SAE Int. Journal of Connected and Automated Vehicles, "Simulation-Based Identification of Critical Scenarios for Cooperative and Automated Vehicles", doi:10.4271/2018-01-1066.

4 authors:

Sven Hallerbach
Carl von Ossietzky Universität Oldenburg

Yiqun Xia
RWTH Aachen University

Ulrich Eberle
Opel Automobile / Groupe PSA

Frank Köster
German Aerospace Center (DLR)

ARTICLE INFO
Article ID: 2018-01-1066

doi:10.4271/2018-01-1066

Simulation-Based Identification of Critical Scenarios for Cooperative and Automated Vehicles

Sven Hallerbach, Yiqun Xia, and Ulrich Eberle, *Opel Automobile GmbH, Germany*
Frank Koester, *German Aerospace Centre, Germany*

Abstract

One of the major challenges for the automotive industry will be the release and validation of cooperative and automated vehicles. The immense driving distance that needs to be covered for a conventional validation process requires the development of new testing procedures. Further, due to limited market penetration in the beginning, the driving behavior of other human traffic participants, regarding a mixed traffic environment, will have a significant impact on the functionality of these vehicles.

In this article, a generic simulation-based toolchain for the model-in-the-loop identification of critical scenarios will be introduced. The proposed methodology allows the identification of critical scenarios with respect to the vehicle development process. The current development status of the cooperative and automated vehicle determines the availability of testable simulation models, software, and components.

The identification process is realized by a coupled simulation framework. A combination of a vehicle dynamics simulation that includes a digital prototype of the cooperative and automated vehicle, a traffic simulation that provides the surrounding environment, and a cooperation simulation including cooperative features is used to establish a suitable comprehensive simulation environment. The behavior of other traffic participants is considered in the traffic simulation environment.

The criticality of the scenarios is determined by appropriate metrics. Within the context of this article, both standard safety metrics and newly developed traffic quality metrics are used for evaluation. Furthermore, we will show how the use of these new metrics allows for investigating the impact of cooperative and automated vehicles on traffic. The identified critical scenarios are used as an input for X-in-the-Loop methods, test benches, and proving ground tests to achieve an even more precise comparison to real-world situations. As soon as the vehicle development process is in a mature state, the digital prototype becomes a "digital twin" of the cooperative and automated vehicle.

History

Received: 26 Apr 2018
Revised: 03 Aug 2018
Accepted: 15 Aug 2018
e-Available: 27 Dec 2018

Keywords

Automated vehicles, Cooperative vehicles, Simulators, Test facilities, Test procedures, Simulation and modeling, Identification and verification, Simulation-based testing, Automotive engineering, Traffic quality

Citation

Hallerbach, S., Xia, Y., Eberle, U., and Koester, F., "Simulation-Based Identification of Critical Scenarios for Cooperative and Automated Vehicles," *SAE Int. J. of CAV* 1(2): 93–106, 2018, doi:10.4271/2018-01-1066.

ISSN: 2574-0741
e-ISSN: 2574-075X

This article is based on and revised or modified from a presentation at WCX18, Detroit, MI, April 10-12, 2018.

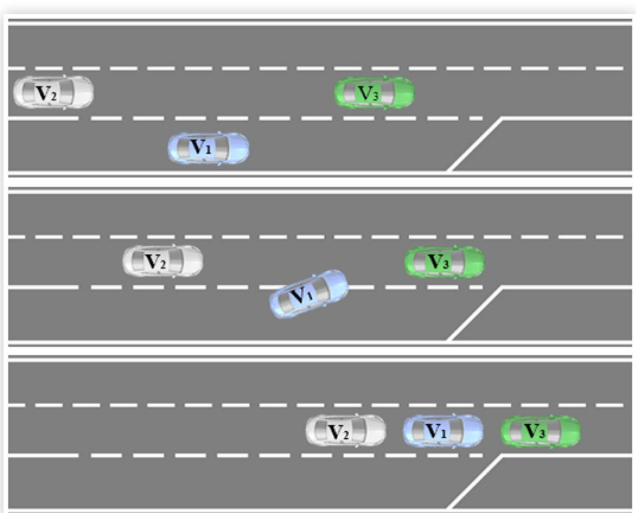
Introduction

The release of cooperative and automated vehicles is impeded due to a small market penetration in the beginning. The task to include these vehicles into an existing infrastructure with human drivers is challenging. It would be simpler to introduce these vehicles without having to consider the driving behavior of other traffic participants. Even though the automation functions are working properly and result in a normative driving behavior, other traffic participants can cause a critical scenario for the automated vehicle. Figure 1 shows such a critical scenario caused by the complex interactions of human drivers.

The initial situation on a German "Autobahn" includes three involved vehicles where vehicle 1 is a cooperative and automated vehicle and the other participants are controlled by humans. The driver in vehicle 2 shows an aggressive behavior, while the driver of vehicle 3 drives defensively. In the beginning, the automation functions of vehicle 1 spot an ideal situation to perform a lane change. While this lane change is performed, vehicle 2 disregards the safety gap and tailgates vehicle 1. At the same time, vehicle 3 is driving extremely careful, which can result in a very serious situation for the automated vehicle shown in Figure 1.

Regardless of who has caused this critical scenario, the vehicle automation has to be capable to handle this scenario properly. Thus, these scenarios are critical and have to be tested. Keeping this in mind, the identification of every possible critical scenario is quite difficult. Even with the use of existing methods, for example, developing a scenario catalogue, gathering expert opinions, or the investigation of existing data, critical scenarios may be missed [1]. Additionally, malfunctions in the automated vehicle can cause critical

FIGURE 1 Critical scenario for automated vehicle V_1, caused by traffic participants due to violation of the required minimum gap. Initial state shown at the top segment, final state shown in the bottom segment.



scenarios. For example, errors can occur in the sensor perception, actuator faults, wrong data in precision maps, etc. The specific influences of these errors have to be investigated, evaluated, and tested.

The main contribution of this article is the composition of a generic simulation-based toolchain for the identification of critical scenarios. This toolchain provides the possibility to use exchangeable automated driving functions, evaluation metrics, and parameter spaces suitable for the intended identification process. To show the capability of the toolchain, a cooperative and automated vehicle, called digital prototype, is embedded in a coupled simulation environment. The performed simulation runs, including the digital prototype, are evaluated by suitable metrics. This approach enables the test and improvement of the automated driving functions depending on requirements. After having reached a mature development stage, such a digital prototype is called a "digital twin" of the cooperative and automated vehicle. In order to provide an overview of the toolchain's capabilities, an exemplary automated driving function on a highway entrance ramp is tested. To identify critical scenarios based on an exemplary metrics set, the requirements presented here are twofold: they are either safety or traffic related. Note, the simulation-based toolchain is generic, meaning that the driving functions, metrics, etc. are exchangeable.

Currently, only a few partial solutions for this formidable challenge exist, mainly provided by academia as well as industrial activities. Insightful work on testing and validating cooperative and automated vehicles in general can be found at [2, 3, 4]. To get a better understanding of scenario-based approaches, we recommend [5, 6]. Recently, new players like Apollo [7] and Nvidia [8] are entering the field with the objective to provide comprehensive simulation platforms in the near future.

The methodology presented here shows a generic methodology to address this overall toolchain approach in a systematic way. Additionally, the proposed toolchain does not depend on a specific simulation tool, vehicle automation function, metrics, spatial application, test method, or development status. All of these aspects can be organized by a modular approach to the scenario identification process.

Simulation-Based Toolchain

The major challenge of releasing cooperative and automated vehicles is the vast driving distance that has to be tested in real traffic as mentioned in [9]. To estimate the effort of a validation process, assessing accident statistics seems reasonable. Nevertheless, the derivation of a general distance based on these assumptions is questionable. In this consideration, neither location nor variation of the driven route is included. If a test is conducted always on the same route, it can easily be argued that the distance of the tests is not a sufficient indicator for the validation of cooperative and automated vehicles.

Further, the identification of critical scenarios is a key factor in the validation of these vehicles. Critical scenarios are defined as scenarios that need to be tested, regardless of whether the requirements are functional or nonfunctional. Aspects like traffic efficiency, driver comfort, etc. are not considered in the estimation of the validation effort so far [10]. Thus, important questions for the release of these vehicles have to be raised:

- Which scenarios have to be tested with respect to the vehicle development process?
- What are the specific functional and nonfunctional requirements for the evaluation?
- Which test should be carried out in what test environment?
- What are the general advantages and constraints of a specific test environment?

Therefore, a generic simulation-based toolchain to address these questions is shown in Figure 2. This toolchain is a significant advancement of our rough concept presented at [1] and allows us to start with a logical scenario [11], which is a scenario description based on parameter spaces, defining a domain that confines possible scenarios.

Table 1 shows an example of a logical scenario and possible parameter spaces.

FIGURE 2 Simulation-based toolchain for the verification and identification of critical scenarios for cooperative and automated vehicles. Automated segments of the toolchain are surrounded by straight lines. Toolchain segments where human intervention is needed are surrounded by dash-dotted lines.

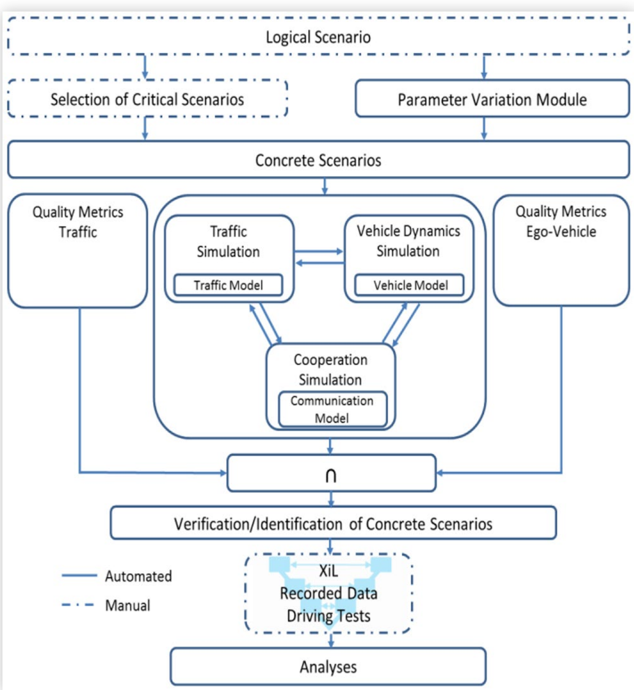


TABLE 1 Logical scenario with parameter spaces (e.g., entering the highway).

Attribute	Parameter space	Unit
Entrance ramp length	l_{min} - l_{max}	m
Number of lanes	N_{min} - N_{max}	n.a.
Speed limit highway	v_{min} - v_{max}	m/s
Traffic flow	Q_{min} - Q_{max}	veh/s
Driver behavior	Defensive - Aggressive	n.a.
Curve radius	r_{min} - r_{max}	m
Coefficient of friction	μ_{min} - μ_{max}	n.a.

Within this parameter space, there are many concrete scenarios [11] that are determined by the parameter space of the logical scenario. The selection of these parameters can be done as shown in the left part of the toolchain by determining critical scenarios using expert opinions and peer review, recorded data, scenario catalogues, etc. The major drawback of these methods is that they neglect critical scenarios. Hence, the toolchain possesses another path with a parameter variation module. In this article, we will mainly focus on the path on the right-hand side of the toolchain. The parameter variation module creates concrete scenarios by changing the parameters of the logical scenario. Parameter variation is achieved by automatically changing the parameters of the logical scenario with a certain bandwidth. This is a straightforward approach to identify critical scenarios over the entire parameter range. A systematic way to achieve more efficiency while performing this task is part of our current research. The major challenge using these approaches is that the critical scenarios and their specific nature are unknown before applying the toolchain, and therefore, it is a difficult task to determine which parameter combination has to be tested more carefully and comprehensively than others.

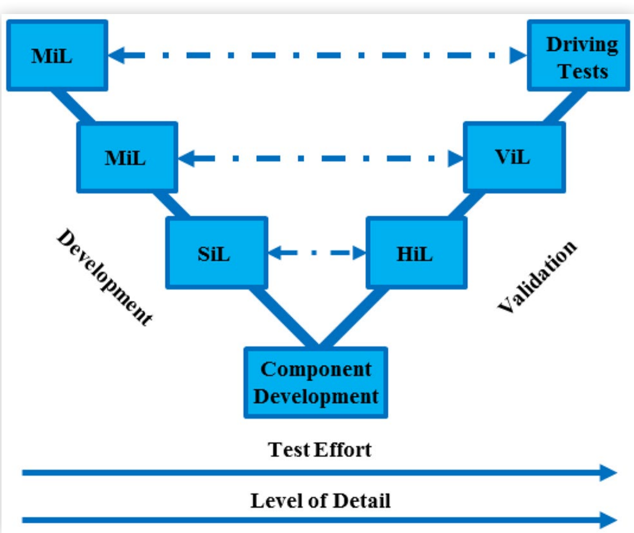
The resulting concrete scenarios are used as an input for the simulation environment. This environment consists of a coupled traffic simulation, a vehicle dynamics simulation, and a cooperation simulation. Each simulation environment has particular advantages and a specific purpose for this method. The traffic simulation provides the surrounding environment for the automated vehicle. The vehicle dynamics simulation contains a detailed model of the vehicle and includes the automation functions that have to be tested. In order to capture the cooperative aspects of these vehicles, the environment provides a cooperation simulation in which cooperative aspects and communication models can be included. An example overview of some state-of-the-art simulation tools compatible with the presented toolchain comprises among others Apollo Simulation, DLR-Simulation of Urban Mobility, IPG CarMaker, Nvidia Drive Constellation, TASS International PreScan, and Vires VTD.

The automated classification of these concrete scenarios into critical or not critical is done with the help of tailored metrics. Metrics are used to evaluate the quality of certain aspects. The toolchain can be used with various types of metrics depending on the domain of interest. In this toolchain, two possibilities of metrics are shown. It is possible to exchange metrics or extend this approach with other metrics depending on the functional and nonfunctional requirements. The evaluation of the simulation using metrics results in a verification or identification of scenarios that are critical, depending on the previous chosen path of the toolchain, and have to be investigated further.

The sole use of a Model-in-the-Loop approach is only sufficient for the development part of the V-Model [12]. The use of a test method is always derived from requirements and the quality of the results. Validation engineers on the right side of the V-Model need far more resilient results than development engineers on the left side designing a basic control concept. Fortunately, the level of detail for simulation models, benches, and driving tests increases along the development process. It is not possible to use a vehicle model with an exact parameter set in early stages of the development process, because the vehicle has not been produced in this phase, components are not yet built, and even the design of the development vehicle is not finished. Nevertheless, development engineers are still able to use vehicle dynamics to develop, for example, a control concept. These models will differ from the actual vehicle model in a later process phase. But the basic concepts developed in an earlier phase can be adapted to changes in the model occurring in subsequent development. Figure 3 shows a V-Model and the resulting test methods.

On the left side of the V-Model, basically only Model- and Software-in-the-Loop tests are possible. As already mentioned, the vehicle components are not yet built. After the component development, it is possible to use more complex models with detailed knowledge about model parameters and component setups. On the right side of the V-Model, benches and driving tests can be applied. With growing knowledge about the vehicle and the use of real components as well as test benches, the level of detail increases, meaning that the validity of the

FIGURE 3 Test methods with respect to the V-Model, see also [12]. MiL = Model-in-the-Loop, SiL = Software-in-the-Loop, HiL = Hardware-in-the-Loop, ViL = Vehicle-in-the-Loop.



tests rises. The drawback of this aspect is that the test effort increases simultaneously. The term “test effort” is mainly determined by costs for simulation, operating of benches, and performing driving tests.

There are various constraints for possible test methods. In addition to the test effort, which is mostly an economic constraint, there are a lot of different technical properties to consider. The limits of each test environment have to be investigated separately. The performance of a simulation strongly depends on the complexity of the used mathematical models. The more detailed the model quality, the more processing power is needed. In addition, the use of mathematical models always has the drawback that these models contain a certain inaccuracy. Further, it is more difficult to perform parameter identification and assess the robustness against parameter changes. Bearing that in mind, driving tests are always valid, and due to the absence of mathematical models, there are no inaccuracies in the test results. Driving tests have drawbacks, besides their economical infeasibility, too. It is not possible to run driving tests faster than real time and some scenarios are too dangerous to be performed safely, when these borderline scenarios are particularly important for validation.

Benches serve as a compromise between simulation and driving tests. Hardware-in-the-Loop methods require the dismantling of vehicle parts. The part of interest is separated from the vehicle and mounted on a bench, which can be controlled by a simulation interface. For that matter, it is possible to simulate a scenario and simultaneously investigate the effects on the hardware. An extension of this is the Car-in-the-Loop method. The benefit of this method is that the actual test vehicle can be mounted on a bench. Dismantling is, therefore, not necessary. This is the closest state before having to perform driving tests. Driving tests considered in the toolchain are not constrained to a certain domain. These tests can be performed on proving grounds and in real traffic. As some tests cannot be performed in real traffic, because this would endanger other traffic participants, proving grounds are therefore a valuable alternative. The benefit of testing in real traffic is the possibility to sample data that shows the realistic behavior of the surrounding world of the vehicle, including other traffic participants, infrastructure, sensor characteristics, and the impact of disturbances due to, for example, map errors or sensor shortcomings.

Recorded data can be obtained from various sources. Depending on the domain of interest, data can be gathered from driving tests, accident studies, traffic observation systems, etc. The use of this data is a good way to compare simulation and driving tests with the existing data. The major drawback can be the incompleteness of the data sets. For example, traffic observation systems are mostly based on camera systems. Therefore, the states of the vehicles are estimated during the process. If recorded data is used, it will be necessary to take a closer look at how the data was acquired.

All of these aspects have an influence on the last part of the toolchain. The analysis is the feedback to the function development and consists of a specific description of the performed tests. Depending on the domain of interest, the analysis looks different. The function development department can choose the quality criteria. The toolchain is a generic setup, and therefore, it is possible to change scenarios, parameters, simulation models, metrics, test methods, and analyses. With that in mind, the toolchain supports a generic way to identify scenarios for cooperative and automated vehicles independent from used vehicles, sensor setups, implemented functions, locations, criticality criteria, test methods, required analyses, etc.

In this article, we will demonstrate the toolchain using the example of automated highway chauffeur [13]. However, it is important to mention that the toolchain can be extended to other functionalities and domains, such as rural roads and, most importantly, urban areas.

Simulation Environment

The basic requirement for the application of a coupled simulation is that the virtual environments are comparable. These environments do not necessarily need to be identical, while the properties depend on the level of abstraction and the objective of the particular simulation. For example, the vehicle dynamics simulation requires more detailed information about the environment than the traffic simulation. Nevertheless, the basic geometrical information has to be similar enough so that the dynamic coupling works in both simulation tools properly. The static coupling is based on a measured data set, provided in the OpenDrive format [14]. For the functionality of the toolchain's simulation environment, the map data has to be converted for each simulation tool. The map formats of the simulation tools differ quite strongly. A scheme of the static coupling and the map data conversion is shown in Figure 4.

The map data is acquired through high-accuracy measurements of the selected road. The Opel proving ground near Frankfurt (Germany) is chosen in this article, and the data is stored in an XML document following the OpenDrive standard. The map data converter shown in Figure 4 is a data

FIGURE 4 Static coupling and map data conversion for the simulation environment.

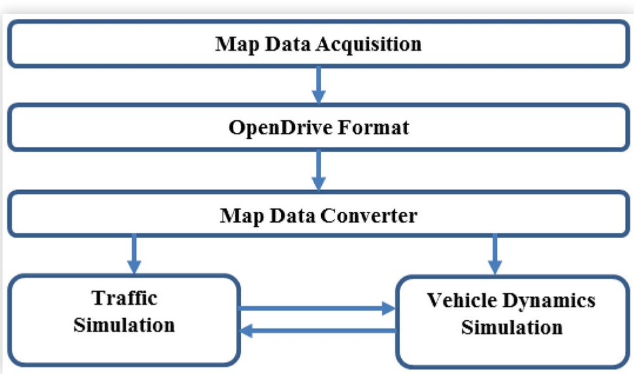
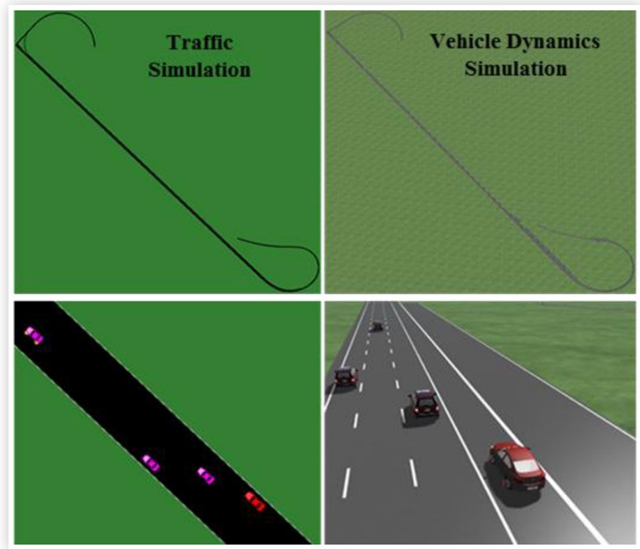


FIGURE 5 Proving ground representation in different simulation environments. Left-hand side: Traffic simulation. Right-hand side: Vehicle dynamics simulation. Top segments: Proving ground overview. Bottom segments: Ego-vehicle surrounded by traffic.

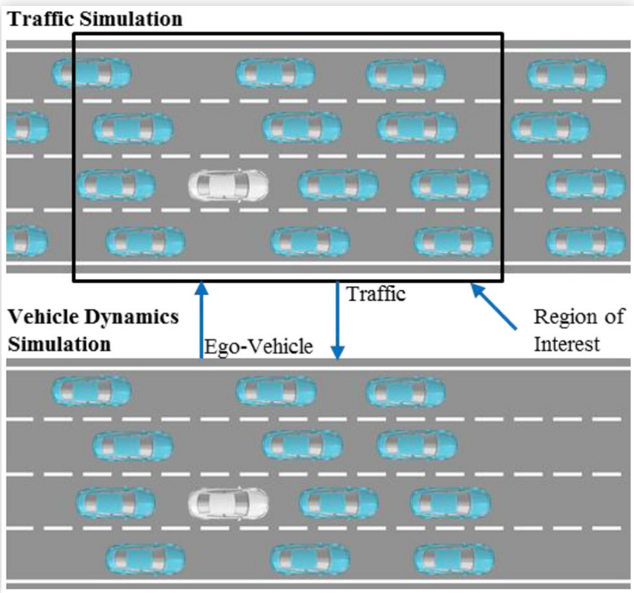


conversion tool parsing the data from OpenDrive format to the specific formats needed in the particular simulation environment. As already indicated, the static environment of each simulation tool has different requirements with respect to accuracy of the converted data. The geometrically consistent virtual environment, with respect to the required accuracy, is the foundation for the coupled simulations used in the toolchain. Further, the consistency of the virtual environment is vital for the implementation of the dynamic coupling. Figure 5 shows the proving ground in both traffic and vehicle dynamics simulation environments.

The dynamic coupling in this article focuses mainly on the traffic and vehicle dynamics simulation. The objective of the simulation framework is to include a cooperative and automated vehicle into a traffic simulation environment. Therefore, traffic participants, for example, passenger cars, busses, pedestrians, etc., are controlled by the traffic simulation, and the virtual automated vehicle comprises driving functions of an existing automated vehicle and can be considered as a digital twin of the real vehicle. For example, vehicle model, sensor setup, and parameters are virtual copies of a real test vehicle. In addition, the use of the same driving functions ensures a similar behavior of the digital twin and the existing vehicle. The perception capabilities of the vehicle depend on the sensor setup. Due to the fact that this setup can change, we decided to introduce an adjustable region of interest, as shown in Figure 6.

The traffic simulation provides the behavior of traffic participants, which is dynamically included in the vehicle dynamics simulation. Simultaneously, the vehicle dynamics simulation provides the behavior of the cooperative and

FIGURE 6 Dynamic coupling with an adjustable region of interest.

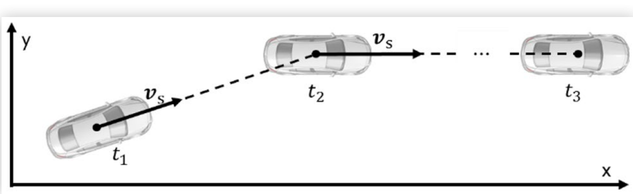


automated vehicle including driving functions, sensor setup, etc. The state of the automated vehicle (ego-vehicle) is used by the traffic simulation to determine the behavior of the surrounding traffic participants. The dynamic coupling ensures that the cooperative and automated vehicle is able to drive in a surrounding traffic environment, which provides a dynamic traffic environment that is able to react according to the driving behavior of the vehicle's driving functions. Each traffic participant possesses its own adjustable driver model.

One of the major issues for the dynamic coupling between traffic and vehicle dynamics simulation is the discrepancy of sample time. The simulation of the vehicle dynamics requires a high sample rate ($T_{S,\,Veh-Dyn} = 0.001$ s) for an accurate calculation of the vehicle behavior, while the traffic simulation works sufficiently with a lower sample rate ($T_{S,Traffic} = 0.1$ s) for traffic evaluation. Therefore, the absent motion of the traffic participants provided by the traffic simulation is predicted with a constant-velocity model [15]. The motion prediction is shown in Figure 7.

This prediction is necessary to avoid large step changes in the sensor perception of the cooperative and automated

FIGURE 7 Motion prediction with a constant-velocity model, required due to the discrepancy of sample times between the simulation environments.



vehicle. Simulation results state that the gap of the simulation environment's sample times is sufficiently small so that the results with enhanced motion prediction do not influence the sensor perception significantly. Besides, this behavior is not unusual for environmental modeling. The use of motion models for the time frame of sensor fusion itself, which generates a new perception for the surrounding environment, is conventional. The interaction between sensor models and virtual environment of the automated vehicle is performed in the vehicle dynamics simulation.

The enhancement of a cooperation simulation is not elaborated in this article. It is worth mentioning that the toolchain is expandable with respect to cooperation simulations including V2V, V2X, etc. Currently, the simulation environment is being enhanced in our group to simulate more than one cooperative and automated vehicle simultaneously. This provides the possibility to evaluate features like cooperative merging, traffic distribution, overtaking, etc.

To complete the simulation environment, functions of the digital twin have to be embedded. It is important that the vehicle model represents the test vehicle sufficiently. If this is the case, the driving functions, such as driving strategy, trajectory planning, vehicle control, maneuver intention prediction, and localization, can be implemented. All of the aspects mentioned above including driving functions complete the simulation environment of the toolchain. The simulation framework builds the foundation of the following investigations and evaluation results presented in this article.

Metrics

In order to identify critical scenarios with a simulation-based toolchain, the term criticality needs to be specified. It is obvious that the understanding of criticality can vary significantly, depending on the specific requirements. The generic design of the toolchain allows the use of different terms of criticality. Usually, criticality is determined by the application of metrics. The best-known criticality metric is called "time to collision" [16]. In the following, a short introduction to some standard safety-related metrics is given, before an approach to developing new metrics will be shown. These new metrics are designed to identify critical scenarios for the cooperative and automated vehicle (ego-vehicle) regarding traffic quality. As already mentioned, the most commonly used safety-related metric is called "time to collision" [16]:

$$TTC = \frac{\Delta p}{v_{ego} - v_{obj}} = \frac{\Delta p}{v_{rel}} \qquad \text{Eq. (1)}$$

where Δp is the difference of the vehicle positions, v_{ego} the ego-vehicle velocity, v_{obj} the object velocity, and v_{rel} the relative velocity between both vehicles. TTC is defined as the time until a collision between the ego-vehicle and an object would occur, if the velocity of both does not change with respect to the point of time when the TTC is calculated [17].

Another standard metric is called "time to brake," which can be defined as [16]

$$TTB = \frac{\Delta p + \frac{v_{rel}^2}{2a_{ego,max}}}{v_{rel}} = TTC + \frac{v_{rel}}{2a_{ego,max}} \quad \text{Eq. (2)}$$

where $a_{ego,max}$ denotes the maximum deceleration the ego-vehicle is able to execute. This safety-related metric is defined as the time span until a collision with 0 m/s is unavoidable, depending on the maximum deceleration ability of the ego-vehicle [18]. The last safety-related metric introduced in this article is called required deceleration and describes the deceleration of the ego-vehicle needed to generate a collision with 0 m/s [19]. This metric can be stated as [16]

$$a_{req} = a_{obj} - \frac{v_{rel}^2}{2\Delta p} \quad \text{Eq. (3)}$$

where a_{obj} is the acceleration of the object.

The introduced toolchain allows for using different metrics depending on the specific requirements. This possibility will be demonstrated in this article. In order to improve the performance and robustness of the identified scenarios, it is convenient to use different metrics simultaneously. Furthermore, the implementation of different metrics allows for collecting more information about an investigated scenario. The introduced standard metrics have one major drawback: they are only defined for the lane the ego-vehicle is driving in. That is not sufficient for automated vehicles, because the relevant traffic participants can enter the ego-vehicle's lane. Therefore, a maneuver intention prediction is used to identify relevant traffic participants, which are performing a lane change toward the ego-vehicle's lane. The concept of using machine learning algorithms and training data to extend the usability of criticality metrics, which are only defined for a single lane, is part of our current research, but will not be elaborated in detail in this article. For further information about maneuver intention prediction and interaction modeling, we recommend [20, 21].

For the investigation of cooperative and automated vehicles and their impact on traffic quality, the use of a metric combination is proposed. This approach aims at collecting more information for the criticality evaluation of a scenario. The benefit of this methodology is that diverse aspects can be analyzed simultaneously, which increases the robustness and validity of the results. The methodology is based on already known traffic quality metrics that are adapted for this application. General requirements for the metrics can be stated as follows:

- Every critical scenario should be identified.
- The "false-positive rate" (FPR) should be low.
- A grading system should be used for the assessment.
- There should be a threshold allowing a binary classification for the combined metrics.

Usually, the investigated time interval for traffic quality varies from several minutes to hours. For our purpose, this time interval is too large and is adjusted to 15 s. The reason for this is that a longer interval would not capture short-term impacts of the automated vehicle, and the assessment whether the ego-vehicle is responsible for the critical scenario or not is difficult to make. The spatial "domain of interest" (DOI) is chosen to be 450 m following the suggestions in the highway capacity manual [22]. To achieve an additional indication about surrounding influences of the vehicle, a moving DOI is introduced. This DOI follows the ego-vehicle and takes the direct surroundings into account. A circle with an adjustable radius moves with the ego-vehicle and considers all traffic participants included in this area. Lastly, there will be an additional moving DOI, which just takes the ego-vehicle into account. Figure 8 shows the overall concept.

The first traffic quality sub-metric is a macroscopic description. The concept uses a fixed DOI and is based on the highway capacity manual [22] and can be calculated as

$$D = \frac{v_P}{S} \quad \text{Eq. (4)}$$

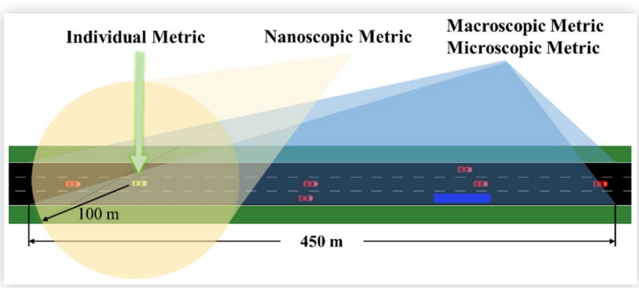
where D is the traffic density, v_P the traffic flow rate, and S the average travel velocity. The traffic density can be compared to an evaluation table grading the traffic quality from A to F, meaning that A is the best grade and F the worst. This metric includes some empirical correction factors used such as peak-per-hour factor, driver-population factor, and a heavy-vehicle-adjustment factor [22].

The second sub-metric used is a microscopic metric introduced by Zhu Weiha et al. [23], considering the velocity deviation and the average velocity in a fixed DOI. The concept aims to interpret the velocity deviation divided by the velocity mean value of the ego-vehicle as an indication of the microscopic traffic quality. The focus on the overall traffic quality requires the enhancement from one vehicle to the consideration of every vehicle, referenced by index j, in the specified DOI and can be stated as [23]

$$CV_j = \frac{\sigma_{vj}}{\overline{v}_j} \quad \text{Eq. (5)}$$

where σ_{vj} is the standard velocity deviation and $\overline{v}_j$ the mean velocity of every vehicle, respectively. The resulting mean coefficient of variation CV is calculated by the mean values of Equation 5. The microscopic metric gives further

FIGURE 8 Domains of interest for different evaluation metrics.



© SAE International

information about the traffic conditions, but still more information is required for the traffic quality evaluation. Therefore, a circular DOI is attached to the ego-vehicle and travels with the ego-vehicle's position. The first moving DOI is a circle surrounding the ego-vehicle that allows for investigating close-range interactions. This metric is called nanoscopic and the calculation is based on velocity deviation and mean value with respect to the DOI and can be written as

$$DV_j = \frac{\sigma_{\text{vCircle,j}}}{\bar{v}_{\text{Circle,j}}} \quad \text{Eq. (6)}$$

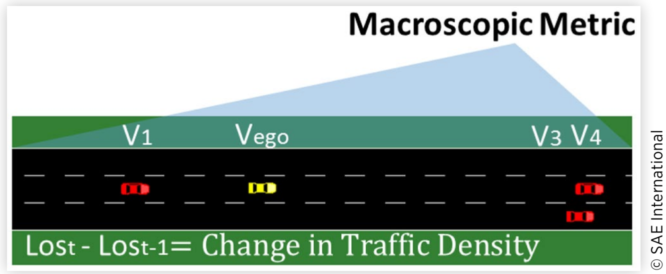
where $\sigma_{\text{vCircle,j}}$ is the velocity standard deviation and $\bar{v}_{\text{Circle,j}}$ is the mean velocity with respect to the DOI. By using the mean value $\overline{DV}$, including every vehicle driving inside the DOI, the overall traffic quality can be evaluated again. The last used sub-metric is called individual metric [24] considering just the ego-vehicle's quantities. The DOI surrounds the ego-vehicle only, considering solely the vehicle's behavior to gather further information for the evaluation process.

The decision whether a scenario is critical or not and if it should be investigated further is binary. It is desirable to have one specific threshold for the differentiation of critical and uncritical scenarios, even though many indicators influence the decision. The individual grades can be used later on to achieve a better understanding of why the scenario is classified as critical. To achieve an overall grading system, it is obvious that every individual metric should have the same specified range. Therefore, every grade will be normalized so that the mathematical set ranges from zero to one, where zero is defined as the best grade and one the worst. The normalized grading system can be stated as follows:

$$\begin{aligned}
G_{\text{mac}} &= \frac{LOS_t - LOS_{t-1}}{5} \\
G_{\text{mic}} &= \frac{\left[\frac{\overline{CV}}{CV_{\text{ref}}} + \left(1 - \frac{\bar{v}}{\bar{v}_{\text{ref}}}\right)\right]}{2} \\
G_{\text{nan}} &= \frac{\left[\frac{\overline{DV}}{DV_{\text{ref}}} + \left(1 - \frac{\bar{v}}{\bar{v}_{\text{ref}}}\right)\right]}{2} \\
G_{\text{ind}} &= \frac{\left[\frac{\sigma_a}{\sigma_{a,\text{ref}}} + \left(1 - \frac{\bar{v}_{\text{ego}}}{\bar{v}_{\text{ref}}}\right)\right]}{2}
\end{aligned} \quad \text{Eq. (7)}$$

The macroscopic grade for our purpose is determined by the change of the traffic quality. This ensures that only negative changes weigh into the overall grade. The denominator of the macroscopic grade is determined by the maximum change of traffic quality between two time intervals from grade A to F according to the highway capacity manual [22]. The other grades are equipped with reference values that represent good traffic quality, which ensures that the metrics are adjustable to the specific situation. The determination of the reference values is based on the observation of representative traffic scenarios.

FIGURE 9 Macroscopic metric including the DOI and traffic quality indicators.



Each individual indicator is observed over the entire range for representative scenarios and adjusted to a subjective evaluation by experts. Attributes with the subscript "ref" are adjustable to different DOIs, for example, rural roads and urban areas.

The set of equations in Equation 7 can be interpreted as a grading system to evaluate different aspects of traffic quality. The macroscopic grade represents the change of traffic density between two time intervals. Therefore, negative changes in traffic density caused by the ego-vehicle result in a critical classification of the scenario. Figure 9 shows the domain of interest and the traffic quality indicator in detail.

The microscopic grade is a trade-off between the coefficient of variation and the mean velocity of a time interval. This trade-off is used to achieve a better understanding of the overall traffic quality. Strong velocity fluctuations and a low average velocity result in a critical scenario, while small fluctuations and high average velocity can be classified as an optimal traffic condition. Figure 10 shows the microscopic metric.

Similarly, the nanoscopic metric and individual grade use the average velocity as an additional indicator. The coefficient of variation and average velocity for the nanoscopic metric are calculated only for the vehicles inside the moving circle, to consider the traffic participants with close-range interactions. Figure 11 represents the concept of the moving DOI to consider close-range interactions between traffic participants inside the moving circle.

The individual metric is calculated by the standard deviation of acceleration and the average velocity of the ego-vehicle itself. Similar to the other metrics, in this case, strong acceleration changes and low average velocity result in critical

FIGURE 10 Microscopic metric including the DOI and traffic quality indicators.

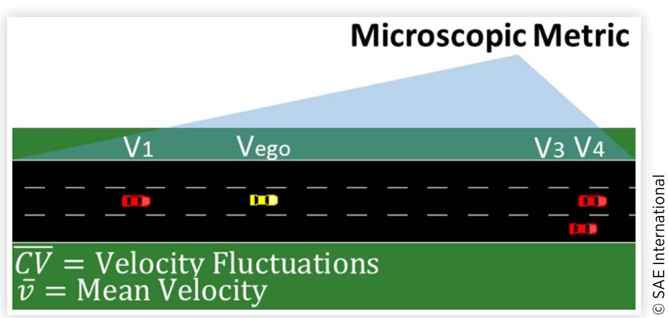
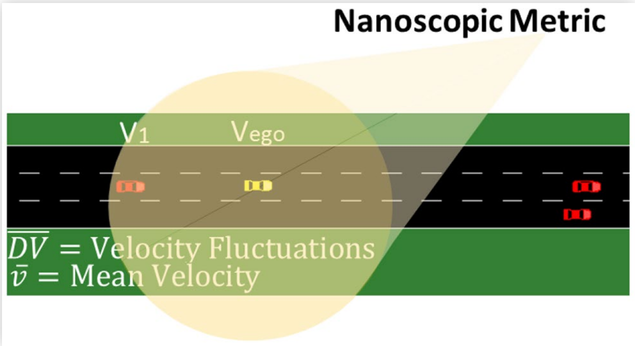


FIGURE 11 Nanoscopic metric including the DOI and traffic quality indicators.



classification regarding traffic quality. The concept of the individual metrics is shown in Figure 12.

The simplest form of an overall grading would be to weigh every single metric equally as shown in Equation 8:

$$G_{\text{final}} = \frac{G_{\text{mac}} + G_{\text{mic}} + G_{\text{nan}} + G_{\text{ind}}}{4} \qquad \text{Eq. (8)}$$

It is obvious that the metrics differ in their sensitivity. Therefore, an optimization of the weighting coefficients is done. The goal is to increase the robustness of the overall evaluation based on training data. The final grade can be rewritten in parameter form

$$G_{\text{final}} = \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4 \qquad \text{Eq. (9)}$$

where $x_1 \dots x_4$ are the individual grades for every metric and $\beta_1 \dots \beta_4$ are the corresponding weighting factors. Figure 13 shows the optimization scheme.

The training data is produced with the coupled simulation environment and evaluated and graded by expert opinions. The quadratic error of the simulation data and the evaluation grade is minimized. The optimization problem can be stated as follows:

$$\min_{\beta} \left\| \boldsymbol{X}\boldsymbol{\beta} - \boldsymbol{y} \right\|_2^2$$

$$\text{s.t.} \quad \boldsymbol{b}_{\text{l}} \le \boldsymbol{X}\boldsymbol{\beta} \le \boldsymbol{b}_{\text{u}}, \sum_{i=1}^{4} \beta_i = 1, \quad 0 \le \beta_i \le 1, \; i = 1, \dots, 4 \qquad \text{Eq. (10)}$$

FIGURE 12 Individual metric with traffic quality indicators.

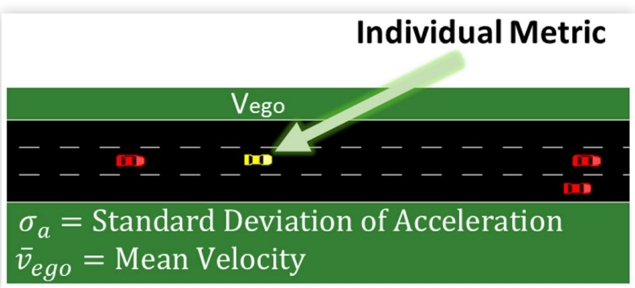
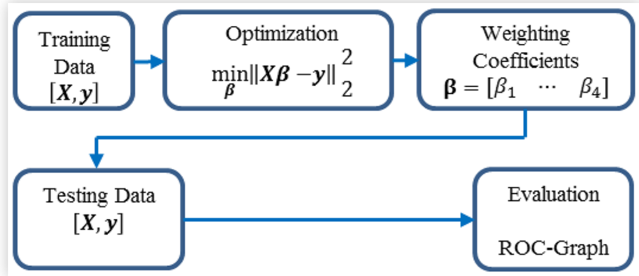


FIGURE 13 Weighting factors optimization and metrics performance evaluation.



where $\boldsymbol{X}$ is a matrix containing the individual grades, $\boldsymbol{\beta}$ a parameter vector including the weighting coefficients, and $\boldsymbol{y}$ a vector with the grades based on expert opinions and constraint vectors $\boldsymbol{b}_{\text{l}} = [0 \;\cdots\; 0]^{\text{T}}$, $\boldsymbol{b}_{\text{u}} = [1 \;\cdots\; 1]^{\text{T}}$. The optimization is based on data from 836 scenarios in a training data set.

This training data set consists of representative critical and uncritical traffic scenarios created by a traffic simulation tool. The traffic density in the training data varies over a bandwidth from low to high. For example, scenarios with traffic congestions, the need for full stops on the highway, strong decelerations, and velocity fluctuations were generated. This data represents a set of possible traffic conditions and is used to evaluate the introduced traffic quality metrics.

The binary threshold is determined by the so-called "receiver operating characteristic" (ROC) graphs and the related confusion matrix shown in Figure 14 [25].

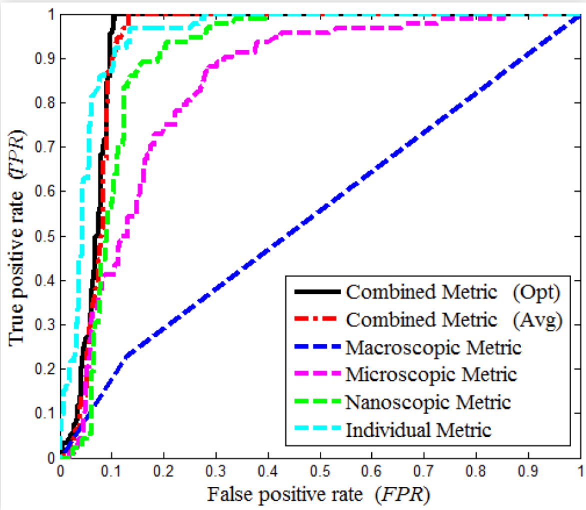
It is desirable to have a "true-positive rate" (TPR) of 100% while keeping the false-positive rate low. Figure 15 shows the results of the optimization in an ROC graph.

It can be seen that the combined and optimized metric (Figure 15, black solid line) shows the best results. The FPR is around 10%, while the requirement of a 100% TPR is fulfilled. Considering a scenario where the ego-vehicle follows a leading vehicle on the same lane and the leading vehicle performs a

FIGURE 14 Confusion matrix for the binary classification of critical scenarios.

	Actual Critical	Actual Not Critical
Predicted as Critical	**True Positive TP**	**False Positive FP**
Predicted as Not Critical	**False Negative FN**	**True Negative TN**

FIGURE 15 Metrics performance results (training data).



braking maneuver causing the ego-vehicle to also brake, the root for the traffic quality decreasing behavior is triggered in this case by the leading vehicle, and therefore, the ego-vehicle is not responsible for the decrease in traffic quality. The metrics evaluation would still classify this scenario as critical, which results in a false-positive case. To prevent similar false-positive flagging, the FPR can be decreased even further by designing a filter, which minimizes the false-positive cases by considering the velocity behavior of the leading vehicle in the immediate past and the gap changes between the ego and the leading vehicle. In some special cases, the use of a filter can result in an undesired omission of some true-positive cases. The decision if a filter should be used depends on the desired results. For the development phase, it is convenient to neglect some special cases in order to achieve a smaller FPR. In the validation process, every true-positive case has to be captured.

The metrics can now be used to identify critical scenarios with the simulation-based toolchain and provide additional information on how the ego-vehicle automation functions influence traffic quality.

Identification of Critical Scenarios

The investigated automated driving function is an SAE Level 3 highway chauffeur [13, 26]. The highway chauffeur is able to perform standard driving tasks on highways such as entering/leaving the highway, overtaking, etc. The cooperative and automated functions are implemented on an Opel Insignia equipped with a sensor and actuator setup suitable for automated driving tasks. The vehicle under test (VUT) possesses features like robust localization, path planning, maneuver prediction, driving strategy, etc. This VUT is embedded as a digital prototype in the simulation environment.

TABLE 2 Logical scenario: entering the highway.

Attribute	Parameter space	Determined by example
Entrance ramp length	l_{min} - l_{max}	410 m
Number of lanes	N_{min} - N_{max}	4
Speed limit highway entrance	v_{min} - v_{max}	36.1 m/s
Traffic flow	Q_{min} - Q_{max}	1 veh/s

The scenario of entering the highway with different exemplary disturbances is chosen as an example use case. It is worth mentioning that this methodology can also be used for other automated driving functions, use cases, and different spatial domains, for example, rural roads and urban areas.

The first step in the toolchain is the definition of a logical scenario and its corresponding parameter spaces. It is obvious that a lot of different parameter spaces can be listed here. In order to keep a clear view only a part of possible, for this use case necessary, parameter spaces are shown in Table 2.

Most parameters are determined by the static attributes of the highway entrance, which in this case is located at the Opel proving ground. Figure 16 shows the initial condition for the simulation runs investigated in this section.

The traffic flow is set to a value, where a lot of traffic participants take part without causing a congestion before the ego-vehicle enters the highway. The behavior of other traffic participants is set to a normative way in the first three scenarios. In the fourth scenario, this behavior is varied. The cooperation simulation is neglected in this example, and therefore, the traffic and vehicle dynamics simulations are used for evaluation. The disturbances can be included in the logical scenario. Depending on the representation, it is difficult to represent every disturbance in Table 2. For example, it would be possible to vary the driving behavior of other traffic participants stepwise from defensive to aggressive. For a full definition of the scenario nomenclature used in this article, the reader is kindly referred to [11].

The evaluation process will be done by using the already introduced metrics for safety and traffic quality. In this case, the focus will be on the identification and evaluation of critical scenarios, where the corresponding thresholds for the criticality classification are $G_{final} = 0.279$, $TTC = 3.9$ s, $TTB = 3.8$ s, and $a_{req} = -2$ m/s^2 [16]. The verification of critical scenarios is not carried out here, and the example is solely based on a Model-in-the-Loop approach. The enhancement of further X-in-the-Loop, recorded data, and driving tests to the methodology is part of our current research. Based on the previous steps, it is possible to analyze the scenarios and prepare test results for the function development.

FIGURE 16 Initial conditions for the performed simulation runs.

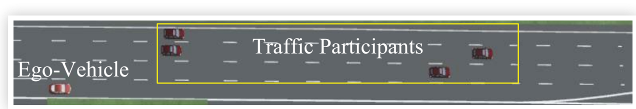


TABLE 3 Concrete scenario: entering the highway without disturbances.

Scenario characteristics	Results
Disturbance	None
Safety metrics	TTC_{crit} = ø, TTB_{crit} = ø, $a_{req,crit}$ = ø
Traffic metrics	G_{final} = 0.15
Criticality	Not critical

The first scenario is carried out with no disturbance at all. The ego-vehicle enters the highway in a normative way. This scenario is chosen to illustrate that the metrics classify a so-called optimal behavior as not critical. The results are shown in Table 3.

As expected, the evaluation results show that the metrics do not deflect (indicated by the empty set ∅) and the corresponding conclusion is a negative criticality classification. In the toolchain, the scenario is not identified as critical and will be neglected. The first run-through the whole toolchain is therefore concluded.

The second concrete scenario remains a highway-entry scenario affected by a sensor error disturbance causing the ego-vehicle's trajectory-following controller to result in fluctuating behavior. To keep the examples sufficiently brief, the same logical scenario is chosen with the same specified attributes. Keeping that in mind, it is possible to carry out the examples directly with the toolchain step "concrete scenario," which is the input for the corresponding simulation environment. Table 4 shows the results of the simulated concrete scenario.

Correctly, the classification result of the scenario is stated as critical. The traffic quality metrics respond to strong fluctuations of the ego-vehicle, and especially the individual metric takes this aspect into account. In this particular simulation run, the safety-related metrics responded as well. In general, this disturbance can lead to safety-critical behavior. Due to usage of the Model-in-the-Loop approach, it is possible to skip the next step and go directly to the analysis step. This critical scenario and the complete simulation data are saved in a database together with a test result description. The function development department can use this simulation data and test result documentation to improve their functionalities.

Cooperative and automated vehicles use precision maps to increase their foresight compared to the operation of their sensor setup only. Precision maps can be used to develop driving strategies. Especially on highway entries, these maps provide helpful information, such as the distance to the end of the lane and the start of the dashed line allowing the vehicle to enter the highway. The third scenario is a consideration of what can happen when the information of the precision map is incorrect. The map thus provokes an error by not allowing our ego-vehicle to enter the highway directly when the dashed line begins, but 40 m later, shortening the possible length of the lane change from 140 m to 100 m. Table 5 contains the outcomes of this scenario.

On account of the map error, the ego-vehicle does not enter the highway and performs a full stop on the entry ramp. The level-3 function demands that the driver takes control while the vehicle blocks the entry ramp. The traffic metrics also correctly classify this scenario as critical. Therefore, the results are saved and passed on to the function development department. The ego-vehicle caused no safety-related problem in this scenario. Therefore, the safety metrics did not deflect, which is also accurate. It is worth mentioning that this behavior has a very negative effect on customer acceptance of these systems. Thus, the reliability of the ego-vehicle functions can be considered as a key requirement for the release.

The fourth scenario highlights the behavior of other traffic participants and their influence on automated driving functions. As already indicated in the introductory example, the driving behavior of other participants can maneuver the ego-vehicle into critical scenarios. Therefore, this driving behavior is changed to aggressive in the traffic simulation. The coupling allows for changing this parameter very easily by adjusting the driver models inside the coupled environment. The results can be seen in Table 6.

It is obvious that the aggressive driving behavior influences the safety-related metrics very strongly. The values of each single metric fall below the critical threshold at a certain time in the simulation. These scenarios are dangerous and have to be tested further. In this simulation run, the traffic quality did not decrease significantly, which correctly led to an uncritical classification result regarding the traffic metrics.

The use of the entire toolchain for different scenarios and the correct classification of the relevance of the scenarios prove

TABLE 5 Concrete scenario: entering the highway with map errors.

Scenario characteristics	Results
Disturbance	Map errors
Safety metrics	TTC_{crit} = ø, TTB_{crit} = ø, $a_{req,crit}$ = ø
Traffic metrics	G_{final} = 0.46
Criticality	Critical

TABLE 4 Concrete scenario: entering the highway with sensor errors.

Scenario characteristics	Results
Disturbance	Sensor errors
Safety metrics	TTC_{crit} = 2.9 s, TTB_{crit} = 1.2 s, $a_{req,crit}$ = −6 m/s^2
Traffic metrics	G_{final} = 0.29
Criticality	Critical

TABLE 6 Concrete scenario: entering the highway with aggressive traffic participants.

Scenario characteristics	Results
Disturbance	Aggressive traffic participants
Safety metrics	TTC_{crit} = 1.9 s, TTB_{crit} = 1.2 s,$a_{req,crit}$ = −5.7 m/s^2
Traffic metrics	G_{final} = 0.19
Criticality	Critical

that this methodology is able to identify critical scenarios. Normative driving behavior is correctly classified as not critical and, therefore, not saved and passed on to the function development department.

Conclusions

In this article, a simulation-based toolchain to identify and verify critical scenarios for cooperative and automated vehicles is introduced. The methodology is illustrated and demonstrated to work with different example scenarios containing sensor errors, map errors, and aggressive driving behavior of other traffic participants. Two different metrics are used for the classification, considering standard safety metrics and newly developed metrics to evaluate traffic quality. Critical scenarios are flagged and the corresponding data is passed on to the function development department as test results in order to improve the implemented automated driving functions.

The results presented in this article show an example for the realization of the generic toolchain and its capabilities. The combination of different metrics to evaluate traffic quality allows us to make a binary classification for the criticality of a concrete scenario. Instead of using every single traffic quality indicator separately, this approach focuses on finding one final grade with a criticality threshold for the overall identification process. Due to that fact, by calculating the final grade and comparing this value with the threshold determined by the optimization scheme using training data, it is possible to decide whether a concrete scenario is critical or not. This approach can be adopted for other metrics as well, for example, driver comfort, fuel efficiency, and driving foresight, just to name a few. Even though the results presented here are examples, the capabilities of the toolchain to identify critical scenarios independent of the implemented driving functions, different simulation tools, the application-specific environment, and the use of various metrics derived from the chosen requirements are demonstrated.

The proposed methodology is limited due to the imperfections of current simulation tools as well as of the simulation models. Especially parameters used for the vehicle dynamics including the sensors are highly sensitive to various real-world conditions, for example, weather, road surface properties, etc. To counter these limitations, we are currently developing a so-called Prototype-in-the-Loop method, where the vehicle dynamics simulation is replaced by a real-world cooperative and automated vehicle. This vehicle is interacting continuously, via both a static and a dynamic coupling, with the traffic simulation while driving on a real-world proving ground.

Regarding future activities, the methodology will be extended with X-in-the-Loop methods, benches, and driving tests to prove the plausibility with a real test vehicle. Such an approach enables a high degree of awareness whether the simulation results are sufficiently robust and provides insight into the interferences and uncertainties of the used mathematical models. The influences of these uncertainties will be investigated further. A proposition can be made about which model precision is required in what part of the V-Model.

Contact Information

Sven Hallerbach
Opel Automobile GmbH, Bahnhofsplatz, D-65423
Ruesselsheim am Main, Germany
Sven.Hallerbach@opel.com

Nomenclature

Symbol/ acronym	Description	Unit
TTC	Time to collision	s
TTB	Time to brake	s
a_{req}	Required deceleration	m/s^2
Δp	Difference of vehicle positions	m
v_{ego}	Ego-vehicle velocity	m/s
v_{obj}	Object velocity	m/s
v_{rel}	Relative velocity between vehicles	m/s
$a_{ego,max}$	Maximum deceleration of the ego-vehicle	m/s^2
a_{obj}	Acceleration of the object	m/s^2
D	Traffic density	veh/m/lane
v_P	Traffic flow rate	veh/s/lane
S	Average travel velocity	m/s
CV_j	Microscopic coefficient of variation	n.a.
σ_{vj}	Microscopic velocity standard deviation	m/s
$\overline{v}_j$	Microscopic mean velocity of every vehicle	m/s
DV_j	Nanoscopic coefficient of variation	n.a.
$\sigma_{vCircle,j}$	Nanoscopic velocity standard deviation	m/s
$\overline{v}_{Circle,j}$	Nanoscopic mean velocity of every vehicle	m/s
LOS_t	Macroscopic grade	n.a.
$CV_{ref}, \overline{v}_{ref}, DV_{ref}, \overline{v}_{ref}, \sigma_{a,ref}$	Reference values for normalization	n.a., m/s, n.a., m/s, m/s^2
$\overline{v}$	Mean velocity of every vehicle with respect to time interval	m/s
$\overline{v}_{ego}$	Average velocity of the ego-vehicle with respect to time interval	m/s

$\boldsymbol{\sigma_a}$	Individual acceleration standard deviation	m/s^2
$\boldsymbol{G_{\text{final}}}$, $\boldsymbol{G_{\text{mac}}}$, $\boldsymbol{G_{\text{mic}}}$ $\boldsymbol{G_{\text{nan}}}$, $\boldsymbol{G_{\text{ind}}}$	Normalized traffic quality grades	n.a.
$\boldsymbol{X}$	Matrix of grades evaluated by metrics	n.a.
$\boldsymbol{\beta}$	Parameter vector	n.a.
$\boldsymbol{y}$	Vector of grades evaluated by experts	n.a.
$\boldsymbol{b}_{\text{l}}$	Vector of lower bound constraints	n.a.
$\boldsymbol{b}_{\text{u}}$	Vector of upper bound constraints	n.a.

References

1. Hallerbach, S., Eberle, U., and Köster, F., "Absicherungs- und Bewertungsmethoden für kooperative und hochautomatisierte Fahrzeuge," *AAET - Automatisiertes und vernetztes Fahren*, Brunswick, Germany, Feb. 8-9, 2017, 368-384, ISBN:978-3-937655-41-3.
2. Koopman, P. and Wagner, M., "Autonomous Vehicle Safety: An Interdisciplinary Challenge," *IEEE Intelligent Transportation Systems Magazine* 9(1):90-96, 2017, doi:10.1109/MITS.2016.2583491.
3. Pütz, A., Zlocki, A., Küfen, J., Bock, J. et al., "Database Approach for the Sign-Off Process of Highly Automated Vehicles," *25th Enhanced Safety of Vehicles Conference (ESV)*, Detroit, 2017.
4. General Motors, "2018 Self-Driving Safety Report," 2018, https://www.gm.com/content/dam/gm/en_us/english/selfdriving/gmsafetyreport.pdf.
5. Ulbrich, S., Menzel, T., Reschka, A., Schuldt, F. et al., "Defining and Substantiating the Terms Scene, Situation, and Scenario for Automated Driving," *IEEE 18th International Conference on Intelligent Transportation Systems*, Canary Islands, Spain, 2015, doi:10.1109/ITSC.2015.164.
6. Menzel, T., Bagschik, G., and Maurer M., "Scenarios for Development, Test and Validation of Automated Vehicles," *2018 IEEE Intelligent Vehicles Symposium*, Changshu, China, 2018, https://arxiv.org/abs/1801.08598.
7. Baidu Artificial Intelligence, "Apollo Simulation," accessed July 27, 2018, http://apollo.auto/platform/simulation.html.
8. Nvidia Cooperation, "Nvidia Drive Constellation," accessed July 27, 2018, https://www.nvidia.com/en-us/self-driving-cars/drive-constellation/.
9. Maurer, M., Gerdes, J.C., Lenz, B., and Winner, H., *Autonomes Fahren - Technische, rechtliche und gesellschaftliche Aspekte - Teil 4 Sicherheit* (Berlin, Germany: Springer Verlag GmbH, 2015), 454-458, doi:10.1007/978-3-662-45854-9.
10. Hallerbach, S., Eberle, U., and Köster, F., "The Challenges of Releasing Cooperative and Highly Automated Vehicles - A Look beyond Functional Requirements," *AmE - Automotive meets Electronics, GMM-Fachbericht, VDE*, Dortmund Germany, Mar. 07-08, 2017, 102-106, ISBN:978-3-8007-4369-8.
11. Bagschik, G., Menzel, T., Reschka, A., and Maurer M, "Szenarien für Entwicklung, Absicherung und Test von automatisierten Fahrzeugen," *11. Workshop Fahrerassistenzsysteme und automatisiertes Fahren, FAS 2017*, Uni-DAS e.V., Walting, Germany, Mar. 29-31, 2017, 125-135, ISBN:978-3-00-055656-2. A related English-language article "Scenarios for Development, Test and Validation of Automated Vehicles" by the Braunschweig Group is to be found at: https://arxiv.org/abs/1801.08598.
12. Winner, H., Hakuli, S., Lotz, F., and Singer, C., *Handbuch der Fahrerassistenzsysteme-Grundlagen, Komponenten und Systeme für aktive Sicherheit und Komfort* Third Edition (Wiesbaden: Springer Fachmedien, 2015), 128-132, doi:10.1007/978-3-658-05734-3.
13. Bartels, A., Eberle, U., and Knapp, A., "Deliverable D2.1. System Classification and Glossary," *Adaptive Consortium*, Wolfsburg, Germany, Feb. 6, 2015, 63.
14. Dupuis, M. et al., "OpenDrive Format Specification, Rev. 1.4," *VIRES-Simulationstechnologie GmbH*, Nov. 4, 2015.
15. Schubert, R., Richter, E., and Gerd, W., "Comparison and Evaluation of Advanced Motion Models for Vehicle Tracking," *11th International Conference on Information*, June 30-July 3, 2008, doi:10.1109/ICIF.2008.4632283.
16. Junietz, P., Schneider, J., and Winner, H., "Metrik zur Bewertung der Kritikalität von Verkehrssituationen und -szenarien," *11. Workshop Fahrerassistenzsysteme und automatisiertes Fahren, FAS 2017*, Uni-DAS e.V., Walting, Germany, Mar. 29-31, 2017, 149-160, ISBN:978-3-00-055656-2.
17. Hayward, J.C., "Near Miss Determination through Use of a Scale of Danger," *51st Annual Meeting of the Highway Research Board*, Washington, DC, Jan. 17-21, 1972.
18. Hillenbrand, J., Kroschel, K., and Schmid, V., "Situation Assessment Algorithm for a Collision Prevention Assistant," in *Proceedings of Intelligent Vehicles Symposium, 2005*, IEEE, Las Vegas, June 6-8, 2005, 459-465, doi:10.1109/IVS.2005.1505146.
19. Karlsson, R., Jansson, J., and Gustafsson F., "Model-Based Statistical Tracking and Decision Making for Collision Avoidance Application," *Proceeding of the 2004 American Control Conference*, Boston, June 30-July 2, 2004, 3435-3440, ISBN:0-7803-8335-4.
20. Augustin, D. and Hallerbach, S., "Interaction-Aware Motion Prediction for Highly-Automated Driving Function on Highways," *AmE - Automotive meets Electronics, GMM-Fachbericht*, VDE, Dortmund Germany, 127-132, Mar. 7-8, 2017, ISBN:978-3-8007-4369-8.
21. Lefèvre, S., Vasquez, D., and Laugier, C., "A Survey on Motion Prediction and Risk Assessment for Intelligent Vehicles," *ROBOMECH Journal* 1(1), 2014, doi:10.1186/s40648-014-0001-z.
22. Transport Research Board- National Research Council, *Highway Capacity Manual* (Washington, DC, 2000). ISBN:0-309-06681-6.

23. Zhu, W., Boriboonsomsin, K., and Barth, M., “Microscopic Traffic Flow Quality of Service from the Drivers’ Point of View,” *Proceedings of the 2007 IEEE, Intelligent Transportation Systems Conference*, Seattle, WA, Sept. 30-Oct. 3, 2007, 47-52, doi:10.1109/ITSC.2007.4357790.
24. Ko, J., Guensler, R., and Hunter, M., “Variability in Traffic Flow Quality Experienced by Drivers: Evidence from Instrumented Vehicles,” Transportation Research Record: Journal of the Transportation Research Board, No. 1988, Transportation Research Board of the National Academies, Washington, DC, 1-9, 2006, doi:10.3141/1988-02.
25. Fawcett, T., “An Introduction to ROC Analysis,” *Pattern Recognition Letters* 27:861-874, Dec. 19, 2005, doi:10.1016/j.patrec.2005.10.010.
26. SAE-International, “Taxonomy and Definitions for Terms Related to On-Road Motor Vehicle Automated Driving Systems,” SAE Standard J3016.201401, Jan. 16, 2014.