietary description files. Each concrete scenario is tested once, as there does not exist considerable aleatory uncertainty in this SiL environment.

Proving ground tests are performed for scenario 1A and 2. There are five samples of scenario 1A (indicated by I to V) and three samples of scenario 2 (indicated by VI to VIII), which allows the computation of scenario distance measure thresholds. Scenario 1A tested on PG also serves as a reference for scenario 1B SiL tests.

In both test environments, SUT network communication is recorded consistently. Furthermore, precise trajectory data using Differential-GPS is recorded on PG, while equivalent ground truth information can be extracted from SiL tests. In order to compute scenario distance measures, all tracks are previously cut using predefined start and end conditions.

Plausibilization Results

Scenario distance measure thresholds are determined based on PG samples. Since in this example all test results are identical in the respective PG tests for scenario 1A, there is only one group of samples. Though this does not apply for scenario 2, we still consider all PG samples as part of an equivalent group as there has been a manual driver brake intervention in case of no ASGA brake intervention (applies to sample VI). The analysis as described in Equation (18) and Equation (19) yields individual thresholds $d_{1,max}$, $d_{2,max}$ and $d_{3,max}$ for scenarios 1A/1B and scenario 2:

Table 1.

Scenario distance thresholds

Scenario	$\mathbf{d_{1,max}}$	$\mathbf{d_{2,max}}$	$\mathbf{d_{3,max}}$
1A/1B	2.434	0.752	0.006
2	6.657	0.443	0.028

Plausibilization is applied to all sample combinations of corresponding scenarios, which yields the following results:

Table 2.

Equivalence of Scenarios 1A and 1B (1/0 indicate whether equivalence of PFC or scenario distance measure is achieved)

Sample combination	**noColl**	**ttcTh**	**aebsW**	**aebsPB**	**aebsFB**	d_1	d_2	d_3	E
1A-I	1	1	1	1	1	1	0	0	0
1A-II	1	1	1	1	1	1	1	1	1
1A-III	1	1	1	1	1	1	1	1	1
1A-IV	1	1	1	1	1	1	1	1	1
1A-V	1	1	1	1	1	1	0	1	0
1B-I	1	1	1	1	1	0	0	0	0
1B-II	1	1	1	1	1	0	0	1	0
1B-III	1	1	1	1	1	0	0	1	0
1B-IV	1	1	1	1	1	0	0	1	0
1B-V	1	1	1	1	1	0	0	1	0

Table 3.

Equivalence of Scenario 2 (1/0 indicate whether equivalence of PFC or scenario distance measure is achieved)

Sample combination	$noColl$	$bsisI$	$bsisW$	$asgaB$	d_1	d_2	d_3	E
2-VI	1	1	1	0	1	1	1	0
2-VII	1	1	1	1	1	1	1	1
2-VIII	1	1	1	1	1	1	1	1

In Table 2 and Table 3, 1/0 indicates whether equivalence of PFC or scenario distance measure is achieved. For scenario 1A, three of five sample combinations are classified as plausible, while the same applies to two of three scenario 2 samples. For scenario 1A, the samples with unsuccessful plausibilization were not equivalent in terms of scenario distance measures. In contrast, scenario 2, one sample is not equivalent due to a mismatch of a PFC

($asgaB$). For scenario 1B, plausibilization is not successful for any sample due to failed equivalence of scenario distance measure 1. A plot to illustrate scenario distance measures for both a plausible (1A-II) and not plausible (1B-II) sample combination (both marked in gray in Table 2) is shown in Figure 4 and Figure 5.

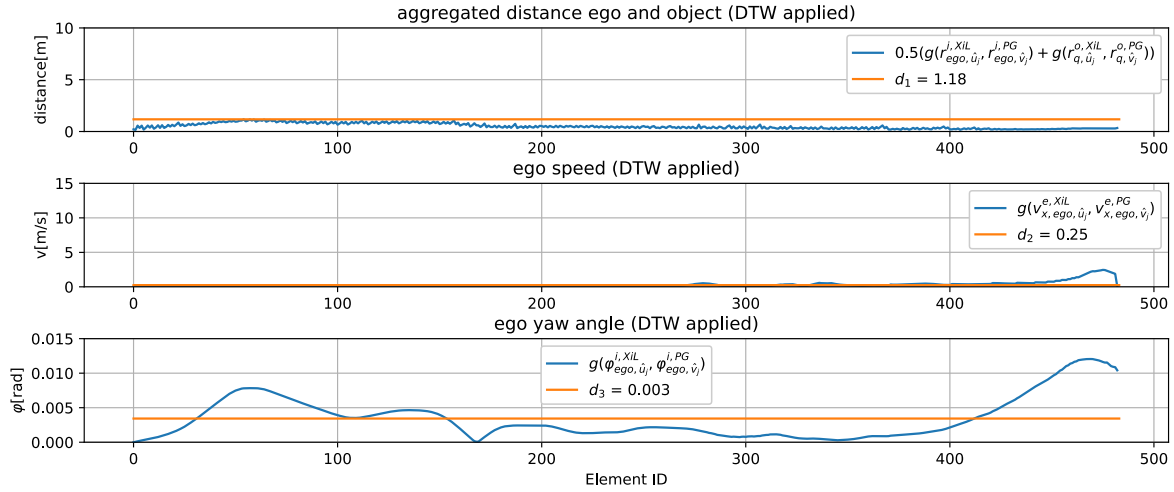

Figure 4. Illustration of scenario distance measures for sample combination 1A-II (plausible).

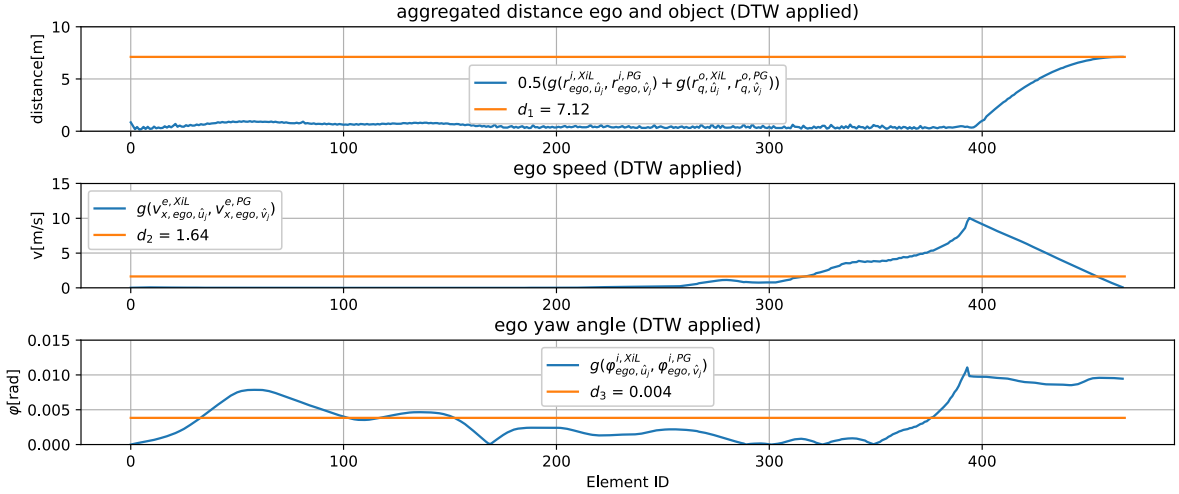

Figure 5. Illustration of scenario distance measures for sample combination 1B-II (not plausible).

The major difference between scenario 1A and 1B is a deviation of d_1, which is caused by a significant increase of the aggregated distance ($0.5 \cdot (g\left(r^{i,XiL}_{ego,\hat{u}_j}, r^{i,PG}_{ego,\hat{v}_j}\right) + g\left(r^{o,XiL}_{q,\hat{u}_j}, r^{o,PG}_{q,\hat{v}_j}\right))$ in Equation (17) in scenario 1B. There is also a higher deviation of the mean ego velocity in scenario 1B. This behavior can be explained by the error injection, which causes the ego vehicle to brake earlier in scenario 1B compared to scenario 1A in both SiL and PG.

Discussion

The presented XiL plausibilization method combines an expert knowledge-based approach (PFC) with a data-driven approach (scenario distance measures). While PFC equivalence ensures that a XiL test yields the same test results as a PG test, scenario distance measures are used to evaluate the trajectories leading to these test results. Both PFC and scenario distance measures can be tailored to application demands, including both testing of ADAS and ADS. Consequently, this dependency may lead to different plausibilization results for the same XiL and PG data. However, this is a desired behavior as requirements for XiL tests may vary by application.

When it comes to the application example presented, the overall plausibilization statement can be considered as reasonable from a visual expert inspection of all scenarios. Both scenario 1A and 2 show a very similar SUT stimulation and behavior in XiL and PG, whereas the plausibilization results of scenario 1B are caused by a different SUT behavior which has been identified in trajectory analysis through scenario distance measures. Even though there is no standardized way to evaluate a plausibilization statement, the presented method is suitable for the use case discussed and can be adapted for further use cases.

While mere plausibilization is not sufficient to statistically compare XiL to PG, it is still a necessary step towards a XiL validation. The fact that plausibilization of scenario 1A does not yield a positive result for all PG samples shows the variance of PG tests and underscores this statement. To argument credibility of a XiL environment it is not expected that all sample combinations are classified as plausible. Statistical evaluation will therefore enhance the plausibilization statement.

CONCLUSIONS AND OUTLOOK

Credibility of XiL tests is a major concern when it comes to their growing deployment to test ADAS and ADS. While direct comparison of a XiL test to a reference PG test is an established procedure, taking into account uncertainty of both XiL and real-world testing is a significant enhancement of existing XiL validation methods. In this paper, we discuss XiL validation and introduce the term plausibilization as a preliminary step towards validation. We further provide a potential method to plausibilize a XiL sample by comparing its PFC and scenario distance measures to a PG sample as a reference. To evaluate usability of this method, it is applied to an ADAS series ECU SiL test. For two concrete scenarios, a successful plausibilization can be executed for a majority of sample combinations. In an additional scenario, which contains an intended error injection, all sample combinations are evaluated as not plausible.

The overall plausibilization method is promising considering the application example discussed. Nevertheless, there is a high dependency on PFC and scenario distance measure selection and definition of scenario distance measure thresholds. As for right now, these steps require a significant amount of expert knowledge and may be biased. For XiL validation, it is furthermore necessary to add steps considering statistical behavior and uncertainty of test and plausibilization results.

REFERENCES

[1] W. Wachenfeld, “How Stochastic can Help to Introduce Automated Driving,” PhD Thesis, TU Darmstadt, 2017.

[2] F. Reisgys, J. Plaum, A. Schwarzhaupt, and E. Sax, “Scenario-based X-in-the-Loop Test for Development of Driving Automation,” in *14. Workshop Fahrerassistenzsysteme und Automatisiertes Fahren*, Bonlanden, 2022.

[3] F. Reisgys, M. Elgharbawy, A. Schwarzhaupt, and E. Sax, “Argumentation on ADAS Simulation Validity using Aleatory and Epistemic Uncertainty Estimation,” in *Proceedings of the Driving Simulation Conference 2021 Europe VR*, Munich, Germany, 2021, pp. 25–32.

[4] T. M. Gasser, A. Seeck, and B. W. Smith, “Framework Conditions for the Development of Driver Assistance Systems,” in *Handbook of Driver Assistance Systems*, H. Winner, S. Hakuli, F. Lotz, and C. Singer, Eds., Cham: Springer International Publishing, 2016, pp. 35–68.

[5] *Regulation (EC) No 661/2009 of the European Parliamant and of the Council*, 2009.

[6] *Commission Regulation (EU) No 347/2012*, 2012.

[7] *Uniform provisions concerning the approval of motor vehicles with regard to the Blind Spot Information System for the Detection of Bicycles: UN ECE R 151*, 2020.

[8] Daimler Truck AG, *Mercedes-Benz Trucks presents two worldwide innovations in their trucks for more safety on the road*. Stuttgart, 2020. Accessed: Nov. 30 2022. [Online]. Available: https://media.daimlertruck.com/marsMediaSite/ko/en/47504429

[9] C. King, L. Ries, C. Kober, C. Wohlfahrt, and E. Sax, “Automated Function Assessment in Driving Scenarios,” in *IEEE 12th International Conference on Software Testing, Verification and Validation*, Xi'an, China, 2019, pp. 414–419.

[10] C. Esselborn, M. Eckert, M. Holzäpfel, E. Wahl, and E. Sax, “Method for a scenario-based and weighted assessment of map-based advanced driving functions,” in *ATZ live, 20. Internationales Stuttgarter*

Symposium, M. Bargende, H.-C. Reuss, and A. Wagner, Eds., Wiesbaden: Springer Vieweg, 2020, pp. 193–207.
[11] T. Plum *et al.*, "A simulation-based case study for powertrain efficiency improvement by automated driving functions," *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, vol. 233, no. 5, pp. 1320–1330, 2019.
[12] L. Westhofen *et al.*, "Criticality Metrics for Automated Driving: A Review and Suitability Analysis of the State of the Art," *Arch Computat Methods Eng*, 2022.
[13] C. Neurohr, L. Westhofen, M. Butz, M. H. Bollmann, U. Eberle, and R. Galbas, "Criticality Analysis for the Verification and Validation of Automated Vehicles," *IEEE Access*, vol. 9, pp. 18016–18041, 2021.
[14] J. C. Hayward, "Near miss determination through use of a scale of danger," 1972.
[15] K. v. Neumann-Cosel, "Virtual Test Drive," PhD Thesis, TU München, 2014.
[16] C. King, L. Ries, J. Langner, and E. Sax, "A Taxonomy and Survey on Validation Approaches for Automated Driving Systems," in *IEEE International Symposium on Systems Engineering (ISSE)*, Vienna, 2020, pp. 1–8.
[17] O. Balci, "Principles and techniques of simulation validation, verification, and testing," in *Winter Simulation Conference Proceedings*, Arlington, VA, USA, 1995, pp. 147–154.
[18] S. Riedmaier, B. Danquah, B. Schick, and F. Diermeyer, "Unified Framework and Survey for Model Verification, Validation and Uncertainty Quantification," *Arch Computat Methods Eng*, 2020.
[19] W. L. Oberkampf, S. M. DeLand, B. M. Rutherford, K. V. Diegert, and K. F. Alvin, "Error and uncertainty in modeling and simulation," *Reliability Engineering & System Safety*, vol. 75, no. 3, pp. 333–357, 2002.
[20] S. Bernsteiner, Z. Magosi, D. Lindvai-Soos, and A. Eichberger, "Radarsensormodell für den virtuellen Entwicklungsprozess," *ATZ Elektron*, vol. 10, no. 2, pp. 72–79, 2015.
[21] A. Schaermann, A. Rauch, N. Hirsenkorn, T. Hanke, R. Rasshofer, and E. Biebl, "Validation of vehicle environment sensor models," in *IEEE Intelligent Vehicles Symposium (IV)*, Los Angeles, CA, USA, 2017, pp. 405–411.
[22] M. Viehof and H. Winner, "Stand der Technik und der Wissenschaft: Modellvalidierung im Anwendungsbereich der Fahrdynamiksimulation," TU Darmstadt, Darmstadt, 2017.
[23] *ISO 19365:2016: Passenger cars - Validation of vehicle dynamic simulation - Sine with dwell stability control testing*, ISO, 2016.
[24] K. Groh, S. Wagner, T. Kuehbeck, and A. Knoll, "Simulation and Its Contribution to Evaluate Highly Automated Driving Functions," *SAE International Journal of Advances and Current Practices in Mobility*, vol. 1, no. 2, pp. 539–549, 2019.
[25] D. Notz *et al.*, "Methods for Improving the Accuracy of the Virtual Assessment of Autonomous Driving," in *8th IEEE International Conference on Connected Vehicles and Expo (ICCVE)*, Graz, Austria, 2019.
[26] C. Stadler, F. Montanari, W. Baron, C. Sippl, and A. Djanatliev, "A Credibility Assessment Approach for Scenario-Based Virtual Testing of Automated Driving Functions," *IEEE Open Journal of Intelligent Transportation Systems*, vol. 3, pp. 45–60, 2022.
[27] S. Riedmaier, J. Nesensohn, C. Gutenkunst, T. Düser, B. Schick, and H. Abdellatif, "Validation of X-in-the-Loop Approaches for Virtual Homologation of Automated Driving Functions," in *11th Graz Symposium Virtual Vehicle*, 2018.
[28] S. Ulbrich, T. Menzel, A. Reschka, F. Schuldt, and M. Maurer, "Defining and Substantiating the Terms Scene, Situation, and Scenario for Automated Driving," in *18th IEEE International Conference on Intelligent Transportation Systems (ITSC)*, Gran Canaria, Spain, 2015, pp. 982–988.
[29] T. Menzel, G. Bagschik, and M. Maurer, "Scenarios for Development, Test and Validation of Automated Vehicles," in *IEEE Intelligent Vehicles Symposium (IV)*, Changshu, 2018, pp. 1821–1827.
[30] J. Bock, R. Krajewski, L. Eckstein, J. Klimke, J. Sauerbier, and A. Zlocki, "Data Basis for Scenario-Based Validation of HAD on Highways," in *27th Aachen Colloquium Automobile and Engine Technology*, Aachen, 2018.
[31] F. Schuldt, "Ein Beitrag für den methodischen Test von automatisierten Fahrfunktionen mit Hilfe von virtuellen Umgebungen," PhD Thesis, TU Braunschweig, 2017.
[32] G. Bagschik, T. Menzel, and M. Maurer, "Ontology based Scene Creation for the Development of Automated Vehicles," in *IEEE Intelligent Vehicles Symposium (IV)*, Changshu, 2018, pp. 1813–1820.
[33] M. Scholtes *et al.*, "6-Layer Model for a Structured Description and Categorization of Urban Traffic and Environment," *IEEE Access*, vol. 9, pp. 59131–59147, 2021.

[34] B. Schütt, M. Steimle, B. Kramer, D. Behnecke, and E. Sax, “A Taxonomy for Quality in Simulation-Based Development and Testing of Automated Driving Systems,” *IEEE Access*, vol. 10, pp. 18631–18644, 2022.
[35] S. Schlesinger *et al.*, “Terminology for model credibility,” *SIMULATION*, vol. 32, no. 3, pp. 103–104, 1979.
[36] R. G. Sargent, “Verification and validation of simulation models,” in *Proceedings of the 2010 Winter Simulation Conference (WSC)*, Baltimore, MD, USA, 2010, pp. 166–183.
[37] J. Langner, K.-L. Bauer, M. Holzapfel, and E. Sax, “A Process Reference Model for the Virtual Application of Predictive Control Features,” in *IEEE Intelligent Vehicles Symposium (IV)*, Las Vegas, NV, USA, 2020, pp. 1759–1764.
[38] K. Backhaus, B. Erichson, S. Gensler, R. Weiber, and T. Weiber, *Multivariate Analysemethoden*. Wiesbaden: Springer Fachmedien Wiesbaden, 2021.
[39] J. Bian, D. Tian, Y. Tang, and D. Tao, “A survey on trajectory clustering analysis,” Feb. 2018. [Online]. Available: http://arxiv.org/pdf/1802.06971v1
[40] L. Ries, P. Rigoll, T. Braun, T. Schulik, J. Daube, and E. Sax, “Trajectory-Based Clustering of Real-World Urban Driving Sequences with Multiple Traffic Objects,” in *IEEE International Intelligent Transportation Systems Conference (ITSC)*, Indianapolis, IN, USA, 2021, pp. 1251–1258.
[41] T. Giorgino, “Computing and Visualizing Dynamic Time Warping Alignments in R : The dtw Package,” *Journal of Statistical Software*, vol. 31, no. 7, 2009.
[42] D. S. Young, “tolerance : An R Package for Estimating Tolerance Intervals,” *J. Stat. Soft.*, vol. 36, no. 5, 2010.
[43] AUTOSAR, “Classic Platform Release Overview,” 2021.
[44] ASAM, “OpenDRIVE Base Standard 1.7,” 2021.