Large-Scale Evaluation of an Active Safety Algorithm with EuroNCAP and US NCAP Scenarios in a Virtual Test Environment – An Industrial Case Study

Christian Berger*, Delf Block‡, Christian Hons‡, Stefan Kühnel‡,
André Leschke‡, Dimitri Plotnikov§, and Bernhard Rumpe§

*Department of Computer Science and Engineering, University of Gothenburg, Sweden
‡Volkswagen AG, Development Car Safety, Electronic and Testing, Wolfsburg, Germany
§Software Engineering, RWTH Aachen University, Germany

***Abstract*— *Context:* Recently, test protocols from organizations like European New Car Assessment Programme (EuroNCAP) were extended to also cover active safety systems.**

***Objective:* The official EuroNCAP test protocol for Autonomous Emergency Braking (AEB)/Forward Collision Warning (FCW) systems explicitly defines to what extent a Vehicle-Under-Test (VUT) is allowed to vary in its lateral position. In addition, the United States New Car Assessment Programme (US NCAP) test protocol has broader tolerance ranges. The goal for automotive OEMs is to understand the impact of such allowed variations on a the overall vehicle's performance.**

***Method:* A simulation-based approach is outlined that allows systematic, large-scale analysis of such influences to effectively plan time-consuming and resource-intense real-world vehicle tests. Our models allow a profound analysis of an AEB algorithm by modeling and conducting more than 3,000 simulation runs with EuroNCAP's dynamic CCRm and CCRb scenarios including those with adopted USNCAP parameters.**

***Results:* Our structured analysis of such test procedures involving dynamic actors is the first of its kind in a relevant industrial setting. Several anomalies were unveiled under US NCAP conditions to support real-world test runs. Hence, we could show that the proposed method supports all possible scenarios in AEB consumer tests and scales as we had to timely process approx. 7.7GB of simulation data.**

***Conclusion:* To achieve the expected performance and to study a system's behavior in potential misuse cases from a functional point of view, large scale, model-based simulations complement traditional testing on proving ground.**

I. INTRODUCTION AND MOTIVATION

Consumer tests like EuroNCAP [1] and US NCAP by the National Highway Traffic Safety Administration (NHTSA) [2] influence market decisions as they help to assess the safety of today's passenger cars. Therefore, the goal for automotive Original Equipment Manufacturers (OEMs) is to achieve a high rating for their vehicles. As such consumer tests are designed to reflect representative examples from relevant real world scenarios, they are only limitedly useful to identify and analyze critical boundary cases.

A. Problem Domain and Motivation

To fully understand a vehicle's behavior in the full range of all possible and relevant scenarios would require too many resources and too much time to be conducted on a proving ground. Therefore, it is more efficient to prepare and identify critical boundary cases and relevant test scenarios in a systematic and structured way by using virtual test scenarios and focus on identified anomalies on the real proving ground.

B. Research Goal and Research Questions

In this article, we are presenting an approach to describe relevant or interesting test scenarios in a model-based way to generate several thousand traffic situations for a virtual test environment. Our goal is to model and efficiently evaluate scenarios to identify focus areas for real world test scenarios.

RQ-1: How can variants in the range of more than 1,000 simulation runs be modeled, executed, and evaluated for consumer test scenarios?

RQ-2: What is the influence of velocity variances and different consumer test settings on the trigger time of an active safety system like an AEB algorithm?

C. Contributions of the Article

The contributions of our work comprise (a) an automated environment to execute and evaluate such model-based scenarios, and (b) results from evaluating an AEB system in more than 3,000 variants of EuroNCAP's CCRm and CCRb scenarios and US NCAP's boundary conditions resulting in over 7.7GB simulation data. In this regard, our work is the first of its kind to systematically model and evaluate such consumer tests in a large-scale, industrial setting.

D. Structure of the Article

Sec. II outlines a relevant selection of related work. In Sec. III, the design of the industrial case study is presented to study large-scale simulations of more than 3,000 test cases for EuroNCAP's CCRm and CCRb scenarios and US NCAP settings. The results are described in Sec. IV and the work is concluded in Sec. V.

II. RELATED WORK

In [3], we presented a concept of simulating active safety systems within the related consumer test scenarios to support the development of such systems during the EuroNCAP. We showed that the test method *Equivalence Class Partitioning (ECP)* is insufficient to establish confidence in predicting

[BBH+15a] C. Berger, D. Block, C. Hons, S. Kühnel, A. Leschke, D. Plotnikov, B. Rumpe:
Large-Scale Evaluation of an Active Safety Algorithm with EuroNCAP and US NCAP Scenarios in a Virtual Test Environment – An Industrial Case Study.
In: Conference on Intelligent Transportation Systems (ITSC'15), pp. 2280–2286, IEEE, 2015.
www.se-rwth.de/publications

the possible residual velocity during EuroNCAP's AEB test procedure. In [4] we described a simulation approach that uses a graph-based model to systematically describe and evaluate the influence of EuroNCAP's tolerance ranges on an AEB system. We extended our graph model to simulate a monotonic increase of the VUT's velocity with respect to a previous simulation run in our previous work [5].

The work of Schuldt et al. presents a modular testing toolbox to systematically evaluate virtual test cases for Advanced Driver Assistance System (ADAS). They explicitly focus on reducing the overall test cases to validate vehicle functions under development in a virtual environment. One aspect aims on the intelligent combination of system components like sensors or actuators with sufficient evaluation criteria within specifically generated test scenarios. The approach is demonstrated on the example of a Construction-Zone-Assist which supports a driver by detecting construction-zone-specific obstacles and keeping the vehicle in its lane [6], [7].

In [8] a simulation-driven approach for evaluating ADAS is presented that includes a model-based architecture with environmental and vehicle component models. To validate those systems, Belbachir et al. designed several evaluation and assessment criteria like detection error rate of obstacles like vehicles as well as pedestrians or estimating the level of driver safety.

The 2007 the DARPA Urban Challenge was a large-scale robotic experiment with many self-driving vehicles in an urban-like environment. As today's active safety systems were influenced by results thereof, the need for a systematic and hardware-independent assessment of algorithms for such intelligent vehicles was already identified and conceptually realized during that period [9], [10].

To support the front-loading process for Software-in-the-Loop (SiL)-tests of ADAS, Martinus et al. developed a Virtual Application Platform (VAP) that simulates the underlying hardware. Thus, new software releases could easily be deployed to the environment, which also includes a vehicle dynamics and an environmental model [11].

The virtual testing approach by Nentwig et al. focuses particularly on camera-based ADAS and uses explicitly the real hardware of the supplier within a Hardware-in-the-Loop (HiL)-testbed [12]. For the purpose of conducting functional tests of those video-based systems, the two software tools Virtual Test Drive (VTD) [13] and Automotive Data and Time Triggered Framework (ADTF) [14], [15], [16] are combined to a simulation environment. In addition, von Neumann-Cosel investigated the potential of VTD to support the simulation of any type of ADAS that perceives its environment and extended the capabilities of the VTD towards that objective [17], [18].

In [19], another approach and simulation framework for video-based systems is presented using the tool-set provided by IPG Automotive GmbH to assess synthetic data from a camera and a radar sensor for validating the data fusion algorithms with respect to time constraints. Holzmann et al. implemented another use case for the IPG simulation environment, which focuses on the evaluation of a chassis control system and integrated a detailed vehicle dynamics model for this purpose [20].

The vehicle dynamics model as part of Chucholowski's work was implemented in a real-time numerical simulation environment and focuses on modeling a passenger car being driven through the ISO slalom test scenario [21]. Another tool-set is presented by Tideman et al. called "PreScan" that aims for functionally evaluating a Lane-Keeping Assist (LKA) within manually created test scenarios [22], [23].

The main difference between the outlined approaches and the one presented in this work can be seen in the way large-scale scenarios can be described systematically and the subsequent quantity of the simulation runs to be processed. Thus, our approach is able to handle thousands of generated test cases in which only very few parameters are varied to backtrace the change between the different test cases. The concept and results presented in this article adapt our previous work and applies it to a broader test catalogue in the case of EuroNCAP and includes a new test catalogue in the case of USNCAP.

III. Simulating Consumer Tests

In the following, we describe briefly the test procedures from EuroNCAP and US NCAP for AEB systems. Afterwards, a technical summary of the simulation environment in use at the industrial partner is given.

A. *EuroNCAP's AEB Test Protocol*

EuroNCAP provides and develops the test procedures for AEB and FCW systems [1]. The goal is to systematically and comparably assess safety features between different types of passenger cars.

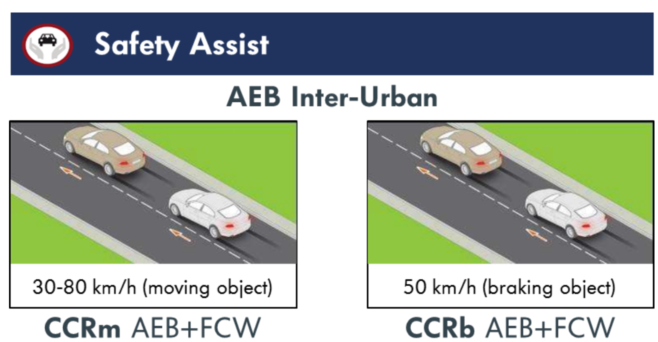

Fig. 1. EuroNCAP's CCRm and CCRb test scenarios assessed as safety assisting systems (based on [24]).

Fig. 1 shows the two main scenarios with moving target objects, which are derived from statistics about car accident data representing the most likely types of crashes in rural and urban areas:

- Car-to-Car-Rear: stationary (CCRs)
- Car-to-Car-Rear: moving (CCRm)
- Car-to-Car-Rear: braking (CCRb)

In [4], we already outlined the main characteristics and underlying tolerance ranges for the CCRs scenario, which is

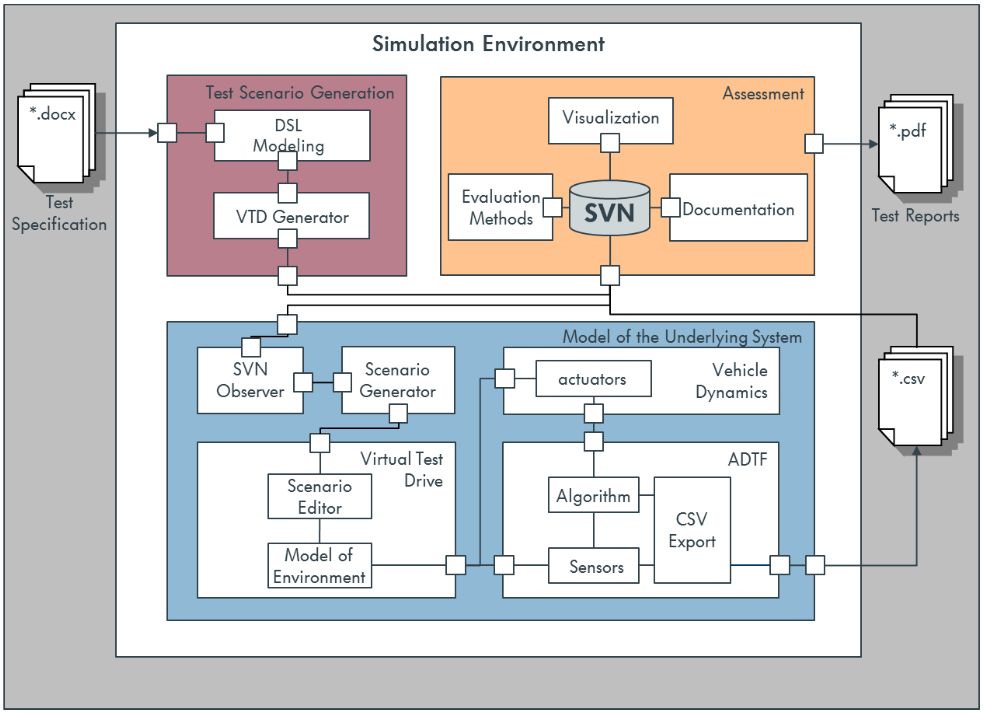

Fig. 2. Simulation Architecture

characterized by a stationary target vehicle being placed in front of the VUT; in the following, we are focusing on the two dynamic scenarios only. The ideal path is a straight line between the center of VUT's front bumper and the center of the target vehicle's rear bumper. The CCRm and CCRb scenarios include a moving target vehicle, which is travelling with either 20 km/h or 50 km/h, respectively. In case of the CCRb scenario, the target vehicle will start to brake with either 2 m/s^2 or 6 m/s^2 while the VUT is travelling with the same speed of 50 km/h in a following distance of either 12 m or 40 m.

Several test parameters can vary in certain ranges in the two EuroNCAP test scenarios. These are as follows:

CCRm:
- velocity of the VUT ($+\ 1.0\ km/h$)
- velocity of target vehicle ($\pm\ 1.0\ km/h$)
- lateral deviation of both vehicles from ideal test path ($0\ \pm\ 0.1\ m$)
- yaw velocity of both vehicles ($0\ \pm\ 1.0\ °/s$)
- steering wheel velocity of both vehicles ($0\ \pm\ 15.0\ °/s$)

CCRb:
- equal to CCRm regarding both vehicles
- the longitudinal distance between both vehicles ($12\ m\ \vee\ 40\ m\ \pm\ 0.5\ m$)

A test case is performed correctly when these parameters are kept within their respective tolerance ranges between the test start (4 s Time-To-Collision (TTC)) and the actual activation of the AEB system. Otherwise, the test case is considered to be incorrect and must be repeated according to EuroNCAP's test protocol.

B. US NCAP's Crash Imminent Braking System Performance Evaluation

In [2] the current working draft of the US NCAP test protocol is proposed assessing Crash Imminent Braking (CIB) systems being a synonym for AEB systems. There are some differences regarding the boundary conditions for a valid test case due to the fact that in the US a human driver is supposed to perform these test trials instead of a driving robot. Thus, the underlying test parameters may vary in a wider range compared to the European counterpart. Considering the moving target scenarios, the main differences are as follows:

- There are less concrete test cases, which means that only six test cases with different velocities of the VUT and the target vehicle will be performed.
- The VUT is travelling towards the nonstationary target with either 25 mph (40.2 km/h) or 45 mph (72.4 km/h).
- Within the scenario of a steadily moving target, the VUT approaches the target with the same two nominal velocities while the target vehicle is travelling with 10 mph (16.1 km/h) or 20 mph (32.2 km/h), respectively.
- The nominal velocities for both vehicles within the scenario with the decelerating target vehicle are either 25 mph (40.2 km/h) or 35 mph (56.3 km/h) and the target vehicle's deceleration rate is ca. 3 m/s^2.
- The maximum yaw rate of both vehicles must not exceed $\pm\ 2.0\ °/s$.

Furthermore, the tolerance ranges also differ slightly from the European counterpart:

- The general velocity of both vehicles is allowed to vary in the range of ± 1.0 mph (1.6 km/h).

- The position in y-direction is allowed to deviate about ± 2 ft (0.6 m) from the ideal path for both vehicles.
- In addition, the relative lateral distance between both moving vehicles is not allowed to deviate more than ± 2 ft (0.6 m).

C. Simulation Environment

Due to the fact, that the final decision regarding the definition of the validity period for each test case has not been made yet, we assume that the same conditions apply as described in the EuroNCAP test protocol.

The simulation technology that is based on VTD [15] and ADTF [14] is provided by the industrial partner. This section outlines the automatization concept that we introduce to scale the simulation approach.

The developed automation concept enables an autonomous execution and subsequent evaluation of modeled test scenarios. It is based on a scenario repository and a repository-observer to detect new scenarios therein. After a new scenario is identified, the observer sends messages to the ADTF and VTD counterparts that invokes the execution of the particular scenario. In order to trace influence of changed parameters on an algorithms's behavior, the repository is implemented using a Subversion system as the backend (SVN) [25]. Thus, simulation results can be directly compared to the previous executions and easily archived.

Our modeling tool for trajectory generation supports consumer test experiments with varying parameter settings such as allowed possible y deviation of VUT, and generates a set of reference trajectories for a particular scenario type, e.g. CCRm or CCRb. A scenario is composed of a trajectory definition for the moving objects and scenario description file defined in a domain specific language instance [26] of the ScenarioDSL. The scenario generator processes a new instance and generates a concrete XML representation based on the scene description for the simulation environment VTD. The ScenarioDSL code generator is based on the MontiCore Language workbench [27], [28], [29].

The simulation runs are recorded as Comma Separated Values (CSV) files and post-processed with Python scripts to analyze the emergency braking times. The corresponding Vehicle-Under-Test configuration serves as input for the analytical residual velocity computation. The scripts are designed in a pipe-and-filter manner that enables exchange of modules, e.g. replacing the braking model. The figure 2 summurizes the proposed architecture.

IV. SYSTEMATIC EVALUATION BY HANDLING A HIGH QUANTITY OF TEST CASES WITH A MOVING TARGET – AN INDUSTRIAL CASE STUDY

Our experimental study is reported according to the guidelines from Jedlitschka et al. [30] and Runeson and Höst [31]. In this industrial case that was designed complementary to our previous work in [4], we focus on dynamic obstacles in a vehicle's surroundings. The design of our experiments bases on EuroNCAP's Car-to-Car-Rear: moving (CCRm) and Car-to-Car-Rear: braking (CCRb) test procedures and considering US NCAP boundary conditions. Our goal with the latter aspect is to generally show the transferability of our method to other consumer test assessments.

A. Experimental Setup

Our experimental setting focuses on the two AEB CCRm and CCRb test scenarios from EuroNCAP, which include a VUT equipped with the active safety system and a moving target vehicle. A distinct description of the experimental setting for each scenario is presented in the following:

CCRm: The test velocity of the VUT will be increased by 5 km/h steps within the interval $[30, 80]$ km/h to cover all possible nominal test cases that may occur during an assessment for the CCRm scenario, to which we refer as the *nominal test velocity*. For each test case a tolerance range of $+1$ km/h for the VUT is valid, and thus, we also vary its velocity in 0.1 km/h steps within the interval of $[0.1, 1.0]$ km/h resulting in 121 test cases; eleven nominal and eleven additional ones with constant test velocities. We oscillate the VUT around the ideal test path with a yaw rate ψ' of 1.0 $°/s$ and a lateral deviation of 0.1 m and 0.3 m, respectively. Due to the fact that a moving target vehicle is part of the scenario with a constant test velocity of 20 km/h, we also conduct two additional simulation runs with a speed of 19 km/h and 21 km/h.

CCRb: The trajectories of the VUT are alternated in lateral position depending on the yaw rate ψ' that varies in 0.2 $°/s$ steps within the interval $[0.0, 1.0]$ $°/s$. The test velocities are fixed to the nominal value of 50 km/h.

The specific points in time are T_0 marking the point in time when the TTC equals 4.0 s, and T_{AEB} representing the point in time when the VUT's active safety system initiates the braking. The test ends when either the test velocity equals or is less than the target's velocity or a collision between both vehicles has occurred, marked as T_{end}.

To address the research questions mentioned in the introduction, we conduct an experimental study, where we oscillate the VUT along the ideal path towards the target vehicle within the allowed tolerance ranges of 0.1 m and 0.3 m for EuroNCAP's CCRm scenario while varying the velocity of both vehicles as mentioned before. For the CCRb scenario, we vary the yaw rate ψ' to investigate the influence of different trajectories.

B. Experimental Procedure

The following steps were performed to conduct a particular experiment.

Scenario Generation: At first, a set of scenarios based on the user parametrization is generated by the modeling tool and placed in the repository. At this stage, the granularity of the vehicles velocity and yaw rate can be adjusted.

Simulation: After placing the scenarios in the repository, the observer module automatically starts the processing chain for the newly added scenarios.

Evaluation: After all scenarios have been processed, a set of Python scripts is run to determine the time point when the active safety algorithm triggers an emergency level;

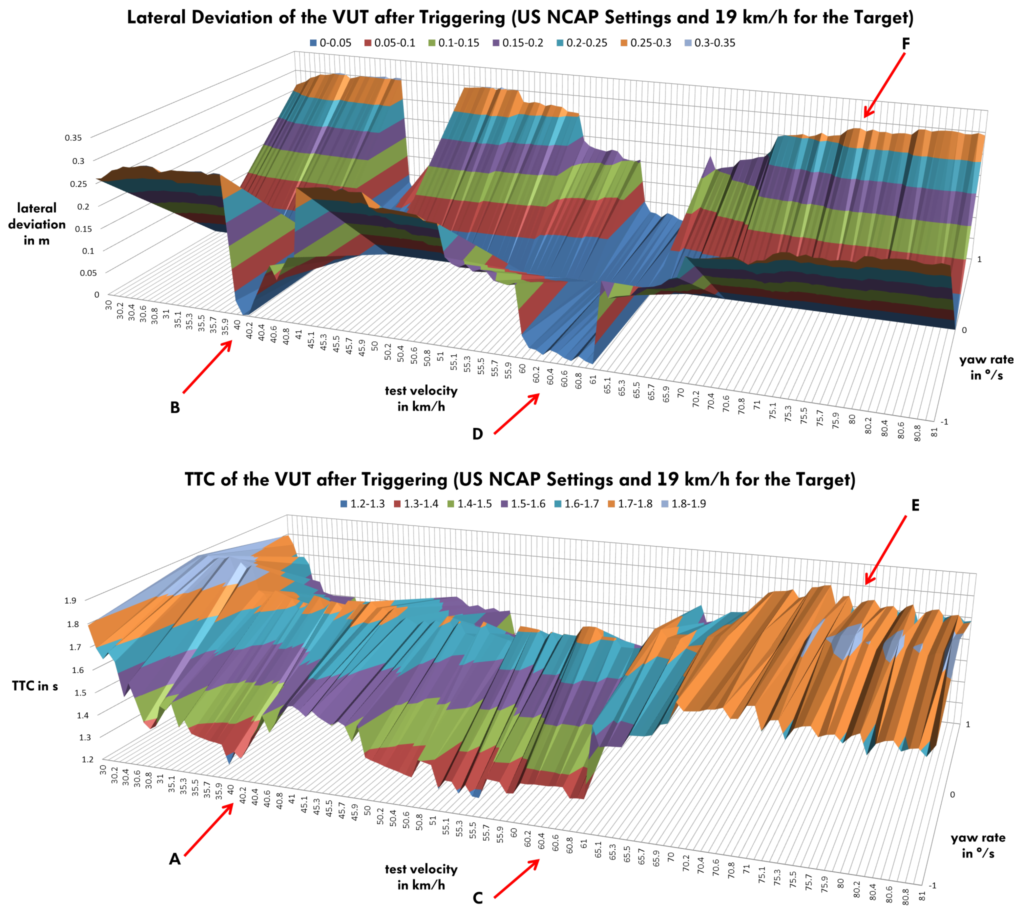

Fig. 3. Charts of the VUT's lateral deviation and TTC at that very moment an AEB algorithm triggers for an emergency braking. In this case, the velocity of the target vehicle is 19 km/h. Three different trajectories are displayed on the z-axis, indicated by the yaw rates $(-1.0; 0.0; 1.0)\ °/s$. The index pair $[A, B]$ marks the point where a delayed triggering can be identified while the lateral deviation from the ideal path is close to 0. A similiar behavior can be identified at the index pair $[C, D]$. This is called an anomaly, because a higher value as a lateral deviation would have been expected. Additional measures should be taken to confirm the such an anomaly as misbehavior. In case of the index pair $[E, F]$ the algorithm reveals a diametral behavior compared to the former described anomalies. Here, the lateral deviation is close to the allowed maximum while the corresponding TTC values are very similiar to each other.

subsequently, the corresponding VUT configuration settings at this point in time are stored as well. Furthermore, the braking model is applied to determine residual braking speeds and TTC values. Finally, the data from all scenarios is aggregated and visualized.

C. Results

In this section, we describe the results from our simulation runs. In total, we processed 2,178 CCRm test cases for our experiment and 88 test cases of the CCRb scenario. We re-run several CCRm test cases another four times for error analysis with a total amount of 1,452 test cases. In total, we simulated 3,718 test cases in this study resulting in over 7.7GB of simulation data; considering an average of one minute for processing a single test case, we needed approx. *62 hours* for all simulation runs.

Because of this amount of data, we can only show and discuss an extract of it in this article to illustrate how to interpret these results. We focus on CCRm test cases with US NCAP settings and a target vehicle velocity of 19 km/h due to a higher lateral deviation of the VUT and hypothetically more interesting effects.

In Fig. 3, the upper chart illustrates the lateral deviation of the VUT on the y-axis after the AEB algorithm triggered an emergency braking for each trajectory indicated by the yaw rate; the x-axis represents each test case indicating the velocity of the VUT.

In the lower chart, the lateral deviation on the y-axis has been exchanged by the TTC when the AEB algorithm sent its signal for activating an emergency braking.

The results of the CCRb test cases are presented in Fig. 4. Due fewer test cases, a tabular representation was chosen.

D. Analysis and Discussion

Fig. 3 reveals that there are several test cases with an unexpected behavior of the AEB algorithm. In case of the 40.1 km/h test run (cf. $[A, B]$), the temporal difference is about $0.4\ s$ between an ideal trajectory and the oscillated ones, while the lateral deviation is close to 0. We declare this an anomaly because we would have expected that there is a higher deviation in case of a lower TTC value due to later existing evasion trajectories (cf. [3]). A similar behavior

test case	-2.0 m/s^2 12.0 m		-2.0 m/s^2 40.0 m		-6.0 m/s^2 12.0 m		-6.0 m/s^2 40.0 m	
ψ' $[^\circ/s]$	y_{dev} $[m]$	TTC $[s]$	y_{dev} $[m]$	TTC $[s]$	y_{dev} $[m]$	TTC $[s]$	y_{dev} $[m]$	TTC $[s]$
-1.0	0.27	0.40	0.25	1.05	0.03	0.62	0.30	1.60
-0.8	0.24	0.41	0.14	1.02	0.01	0.64	0.30	1.51
-0.6	0.20	0.41	0.10	1.18	0.01	0.67	0.27	1.55
-0.4	0.12	0.45	0.25	1.15	0	0.67	0.22	1.51
-0.2	0.04	0.52	0.30	1.05	0	0.69	0.11	1.60
0	0	0.55	0	1.24	0	0.69	0	1.78
0.2	0.04	0.52	0.30	1.05	0	0.69	0.11	1.60
0.4	0.12	0.45	0.25	1.15	0	0.67	0.22	1.51
0.6	0.20	0.41	0.10	1.18	0.01	0.67	0.27	1.51
0.8	0.24	0.41	0.14	1.02	0.01	0.64	0.30	1.51
1.0	0.27	0.40	0.25	1.05	0.03	0.62	0.30	0.60

Fig. 4. Results of the CCRb test cases with a varying yaw rate ψ'. y_{dev} indicates the lateral deviation when the AEB algorithm triggers emergency braking; TTC indicates the time-to-collision with respect to the target vehicle.

can be noticed through the 60 $km/h + x, x \in [0; 0.9]$ test cases (cf. $[C, D]$) that should be further investigated by additional simulation runs and real test runs on proving grounds including an intensive discussion with the supplier of the algorithm. In case of the 70 km/h test cases and above, the delta between the outer and inner trajectories is smaller, which could be the expected behavior due to functional requirements. Hence, we would not declare it as an anomaly, but it should be further discussed by Original Equipment Manufacturer (OEM) and supplier engineers.

In Fig. 4, the results show that the ideal trajectory is always the best case to maximize the TTC and hence, to minimize the residual velocity of the VUT later on. But it also revealed that the 12 m distance test cases are harder to handle for the algorithm than the 40 m distances due to a significant lower TTC when triggered. Without further information from the supplier, the reason for this behavior may remain unclear. Because the delta is more than twice, it should be declared as anomaly as well.

E. Threats to Validity

To finalize the description of the results and our analysis, we briefly discuss potential threats to the validity of our study according to Runeson and Höst [31].

The entire design of our experiments was based on the available EuroNCAP and US NCAP test protocols for active safety systems. Therefore, threats to the internal validity in terms of choosing a setup that might favor our system under test can be ruled out as these official test organizations will conduct the tests in a comparable manner. Our experiments was conducted with tools provided by our industrial partners; in a very few cases we recognized communication latencies between the simulation components. We analyzed this effect and we were able to estimate its impact: For 1,815 simulation runs partitioned into five runs with the same settings, we could quantify the effect to less than 4.6% of the simulation runs.

We further consider our experiments as valid regarding its construction as we varied the vehicle's respective behavior in following a prescribed trajectory; on real proving grounds, one can expect that a driving robot used in real experiments will also slightly oscillate around the ideal trajectory. Regarding external validity, the modeling approach is independent from the simulation environment in use; though, the specific characteristics of the simulation tools in use at the industry partner need to be validated on the proving ground.

V. CONCLUSIONS

In this article, we present our approach to model several thousand test scenarios for vehicle consumer tests on the example of EuroNCAP and US NCAP. As these assessments will be of increasing importance in the foreseeable future, any potential impact of variations from the perfect trajectory on a safety system caused be the test equipment like driving robots is of apparent interest for automotive OEMs to achieve and preserve the highest ratings.

We showed how a large-scale, model-based, and fully automated simulation approach could be successfully used to unveil unexpected anomalies in two of the most important automotive consumer tests. For example, in case of the 40.1 km/h test case (cf. $[A, B]$) an anomaly was discovered, which would be difficult and time-consuming to be caught on proving grounds. Therefore, such an approach is complementary to real world testing to both, better prepare critical test scenarios, but also to study effects at large scale for different vehicle classes, sensor combinations, and consumer tests, which would be otherwise not possible.

REFERENCES

[1] T. European New Car Assessment Programme, "Test protocol - aeb systems," EuroNCAP, Test Protocol 1.0, July 2013.

[2] T. National Highway Traffic Safety Administration, "Crash imminent braking system performance evaluation (draft)," NHTSA, Test Protocol, August 2014.

[3] D. Block, S. Heeren, S. Kühnel, A. Leschke, B. Rumpe, and V. Serebro, "Simulations on consumer tests: A perspective for driver assistance systems," in *Proceedings of International Workshop on Engineering Simulations for Cyber-Physical Systems*, ser. ES4CPS '14. New York, NY, USA: ACM, 2014, pp. 38:38–38:43. [Online]. Available: http://doi.acm.org/10.1145/2559627.2559633

[4] C. Berger, D. Block, S. Heeren, C. Hons, S. Kühnel, A. Leschke, D. Plotnikov, and B. Rumpe, "Simulations on consumer tests: A systematic evaluation approach in an industrial case study," in *Proceedings of 2014 IEEE 17th International Conference on Intelligent Transportation Systems*, ser. IEEE ITSC '14. Piscataway, NJ, USA: IEEE Press, 2014, pp. 1474–1480. [Online]. Available: http://dx.doi.org/10.1109/ITSC.2014.6957640

[5] ——, "Simulations on consumer tests: Systematic evaluation of tolerance ranges by model-based generation of simulation scenarios," in *30. VDI-VW-Gemeinschaftstagung Fahrerassistenz und Integrierte Sicherheit*, ser. VDI-Berichte; 2223. Düsseldorf, Germany: VDI-Verlag, 2014, pp. 403–418. [Online]. Available: http://d-nb.info/1063049393

[6] F. Schuldt, B. Lichte, M. Maurer, and S. Scholz, "Systematische auswertung von testfällen für fahrfuntionen im modularen virtuellen testbaukasten," in *9. Workshop Fahrerassistenzsysteme*, ser. FAS '2014, vol. 9, 2014, pp. 169–178. [Online]. Available: http://www.uni-das.de/documents/FAS2014/FAS2014_Tagungsband.pdf

[7] F. Schuldt, F. Saust, B. Lichte, and M. Maurer, "Effiziente systematische testgenerierung für fahrerassistenzsysteme in virtuellen umgebungen," in *Automatisierungssysteme, Assistenzsysteme und eingebettete Systeme für Transportmittel*, 2013.

[8] A. Belbachir, J.-C. Smal, J.-M. Blosseville, and D. Gruyer, "Simulation-driven validation of advanced driving-assistance systems," *Procedia - Social and Behavioral Sciences*, vol. 48, no. 0, pp. 1205 – 1214, 2012, transport Research Arena 2012. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1877042812028327

[9] F. W. Rauskolb, K. Berger, C. Lipski, M. Magnor, K. Cornelsen, J. Effertz, T. Form, F. Graefe, S. Ohl, W. Schumacher, J.-M. Wille, P. Hecker, T. Nothdurft, M. Doering, K. Homeier, J. Morgenroth, L. Wolf, C. Basarke, C. Berger, T. Gülke, F. Klose, and B. Rumpe, "Caroline: An autonomously driving vehicle for urban environments," in *The DARPA Urban Challenge*, ser. Springer Tracts in Advanced Robotics, M. Buehler, K. Iagnemma, and S. Singh, Eds., vol. 56. Springer Berlin Heidelberg, 2009, pp. 441–508.

[10] C. Berger, *Automating Acceptance Tests for Sensor- and Actuator-based Systems on the Example of Autonomous Vehicles*. Aachen, Germany: Shaker Verlag, Aachener Informatik-Berichte, Software Engineering Band 6, 2010.

[11] M. Martinus, M. Deicke, and M. Folie, "Virtual test driving hardware-independent integration of series software," *ATZelektronik worldwide*, vol. 8, no. 5, pp. 16–21, 2013. [Online]. Available: http://dx.doi.org/10.1365/s38314-013-0193-7

[12] M. Nentwig, R. Schieber, and M. Miegler, "Hardware-in-the-loop testing of advanced driver assistance systems," *ATZelektronik worldwide eMagazine*, vol. 6, no. 4, pp. 10–15, 2011. [Online]. Available: http://dx.doi.org/10.1365/s38314-011-0034-5

[13] Vires Simulationstechnologie GmbH, "Virtual Test Drive," online, Accessed at 03.06.2014. [Online]. Available: http://vires.com/products.html

[14] Audi Electronics Venture GmbH, "ADTF Software-Umgebung für Applikationen und Tools," online, Accessed at 23.01.2015. [Online]. Available: http://www.audi-electronics-venture.de/aev/brand/de/leistungen/Entwicklungstools/adtf.html

[15] ——, "Virtual Test Drive - Simulation und Absicherung vorausschauender Funktionen in virtuellem Umfeld," online, Accessed at 23.01.2015. [Online]. Available: http://www.audi-electronics-venture.de/aev/brand/de/projekte/virtual_test_drive.html

[16] Elektrobit Corporation, "EB Assist ADTF - Driver Assistance Applications and Safety System Development," online, Accessed at 23.01.2015. [Online]. Available: http://automotive.elektrobit.com/driver-assistance/eb-assist-adtf

[17] K. v. Neumann-Cosel, M. Nentwig, D. Lehmann, J. Speth, and A. Knoll, "Preadjustment of a vision-based lane tracker," in *Proceedings on Driving Simulation Conference Europe, 2009, Monaco*, 2009.

[18] K. v. Neumann-Cosel, "Virtual test drive," Dissertation, Technische Universität München, München, 2014.

[19] B. Schick and S. Schmidt, "Evaluation of video-based driver assistance systems with sensor data fusion by using virtual test driving," in *Proceedings of the FISITA 2012 World Automotive Congress*, ser. Lecture Notes in Electrical Engineering, vol. 196. Springer Berlin Heidelberg, 2013, pp. 1363–1375.

[20] H. Holzmann, K.-M. Hahn, and M. K. B. Klein-Ridder, "Virtual test driving at general motors europe," *ATZelektronik worldwide*, vol. 3, 2008.

[21] C. Chucholowski, M. Vögel, O. von Stryk, and T.-M. Wolter, "Real time simulation and online control for virtual test drives of cars," in *High Performance Scientific and Engineering Computing*, 1999.

[22] M. Tideman, "Scenario-based simulation environment for assistance systems," *ATZautotechnology*, vol. 10, 2010.

[23] M. Tideman, R. Bours, H. Li, T. Schulze, and T. Nakano, "Integrated simulation toolset for ada system development," in *Proceedings of the FISITA 2012 World Automotive Congress*, ser. Lecture Notes in Electrical Engineering. Springer Berlin Heidelberg, 2013, vol. 200, pp. 25–36. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-33838-0_3

[24] W. Hulshof, I. Knight, A. Edwards, M. Avery, and C. Grover, "Autonomous emergency braking test results," in *Proceedings of the 23rd International Technical Conference on the Enhanced Safety of Vehicles (ESV)*, 2013.

[25] The Apache Software Foundation, "Apache Subversion," online, Accessed at 23.01.2015. [Online]. Available: https://subversion.apache.org/

[26] S. Völkel, *Kompositionale Entwicklung domänenspezifischer Sprachen*, ser. Aachener Informatik-Berichte, Software Engineering. Aachen, Deutschland: Shaker Verlag, 2011, no. 9. [Online]. Available: http://www.se-rwth.de/phdtheses/Diss-Voelkel-Kompositionale-Entwicklung-domaenenspezifischer-Sprachen.pdf

[27] H. Krahn, B. Rumpe, and S. Völkel, "Monticore: Modular Development of Textual Domain Specific Languages," in *Proceedings of the 46th International Conference Objects, Models, Components, Patterns (TOOLS-Europe)*, ser. LNBIP, R. Paige and B. Meyer, Eds., vol. 11. Zurich, Switzerland: Springer, Germany, July 2008, pp. 297–315. [Online]. Available: http://www.se-rwth.de/ rumpe/publications20042008/MontiCore-Modular-Development-of-Textual-Domain-Specific-Languages.pdf

[28] ——, "MontiCore: a Framework for Compositional Development of Domain Specific Languages," *International Journal on Software Tools for Technology Transfer (STTT)*, vol. 12, no. 5, pp. 353–372, September 2010. [Online]. Available: http://www.se-rwth.de/publications/MontiCore-a-Framework-for-Compositional-Development-of-Domain-Specific-Languages.pdf

[29] ——, "Integrated Definition of Abstract and Concrete Syntax for Textual Languages," in *Proceedings of Model Driven Engineering Languages and Systems (MODELS'11)*, ser. LNCS, G. Engels, B. Opdyke, D. Schmidt, and F. Weil, Eds., vol. 4735. Nashville, TN, USA: Springer, Germany, October 2007, pp. 286–300. [Online]. Available: http://www.se-rwth.de/ rumpe/publications20042008/Integrated-Definition-of-Abstract-and-Concrete-Syntax-for-Textual-Languages.pdf

[30] F. Shull, J. Singer, and D. I. K. Sjø berg, Eds., *Guide to Advanced Empirical Software Engineering*. London: Springer London, 2008. [Online]. Available: http://www.springerlink.com/index/10.1007/978-1-84800-044-5

[31] P. Runeson and M. Höst, "Guidelines for conducting and reporting case study research in software engineering," *Empirical Software Engineering*, vol. 14, no. 2, pp. 131–164, Dec. 2008. [Online]. Available: http://link.springer.com/10.1007/s10664-008-9102-8