2019 IEEE Intelligent Vehicles Symposium (IV)
Paris, France. June 9-12, 2019

Towards Cross-Verification and Use of Simulation in the Assessment of Automated Driving

Sebastian Wagner*, Korbinian Groh*, Thomas Kühbeck, Alois Knoll

Abstract—**One remaining challenge for Automated Driving (AD) that remains unclear to this day is its assessment for market release. The application of previous strategies derived from the V-model is infeasible due to the vast amount of required real-road testing to prove safety with an acceptable significance. A full set of requirements covering all possible traffic scenarios for testing and AD system can still not be derived to this day. Several approaches address this issue by either improving the set of test cases or by including other virtual test domains in the assessment process. However, all rely on simulations that can not be validated as a whole and therefore not be used for proving safety.**

This work addresses this issue and exhibits a method to verify the use of simulation in a scenario-based assessment process. By introducing a pipeline for reprocessing real-world scenarios as test cases we demonstrate where errors emerge and how these can be isolated. We unveil an issue in simulation which may cause behavior changes of the AD function in resimulation and thus makes the straight forward use of simulation in the assessment process impossible. A solution promising to minimize reprocessing errors and to avoid this behavior change is presented. Finally, this enables the local variation of real-world driving tests in a solely simulative context yielding verified and usable results.

Index Terms—**Autonomous vehicles, Vehicle safety, Risk analysis, Performance analysis**

I. Introduction

While recent advances in Automated Driving (AD) move towards SAE level 3 and higher [1]–[3], it remains unclear how to assess them for the release on the market [4]. Despite the regulations for the release being well defined in ISO 26262 [5], the estimated volume of required real-road testing is $5 \cdot 10^9 km$ [6], [7], which is infeasible to be covered under economical and temporal aspects. Additionally, it does not guarantee the coverage of every possible situation. Hence, current research in assessment methods tends towards scenario-based testing [8]–[10].

Preliminary to the assessment of AD, Advanced Driver-Assistance Systems (ADAS) were tested by means of the V-model [11]. Herein, real-world testing is substituted by Simulation-in-the-Loop (SIL), Vehicle-in-the-Loop (VIL) or Hardware-in-the-Loop (HIL) tests in early development stages. Nevertheless, real-world testing remains the last instance before release. This is possible as an ADAS is generally subject to a limited scope of driving scenarios and the driver is still available as a fallback layer. However, the scope of AD is unlimited with respect to driving scenarios and, therefore, results in a large amount of required real-road testing mentioned before. Also, the driver has to take less responsibility through the increasing SAE levels, drops out as fallback to the driving task at level 3 and finally is obsolete in level 5.

To cope with this problem, the following three approaches emerge current research. The first tries to minimize the number of test cases by avoiding redundancies [8]. Starting the test case generation from observed critical scenarios is the second attempt [9], [12]. Both attempts rely on the fact that a generated test case can be conducted on real roads, which is not entirely true. A specific scenario can not exactly be reconstructed in reality, but in virtual testing domains like SIL, VIL or HIL. As a consequence, the last approach suggests enabling those virtual domains to assist in the assessment process [10].

Simulation can never be validated in its entirety [13] as errors emerge due to modeling and simplifying the reality. However, it is possible to compare a test case in reality to the simulation and verify the simulation locally. Since it is not possible to reconstruct a virtual test case on real roads with adequate precision, reprocessing of actual driving tests seems more promising. Still, errors emerge along the pipeline of reprocessing. It has been shown that quantification of those is possible and supports the local cross-verification of test cases [14].

In this work, we introduce a pipeline for reprocessing driving scenarios recorded in the real world. Based on a deeper analysis of errors, the following hypotheses are established:

1) The reprocessing error can be quantified and tracked back to its sources
2) The local sensor offset causes behavior changes in the reprocessing that can be avoided by scenario-based sensor models
3) A locally verified simulation can enable valid local scenario variations that participate in the assessment process

The rest of this work is structured as follows. Section II describes the reprocessing pipeline in detail before the subsequent analysis isolates the major error sources in section III. Afterward, in section IV, an investigation of the error causes leads to the use of scenario-based sensor models. Under the assumption of a locally verified simulation, section

* These authors contributed equally to this work

S. Wagner and A. Knoll are with the Chair of Robotics, Artificial Intelligence and Embedded Systems, Department of Informatics, Technische Universität München, Munich, Germany

K. Groh is with the BMW AG, Munich, Germany

T. Kühbeck is with BMW NA, Mountain View, USA

978-1-7281-0560-4/19/$31.00 ©2019 IEEE

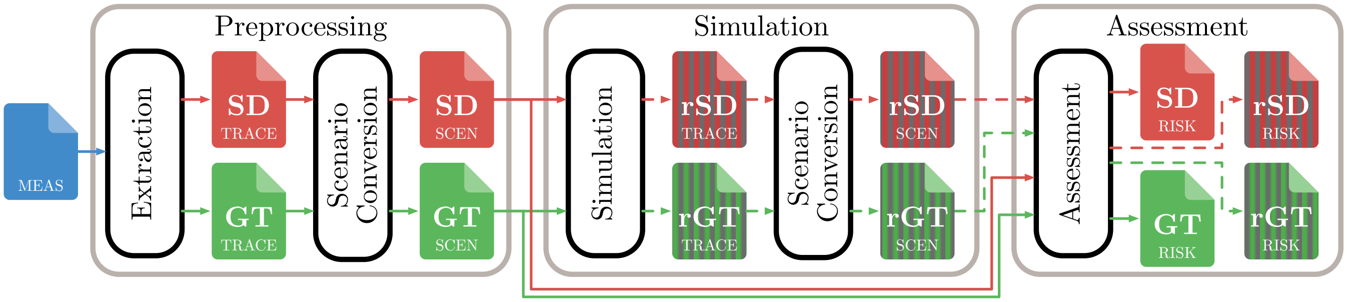

Fig. 1. Flow chart of the reprocessing pipeline presented in this work. Both Ground Truth (GT) and Sensor Data (SD) traces are extracted from a measurement of the prototype vehicle and converted into the according scenario description. A resimulation of the latter provides traces and scenarios for GT (rGT) and SD (rSD) respectively. The assessment evaluates the behavior of the AD function in all available scenarios.

V provides prototypic results of a local scenario variation and its benefits to the assessment of AD. Finally, the conclusion is drawn in section VI.

II. Reprocessing Pipeline

In this section, the individual components of the applied reprocessing pipeline are discussed in detail. Fig. 1 shows a flow chart of the complete reprocessing pipeline and its three major modules - preprocessing, simulation and assessment. The starting point is a measurement from the prototype setup explained later in section II-A containing the recorded real-world measurements. The reprocessing module extracts relevant information from the measurement and converts the traces into a scenario description format. This format contains all relevant information to be interpreted and resimulated by a simulation framework with the same embedded AD function as in the real-world experiment. Resulting traces are again converted into the scenario description to be evaluated in the assessment module based on a sensitive accident risk measure and compared to the original ones. The described pipeline is executed in parallel for recorded Sensor Data (SD) and for Ground Truth (GT). Further, the resimulations of GT and SD are addressed as rGT and rSD respectively in the remainder of this work.

The focus of this section is on the functionality and necessity of each component, whereas the resulting approximation errors in comparison to the real-world test and their propagation through the reprocessing pipeline is covered in the following section III.

A. Measurement Setup

Before discussing each component individually the creation of an initial measurement by a real-world experiment is reviewed. As illustrated in the flow chart of Fig. 1, the measurement contains the SD meaning the information from the vehicles standard sensor equipment as well as the GT information. The latter is an approximation as reality can not be described in arbitrary detail. However, the applied Differential Global Positioning System (DGPS) depicted in Fig. 2 allows an accuracy of $1.0cm$ for the position, $0.05km/h$ for the velocity and $0.1rad/\pi$ for the vehicle heading through a reference antenna with a known position. In comparison to other error magnitudes discussed later, this can be interpreted as GT information.

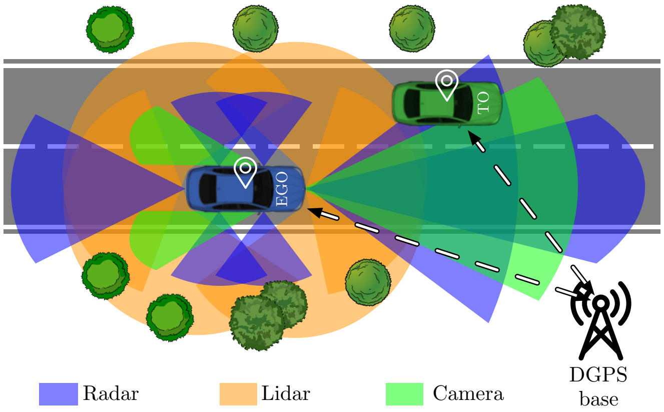

Fig. 2. Sensor setup of the prototype vehicle used for recording the measurements in this work. The depicted cones only show the schematic sensors arrangement and field of view for the tree sensor types Radar, Lidar and camera. GT is obtained through a DGPS system using a reference antenna with known position.

An AD function strongly relies on two kinds of measurements. The first is the aggregation of the data of the vehicle equipped with the AD function itself, namely the EGO vehicle, which is required for self-localization and path planning. Secondly, the understanding of its surrounding including the road, obstacles and other traffic objects (TOs) defines the basis for the AD function's decisions. Hence, a wide range of sensors is key components to the system [15]. The applied sensor setup for the EGO vehicle shown in blue is illustrated in Fig. 2. It includes Radar, Lidar and camera systems. The sensor fusion of the AD function takes the provided information of all three systems with an individual confidence level into consideration and generates the fused object list [16]. Within this list, each detected object is assigned a unique ID and its corresponding dynamic and static quantities describing its state. In combination with the localization on the map, the object list including predictions of its states illustrates the environment model for the AD function. Further, planning and decision making in the function relies on the odometry data of the EGO vehicle and the generated environment model and is referred to as SD.

The accuracy among the acquired data for a single object in the SD varies depending on the sensor setup. In general, the position measurement is the most accurate quantity whereas velocity or especially heading have a larger error. In case of velocity and heading it is challenging to obtain meaningful values from cameras or point clouds and thus these quantities are mostly calculated by changes in position. As a consequence, this calculation method leads often to a time delay and an inconsistency between the measured position and velocity or heading values in the object list. Section III gives a concrete overview of the sensor errors which are present in the acquired data by directly comparing the SD to the GT in terms of position, velocity and heading.

B. Scenario Description

An important role in the reprocessing pipeline is the description of a scenario. Particularly, a scenario in the automotive context is a timely series of scenes, each containing static and dynamic properties of the traffic participants and their surroundings [17]. In addition, the description has to fulfill certain requirements [14] to be in accordance with a suitable assessment method for AD [7]:

- The description has to be unambiguous concerning all objects and their temporal sequences.
- The description has to be consistent in case of redundant information.
- The degree of description accuracy has to be sufficient to not affect the performance of the AD function by simplifying relevant information.

The first requirement demands an unambiguous description in the manner of a recipe on how the scenario is designed. This means there must not be an uncertainty in the description of a scenario which could be interpreted differently in any simulation framework. The results from two individual simulation frameworks will most likely differ but this is part of the simulation error and not an issue of the scenario description. This can be solved by a standardized implementation of the scenario interpretation module within the simulation framework, for example, a trajectory follower for the simulated TO. A further assessment and performance comparison of the simulation results would be meaningless as, indeed, already the underlying scenario is not the same due to the free interpretation possibility. Thus, the missing unambiguous feature would also prevent the aggregation of results from different sources like for example a critical scenario database of different manufacturers and the associated knowledge gain.

The second required feature for the description is a consistent representation of the listed quantities [10]. This relates especially to the representation of an object trajectory. In general, a list of position points with corresponding timestamps is under the condition of a sufficient sampling rate equally suited as a time depended velocity and heading curve including a starting position. Of course, there is no reason not to include redundant representations but they have to be consistent among each other. The arising problem from this consistency requirement happens by converting measured data into the description. As already mentioned in section II-A, the quantities in the object list of SD are not consistent and one has to choose a certain representation. Indeed, the AD function is planning on a scenario which is not existing in reality as its quantities are inconsistent. Again the precise errors caused by converting GT and SD into the scenario description is covered in section III.

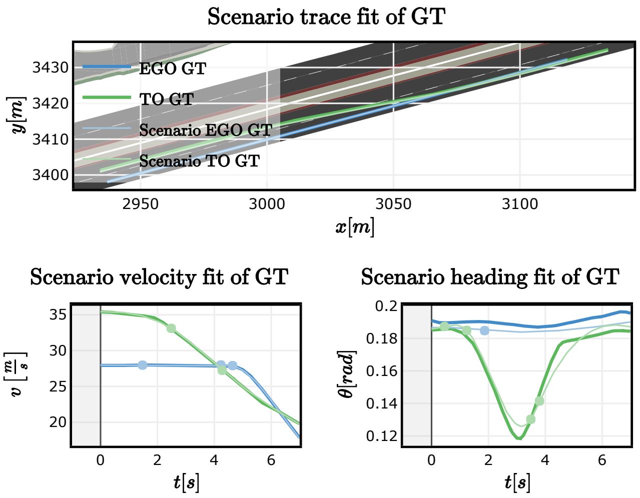

Fig. 3. The scenario description fit on GT data for position, velocity and heading is shown as thin green lines for TO and thin blue lines for EGO. This fit is superimposed on the actual GT data shown as thicker lines.

Lastly, the representation of traces in the description chosen for this work is discussed. Considering the accuracy of the position coordinates in the measured data on the one hand and the clearer and less inflated structure of a time-dependent curve on the other hand, leads to both benefits being combined for a description of traces. This means, the description uses the curve representation in terms of velocity and heading but the data is obtained by fitting the measured position data via equation (1).

$$\begin{aligned} \dot{x}_1(t) &= v(t) \cdot \cos(\theta(t)) \qquad & \dot{\theta}(t) &= \omega \\ \dot{x}_2(t) &= v(t) \cdot \sin(\theta(t)) \qquad & \dot{v}(t) &= a \end{aligned} \tag{1}$$

The velocity and heading curves are implemented as independent splines of cubic polynomials. The number of spline sections for velocity and heading is dynamic and automatically determined by increasing the number of sections until a desired threshold in the cost function is reached. This threshold is engineered to be small enough to be negligible compared to the inconsistency error raised by the conversion into the scenario description. Now fitting the splines formulates as an optimization problem with their spline parameters as optimization target. Within the cost function, heading and velocity splines are integrated by the use of equation (1) yielding position values. Then, the root-mean-square error of the Euclidean distance to the measured position results in the cost value for minimization.

Fig. 3 shows an exemplary scenario fitting on GT data. On the bottom, the measured velocity and heading values, which are not part of the fitting process itself, are superimposed with the resulting splines from the position-based fit. The upper plot compares the position resulting from the integra-

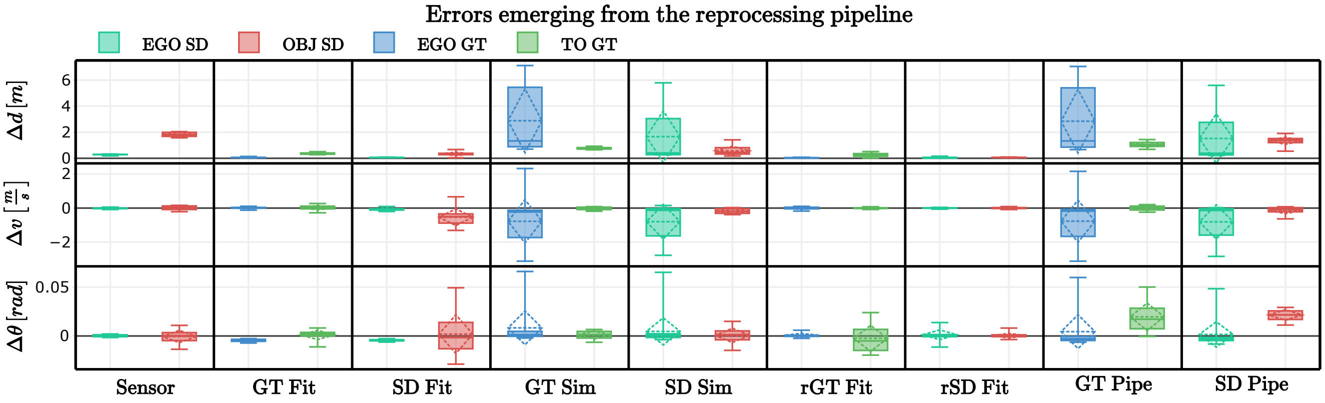

Fig. 4. Errors emerging from the reprocessing pipeline for distance d, velocity v and heading θ as rows. The columns represent the steps of the pipeline for either GT or SD. Each cell contains boxes for the EGO vehicle and TO colored according to their type respectively. The boxes themselves enframe lower (25%) to upper (75%) quantile with whiskers to the full extent. Error mean and standard deviation are embedded as diamonds in the boxes.

tion of those splines trough equation (1) with the original data and shows that errors in the trajectory can hardly be observed. Hence, the shown scenario description is capable of representing the scenario while being consistent and unambiguous. The exact errors of this fit and the difficulty with inconsistencies in SD are further considered in section III.

C. Resimulation

The next step in the reprocessing pipeline is the resimulation of the described scenario. The simulation framework will guide the described objects along their predefined time-depended path given by the velocity and heading spline from the scenario description. Besides its starting conditions, the AD function in the EGO vehicle is free to perform during the resimulated scenario. Of course, the applied function is the same as the one used in the real-world experiment. The quality of the underlying physical models, which are responsible for the movement of the EGO vehicle according to the desired path of the AD function in the simulation framework, is partly responsible for the simulation error. Another part of this error is the capability of the simulation of following the scenario description in terms of matching the starting conditions and preserving object trajectories.

D. Risk Assessment

A useful test framework should always provide measures for quantifying the test results in an understandable manner. A test's passed or failed condition can hardly be derived by just inspecting the measured or simulated traces itself. Hence, the very last part of the presented pipeline calculates a measure assessing the behavior of the AD vehicle. Certainly, a wide range of measures concerning accident risk, passenger comfort and compliance with traffic laws, etc. is required for the final release of AD [10]. For this proof of concept, we focus on a single measure concerning accidents as avoiding those is the primary goal of the function.

The conversion from pure trace to such a measure is described in [18]. Herein a set of possible predictions for TOs defines the possible outcomes of a driving scene – a single step in time of the scenario – at first. These predictions incorporate naturalistic driving behavior derived from the euroFOT large scale driving study [19] and the vehicle model from equation 1. This set is counterchecked with the planned path of the EGO vehicle for possible accidents in the near future. Evasion trajectories for every possible accident lead to the Time-to-React (TTR) value. The weighted TTRs are consolidated considering the trajectorie's probability into a single value of risk for the regarded scene. Iteration over all scenes of the scenario results in a time series of accident-related risks.

In this framework, the measure is computed on the data from the scenario description as it is physically consistent, which is crucial for the underlying predictions. As a low accident risk is a direct evidence for a well-performing AD function, this measure is suitable for the assessment pipeline.

Having discussed all required components for the reprocessing pipeline, a deeper insight on arising errors is necessary to verify the usability of the simulation.

III. Error Propagation

The introduced pipeline enables the virtual reprocessing of real-world experiments that are necessary to locally verify the simulation. Certainly, virtual testing will never reveal the exact same result as a real-road drive. In this section, the sources and magnitudes of errors emerging from the reprocessing pipeline are discussed. After every step of the pipeline, there exists a representation of vehicle traces. This enables isolation and quantification of the errors by comparing input and output traces of every step independently. The used test case for this example is a cut-in and brake maneuver, in which the EGO vehicle drives $100 km/h$ autonomously on the right lane and is inferred by a TO cutting into the lane from the left followed by a braking maneuver. Fig. 4 accompanies this section by showing the magnitudes of errors for the position Δd, velocity Δv and heading $\Delta\theta$ in each row. The columns represent the steps in which the errors emerge, separated by SD and GT, while each cell contains box plots for the EGO and TO errors. The boxes itself range

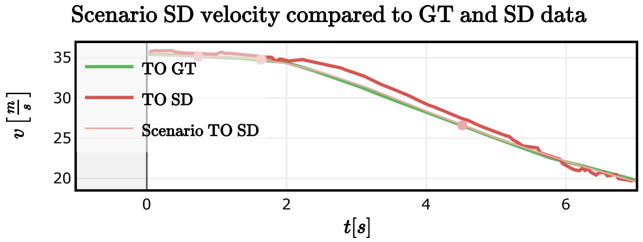

Fig. 5. The fitting of SD is compared to GT and SD traces. This fit converges towards GT and hence corrects the error in SD.

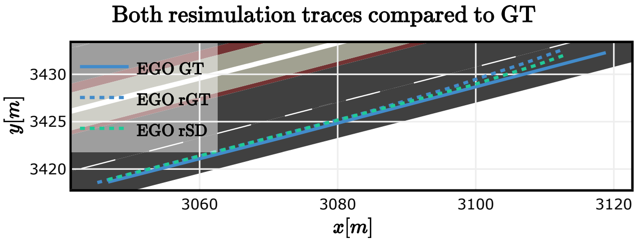

Fig. 6. The GT trace is compared to those of GT and SD resimulation for the last 3s of the scenario. In both resimulations, the EGO behaves differently to the measured GT by initiating a lane change to the left.

from the lower (25%) to the upper (75%) quantile and the whiskers depict the complete extent of the error. Lastly, the inner diamond of the boxes represent the errors mean and standard deviation.

The first error emerges through sensors equipped by the EGO vehicle and is depicted in the first column of Fig. 4. As already explained in section II-A, the prototype vehicle features both standard AD sensors and reference sensor technology considered as SD and GT respectively. The self-measurement thus the odometry of the EGO vehicle reveals negligible offsets, whereas the TO imposes greater difficulty to the sensor setup. While Δv and $\Delta\theta$ are unbiased, the position offset of the TO Δd shows a continuous offset of $\sim 1.4m$, which in this scenario makes the TO appear farther away than it is in reality. Hence, the SD scenario seen by the EGO vehicle and the GT scenario differ significantly from each other. Therefore, the subsequent errors of the pipeline are presented for both cases individually.

In the next step, the data is made interpretable for the simulation framework by fitting into the scenario description format from section II-B. Columns 2 and 3 of Fig. 4 reveal that the errors arising by fitting the GT are significantly lower as with SD. While the reference sensors deliver accurate and physically consistent data, which enable a precise fitting of the scenario, the SD is not as consistent. However, an unambiguous scenario description must be consistent and errors emerge by compromising the inconsistency within SD. Fig. 5 compares GT and SD velocity to the scenario fit of SD. As the fitting is performed solely on position values over time, the inconsistency of velocity not matching those position values is removed. Furthermore, due to the position errors low variance, the velocity is "corrected" towards the GT during fitting. Solely the position offset of the sensor error explained before remains in the SD scenario besides the fitting error itself.

Additional errors arise from the simulation of the scenarios itself due to inaccuracies in modeling the physical world. They are presented in columns 4 and 5 of Fig. 4. The error of EGO and TO have to be considered apart from each other. While the TO error is only defined by the capability of the simulation to follow the predefined traces in the scenario, the EGO vehicle is free in its movement and decisions. For GT the vehicle is presented with accurate data it has not seen in the real driving test. For example, the described position offset is not present. Hence, in this scenario, the TO appears closer to the EGO vehicle than the sensors have seen it. Fig. 6 compares the GT trace of the EGO vehicle to both GT and SD resimulated traces. The more accurate detection of the TO causes a behavior change in the EGO vehicle already observed in [14]. Due to the closer appearance of the TO to the EGO vehicle and the greater deceleration, the AD function decides to avoid the TO on the left instead of just braking. This change is also present in the resimulation of SD. Although the position offset is present in this case, the correction of velocity reveals the deceleration earlier to the EGO vehicle due to the explained removal of the delay. Due to the position offset being present in the SD resimulation, its error is smaller in the SD pipeline than it is in the GT pipeline.

Afterward, the simulated traces are fitted into the scenario description again to make the assessment and comparability to the original scenario easier. As the simulation provides smooth and consistent data in both cases, the scenario fitting errors depicted in columns 6 and 7 of Fig. 4 are negligibly low.

Lastly, columns 8 and 9 summarize the errors of the whole pipelines from the GT information to either rGT or rSD scenario fit. It becomes apparent that the pipeline errors major influence is the simulation error with the extent being lower in the SD pipeline.

Summarizing, the errors in the reprocessing pipelines are constructed by

$$\begin{aligned}\Delta\mathrm{RP}_{\mathrm{GT}} &= \Delta\mathrm{FIT}_{\mathrm{GT}} + \Delta\mathrm{SIM}_{\mathrm{GT}} + \Delta\mathrm{FIT}_{\mathrm{rGT}}\\ \Delta\mathrm{RP}_{\mathrm{SD}} &= \Delta\mathrm{S} + \Delta\mathrm{FIT}_{\mathrm{SD}} + \Delta\mathrm{SIM}_{\mathrm{SD}} + \Delta\mathrm{FIT}_{\mathrm{rSD}},\end{aligned} \tag{2}$$

where $\Delta\mathrm{FIT}_{\mathrm{GT,SD,rGT,rSD}}$ are the fitting errors for the respective traces, $\Delta\mathrm{SIM}_{\mathrm{GT,SD}}$ are the simulation errors and $\Delta\mathrm{S}$ is the sensor error. The components $\Delta\mathrm{FIT}_{\mathrm{GT}}$, $\Delta\mathrm{FIT}_{\mathrm{rSD}}$, and $\Delta\mathrm{FIT}_{\mathrm{rGT}}$ are negligibly low. $\Delta\mathrm{SIM}_{\mathrm{GT}}$ is the main influence on the GT reprocessing pipeline error and is caused by the EGO vehicle seeing the GT as opposed to reality, where SD is present. Contrarily, the SD reprocessing pipeline has multiple significant errors. The deleted inconsistencies are expressed in $\Delta\mathrm{FIT}_{\mathrm{SD}}$, which in succession also causes the behavior change manifested in $\Delta\mathrm{SIM}_{\mathrm{GT}}$. Lastly, $\Delta\mathrm{S}$ presents the offset to the actual GT scenario. Hence, the error progression can approximately be reduced to

$$\begin{aligned}\Delta\mathrm{RP}_{\mathrm{GT}} &\approx \Delta\mathrm{SIM}_{\mathrm{GT}}\\ \Delta\mathrm{RP}_{\mathrm{SD}} &\approx \Delta\mathrm{S} + \Delta\mathrm{FIT}_{\mathrm{SD}} + \Delta\mathrm{SIM}_{\mathrm{SD}}.\end{aligned} \tag{3}$$

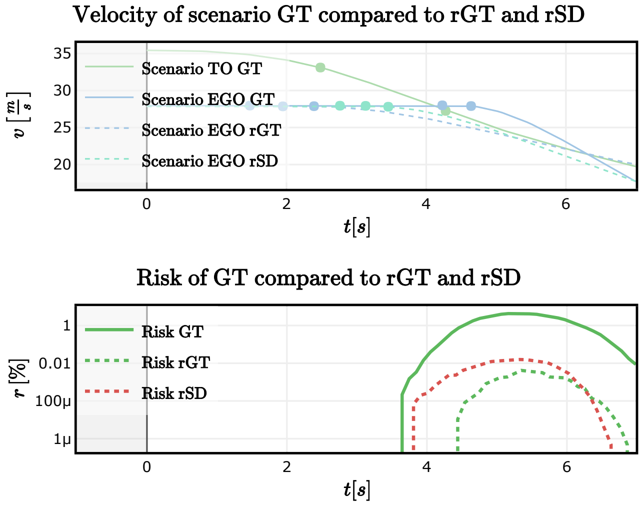

Fig. 7. The EGO vehicles GT velocity trace compared to both resimulations shows the behavioral changes caused by errors in the pipelines in the upper half. For reference, the TO's velocity is also shown. The lower half expresses this effect on the risk measure for the same case.

Both pipelines show significant changes in the outcome with respect to the EGO vehicle's behavior, making a straight forward use of the simulated results in the process of AD assessment not suitable. However, the sources and the cause of the errors can be isolated and quantified, confirming hypothesis 1. The next section discusses an approach to use the gathered information for improving and locally verifying the simulation.

IV. Towards Local Cross-Verification of Simulation

So far a reprocessing pipeline and an in-depth analysis of the arising errors is presented. It is shown, that due to these errors a behavioral change of the AD function can happen in the virtual world. Such changes put the simulated scenario in a different category than the measured one, making a direct comparison for local verification meaningless. This section analyses the cause of the changes and suggests a solution to avoid those.

The upper half of Fig. 7 compares the velocity profiles of the GT scenario to both resimulations of GT and SD for the known measurement of section III. We observe the reaction to the interfering TO as drops in the velocity profiles of the EGO vehicle in GT, rGT and rSD shown in solid blue, dashed blue and dashed cyan respectively. Comparing the resimulated GT behavior to the actual behavior reveals an earlier reaction of approximately $1s$ in the simulation:

$$\Delta t_{\text{react,GT}\leftrightarrow\text{rSD}} = t_{\text{react,GT}} - t_{\text{react,rGT}} \approx 1s. \tag{4}$$

Considering the observed over-approximation of $\sim 1.4m$ in the TOs distance in SD and the EGO vehicles speed of $100km/h$, the reaction should only happen

$$t_{\Delta_{\text{S,d}}} = \frac{\Delta_{\text{S,d}}}{v_{\text{EGO}}} \approx 0.05s \tag{5}$$

later for resimulation. However, the observed reaction delay is by far greater

$$\Delta t_{\text{react,GT}\leftrightarrow\text{rSD}} \gg t_{\Delta_{\text{S,d}}}. \tag{6}$$

Now comparing rGT to rSD with respect to the same reaction criteria, the observed difference reveals approximately the calculated difference from the miss-sensed position of the TO:

$$\Delta t_{\text{react,rGT}\leftrightarrow\text{rSD}} \approx t_{\Delta_{\text{S,d}}} \tag{7}$$

In section III it is already observed that the sensor measurements contain physical inconsistencies that can not be reproduced within the simulation, causing the simulation error to dominate the overall pipeline error. This inconsistency manifests in the observed scenario by timely delays of the TO's velocity and heading, which is neglected by the choice of generating the scenario description in both GT and SD. As most of the reaction difference can not be explained by the offset in vehicle distance in GT and SD, which still remains in the scenario description, it becomes obvious that the dynamic sensor error is the main cause of the behavioral change.

This observation is further reinforced by the lower half of Fig. 7, where the comparison of the same scenario is shown for the risk measure of section II-D. While both resimulations, rGT depicted in dashed green and rSD in dashed red, remain in the same magnitude, while the original GT risk in solid green is more than a magnitude greater. Due to the nature of the applied risk measure and the resemblance as accident probabilities as rare events, the shown values differ in magnitudes. Hence, a logarithmic scale is suitable to make the measure comparable.

It becomes apparent that without an appropriate representation of the sensors in the simulation the local cross-verification is not possible. Just adding the known and observed offset in the simulation might offer reasonable results for this exact scenario, but not for subsequent variations in the local subspace. Additionally, this offset is only known with present reference GT sensors. Hence, the deployment of suitable sensor models is not evitable.

The given observations show that for this specific scenario time delays, as manifested in the dynamic sensor error, biased offsets and sensor noises are present. Especially the time delay and bias are dependent on the driven scenario itself, which suggests the use of scenario-based sensor models to improve the simulation. Such a model is assumed to be derivable from a reasonable set of real-world test drives with the same equipped reference sensors, as the errors are dependent on the static and dynamic properties of the TO and interpolation is possible to a certain degree. A model featuring both static and dynamic errors is applicable to the reprocessing of GT, while SD only requires the dynamic part as static errors are preserved.

It is also apparent that the risk measure from Fig. 7 is robust against static errors such as $\sim 1.4m$, revealing only minor changes. Hence, eradicating the behavioral changes in the simulation through the suggested sensor models promises accurate local simulative results in risk measure with the present reprocessing pipeline.

Having the same behavior in both reality and simulation makes measuring the distance of the results and hence locally verifying the simulation meaningful, encouraging hypothesis 2. Combined with the described sensor models, variations in the local space of the scenario promise deeper insight into the AD function behavior and deliver valid virtual test kilometers.

V. Simulative Assessment through Variation

So far by just accurately reprocessing a scenario from reality, no additional information with respect to the assessment of the AD function itself is gathered. The scenario can also be assessed with its real-world information. Based on the assumption that if the simulation can reprocess a given scenario accurately enough, local variations in purely a simulative domain also provide valid results. Exploring the local scenario space does not only generate additional test kilometers for the statistical reliability proof, but it also provides information about the local risk gradient for critical scenarios. The purpose of this kind of explorations is not to find completely new kinds of scenarios but to create meaningful, physically feasible and slightly changed derivatives of the original scenario. The confidence in such variations is coupled to the simulations local accuracy derived from the previous cross-verification.

The generation of a representative ensemble composed of locally related scenarios is a sophisticated task. Only little research is available in this area [20]–[22] and most of it targets global variation and test case generation, which is not suitable for this work. A simple variation of certain parameters in the description is not valid as many parameters are not independent of each other. The most complicated process is the meaningful variation of an object trajectory as it is expressed by parameters which have to preserve the C^2 continuity of the velocity and heading spline. In detail, variating the position coordinates in order to form a meaningful cut-in geometry without taking care of the temporal component will lead to jumps in the velocity or heading over the time profile. Therefore, geometrical variations to the cut-in geometry are only applied locally around the point of crossing the lane of the EGO. This is established by fitting a cubic Bèzier spline to the geometry of the trajectory. The trajectory is divided into equitemporal sections with the start and end point of each section marking the two fixpoints of a Bèzier curve. The two remaining support points which are required for a cubic Bèzier curve are optimized under the conditions of C^2 continuity between two neighboring Bèzier curves. To yield a unique solution for this optimization problem, four additional constraints have to be added. These can be utilized to fix the velocity and heading values at the start of the first and at the end of the last Bèzier curve. Based on the constructed Bèzier curve only certain fix points can be variated which after the optimization process leads mainly to a local change of the trajectory shape. In the case of a cut-in scenario, we choose fix points around the lane change for variation. Bèzier curves which are further from the variation located are less affected by trajectory changes.

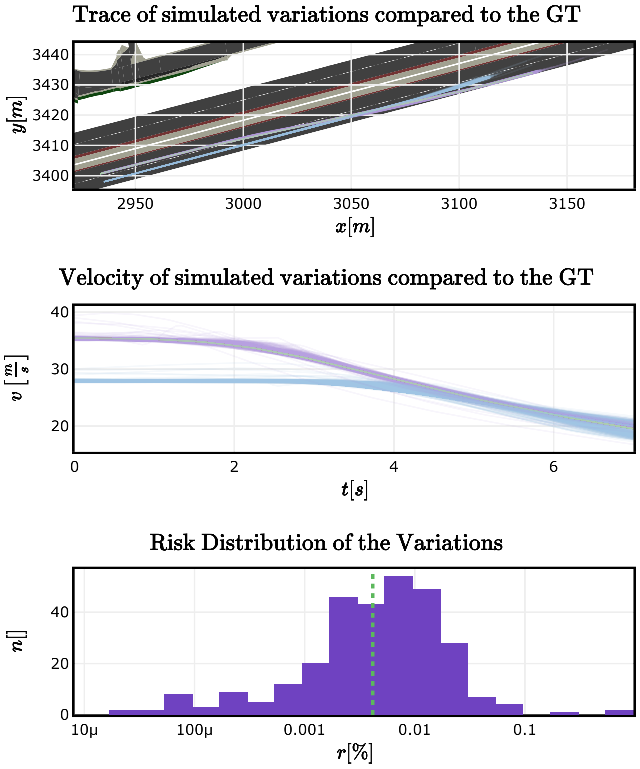

Fig. 8. The simulations of 300 variations of the original scenario are compared to the resimulation of GT by trace (upper), velocity (middle) and risk (lower). The variations of the TO are shown in purple and the for the EGO in blue. The risk of the rGT scenario is shown as reference in green.

However, the newly generated shape of the trajectory still lacks a smooth temporal representation for heading and velocity around the variation as the varied fix points keep their original time component. A solution to this problem is exploiting the smooth nature of splines by applying again the fitting algorithm explained in section II-B. This method works only under the condition that the effects of the variation are small and locally limited in comparison to the length of the trajectory. This way, it is more costly for the loss function to make many changes along the only slightly variated trajectory parts than it is to fix the strongly variated sections. Besides the smoothing effect of the fitter, the trajectory is already expressed in the representation required for the scenario description.

Certainly, this method of variation is not optimal and might not always yield naturalistic and physically realistic trajectories. Nevertheless, for this proof of concept, it is sufficient as we only want to demonstrate that a purely local variation can support the assessment of AD. For that reason, the variation is not verified in more detail concerning a realistic representation of the local scenario space.

An example variation for the previously observed scenario is given in Fig. 8. The upper and middle part show the traces and velocities of 300 variations of the TO in purple and the simulated EGO vehicle in blue. Close local variation can be observed from the closeness of the paths in the trace plot on the road. Similar velocity profiles and traces prove the variation's affiliation to the same scenario category and hence are comparable to the original resimulated GT. However, a few outliers are present due to the non-optimal variation used. Also, the reaction time of the simulated EGO vehicle varies according to the different TO behavior. As the sensor models suggested in section IV are not yet implemented, the EGO vehicle is expected to behave similarly to the resimulated GT and not the measured GT. The lower part of Fig. 8 superimposes the distribution of the risk measures maximum over the actual maximum from GT. It can be observed that also the accident-related risk of the variations assemble around the rGT marked in green, emphasizing the validity of the simulative results and local variation.

Summarizing, this section shows that local variations around a given verified scenario can yield additional valuable and valid information to the assessment of AD, confirming hypothesis 3. However, questions concerning the valid range of this local variation combined with a measure of validity and the method behind it itself still remain open to further research. Also, it has to be guaranteed that naturalistic object trajectories are provided.

VI. Conclusion and Future Work

After declaring the necessity of virtual test domains, this work introduces a reprocessing pipeline with the goal to locally verify simulation for its deployment in the assessment process of AD. An in-depth investigation of arising errors reveals the major sources of those and isolated their cause. Realizing that the latter even cause behavioral changes in the AD function concluded that a straight forward cross-verification of simulation with real-world driving tests is meaningless. Further, the behavioral changes are investigated and scenario-based sensor models are suggested as a promising relief to this issue. Based on the assumption of a locally cross-verified simulation, a prototypic local scenario variation is applied. The results show that even a simple variation can provide additional test kilometers and is a promising starting point to gather more information about the risk measures local gradient.

Overall, the presented method to include simulation as a valid component in the assessment process promises to produce relief to the required amount of test kilometers. The proof of concept is investigated on exemplary real data with corroborative results. The remaining required parts to fulfill this goal have been derived. Hence, future work and research have to focus on developing and validating the suggested scenario-based sensor models. Additionally, the applied variation needs to be improved regarding its range of validity and compliance to naturalistic TO behavior. Finally, a measure of confidence in the simulative kilometers bound to the cross-verification is desirable.

Acknowledgment

The authors gratefully acknowledge the financial support by the BMW Group within the CAR@TUM project.

References

[1] SAE International, "Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles," SAE international, Tech. Rep. J3016, Sep. 2016.

[2] E. Taylor, "BMW says self-driving car to be level 5 capable by 2021," Apr. 2017, https://ca.news.yahoo.com/bmw-says-self-driving-car-level-5-capable-140103759–finance.html, accessed 24 Aug. 2018.

[3] F. Lambert, "Tesla CEO Elon Musk: 'self-driving will encompass all modes of driving by the end of next year," Mar. 2018, https://electrek.co/2018/03/11/tesla-ceo-elon-musk-self-driving-next-year/, accessed 16 Oct. 2018.

[4] W. Wachenfeld and H. Winner, "The New Role of Road Testing for the Safety Validation of Automated Vehicles," in *Automated Driving*. Springer, 2017, pp. 419–435.

[5] "ISO/DIS 26262-1 - Road vehicles — Functional safety," 2011.

[6] H. Winner and W. Wachenfeld, "Absicherung automatischen Fahrens," Munich, 2013.

[7] W. Wachenfeld and H. Winner, "Die Freigabe des Autonomen Fahrens," in *Autonomes Fahren*. Springer, 2015, pp. 439–464, doi:10.1007/978-3-662-45854-9_21.

[8] C. Amersbach and H. Winner, "Functional Decomposition: An Approach to Reduce the Approval Effort for Highly Automated Driving," in *8. Tagung Fahrerassistenz*, 2017.

[9] General Motors Company, "Self-driving safety report," Detroit, 2018, https://www.gm.com/content/dam/company/docs/us/en/gmcom/gmsafetyreport.pdf, accessed 16 Oct. 2018.

[10] K. Groh, T. Kuehbeck, M. Schiementz, and C. Chibelushi, "Towards a Scenario-Based Assessment Method for Highly Automated Driving Functions," in *8th Conference on Driver Assistance*, Munich, Sep. 2017.

[11] H. Winner, S. Hakuli, and G. Wolf, *Handbuch Fahrerassistenzsysteme: Grundlagen, Komponenten und Systeme für aktive Sicherheit und Komfort*. Springer-Verlag, 2011.

[12] T. Helmer, K. Kompaß, L. Wang, T. Kühbeck, and R. Kates, "Safety Performance Assessment of Assisted and Automated Driving in Traffic: Simulation as Knowledge Synthesis," in *Automated Driving*. Springer, 2017, pp. 473–494.

[13] D. J. Murray-Smith, *Testing and Validation of Computer Simulation Models*. Springer, 2015, doi:10.1007/978-3-319-15099-4.

[14] K. Groh, S. Wagner, T. Kuehbeck, and A. Knoll, "Simulation and its contribution to evaluate highly automated driving functions," in *SAE Technical Paper*. SAE International, 04 2019.

[15] E. M. Guillaume Girardin, "Sensors and Data Management for Autonomous Vehicles," Yole Développement, Tech. Rep., 2015.

[16] K. Banerjee, D. Notz, J. Windelen, S. N. Gavarraju, and M. He, "Online Camera LiDAR Fusion and Object Detection on Hybrid Data for Autonomous Driving," in *2018 IEEE Intelligent Vehicles Symposium, IV 2018, Changshu, China, June 26-29, 2018*. IEEE, 2018.

[17] S. Ulbrich, T. Menzel, A. Reschka, F. Schuldt, and M. Maurer, "Defining and substantiating the terms scene, situation, and scenario for automated driving," in *Intelligent Transportation Systems (ITSC), 2015 IEEE 18th International Conference on*. IEEE, 2015, pp. 982–988.

[18] S. Wagner, K. Groh, T. Kuehbeck, M. Doerfel, and A. Knoll, "Using Time-To-React Based on Naturalistic Traffic Object Behavior for Scenario-Based Risk Assessment of Automated Driving," in *Proceedings of the IEEE Intelligent Vehicle Symposium*. Changhsu, China: IEEE, 2018, pp. 1521–1528.

[19] C. Kessler and A. Etemad, "European Large-Scale Field Operational Tests on In-Vehicle Systems - SP 6 D6.8 FOT Data," Ford Research & Advanced Engineering Europe, Tech. Rep., Jun. 2012.

[20] A. Andrews, M. Abdelgawad, and A. Gario, "Towards world model-based test generation in autonomous systems," in *2015 3rd International Conference on Model-Driven Engineering and Software Development (MODELSWARD)*, Feb. 2015, pp. 1–12.

[21] C. Sippl, F. Bock, D. Wittmann, H. Altinger, and R. German, "From simulation data to test cases for fully automated driving and ADAS," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9976 LNCS, pp. 191–206, 2016.

[22] E. Rocklage, H. Kraft, A. Karatas, and J. Seewig, "Automated scenario generation for regression testing of autonomous vehicles," in *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, Oct. 2017, pp. 476–483.