

# Novel Hybrid-Testing Paradigms for Automated Vehicle and ADAS Function Development



Selim Solmaz, Franz Holzinger, Marlies Mischinger, Martin Rudigier, and Jakob Reckenzaun

## 1 Introduction

The increasing challenge in the calibration, testing, and validation of ADAS functions, active safety systems, and technologies for autonomous driving in the different development cycles requires novel and alternative solutions. Especially the verification and testing of the ADAS/AD functions with respect to the expected vehicle behavior poses a major difficulty during the development of such systems. In order to ensure accurate vehicle behavior, several tests must be conducted under safe driving conditions. The conventional approach for doing this is to test the developed vehicle within closed proving grounds accompanied by limited performance evaluations and tests on public roads. However, the coverage and extent of these evaluations are often very limited, as they are in general too costly and often infeasible to perform for verifying every ADAS function before commissioning. The current approach in the automotive industry is to employ simulation tools to verify such systems. Due to the lack of one-to-one correlation between simulation models and real vehicle measurements, simulation-only approaches are not sufficient for verification and testing. Given these facts, there is an increasing interest for novel testing concepts combining real-world testing and simulation, particularly for the development of automated vehicles and related technologies. Motivated by these facts, we introduce two novel testing concepts proposed for the development and testing of ADAS/AD functions in this chapter.

---

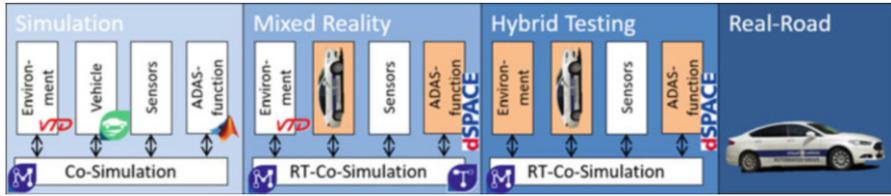
S. Solmaz (✉) · F. Holzinger · M. Mischinger · M. Rudigier · J. Reckenzaun  
Virtual Vehicle Research GmbH, Graz, Austria  
e-mail: [selim.solmaz@v2c2.at](mailto:selim.solmaz@v2c2.at); [franz.holzinger@v2c2.at](mailto:franz.holzinger@v2c2.at); [marlies.mischinger@v2c2.at](mailto:marlies.mischinger@v2c2.at);  
[martin.rudigier@v2c2.at](mailto:martin.rudigier@v2c2.at); [jakob.reckenzaun@v2c2.at](mailto:jakob.reckenzaun@v2c2.at)

© Springer Nature Switzerland AG 2021  
U. Z. A. Hamid, F. Al-Turjman (eds.), *Towards Connected and Autonomous Vehicle Highways*, EAI/Springer Innovations in Communication and Computing,  
[https://doi.org/10.1007/978-3-030-66042-0\\_8](https://doi.org/10.1007/978-3-030-66042-0_8)

The first testing concept involves a novel steerable rolling test bench (DÜRR X-Road-Curve System; see [1]) with a corresponding methodology for development, testing, and validation of ADAS/AD systems utilizing a real vehicle in the loop. This novel test bench allows independent steering of the front wheels by rotating the rollers about respective vertical axis. The regulation of speed and angle of these rollers keeps the vehicle position relatively fixed on the rolling test bench. This consequently enables lateral and longitudinal inputs by the driver or the automated vehicle control system. In this chapter, a practical implementation of this test bench for the functional test approach for camera-based ADAS/AD systems is demonstrated. Particularly, it is shown how Lane Keeping Assistance (LKA) and Adaptive Cruise Control (ACC) functions are tested on this vehicle-in-the-loop setup [2]. In the suggested solution, Model.CONNECT™ co-simulation platform is used as the central medium of integration [3]. As part of the solution, the measured driving status from the test bench (in the form of wheel speeds and steering angle) is fed into a vehicle dynamics simulation, which in return calculates the dynamic response of the vehicle. The vehicle states and dynamic parameters are fed into an environment simulation software, which generates a video output in a virtual world and models a driving scenario. The video output is then shown on a monitor in front of the intelligent traffic camera installed in the car. This represents an over-the-air (OTA) stimulation of the camera input [4]. The camera provides object lists as input for LKA and ACC algorithms running on the embedded ECU on the car [2, 5]. Based on the simulated virtual world, the car can keep to the lane or the distance between ego and target vehicle autonomously, solely relying on the video stream shown on the monitor. The concept was introduced recently in a seminal paper by the authors in [6].

The second testing concept on the other hand is an alternative concept for ADAS/AD system testing solution named the “hybrid testing,” where a real vehicle is combined with virtual ones in a co-simulation framework. Hybrid testing is a novel testing methodology, which was developed in the scope of the EU-H2020 project INFRAMIX to test several mixed traffic scenarios involving autonomous and manual driven vehicles and is based on the Model.CONNECT™ co-simulation platform. In some respects, hybrid testing is similar to the vehicle-in-the-loop (i.e., SIL, VEHIL, or VIL) systems, where real hardware and virtual components are combined in the same simulation framework, which in this case is based on the co-simulation platform Model.CONNECT™ at its core. This is also a generalization of hardware-in-the-loop (HIL) testing, specifically involving a test vehicle as the “hardware” in a feedback loop with the simulated modules. Hybrid testing methodology involves the real vehicle driving in an enclosed proving ground using the real ADAS functions running on its real-time ECU, whereas the driving scenarios and the surrounding traffic as well as the sensor data are simulated on the co-simulation platform using dedicated software components.

Figure 1 shows how these new test concepts can be integrated into the verification and validation process of ADAS/AD development, from simulation-only tests, via



**Fig. 1** Consistency of the development process with staggered evaluation steps

mixed-reality tests on a test bench and hybrid tests with the vehicle and simulated sensors on the road. Simulation-only concepts are fast and, compared to test bench and road test, a cheap method to verify ADAS/AD functions. Multiple parallel simulations allow to test a large amount of test scenarios in a short time. Thereby, all functions can be integrated in one simulation environment like CarMaker [7]. An alternative approach is the modular integration of various simulation components with the use of co-simulation.

Selected and critical scenarios can be tested with mixed-reality methods on a test bench, where the real vehicle is integrated in a virtual environment via co-simulation framework. This allows to analyze the behavior on real vehicle dynamics in a limited extent. The limitation comes from the fact that the longitudinal dynamics are fully represented, whereas the lateral dynamics are represented only in the tire forces. In contrast to hybrid tests or testing on a proving ground, the mixed-reality tests have the advantage to reproduce test scenarios in a very flexible and continuous fashion. In a next step, the vehicle with stimulated sensors can be verified on a real road with hybrid testing. This is an extension of the test bench tests and allows additional analyses, especially regarding vehicle dynamics, before finally the vehicle with real components is tested on the road. This continuous testing enables consistent validation and verification during the development of ADAS/AD systems.

This chapter is organized as follows: First we introduce the ADAS functions utilized in the test vehicle in Sect. 2. We then give an overview of the conventional simulation-based ADAS development in Sect. 3. Next, in Sect. 4, we introduce our novel mixed-reality test bench implementation utilizing a co-simulation platform and a rolling chassis-dynamometer. Subsequently, we introduce the hybrid testing methodology in Sect. 5. Based on the introduced two novel testing paradigms, we describe how the proposed techniques can be utilized during the development and verification of the ADAS/AD functions before they are deployed on the end product, in both Sects. 4 and 5, respectively. It is discussed finally how these systems are likely to reduce the amount and complexity of the on-road testing of ADAS/AD systems thereby increasing the overall safety, security, and quality while also reducing the time to market of these systems.

## 2 ADAS Function Control Modules and Subsystems

### 2.1 Motorway Chauffeur (MWC)

Motorway Chauffeur (MWC) is an experimental SAE Level 3/3+ equivalent autonomous driver assistance technology developed in-house at the Virtual Vehicle Research GmbH. The MWC combines a number of separate control subsystems or advanced driver assistance (ADAS) technologies or Automated Driving (AD) functions, which include a longitudinal and a lateral guidance system. Speed and target distance regulation in the form of Adaptive Cruise Control (ACC) and lane and target trajectory following in the form of Lane Keeping Assistance (LKA) was developed in-house. An essential component of the MWC is the trajectory planner (TP) performing the strategic planning of the lane change maneuvers while utilizing the lateral and longitudinal tracking controllers. The general structure and architecture of the MWC is shown in Fig. 2.

The implementation of the MWC controller is based on MATLAB/Simulink, where the ADAS components were developed in a simulation environment involv-

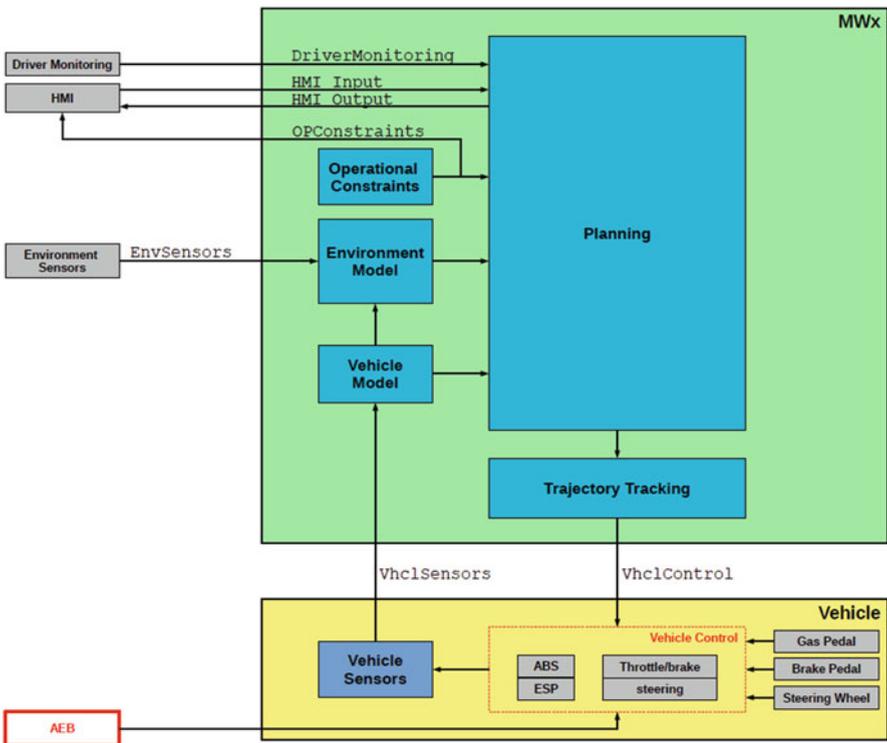


Fig. 2 Virtual Vehicle Research GmbH Motorway Chauffeur (MWC) Architecture

ing MATLAB/Simulink and CarMaker [7], which was quite useful for virtual development and testing before implementation on the test vehicle. Once the respective ADAS components comply with certain performance criteria in simulation, they are exported to embedded C/C++ code to run on DSpace MicroAutobox-II real-time target ECU hardware found on the test vehicle. See Sect. 5.2 for a detailed description of the test vehicle setup. As was experienced during the development process, there are however many limitations and differences imposed by a simulation-only development, particularly in modeling sensor signals with regard to uncertainties, biases, as well as limited and varying data rates. This, on the other hand, results in different behavior and/or reduced performance between simulation-developed and real-life implemented algorithms.

## 2.2 Adaptive Cruise Controller (ACC) Subsystem

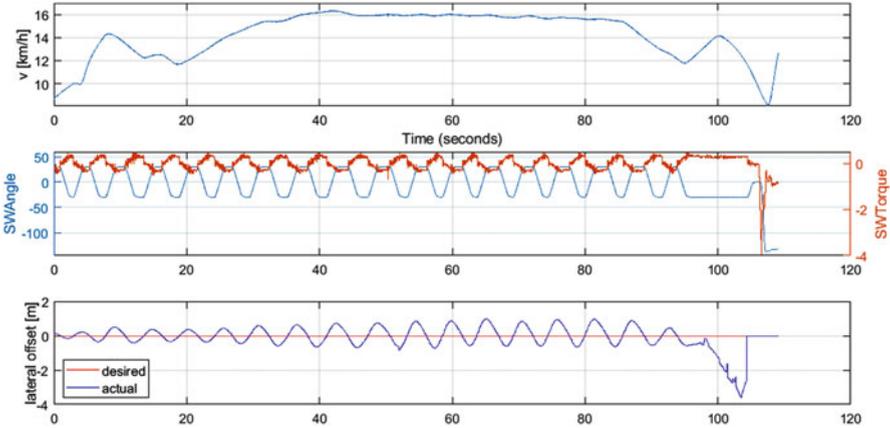
The Adaptive Cruise Control is an ADAS function to assist the driver in longitudinal control of the vehicle. The ACC adapts its driving behavior to the current traffic situation by keeping a safe distance to a possible leading vehicle or driving with a desired speed. As the ACC is a well-developed system, almost every major automotive manufacturer offers nowadays such an ADAS function. An overview over the existing types and technologies can be found in recent compilation books such as [8, 9].

The current implementation of the ACC was developed and integrated as part of the MWC according to the international standard ISO 22179 (intelligent transport systems, full speed range adaptive cruise control (FSRA) systems, performance requirements and test procedures). The core of the ACC controller consists of a state machine that is responsible for switching between different modes. These modes include car-following, cruising, standby, and inactive. For each of these specific modes, standard pole placement based on P- and PI-controllers was developed, respectively. Additionally, models for sensor fusion and target selection complement this ADAS function.

## 2.3 Lane Keeping Assistance (LKA) Controller Subsystem

A flatness-based bumpless controller design is used for the LKA. Therefore, a linear single track vehicle model is used for the lateral vehicle dynamics [10] in combination with a linearized lane keeping model [11]; more details and analysis can be found in [12, 13].

The LKA controller developed in MATLAB/Simulink was compiled on a MicroAutobox-II hardware and utilized the Dataspeed ADAS Kit [14] for controlling the vehicle steering. The measured variable of the controller was the relative position of the ego vehicle from the MobilEye ADAS camera [4].



**Fig. 3** The log of the variables relevant to LKA function during an early successful test run

The first tests on the test bench with the real vehicle indicated an insufficient dynamic behavior. The steering wheel angle and the steering wheel torque in the second plot of Fig. 3 show oscillations with a vehicle speed greater than 30 km/h. The third plot in Fig. 3 illustrates an unstable behavior of the lateral vehicle position. The oscillations are caused by structural delays of the closed loop in the range of 300 ms and could not be reduced by adjusting the parameters of the controller.

### 2.3.1 Delay Compensation Utilizing Kalman Predictor

To improve the LKA controller behavior, i.e., to reduce the lateral oscillation due to the additional delay in the closed loop, a Kalman filter prediction based on the lane tracking model [11] is used. Based on the estimated states and the last measured values, the 300 ms delay can be compensated by an  $n$ -step predictor. During the prediction of the  $n$ -steps, it is assumed that the input data does not change. According to the measurements, this assumption is generally not valid. Nevertheless, the implementation shows significant improvements of the closed loop control behavior. A detailed analysis of the Kalman filter implementation was discussed in [15].

## 2.4 Trajectory Planner

The trajectory planner uses inputs from several sources to compute a speed and position trajectory. There are three sources for this: the ego vehicle dynamics, the environment objects, and the traffic objects. From the ego vehicle, it uses vehicle dynamics variables like longitudinal speed  $v_x$ , lateral speed  $v_y$ , accelerations, and

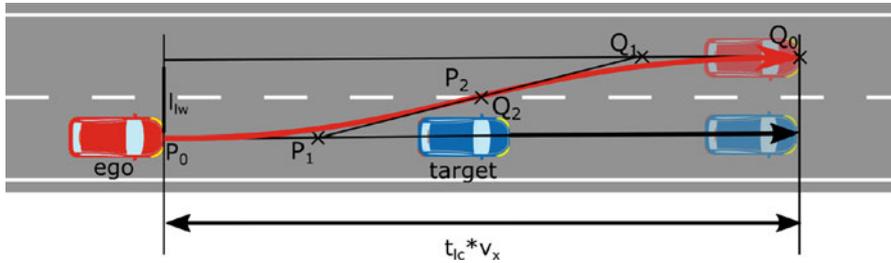


Fig. 4 Lane change: Bezier curve based on lane change trajectory

yaw rate  $\dot{\varphi}_z$ . The vehicle sensor provides information about road marks like position, line type, and visibility as well as information about the nearby traffic objects (surrounding other vehicles) like position, velocity, and type. The trajectory planner described here was designed for simulation of motorway driving with well-defined and visible road markings. As a simplification, the trajectory planner drives the AV in the middle of the lane. Only when a lane change is performed, the AV leaves the middle of the lane and is transferred to the new lane. This transfer follows trajectory based on a Bezier curve.

The LKA controller of the MWC is used to follow the actual lane and also to perform the lane changes. Besides the vehicle dynamic variables, the reference trajectory that implies the description of the line to follow and an offset  $y_L$  are the inputs for the LKA. With these two inputs, the lane change is generated. While  $t < t_{LC}/2$ , the reference trajectory follows lane 1 (old lane), and with  $t_{LC}/2 < t < t_{LC}$ , it follows lane 2 (new lane), and the demanded offset  $y_L$  follows a Bezier curve. Figure 4 shows such lane change maneuver. At time  $t_0$  the ego vehicle starts the lane change maneuver at position  $P_0$ , and completes the maneuver at time  $t_1$  and position  $Q_0$ . The lane change duration  $t_{lc}$  is a parameter of the trajectory planner.

Using the symmetry with respect to  $P_2 = Q_2$  and splitting the curve in two parts leads to two quadratic Bezier curves. The trajectory of the centerline of the lane and the spline are superposed to form the resulting trajectory. This works fine for straights and curves with low curvature. This does not work however for narrow curves that are not part of a highway and also for evasive maneuvers. Yet these extreme situations are outside the scope of this implementation. The decision if a lane change is initiated is taken by a network of tactical decisions. The basis of these decisions is a matrix, which describes where a target vehicle from the surrounding traffic is located. Each matrix element holds the ID of the nearest vehicle in this area. First, each of the surrounding vehicles is assigned to a lane with the help of the road markings. After that, this information is used to allocate the vehicles in the obstacle matrix. With the known surrounding, lane change requests are generated, and predefined rules are checked to ensure that a lane change will not cause an accident.

The trajectory planner uses the rightmost lane if it is free from other vehicles. If a slower vehicle prohibits the AV from reaching its free flow speed, a lane change request to the left lane is generated. When not driving on the rightmost lane, the trajectory planner forces the AV to change to the right lane, should this lane be free in front of the AV. Another reason for a lane change is merging into the main traffic on an on-ramp. There is also the possibility of an externally triggered lane change, e.g., from a traffic management center via I2V communication.

In case of a lane change request, a set of rules has to be checked. This set of rules consists of discrete decisions like the availability of a lane or free space on the desired lane. These rules can also be parameterized, e.g., the necessary time gap to a leading vehicle can be chosen by setting the appropriate parameter. When these checks for a lane change are passed, the lane change will be started. The trajectory planner passes a desired lateral lane position to the trajectory execution controller of the MWC. The core of the speed controller is the ACC function of the MWC. While using tactical decisions and the obstacle matrix of the lateral controller, an appropriate target is selected. Usually, the target vehicle is the vehicle on the same lane in front of the ego vehicle. However, in case of a lane change, the desired lane is taken into account. If there is a vehicle on the target lane, this is chosen as the target vehicle, or if there are no vehicles on the desired lane, the desired velocity for speed following (i.e., cruise control function) is chosen. In special situations, separate controllers override the ACC function, e.g., an emergency brake controller.

### 3 Simulation-Based Development of ADAS Functions

Simulation has become an indispensable part of vehicle development in general. Specifically, in the case of ADAS and AD function development, it is a necessary technique. The number of tests to verify and validate the functionality and safety-relevant features of ADAS/AD functions in every conceivable driving scenarios is not feasible without simulation. Therefore, virtual vehicle prototypes in traffic and environment platform are used. A list of simulation tools, which provide 3D-vehicle dynamics systems, is given in [7]. These provide an internal simulation environment and traffic simulation as well as sensor simulation and also support the possibility to integrate these models from third-party tools.

Modular integration of simulation components utilizing co-simulation, e.g., with [3], is an alternative concept. This modular approach allows a consistent usage of components from simulation-only test, via mixed-reality and hybrid tests, as illustrated in Fig. 1. The adjustments of the simulation components from one test phase to the next can thus be kept to a minimum.

It is clear that the amount of tests required to verify ADAS/AD systems cannot be handled without simulation. Specifically, in the early phases of the ADAS/AD development process, the benefits of the simulation is apparent. The interactions of sensors, actuators, control strategies, and vehicle dynamics can be analyzed by utilizing corresponding mathematical models, and without needing

real components. Also, instead of having to test the components on a real vehicle on the road, the interaction can be pre-tested on any scenario in the simulation. Nevertheless, the more progress is made in the development process, the more difficult it becomes to ensure validation with simulation-only concepts. The use of simulation-only concepts is therefore no longer sufficient, especially at the limits of vehicle dynamics, which makes the verification and validation with real components essential. Since simulation-based development is not the main focus of this chapter, and also the fact that there is a vast literature on this topic, we conclude its discussion here after its brief introduction.

## **4 A Framework for ADAS Function Development Utilizing Chassis Dynamometers**

Engineering and development of ADAS functions as well as AD technologies need extensive testing and validation cycles, much more so than traditional automotive safety systems, before their approval and deployment to the end users. This process is usually quite time- and cost-intensive utilizing the traditional automotive testing and validation approaches. Therefore, it is in the interest of ADAS/AD system suppliers and OEMs alike to speed up the time-to-market of such systems, without jeopardizing the compliance with expected safety and performance levels [16]. According to the traditional methods, the required testing and validation effort for the ADAS/AD systems is often performed in enclosed proving grounds, as well as utilizing limited trials on public roads. Such conventional approaches however can only provide limited performance evaluations since the associated costs of such testing methods are quite high, while their corresponding coverage is low. As a remedy, simulation techniques are quite often employed by system developers to extend the scenario testing of the ADAS/AD functions, which help to enrich performance evaluations as well as to provide testing possibilities for safety critical edge scenarios. For a sufficient validation, only a subset of scenarios is mandatory, due to the assumption that only a small number of test scenarios lead to potentially faulty behavior of the ADAS/AD functions [17]. Regardless, simulative analysis alone cannot sufficiently answer the testing and validation requirements of ADAS/AD systems for every possible driving scenario before their commissioning, since there are various nonlinearities and uncertainties that cannot be modelled in simulation.

To cater for the testing and validation task of ADAS/AD functions, novel testing approaches have been proposed in the recent years, where simulation is combined with hardware components in various applications. For this purpose, hardware-in-the-loop (HIL) techniques are commonly utilized for the development and testing of automotive internal combustion engines [2], electronic-control-units (ECUs) [5, 18], as well as components for electric vehicles and electrified drive-trains [19]. The integration of an entire automotive vehicle into a simulation environment is

called “vehicle-in-the-loop” (or VEHIL) demonstrated in a seminal work by TNO [20]. This concept was used in a follow-up study to develop and test automotive longitudinal and lateral control systems; for more details, see [21]. Therefore, an entire vehicle on a rolling test bench is combined with a simulated environment. Based on this concept, a research, development, and testing platform for emergency braking and pre-crash systems was derived [22–24]. Similar VEHIL concepts have been developed in recent years by different suppliers, e.g., the DRIVINGCUBE from AVL [25] with a focus on the development and validation of ADAS/AD functions.

Besides the concept of testing the entire vehicle on the test bench, VEHIL tests are performed on a test track (proving ground), where the sensor data are provided by the simulation of the environment; details can be found in [22–24].

Consequently, the aim of this section is to introduce a new steerable rolling-dynamometer test system along with the software components as well as the demonstration of a suitable testing methodology, which can cater for the testing and validation requirements for ADAS/AD systems and automotive active safety systems. Conventional rolling test benches do not support the steering of the vehicle. The additional integrated rotatable front rollers of the novel test bench allows a steering of the front wheels and enables the longitudinal and lateral road force emulation on the vehicle. This testing system allows, for instance, the tuning of Lane Keeping Assistance (LKA) and Adaptive Cruise Control (ACC) functions, which was demonstrated with the *VIRTUAL VEHICLE* Automated Drive Demonstrator (ADD); see [26]. Therefore, the driving scenario was simulated, and the environment was visualized on a screen to stimulate the camera system of the ADD vehicle with the driving functions on the test bench. This framework allows to simulate different driving conditions and to reproduce critical driving scenarios for tuning and validation of ADAS functions.

#### ***4.1 X-Road-Curve Steering Chassis Dynamometer***

In order to guarantee the quality and functionality of vehicle features of series vehicles, testing at the end of the production line is essential. In addition to quality control tests, the alignment of the chassis and wheels, the adjustment of the headlamps, and driving safety tests like the check of the ABS system are performed. Therefore, a rolling test bench (i.e., chassis dynamometer) is used at the end of the assembly line (i.e., end-of-line, EOL) to test the series vehicle. Typically, these tests of the EOL control step take about 1.5 to 5 min per vehicle. If problems are detected on the vehicle during the EOL inspection, they are examined at the rework station and repaired if necessary. The development of ADAS functions as well as automated driving functions is tremendously challenging as well as a complex process, and therefore new approaches are urgently needed. These new methods have to be incorporated in different stages of the development starting from design, calibration, test procedure, up to the production and the EOL processes [27].

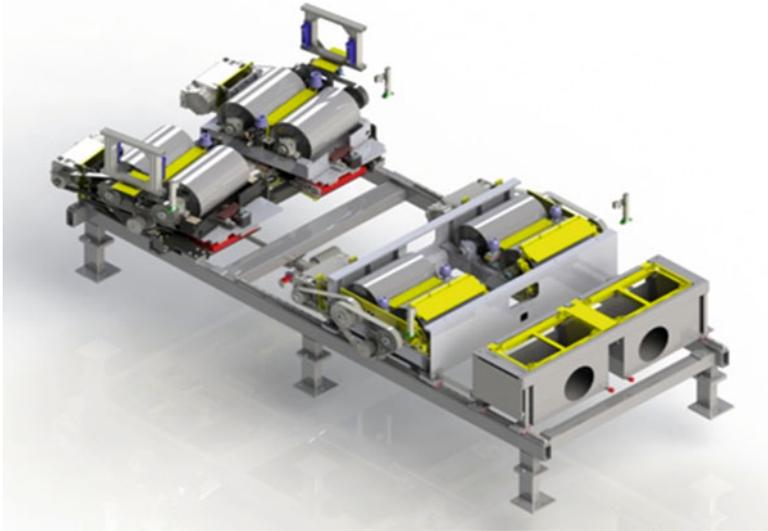


Fig. 5 DÜRR X-Road-Curve Test Bench platform 3D-CAD drawing

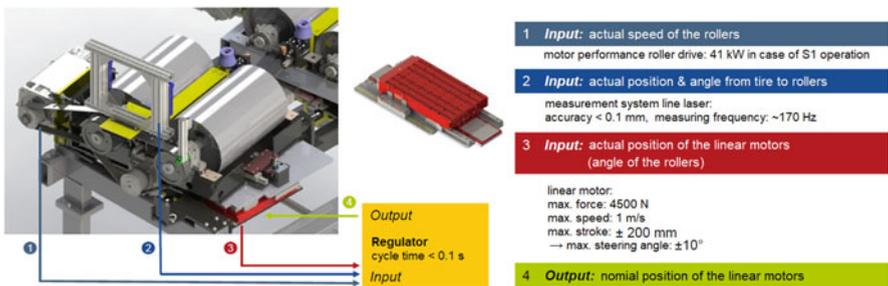
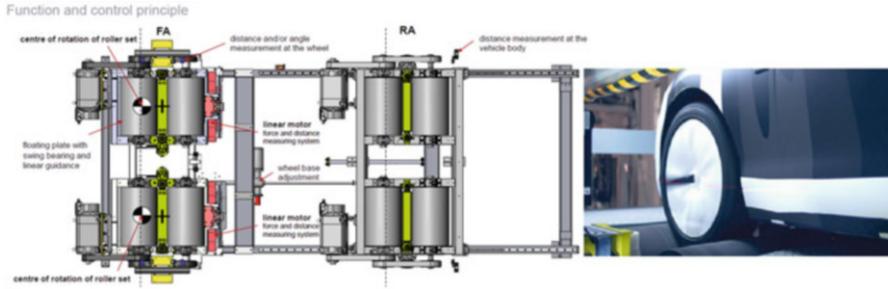


Fig. 6 Detailed view of the steering front roller set together with the system input-output limitations

In order to handle these new challenges, DÜRR Industrial Products Company has developed a novel chassis dynamometer system called X-Road-Curve [1]. Therefore, the traditional EOL brake/ABS chassis dynamometer has been extended to test more vehicle functions. A 3D CAD drawing of this new dynamometer is shown in Fig. 5. The new test bench has independent rotating rollers on the front axle. These automatically keep the vehicle in the center of the test bench without the necessity for a driver to be in the vehicle. In addition, it enables tests to be carried out with steering movements. These novel test bench technique allows testing and developing of driving functions for autonomous vehicles.

Figure 6 shows the steering front rollers. The cycle time of the controller is less than 0.1 s. In Fig. 7, the top-down view of the chassis dynamometer with the rotating plates is illustrated. Linear actuators are used to move the plates. Distance sensors



**Fig. 7** Functional and control principle of the steering dynamometer

at the side of each tire measure the position of the wheels. This distance is used as feedback signal for the controller to rotate the plates and keep the vehicle in the middle of the test bench.

The front rollers have a maximum rotation angle of  $\pm 10^\circ$ , each. Depending on the steering ratio of the vehicle, this is about  $200^\circ$  at the steering wheel. The maximum velocity of the vehicle on the test bench can be up to 170 km/h for driving straight and is limited to 130 km/h with steering movements. Each roller is powered by a 40 kW motor with a peak force of 6000 N.

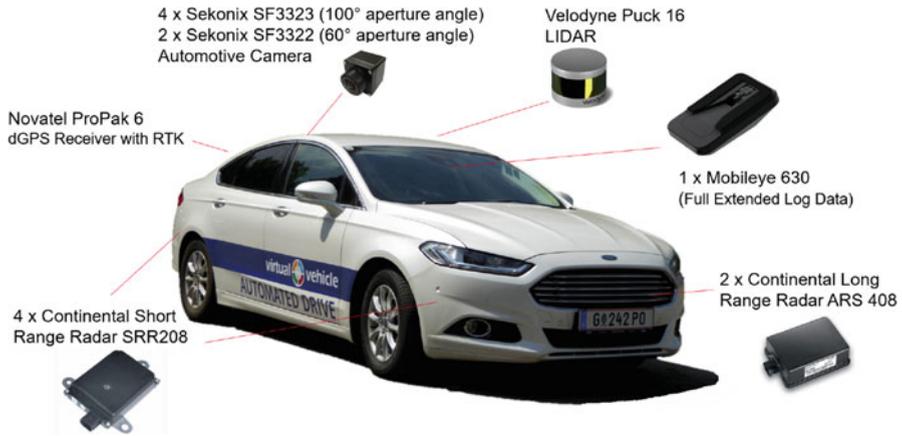
## 4.2 ADAS/AD Test Bench System Description

At the beginning of the section, a brief description of the used components in the DÜRR X-Road-Curve test bench is given. Then, various development methods (Simulation, HIL, VIL) are described, and their characteristics are discussed. In the end, the actual testbed for ADAS function development, testing, and calibration is described as combination of the different components and application of the described methods.

The first key element of the DÜRR X-Road-Curve is the *Virtual Vehicle Automated Drive Demonstrator (ADD)*. The underlying platform of the ADD is a standard 2016 Ford Mondeo where additional hard- and software components were mounted to enable automated control of the vehicle (shown on Fig. 8). Automated drive capabilities are primarily enabled using the Data Speed ADAS Kit [14] and various other HW components depicted. The driving functions are running on a D-Space MicroAutobox-II [28] control unit (not shown on Fig. 8) that is used as the onboard processing unit. Out of the whole set of sensors, in this specific application, only the Mobileye traffic camera [4] was used.

The second key component of the test bench is the DÜRR chassis dynamometer that was described in the previous section in more detail.

The third key element is the Model.CONNECT<sup>TM</sup> [3] co-simulation framework to link the physical components with software and simulation modules.



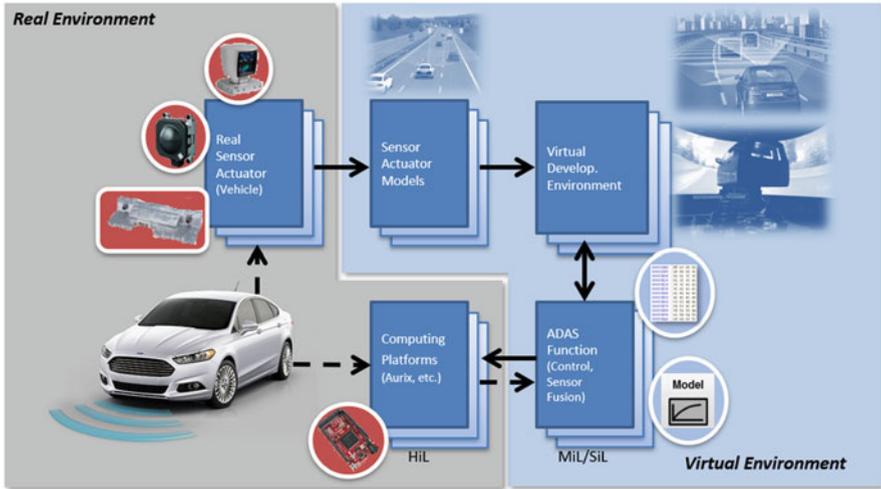
**Fig. 8** Virtual vehicles Automated Drive Demonstrator (ADD) with the hardware breakdown layout

When automotive engineers need to choose the appropriate testing and development method, the development stage and the development scope define the requirements regarding testing cycle time and accurateness. In early development stages for both – system- and competent-level development – usually pure simulation tools are used. Simulation is limited as for every model assumptions are made. All models are abstractions from reality due to an intentional or non-intentional neglect of physical processes and effects happening in reality.

For prototype testing or if certain effects cannot be modeled in the desired accuracy in simulation, HIL methodology could be applied. In a HIL setup, a simulation model is replaced by a physical component. In this setup, the input that goes into the component is emulated in simulation, while the output of the real component is again fed into and continuously processed by the coupled simulation [2, 18]. HIL testing enables to take a detailed and accurate look on a certain component, with the opportunity of a high number of repetitions in a short time. As HIL testing is usually done in a lab setting, HIL comes with a relatively low cost. Apart from component behavior, it also gives evidence on the overall system behavior, within the range of the capability of the surrounding simulation.

A generic mixed-reality testing HIL setting is shown in Fig. 9, where real hardware components are embedded into a co-simulation framework enabling a bi-directional communication between the virtual environment and physical components. Over the last decades, HIL setups have often been used in the design phase of novel products and components, especially in the design and development of various active safety systems. In these lab studies, vehicle dynamics signals output from simulation are transmitted to the controller to test the response of the controller and/or the embedded software stacks running on the controller [29].

More than one component can replace simulation models. When the whole vehicle is tested, as an ultimate extension of HIL, the term vehicle-in-the-loop



**Fig. 9** Mixed-reality testing concept of real and virtual components

(ViL) is commonly used. In this specific testing methodology, a real vehicle on a test bench and a simulation environment are coupled and exchange data during every simulation time step. The idea behind this is based on the direct link between vehicle movement in simulation when operating the vehicle on the test bench. One requirement for a test bench setup is the ability to mimic road properties as indicated and given by the virtual environment. If an ADAS function shall be tested in such a framework, conventional test benches are often not sufficient due to the lack of emulating steering behavior. For simpler active safety systems, the necessary signals for the system to operate can simply be acquired from the virtual simulation environment. In contrast, automated driving functions need to calculate a model of their surrounding environment to plan and execute suitable reaction.

The implementation of the perception testing was a major challenge in the development of the test bench. AD functions require relevant input signals about its surroundings through sensors, so HiL testing of sensors, sensor components, and perception systems became an important work package. To create the needed input signals, different techniques have been proposed in the literature [25]. All examples from literature first simulate the static (road, lanes) and dynamic (objects, vehicles, VRUs) ground truth data in a 3D virtual environment.

From the simulated ground truth, there are two approaches to generate the sensor input for the driving function in a lab or on a test bench. The first approach is to utilize the simulated sensor output. Thereby the sensor model derives the input for the ADAS function in simulation and provides this information via an interface. In the second approach, a physical sensor is stimulated using the physical stimulation of the real vehicle sensor and passes the signal to the platform running the driving function. This approach is used in this demonstration.

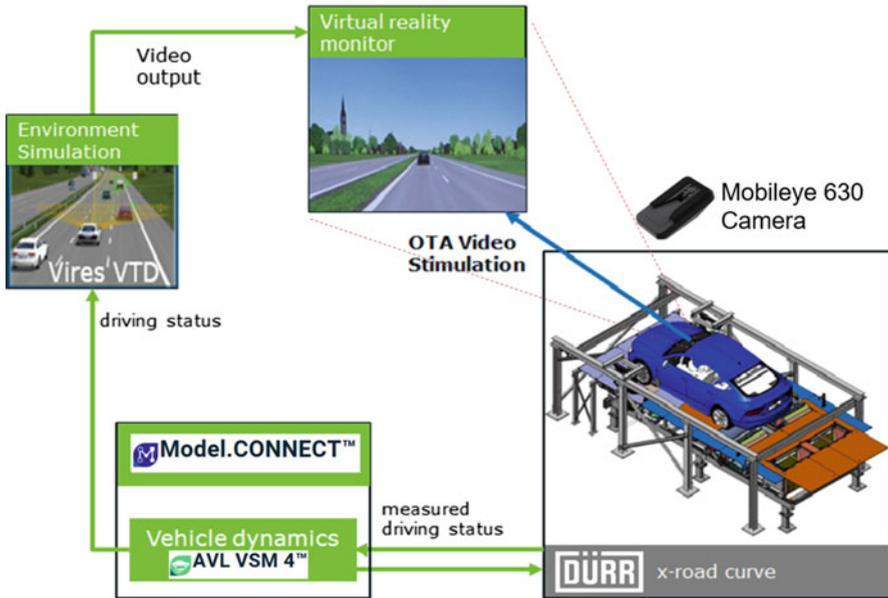


Fig. 10 DÜrr X-Road-Curve VIL ADAS function testing and calibration setup

In the DÜRR X-Road-Curve, a Mobileye camera provides the input for the lateral and longitudinal control of the vehicle. In this testbed, a big screen showing a representation of the scenery was used. The distance to the screen needs to be calibrated, so that the screen fits the camera’s field of view and that the distance to the objects shown on the screen fits the estimates from the camera sensor. The environment perception of a driving function is an important element of its architecture. For all automated vehicles that follow the sense-plan-act cycle, the decision-making step usually follows the perception of the surrounding environment. After calculating a model of the environment, the path planning can be initiated on strategic (route), tactical (e.g., lane selection), and reactive (e.g., immediate obstacle avoidance) level.

The full architecture of the mixed-reality VIL simulation system implementation is shown in Fig. 10. The core of the testbed is the DÜRR X-Road-Curve steerable chassis dynamometer. The physical testbed is coupled with a Model.CONNECT™ co-simulation framework and a VR monitor to create stimuli for the Mobileye camera. This integrated mixed-reality concept of three different components enables testing of ADAS functions.

Signals from the real car and the physical part of the test bench drive the vehicle dynamic simulation. In this specific application use case, the vehicle dynamics is simulated in AVL-VSM software as part of the Model.CONNECT™ environment. The vehicle dynamics block controls the environment simulation too, to emulate the driving scenario and a real-time 3D visualization of the virtual environment. This 3D virtual twin is shown on a television screen, which is placed in front of the vehicle. It is also placed in the field of view of the Mobileye camera that is mounted



**Fig. 11** Mixed-reality testing system implementation visuals

on the windshield. When all components are properly calibrated to an integrated, mixed-reality simulation test bed, the vehicle assumes that it drives on a real road.

By connecting all these individual parts, the speed and directional vehicle motion on the testbed is realized and duplicated within the digital twin. In this specific test setup, two ADAS functions were running on the vehicle under test.

- *Adaptive Cruise Control (ACC)*: vehicle holds a pre-set velocity when no car drives in front and adapts its velocity to a lead car
- *Lane-Keeping Assistant (LKA)*: holds the vehicle within a specific lane

In this demonstration project, we used the Dataspeed ADAS Kit [14] to control the ADDs steering actuator enabling a lateral and longitudinal movement of the vehicle. A Mobileye ADAS camera [4] was used for receiving the relative position information of the ego-vehicle with regard to the lanes markings as well as distance and speed of target vehicles in front. The ACC and LKA functions were written in MATLAB/Simulink code and exported to real-time C/C++ for the specific DSpace MicroAutobox-II hardware that they run on.

We demonstrated that the proposed setup is well suited for consistent testing of ADAS functions during development. The same setup could be utilized in the end-of-line production process, where the final checks of the system are performed before delivering the car to its customer. A further possible application could be yearly inspections performed by car repair shops. For a deeper understanding of the test setup implementation, visuals showing the DÜRR X-Road-Curve test bench in its demonstrated version are given in Fig. 11.

### 4.3 Cross-Domain Integration

Compared to real road ADAS/AD testing, the vehicle on the test bench has a reduced dynamic behavior. Due to the reduced vehicle body movement, physical system signals like longitudinal and lateral acceleration or yaw rate are not measurable. These physical signals have to be provided by simulation, where the remaining signals are modelled in a dynamic subsystem or via stimulation, where pre-recorded measurements are used to stimulate the missing signals. For the integration of the simulation models and the real vehicle on the test bench, a co-simulation framework (Model.CONNECT™, [3]) is used. It allows the integration of different simulation tools with different communication media and interfaces to the vehicle and the testbed. In addition to the simulation of the missing physical sensor signals, the co-simulation framework also provides the environment of the vehicle via a screen and the additional road specific loads for the test bench (e.g., road gradient). The overall integration consists of the mechanical integration of the vehicle and the test bench between the wheels and the rollers and the steering capability of the X-road-Curve for the steering maneuvers. In addition to that, there is a visual integration between the simulated environment (road, traffic, traffic sign, etc.) and the camera system of the ADAS/AD function. The several hardware components (ADAS-KIT, MWC, camera) and the software components (vehicle dynamic simulation, environment) communicate via an electric-bus integration with different protocols (CAN, Ethernet). Figure 12 shows the schematic integration of the VIL. The missing physical signals of the vehicle are simulated in a vehicle dynamic model in VSM. Therefore, VSM is integrated into the co-simulation platform Model.CONNECT™, which ensures the data exchange between the components at certain coupling times. The VSM model acts like an observer and provides the missing signal via CAN

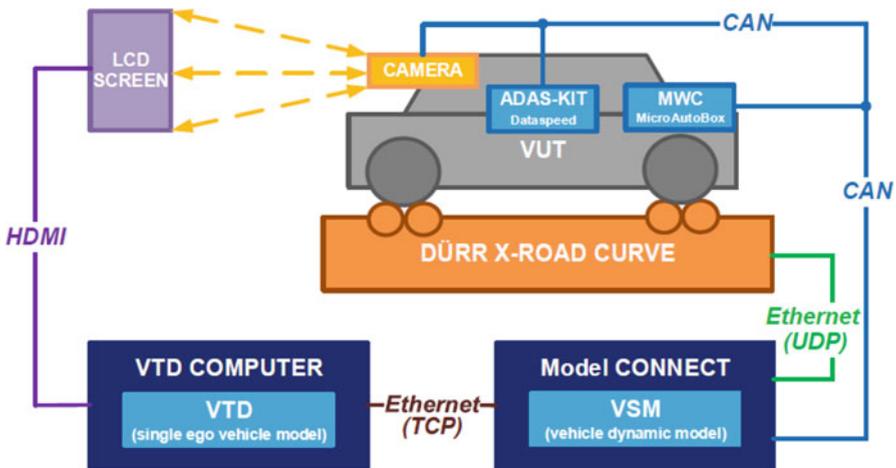


Fig. 12 Topology of the VIL test bench setup

to the control units for the ADAS function. The current speeds of the test bench rollers are sent to the dynamic vehicle via an Ethernet/UDP communication. The environment is simulated in VTD on a separate workstation and communicates via Ethernet/TCP with dynamic vehicle model. VTD also provides the road and traffic information to the real vehicle on the test bench via an LCD screen, where environment from the vehicle point of view is illustrated. The camera in the vehicle films the screen, and the ADAS function acts regarding the information from the camera and dynamic behavior of the vehicle.

The integration of software with real-time components leads to a number of challenges regarding the coupling (e.g., round-trip-time, coupling error, etc.). These challenges were already discussed by the authors; see [6].

#### 4.4 Implementation and Test Results

For testing the self-developed ADAS/AD functions described in Sect. 2, a test scenario was defined, which demonstrates the ACC functionality on the one hand and shows the LKA function on the other. For this purpose, a driving cycle involving a complete virtual circuit was created. The driving cycle modelled in *VDT* [30] is illustrated in Fig. 13. The straight lane in the top section of the scenario is for the ACC tests and the bottom track with a curvy road is for demonstrating the LKA functionality.

##### 4.4.1 Adaptive Cruise Control (ACC) Testing Scenario

The ACC function is evaluated on a straight lane in the top section of the test scenario in Fig. 13. The test consists of two vehicles. The first vehicle is the

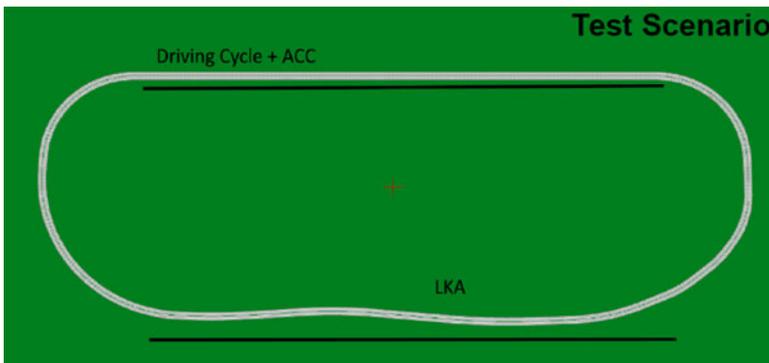


Fig. 13 Driving cycle to test ACC and LKA functions on the mixed-reality test bench

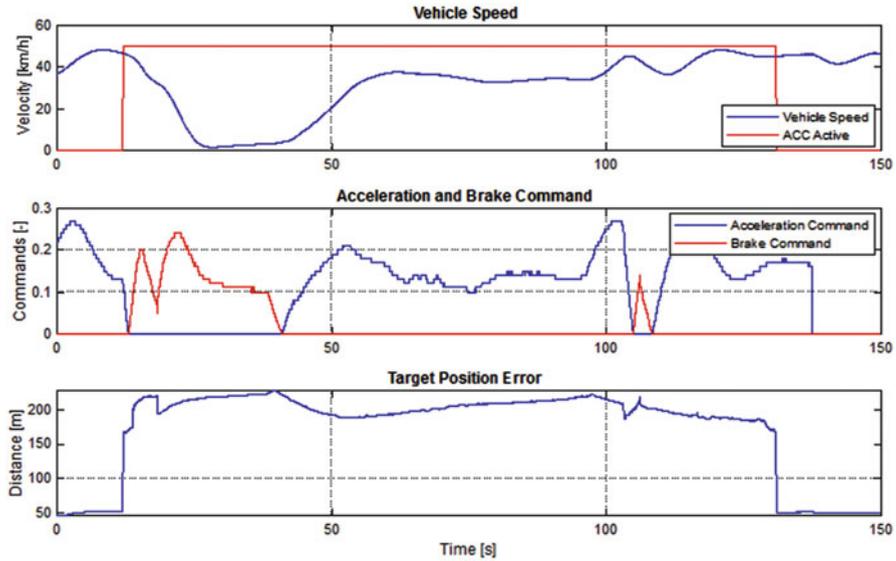
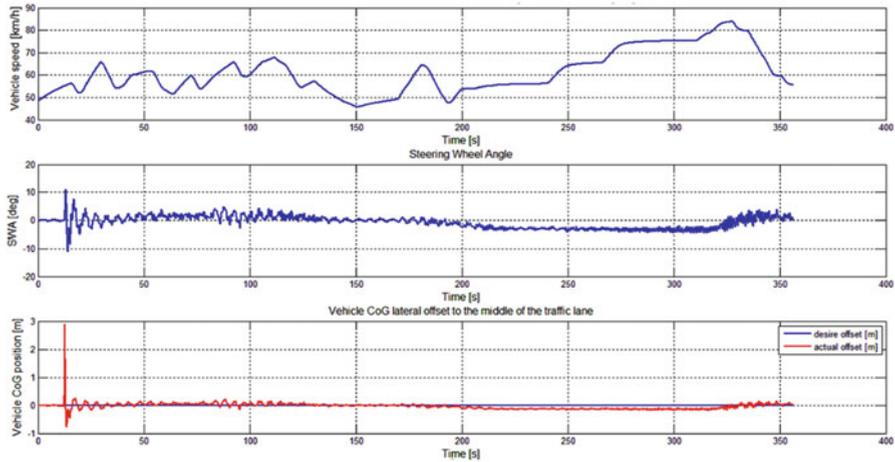


Fig. 14 Vehicle ACC function test results on the DÜRR X-Road-Curve mixed-reality test bench

Vehicle under Test (VuT), with activated ACC function and a demand velocity of  $v_d = 50$  km/h. In front of the VuT stands a second vehicle (target vehicle) on the same lane. When the VuT approaches the target vehicle, it decelerates close to  $v = 0$  km/h until the target vehicle accelerates first to  $v = 30$  km/h and after  $t = 50$  s to  $v = 100$  km/h. The VuT follows the target vehicle with its velocity until the target vehicle accelerates and leaves the visual range of the VuT. Due to the lower-demand velocity, the VuT accelerates to  $v_d = 50$  km/h. The results of the ACC test scenario are shown in Fig. 14. The first plot shows the velocity of the VuT and the ACC activation signal. The acceleration and braking commands of the ACC function are shown in the second plot, and the last plot shows the distance to the target vehicle (position error). After activation of the ACC function and identification of the target vehicle, the VuT reduces the velocity and stands almost still behind the target vehicle. The VuT accelerates and follows the target vehicle with  $v = 30$  km/h at about  $t = 50$  s. The target vehicle speeds up to  $v = 100$  km/h at about  $t = 100$  s, and the VuT accelerates to the demand velocity  $v_d = 50$  km/h and remains at this speed.

#### 4.4.2 Lane Keeping Assistance (LKA) Testing Scenario

On the track at the bottom of the driving cycle in Fig. 13, the LKA scenario is tested. The part of the track consists of slightly curved sections. Figure 15 shows the results of the LKA test scenario on the DÜRR X-Road-Curve test bench. The plot on the top



**Fig. 15** Vehicle LKA test on DÜRR X-Road-Curve mixed-reality test bench utilizing vehicle steering angle

shows the vehicle velocity of the VuT. The steering wheel angle (SWA) is illustrated in the second plot. The last plot shows the position of the vehicle (center of gravity) relatively to the lane-center line. The LKA is activated at about  $t = 15$  s after a long curve. The scenario starts with a right turn, changes into a left turn, and ends again in a right turn. The VuT accelerates at the second half of the track. The VuT follows the course of the road, but a high-frequency oscillation is observed at the steering angle, which rises with increasing speed. It has been shown that this oscillation does not occur during real road tests even at higher speeds; see [6]. This leads to the conclusion that the oscillations are caused by imperfections of the coupling between the vehicle with the test bench and the simulation (e.g., communication delay, measurement noise, etc.). Additional investigations are necessary for a detailed analysis of these imperfections.

## 5 A Vehicle-in-the-Loop Testing Method for the Development of Automated Driving Functions

The development of automated driving functions and technologies poses various challenges even at the development stages. Testing the automated or ADAS function and verifying the vehicle behavior in multiple traffic scenarios poses a major difficulty during the development cycle of such systems. In order to ensure accurate vehicle behavior, several tests must be conducted under safe driving conditions. However, these are in general too costly and are often infeasible to perform for verifying every ADAS function before deploying such systems on production

vehicles. The current approach in automotive industry is to employ simulation tools to verify such systems against certain performance criteria. However, simulation-only verification and testing also have their limitations.

Advanced testing methodologies that bring together the advantages of virtual and real tests have been developed in the past. Real-time HIL simulation is commonly used in automotive engineering, e.g., in engine development [2] and electronic control unit (ECU) development [5, 18] as well as for the development and testing of electric vehicle components [19]. An extension of the HIL-simulation technique on the vehicle level was proposed as vehicle-in-the-loop (VEHIL) simulation in the seminal work by TNO [20] and conceptualized as testing option for longitudinal and lateral control systems in [21].

Several examples of combined virtual and real testing have been reported in the literature [16, 31–33]. Some end products are also available offered for this purpose by commercial vehicle dynamics software producers [34]. Also there are some recent use case demonstrations for the utilization of the combined virtual and real testing concept by some automotive manufacturers in a series of publications [35–39].

As an alternative implementation of the combined or mixed reality for automotive testing, we developed in the scope of the EU-Horizon 2020 funded project [40] an alternative testing concept for ADAS and automated driving systems testing named the “hybrid testing” methodology, where a real vehicle can be combined with virtual ones in a co-simulation framework for analyzing ADAS system performance in virtually created traffic scenarios. Hybrid testing is a novel testing methodology based on the Model.CONNECT™ co-simulation platform [41–44], which was implemented and extended in the scope of the INFRAMIX project to test several mixed traffic scenarios involving autonomous and manually driven vehicles.

In some respects, hybrid testing is similar to the VEHIL or VIL system solutions, where real components and simulated components are combined, however with a distinct difference in utilization of the co-simulation platform Model.CONNECT™ at its core. This is also a generalization of hardware-in-the-loop (HIL) testing, specifically involving a test vehicle as the “hardware” in a feedback loop with the simulated modules. Other publications suggest calling it scenario-in-the-loop testing. In the hybrid testing methodology, a real car drives in an enclosed proving ground using the real ADAS functions running on its real-time ECU (dSPACE-MicroAutobox-II), whereas the driving scenarios and the surrounding traffic as well as the sensor data are simulated on the co-simulation platform using dedicated software components.

In the developed hybrid testing solution, a real-life automated vehicle (AV), which is a generic test vehicle, is driven on a proving ground, where the required perception sensors inputs are provided by a real-time co-simulation of static and dynamic environment using SUMO [45] traffic simulation software. The utilized automated driving function is an in-house developed SAE Level 3 ADAS function with lateral and longitudinal tracking as well as lane change decision capabilities (i.e., Motorway Chauffeur or MWC), enabling the AV to have Adaptive Cruise

Control (ACC), Lane Keeping Assistance (LKA), and trajectory planning (TP) functionalities.

In this chapter we describe the implementation details of the hybrid testing methodology and its corresponding components together with the use case implementation for testing and evaluation of a trajectory planning algorithm.

### 5.1 Submicroscopic Co-simulation Framework and Hybrid Testing Methodology

In the scope of the hybrid testing methodology, one of the core components is the test vehicle driven in an enclosed proving ground and is interacting, via a coupled submicroscopic co-simulation framework, inside a virtual traffic environment. The sensor signals of the vehicle are simulated. That means, virtual static objects such as lane markings, traffic signs, and dynamic virtual traffic are provided by the submicroscopic co-simulation. The current vehicle position is measured by a GPS sensor to localize the ego vehicle on a corresponding digital HD map, which is linked by the co-simulation framework. The position and other vehicle states of the VuT are transmitted to the traffic simulation component, and based on this, the virtual traffic reacts accordingly. The concept of the hybrid testing methodology is shown in Fig. 16, where virtual and real components are illustrated as two interconnected parts of the environment. The coupling of physical and virtual components enables a testing environment for evaluating ADAS functions under realistic dynamic traffic conditions.

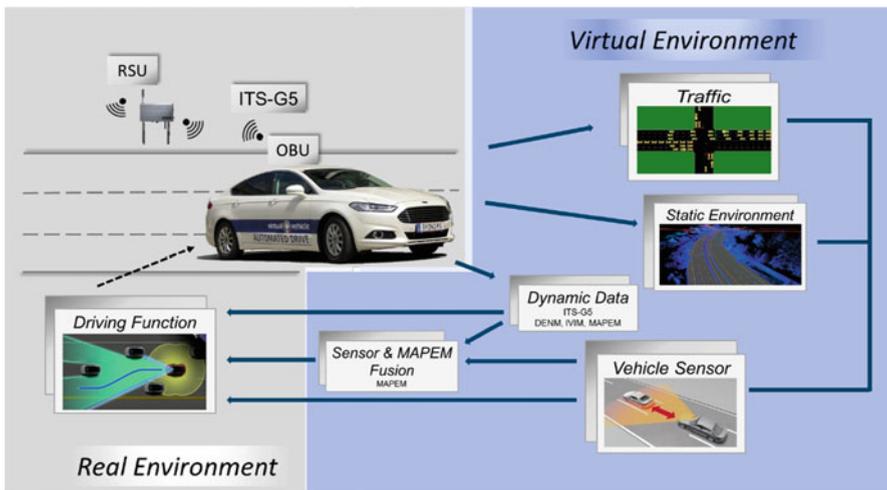


Fig. 16 Hybrid testing concept

The test vehicle is equipped with an ADAS-Kit that enables controlling the car from an algorithm running on a development ECU. The details of the test vehicle and its hardware setup are described in Sect. 5.2. In addition to the test vehicle, there are two further components that belong to the real environment: the *Road Side Unit* (RSU) [46] and the *On-Board Unit* (OBU) [46]. Both are ITS-G5 communication devices from Siemens Mobility and enable connectivity as well as information support to the VuT. The RSU is placed on the proving ground and sends out C-ITS messages over the ITS-G5 communication channel. The OBU that is integrated inside the car receives and interprets the messages and directs them to the driving function.

Since the most important physical components are described in separate sections, the following subsections explain specifically the submicroscopic co-simulation framework and its various components. Each of these software components was developed for a specific task and has particular responsibilities as described in detail below.

### 5.1.1 Traffic Simulation

As microscopic simulation module, the SUMO traffic simulator is used. SUMO provides realistic mobility pattern for a multitude of vehicles and generates the overall traffic flow in our investigations. Each vehicle in the simulation is modeled as an individual agent using simplified car-following models. Additionally, SUMO provides an interface to control individual vehicles. Therefore, the current vehicle position, the vehicle speed, and the orientation have to be transmitted to SUMO. The most important prerequisite for the exchange of the vehicle position to SUMO is the match of the SUMO coordinate system, the map which is used in the submicroscopic co-simulation as well as the GPS coordinates of the real car.

### 5.1.2 Static Environment

The main challenge of the *Static Environment* is to provide relevant road infrastructure information to the ego vehicle. This means a highly accurate course of all lanes on the current road section and positions of the traffic signs in front of the ego vehicle. A widely used format for the description of static infrastructure elements, like lane courses and their attributes as well as traffic signs, is the OpenDRIVE [2] file format, which is also the input for the *Static Environment*. The *Static Environment* can be parameterized by the length  $l$  of a searching box which is depicted in Fig. 17. According to the input of the *Static Environment* and the parameterization of the dashed box, the model will filter all information about the lanes inside the dashed box such as the x/y coordinates, the road mark type, the lane id, and the road id. This detailed description of the lanes inside the dashed box as well as the signals will be sent to the *Sensorsystem*.

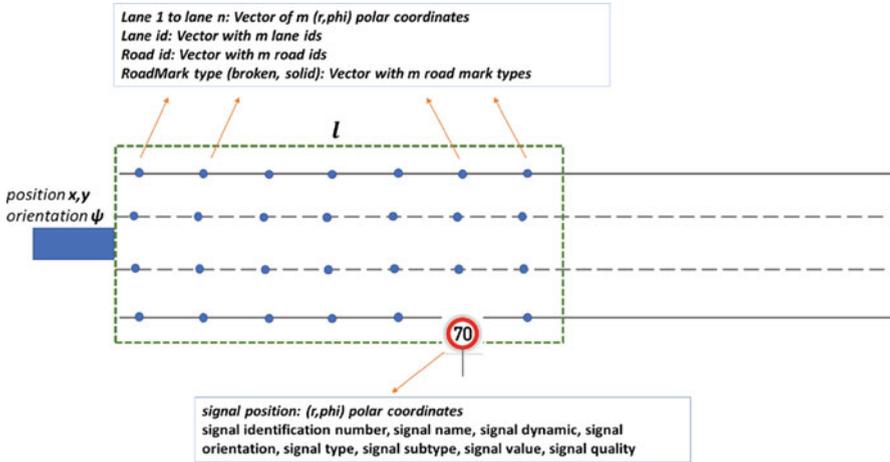


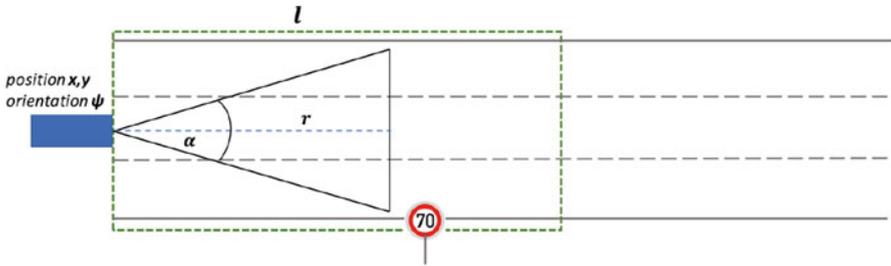
Fig. 17 Output of the *Static Environment*

Figure 17 shows the output of the *Static Environment* which is transmitted to the *Sensorsystem*. Every lane is described by a set of polar coordinates and specific attributes such as the lane id, the road id, and the road mark type. There are nearly the same parameters for the description of traffic signals such as polar coordinates for the signal position, signal ids, and so on.

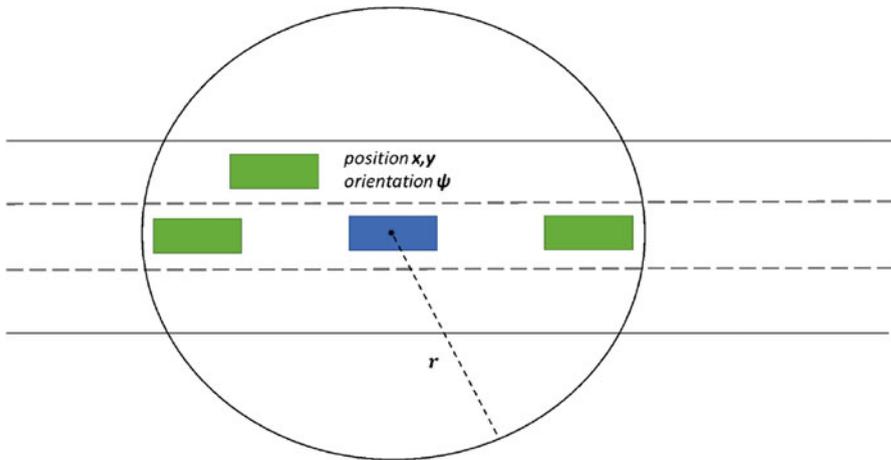
### 5.1.3 Sensorsystem

The *Sensorsystem* is used for ongoing monitoring of the vehicle's environment. The *Sensorsystem* provides continuously updated information regarding the traffic situation to the automated driving function so that the trajectory planning algorithm can be adapted to the current traffic situation. The *Sensorsystem* is a 2D sensor with ideal characteristics. That means neither non-linearity, signal noise, nor environment influences are modeled. The *Sensorsystem* in the submicroscopic co-simulation includes two sub-sensors. The first one is the *Static Environment Sensor*, and the second one is the *Dynamic Environment Sensor*. The *Static Environment Sensor* can be parameterized by its range  $r$  and azimuth angle  $\alpha$ , as can be seen in Fig. 18. The horizontal field of view of the *Static Environment Sensor* corresponds to the real field of view of a camera inside the car which detects lanes and signs in front of the car. This sensor is parameterized with a range of 150 m and an azimuth angle of  $40^\circ$ .

The input from the *Static Environment* can be described as a part of a static map which includes the lanes and signs in front of the vehicle. The main task of the *Static Environment Sensor* is to decide which of the information from the map is located inside the sensors field of view and can be seen by the vehicle. The output of the *Static Environment Sensor* are lanes and signs which are inside its field of view; see Fig. 18.



**Fig. 18** Output of the *Static Environment Sensor* is the information inside the rectangular shape



**Fig. 19** *Dynamic Environment Sensor* – field of view

The *Dynamic Environment Sensor* receives dynamic input via the coupling interface from the SUMO simulation. Since the driving function needs the information about vehicles in front of the vehicle as well as in the rear section of the vehicle, this sensor is parameterized in a way that vehicles can be detected in a range of 200 m with a field of view of 360°; see Fig. 19. Additionally, to each vehicle position  $(x, y)$ , the velocity  $v_x$  in longitudinal direction and the velocity  $v_y$  in lateral direction and the vehicle width are transmitted via the interface to the microscopic simulation SUMO. To decide which information is inside the sensors' visual range is the main task of the vehicle sensor.

### 5.1.4 Dynamic Data

The *Dynamic Data* model is responsible for the handling of dynamic data information which is received via OBU. The tasks of the *Dynamic Data* model range

from revision of received messages to transmission of ADAS-relevant information from that messages to the real environment. This model has a mechanism which continuously revises an interface for new incoming messages. In case of a received message, the next step is to parse the message and extract particular information of it. After verification of the message content and its validity inspection for the current vehicle state, the information and a validity flag are forwarded to the driving function of the real environment, and the VuT will act accordingly to that message.

## 5.2 Test Vehicle and Experimental Setup

The test vehicle used for the testing of in-house developed ADAS functions is based on a Hybrid Ford Mondeo MY2016 platform. This vehicle has an ADAS kit build by Dataspeed Inc. [14] and is a comprehensive add-on hardware and software solution that allows the full control of the throttle, brake, steering, and shifting of the test vehicle. The picture of the vehicle along with the installed sensor hardware is shown in Fig. 8. Among the indicated sensors in the figure, only the dual-antenna RTK-GPS is utilized in the scope of hybrid testing. The test vehicle also has many computational hardware for implementation, where most of the development ECUs are installed at the trunk of the vehicle. Of primary importance in this list is the Dataspeed CAN interface, which enables the access to the onboard vehicular sensors and controls. Using this interface, the throttle, brake, and steering as well as other parameters can be controlled using a secondary ECU. The data rates for the control-specific parameters provided by the Dataspeed CAN interface can be seen in Table 1.

**Table 1** Dataspeed CAN data and the corresponding data rates

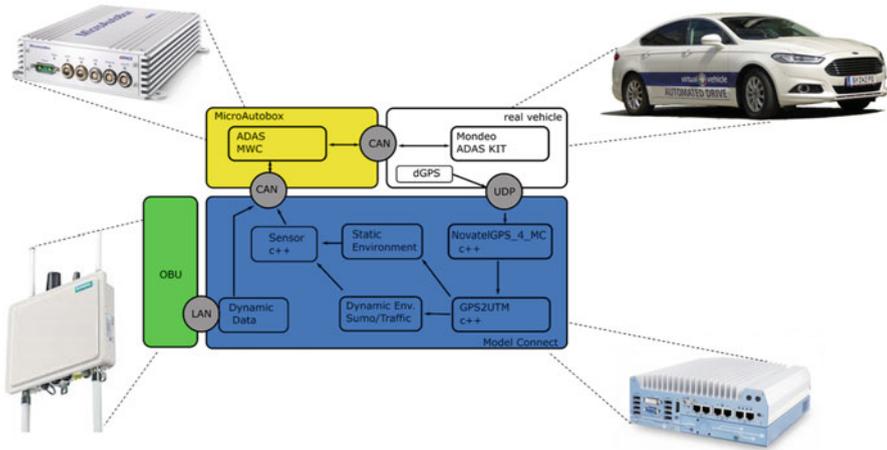
Feature	Specification
Platform	Ford CD4
Initial release date	Oct 2015
Throttle control frequency	50 Hz
Brake control frequency	50 Hz
Steering control frequency	50 Hz
Steering angle control	Yes
Steering torque control	No
Gear shift control (PRNDL)	Yes
Turn signal control	Yes
ULC (speed control)	Yes
Individual wheel speeds	100 Hz
3-Axis accelerometer	100 Hz
Roll and yaw rate gyro	100 Hz
Parking SONAR Sensors	5 Hz
Tire pressures	2 Hz
GPS	100 Hz

### 5.2.1 Test Vehicle Setup

The test vehicle also houses a dSPACE MicroAutobox-II (ds1401) real-time ECU [28] that runs the ADAS functions in real time based on the provided sensor data from onboard sensors as well as the simulated one from the co-simulation platform. The ADAS functions for ACC, LKA, and TP are implemented in MATLAB/Simulink and later exported to C++ code that is automatically generated by the MATLAB/Simulink embedded coders to run on the dSPACE MicroAutobox real-time hardware. Therefore, the ADAS functions running on the MicroAutobox ECU provide the driving commands to the vehicle's actuators through the Dataspeed CAN interface.

In hybrid testing, the co-simulation framework runs on the Nuvo 7006 PC that is located inside the vehicle, and the RSU (Road Side Unit) is located outside the vehicle as part of the test-site infrastructure. The PC in the VuT is responsible for running the traffic and environment simulation, sensor models, and object list generation algorithms in a co-simulation environment while also being connected in real time to the CAN bus of the VuT to collect variables relevant for representation of the VuT in the traffic simulation on the PC. During hybrid testing, the ADAS control functions are based on the vehicle (specifically running on the dSPACE MicroautoBox-II platform) as opposed to having an ADAS function block along with vehicle dynamics simulator in the submicroscopic co-simulation framework. Additionally, the car is equipped with an OBU for bidirectional communication with the RSU. In this context, the communication between the RSU and OBU is unlimited, and there can be persistent bidirectional link between the RSU and the OBU during the demonstrations of the specific hybrid testing use case demonstrations planned. That is, when a pre-specified traffic control message is sent by the RSU to the VuT, it will act according to this message as soon as the message is received by the OBU in VuT. The complete hardware and software architecture and the internal communication structure implemented on the Ford Mondeo test vehicle for realizing the hybrid testing methodology are seen in Fig. 20.

In the scope of the hybrid testing methodology, the sensor signals of the test vehicle are simulated, and they sense the virtual objects created in the simulated static environment as well as the dynamic traffic components. The simulated sensor signals exclude the GPS sensor data, which provides survey-grade (i.e., cm-level accurate) position and heading information from the real vehicle to localize the ego vehicle on the digital HD map (i.e., the static environment) used in the co-simulation platform. The position and other vehicle states of the VuT are sent to the traffic simulation, where the simulated vehicles in the traffic simulation react to the VuT. The environmental information such as the lane markings, road curvatures, etc. as well as other traffic participants are only virtually present. The virtual dynamic objects consist of the vehicles of the surrounding traffic, while the static environment consists of features such as the road markings and traffic signs. The concept of the hybrid testing methodology is shown in Fig. 16, where virtual and real components are illustrated as lumped at the right- and left-hand sides of the figure, respectively.



**Fig. 20** Hardware and software architecture for hybrid testing

The use of virtual sensors naturally imply that the vehicle is not able to sense real objects and real road markings in the testing area (i.e., proving ground). Therefore, hybrid testing needs to be conducted on an open area, large enough to cover the virtual test track with additional buffer zones around the road layout, for safety purposes. Also, real infrastructure components are present in the form of RSU that sends out C-ITS messages over the ITS-G5 communication channel to an OBU that is integrated to the vehicle. The real vehicle reacts to the virtual elements accordingly while also taking into account the real-world communication via RSU. Moreover, using the OBU, the status of the real vehicle can be sent back to the RSU for further processing as may be required from the use case scenario. When approaching the area of interest (e.g., the road works zone), the automated vehicle receives a C-ITS message from the RSU, which will then react according to this message, considering the surrounding virtual traffic.

The results of hybrid testing are logs of positions and velocities of the VuT and the simulated traffic, as well as the logs of the C-ITS message transmissions. Out of these logs, behavior of the VuT can be investigated. The logged data either come from the simulation tool used for the virtual traffic (SUMO) or the vehicle sensors itself (inertial measurements such as acceleration, yaw rate, steer angle, etc.). During the test runs, the vehicle states (position, velocity, acceleration, heading, yaw rate, etc.), the states of the surrounded vehicles, as well as the C-ITS messages, are recorded and can be used for adapting and validating simulation parameters. Signal processing and evaluation tools are implemented in MATLAB and Microsoft Excel. The resulting investigations can be used to verify the vehicle behavior (both VuT and the simulated vehicles) in the simulation (parameters of the car-following and lane change model in SUMO) or lead to adaptations of the vehicle behavior in the real vehicle. With the findings of hybrid testing, both the simulation and the automated driving function can be validated and improved.

The time frame for hybrid testing scenarios is between 30 and 60 s depending on the desired velocity and as restricted by the test site physical limitations. The actual useable stretch of the road section (ÖAMTC Lang/Lebring test track) used during the hybrid testing demonstrations is about 250 m, and assuming an average speed of 50 km/h during a hypothetical test, the entire stretch is covered in less than 30 s.

### 5.2.2 HD Map Generation as Preparation for Hybrid Testing

Before starting with hybrid testing, the preparation of an HD map was required. The OpenDRIVE standard, which is an xml-compliant scheme, provides a detailed representation of static road infrastructure elements such as lane markings as well as traffic signs and enables safe vehicle guidance along pre-calculated trajectories. Due to the high level of detail in this format and also SUMO's import mechanisms for that file format, the OpenDRIVE standard is used as basis in the submicroscopic co-simulation. After local reference measurements at the proving ground, the OpenDRIVE file was created semi-automatically based on these measurements, as well as OpenStreetMap™ and Google Maps™. As microscopic simulations do not require that high-level detailed maps in order to model traffic, SUMO provides a conversion tool for OpenDRIVE files to generate SUMO-specific net files. Figure 21 shows the test area as satellite picture and the corresponding SUMO net file. One of the fundamental challenges is the mapping of the current vehicle position, measured by a GPS sensor, as the OpenDRIVE file uses UTM coordinates and the traffic simulation uses a SUMO-specific coordinate system.

The preparation of the test vehicle and the HD maps of the corresponding proving ground enabled the conduction of different scenarios. A description of these scenarios together with an analysis of the results is given in the following section.

## 5.3 Scenario Description, Implementation Results and Analysis

Various number of use case studies were conducted during the hybrid testing trials, and two specific use case examples are discussed in this section. The following experiments are focusing specifically on lane change and merging maneuvers in combination with C-ITS messages. The initiation of lane changes is triggered by the trajectory planner as a result of the road layout as well as the traffic situation at hand. While the vehicle drives on the real test track, it is fully aware of the presence of virtual vehicles close to it and takes their behavior consequently into account. The use case studies were extended with communication elements to investigate the reaction of the vehicle on specific C-ITS messages for speed advices. Besides the speed information, this messages include also GPS positions of a *relevance zone* as seen in Fig. 21. Here, what is meant by the *relevance zone* is the zone for which the C-ITS message information is valid. In what follows, we describe two example

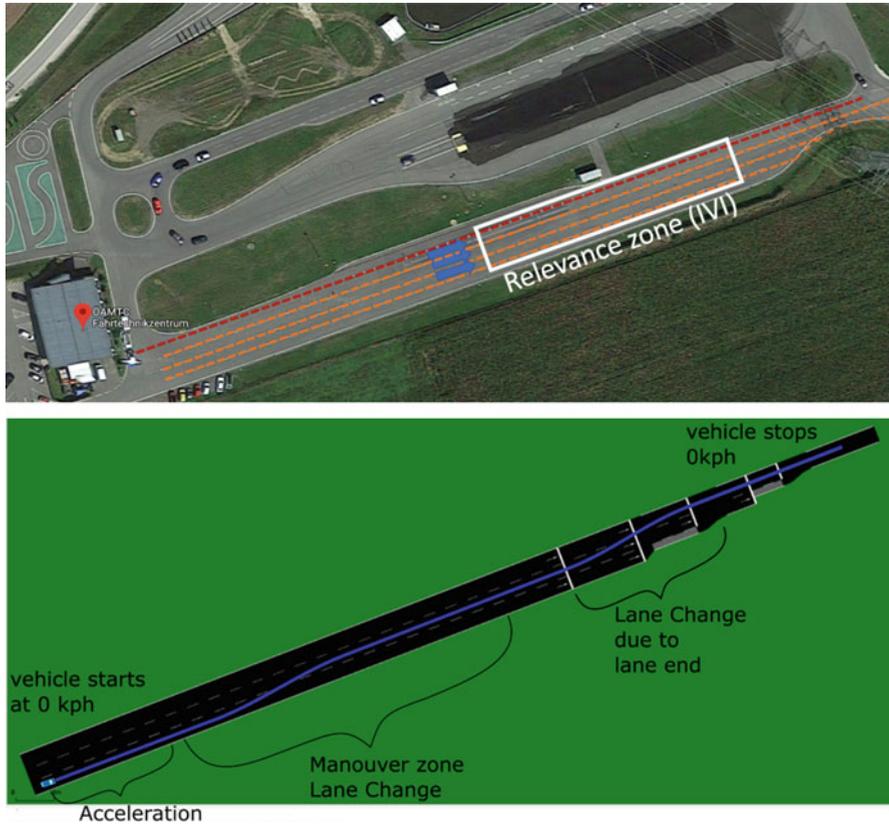


Fig. 21 Maneuver zone satellite picture and SUMO net file

scenarios and the corresponding result as part of this study. In these example test runs, the VuT starts on the second lane (from the right), which corresponds to the first lane of the main road, and receives speed advice and acts accordingly.

### 5.3.1 Scenario I

This scenario shows that the VuT is capable to react accordingly to a speed advice when the VuT is driving on the main road. In this experiment, virtual traffic is not present. The VuT drives initially with a speed of 30 km/h on the main road when it receives a message via the OBU, which was sent from the RSU. In Fig. 22a, we see the desired and the actual velocity history of the ego vehicle during this experiment. We can see that the VuT adapts accordingly to the speed advice of 50 km/h when it reaches to the relevance zone. In Fig. 22b, we show the commended steering angle output from the vehicle as a result of the trajectory planner control function. Here the lane change action is clearly visible.

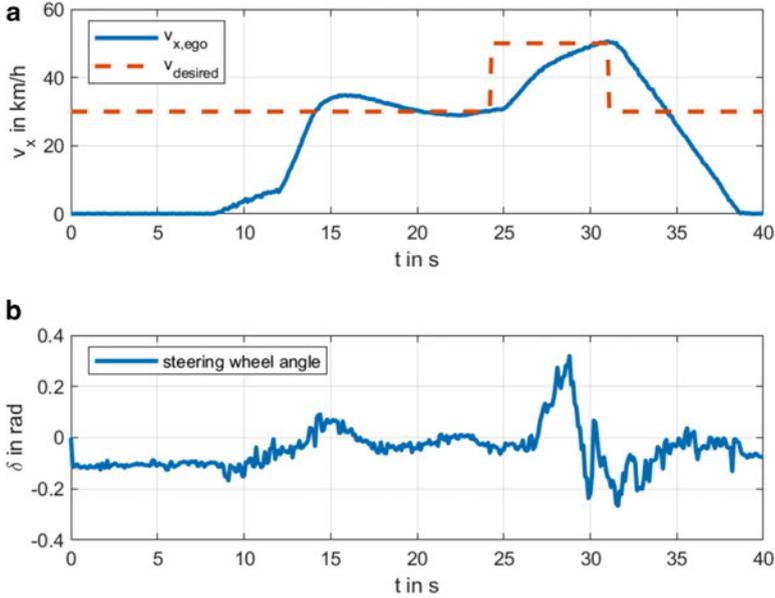
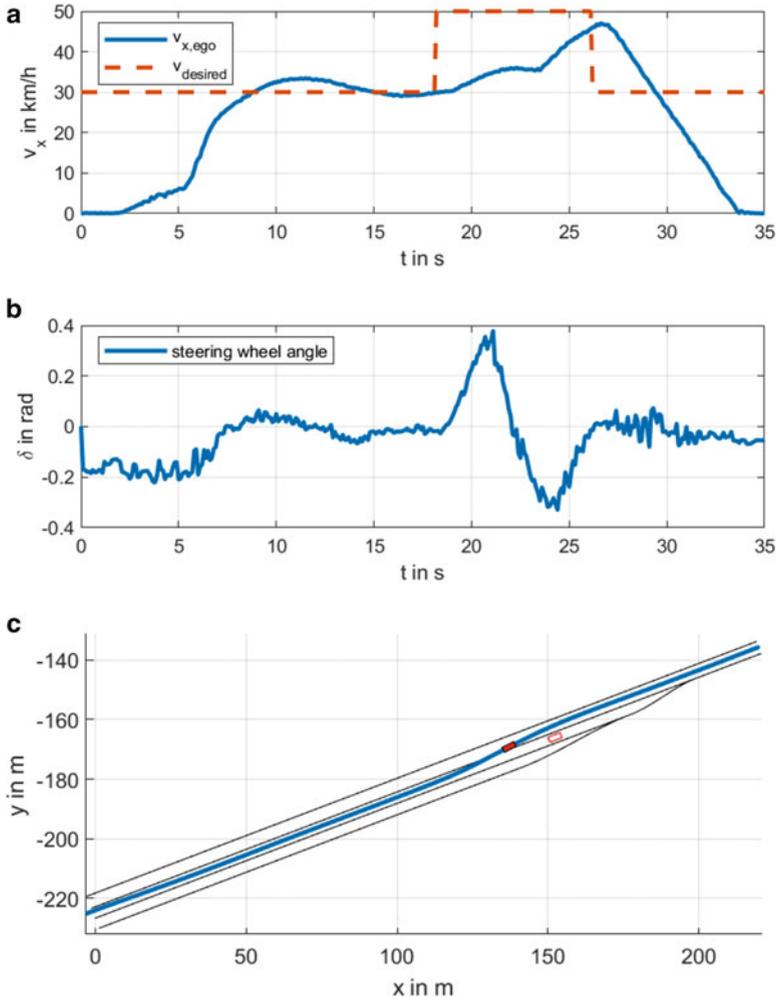


Fig. 22 Vehicle dynamic parameters of scenario I of hybrid testing. (a) Velocity (b) Steering angle

### 5.3.2 Scenario II

The next scenario shows that the VuT can react to a C-ITS message also when traffic is present in the near vicinity. As in the previous scenario, the VuT is driving with a speed of 30 km/h initially on the main road when it receives a message with a speed advice of 50 km/h. But now, a leading vehicle hinders the VuT from reaching the speed advice since it is driving slower. As a result, the VuT was not able to reach the speed advice of 50 km/h, and it adapted its speed to the leading vehicle. Moreover, in Fig. 23b, c, we observe that the VuT conducted an overtaking maneuver and passed the leading target vehicle at the end of the stretch as was expected from the trajectory planning algorithm. The resulting steering angle variation during this experiment is shown in Fig. 23b. Finally, in Fig. 23c, we illustrate the path and a snapshot of the positions of the ego vehicle (VuT) and the leading target vehicle during the test scenario, where the solid rectangle depicts the VuT and the empty rectangle designates the target vehicle.

In the second experiment, the trajectory planner had to take into account the distance to the leading target vehicle and had to adapt the VuT's driving behavior to keep a safe distance. Therefore, further parameters of the trajectory planner were measured during the test and evaluated afterward utilizing the co-simulation framework. Specifically, for the trajectory planner, we evaluated the *minimum distance gap* in meters, the *minimum time gap* in seconds, as well as the *minimum time to collision* in seconds. The results of these parameters are gathered in Table 2.



**Fig. 23** Vehicle dynamic parameters of scenario II of hybrid testing. (a) Commanded and actual velocity variations (b) Measured steering wheel angle (c) Vehicle path along the HD-Map

**Table 2** Distance measurements – scenario II

Minimum distance gap [m]	Minimum time gap [s]	Minimum TTC [s]
11.7	1.17	4.85

Depending on the KPIs for the ADAS function under test, many test metrics and parameters can be calculated and evaluated during hybrid testing to ensure conformity to the performance objectives.

Following the process from simulation to real-world testing presented in Sect. 1, this section described the real-world testing aspects. The following section con-

cludes the outcome from this and all previous sections and provides an outlook to future work.

## 6 Conclusions and Outlook

In this chapter, we presented the implementation details of two novel testing and verification methods for the development and testing of ADAS/AD functions, both utilizing and expanding the idea of combining real-world testing with simulation elements.

The first testing/verification concept involves a steerable rolling dynamometer (DÜRR X-Road-Curve chassis test bench) system together with a vehicle-in-the-loop solution with OTA camera stimulation. It has been demonstrated that the suggested VEHIL solution can potentially be utilized to replace on-road tests and enable repeatable batch scenario testing opportunities. It is also possible to use the proposed solution to check ADAS/AD functions for robustness, integrity, and functional safety, which is normally not easily achievable with traditional road testing method. This test bench solution and the corresponding methodology can also be used for endurance testing of ADAS/AD functions. Therefore, the suggested mixed-reality test bench system can provide an intermediate step in ADAS/AD system development, testing, and validation.

In the second testing/verification method, we presented the generalization of the co-simulation framework together with a vehicle-in-the-loop testing concept and named this “hybrid testing,” where a real vehicle can be combined with virtual ones on a real test track (or proving ground) to evaluate and validate ADAS functions in realistic traffic scenarios. Specifically, we have described the co-simulation framework with all its sub-components, which enables real-time integration between the traffic scenario simulation and the real-life test vehicle along with the ADAS functions running on it.

There are open problems related to the test bench implementation that clearly need further investigation. First of all, there is an observed oscillatory behavior, which is possibly caused by the multiple feedback loops that are an inherent part of the system. These feedback loops exist within the X-Road-Curve chassis dynamometer, the Model.CONNECT<sup>TM</sup> co-simulation platform that interface the environment simulation and vehicle dynamic components, as well as in the intelligent ADAS camera (MobilEye) utilized as part of the driving function. The inherent complex interactions among these multiple dynamic and interlocking loops are an open scientific problem that needs closer investigation. Another open problem is related to the utilized camera stimulation. Many ADAS/AD systems utilize not only cameras but also combinations of other sensors such as radars, lidars, and/or ultrasonic sensors. Stimulation and/or simulation of other sensors in the scope of a consistent testing methodology is another open problem. These are therefore some of the possible directions of research on this specific implementation.

The hybrid testing implementation on the other hand demonstrated that successful and repeatable virtual driving scenarios are achievable on a proving ground for the functional testing of ADAS/AD functions. The suggested testing method can be automated and performed as batch tests, if the effect of various input parameters needs to be evaluated and/or a statistical evaluation is required. Moreover, scenarios involving the interaction of VuT with new road control elements (e.g., ITS-G5, 5G, and/or VMS) and traffic management centers can also be set up and evaluated. Hybrid testing can also be extended, likewise, in various directions. This proof-of-concept implementation utilized only a traffic simulator (SUMO) for the scenario modeling, which as a first step can be improved by the integration of a photo-realistic environment simulator (such as CARLA, Unity 3D, LG Simulator, etc.) to the co-simulation framework. Additionally, it is possible to include physical sensor models to the co-simulation framework for use of the methodology for perception algorithm development. As a further extension, it is possible to include real sensor equipment to the co-simulation framework to perform enhanced reality scenario testing of ADAS/AD functions for standard compliance evaluations and validation. Finally, it is also interesting to compare the same test scenario in simulation, test bench, and hybrid testing evaluations to analyze the quantitative correlation between these testing options. These possible extensions shall be investigated for future work.

**Acknowledgments** This work is partially funded by the INFRAMIX project and was accomplished at the VIRTUAL VEHICLE Research GmbH in Graz, Austria. INFRAMIX has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement no 723016. The authors would like also to acknowledge the partial financial support of the COMET K2-Competence Centers for Excellent Technologies program of the Austrian Federal Ministry for Transport, Innovation and Technology (BMVIT), the Austrian Federal Ministry of Science, Research and Economy (BMWFW), the Austrian Research Promotion Agency (FFG), the Province of Styria, and the Styrian Business Promotion Agency (SFG). Content of the paper reflects only the authors' view. The Innovation and Networks Executive Agency (INEA) is not responsible for any use that may be made of the information it contains. The authors would like to thank all partners within INFRAMIX consortium for their cooperation and valuable contributions. The authors would furthermore like to express their thanks to their supporting industrial project partners, namely, AVL List GmbH and DÜRR Assembly Products GmbH.

## References

1. X-Road-Curve dynamically tests autonomous cars, (2018). <https://www.durr-group.com/en/duerrmore/autonomous-driving/x-road-curve/>, Accessed on 11 Apr 2018
2. T.E. Passenbrunner, M. Sassano, H. Trogmann, L. del Re, M. Paulweber, M. Schmidt, H. Kokal, Inverse torque control of hydrodynamic dynamometers for combustion engine test benches, in *Proceedings of the 2011 American Control Conference*, 2011, pp. 4598–4603
3. AVL-Model.CONNECT™, the neutral model integration and co-simulation platform connecting virtual and real components. 2018. <https://www.avl.com/-/model-connect>. Accessed on 11 Apr 2018

4. MOBILEYE Inc., sensing the future. 2018. <https://www.mobileye.com/>. Accessed on 11 Apr 2018
5. D. Fodor, K. Enisz, Vehicle dynamics based ABS ECU verification on real-time hardware-in-the-loop simulator, in *2014 16th International Power Electronics and Motion Control Conference and Exposition*, Sept 2014, pp. 1247–1251
6. S. Solmaz, F. Holzinger, A novel test bench for development, calibration and functional testing of ADAS/AD functions, in *2019 IEEE International Conference on Connected Vehicles and Expo (ICCVE)*, Nov 2019, pp. 1–8
7. IPG Automotive, *Reference Manual*, 4th edn. (2012)
8. D. Watenig, M. Horn, *Automated Driving. Safer and More Efficient Future Driving* (Springer, 2017)
9. H. Winner, S. Hakuli, F. Lotz, C. Singer, *Handbook of Driver Assistance Systems* (Springer International Publishing Switzerland, 2016)
10. R. Rajamani, *Vehicle Dynamics and Control* (Springer US, 2012)
11. J. Kosecka, R. Blasi, C.J. Taylor, J. Malik, A comparative study of vision-based lateral control strategies for autonomous highway driving, in *Proceedings. 1998 IEEE International Conference on Robotics and Automation (Cat. No.98CH36146)*, vol. 3, May 1998, pp. 1903–1908
12. G. Nestlinger, M. Stolz, Bumpless transfer for convenient lateral car control handover, *IFAC-PapersOnLine*, vol. 49, no. 15, pp. 132–138, 2016. 9th IFAC Symposium on Intelligent Autonomous Vehicles IAV 2016
13. G. Nestlinger, Modellbildung und Simulation eines Spurhalte-Assistenzsystems, Master's thesis, Technische Universität Graz, 2013
14. Dataspeed Inc. We make robots move and cars drive by themselves. <http://dataspeedinc.com/>. Accessed 04 Nov 2018
15. S. Solmaz, G. Nestlinger, G. Stettinger, Improvement of lane keeping assistance ADAS function utilizing a Kalman filter prediction of delayed position states, in *2019 IEEE International Conference on Connected Vehicles and Expo (ICCVE)*, Nov 2019, pp. 1–8
16. M. Paulweber, K. Lebert, *Mess- und Prüfstandstechnik: Antriebsstrangentwicklung · Hybridisierung · Elektrifizierung*. Der Fahrzeugantrieb (Springer Fachmedien Wiesbaden, 2014)
17. H. Beglerovic, S. Metzner, M. Horn, Challenges for the validation and testing of automated driving functions, in *Proceedings of the AMAA 2017 Conference*, 2017
18. T. Hwang, J. Roh, K. Park, J. Hwang, K.H. Lee, K. Lee, S. Lee, Y. Kim, Development of HILS systems for active brake control systems, in *2006 SICE-ICASE International Joint Conference*, Oct 2006, pp. 4404–4408
19. J.D. Moore, G.M. Molen, SIL, HIL, and vehicle fuel economy analysis of a pre-transmission parallel PHEV, in *2013 World Electric Vehicle Symposium and Exhibition (EVS27)*, pp. 1–7, Nov 2013
20. L. Verhoeff, D.J. Verburg, H.A. Lupker, L.J.J. Kusters, Vehil: a full-scale test methodology for intelligent transport systems, vehicles and subsystems, in *Proceedings of the IEEE Intelligent Vehicles Symposium 2000 (Cat. No.00TH8511)*, pp. 369–375, Oct 2000
21. D.J. Verburg, A.C.M. van der Knaap, J. Ploeg, Vehil: developing and testing intelligent vehicles, in *Intelligent Vehicle Symposium, 2002. IEEE*, vol. 2, pp. 537–544, June 2002
22. O.J. Gietelink, D.J. Verburg, K. Labibes, A.F. Oostendorp, Pre-crash system validation with PRESCAN and VEHL, in *IEEE Intelligent Vehicles Symposium, 2004*, pp. 913–918, June 2004
23. J. Ploeg, F.M. Hendriks, N.J. Schouten, Towards nondestructive testing of pre-crash systems in a HIL setup, in *2008 IEEE Intelligent Vehicles Symposium*, pp. 91–96, June 2008
24. F. Hendriks, M. Tideman, R. Pelders, R. Bours, X. Liu, Development tools for active safety systems: PRESCAN and VEHL, in *Proceedings of 2010 IEEE International Conference on Vehicular Electronics and Safety*, pp. 54–58, July 2010
25. C. Schyr, A. Brissardt, Drivingcube – a novel concept for validation of powertrain and steering systems with automated driving, in *Proceedings of the 13<sup>th</sup> International Symposium on Advanced Vehicle Control AVEC2016*, Sept 2016

26. VIRTUAL VEHICLE automated drive demonstrator (ADD), an open research platform for automated driving, 2018. <https://www.v2c2.at/automated-driving/>. Accessed on 11 Apr 2018
27. T. Weck, J. Neumann, How to calibrate and test advanced driver assistance and autonomous driving systems EoL, in *Proceedings of the 19. VDI-Kongress SIMVEC – Simulation und Erprobung in der Fahrzeugentwicklung*, 2018
28. DSPACE Inc., Microautobox 2 Kompaktes, robustes Prototyping-System für den Fahrzeugeinsatz. 2018. <https://www.dspace.com/>. Accessed on 11 Apr 2018
29. O. Gietelink, J. Ploeg, B.D. Schutte, M. Verhaegen, Development of advanced driver assistance systems with vehicle hardware-in-the-loop simulations. *Veh. Syst. Dyn.* **44**, 569–590 (2006)
30. VTD – Vires Virtual Test Drive, 2018. <https://vires.com/>. Accessed on 11 Apr 2018
31. T. Bock, M. Maurer, F. van Meel, T. Müller, Vehicle in the Loop, Ein innovativer Ansatz zur Kopplung virtueller mit realer Erprobung, *ATZ – Automobiltechnische Zeitschrift*, vol. 110, pp. 10–16, Jan 2008
32. S.A. Fayazi, A. Vahidi, Vehicle-in-the-loop (vil) verification of a smart city intersection control scheme for autonomous vehicles, in *2017 IEEE Conference on Control Technology and Applications (CCTA)*, pp. 1575–1580, Aug 2017
33. H. Németh, A. Hányi, Z. Szalay, V. Tihanyi, B. Tóth, Proving ground test scenarios in mixed virtual and real environment for highly automated driving, in *Mobilität in Zeiten der Veränderung*, ed. by H. Proff (Springer Gabler, Wiesbaden, 2019), pp. 200–2009
34. VIL-Systeme: Einfach reale mit virtuellen Fahrversuchen verbinden, 2019. <https://ipg-automotive.com/de/produkte-services/test-systems/vil-systeme/>. Accessed on 23 Aug 2019
35. T. Kim (Hyundai MOBIS), Vehicle-in-the-loop (Vil): Validating driver assistance systems with synchronous virtual and real test drives, *dSpace Magazin*, vol. 2, 2019
36. U. Eberle, S. Berger, S. Hallerbach, Erprobung eines Fahrzeugs mit simulierten Verkehrsteilnehmern. Patent, Oct 2019. Patentnummer: DE102018003130A1
37. S. Hallerbach, Y. Xia, U. Eberle, F. Koester, Simulation-based identification of critical scenarios for cooperative and automated vehicles. *SAE Int. J. CAV* **1**(2), 93–106 (2018)
38. D. Augustin, J. Schucker, J. Tschirner, M. Hofmann, U. Konigorski, A simulation-based reinforcement learning approach for long-term maneuver planning in highway traffic scenarios, in *VDI Berichte Nr. 2349*, pp. 47–60, 2019
39. S. Hallerbach, U. Eberle, Simulation-based testing of cooperative and automated vehicles. Conference-Presentation at KO-HAF final event, 09 2018. Accessed 21 Jan 2020
40. Horizon 2020-INFRAMIX, Preparing road infrastructure for mixed vehicle traffic flows. <https://www.inframix.eu/>. Accessed 10 Jan 2020
41. G. Stettinger, M. Benedikt, M. Horn, J. Zehetner, C. Giebenhain, Control of a magnetic levitation system with communication imperfections: A model-based coupling approach. *Control. Eng. Pract.* **58**, 161–170 (2017)
42. G. Stettinger, M. Horn, M. Benedikt, J. Zehetner, A model-based approach for prediction-based interconnection of dynamic systems, in *53rd IEEE Conference on Decision and Control*, pp. 3286–3291, Dec 2014
43. M. Tranninger, G. Stettinger, M. Benedikt, M. Horn, Diagnosis of interconnected systems via well tuned model-based coupling algorithms, *IFAC-PapersOnLine*, vol. 51, no. 24, pp. 1271–1277, 2018, in *10th IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes SAFEPROCESS 2018*
44. G. Stettinger, M. Horn, M. Benedikt, J. Zehetner, Model-based coupling approach for non-iterative real-time co-simulation, *2014 European Control Conference (ECC)*, pp. 2084–2089, 2014
45. P.A. Lopez, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y. Flötteröd, R. Hilbrich, L. Lücken, J. Rummel, P. Wagner, E. Wiessner, Microscopic traffic simulation using sumo, in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pp. 2575–2582, Nov 2018
46. Siemens Mobility GmbH, Sitraffic Vehicle2x. <https://new.siemens.com/global/de/produkte/mobilitaet/strassenverkehr/connected-mobility-solutions/sitraffic-vehicle2x.html>. Accessed 23 Jan 2020