Driving Simulation Technologies for Sensor Simulation in SIL and HIL Environments

Conference Paper · September 2018

6 authors, including:

Gregor Sievers
dSPACE GmbH

Michael Peperhowe
dSPACE

Herbert Hanselmann
dSPACE GmbH

Driving Simulation Technologies for Sensor Simulation in SIL and HIL Environments

Dr. Gregor Sievers[1], Caius Seiger[1], Michael Peperhowe[1], Holger Krumm[1], Dr. Sebastian Graf[1]

(1) dSPACE GmbH, Rathenaustraße 26, 33102 Paderborn, Germany
e-mail: {GSievers, CSeiger, MPeperhowe, HKrumm, SGraf}@dspace.de

Abstract – The market introduction of autonomous vehicles is on the horizon. To fulfil the requirements of functional safety for autonomous driving (ISO 26262) the autonomous car needs to be test-driven by means of hundreds of millions of kilometers. To overcome the burdens of real test drives on public roads according to safety, costs, and time to market, virtual test methodologies can offer a way-out. Virtual driving tests can be divided in Software-In-The-Loop (SIL) and Hardware-In-The-Loop (HIL) simulation. This work presents a unified toolchain for SIL and HIL validation of autonomous driving functions. This tool chain allows for multiple injection options (e.g., object list, raw data, and over the air) for testing all different level of a sensor's signal processing chain. The work is finally concluded by one example test system for camera raw data injection.

Keywords: *Autonomous Driving, Environment Simulation, Software-In-The-Loop (SIL), Hardware-In-The-Loop (HIL)*

Introduction

Developing functions for autonomous cars requires detailed and thorough testing to meet the requirements of functional safety ISO 26262 [ISO11, Bir13]. While test drives allow for a high level of test coverage, they result in an increased risk for the test driver and high costs per test kilometer. Due to the requirements of the ISO 26262 a huge amount of test kilometer is required. At this not only the amount of kilometer is of interest, but also the identification and run of representative, relevant, and "critical" kilometers.

On this occasion driving tests in the lab can significantly help. On the one hand driven test kilometer can be standardized, on the other hand costs can enormously be reduced with respect to 24-7 testing. In principle virtual test drives can be divided into two stages.

Software-In-The-Loop (SIL) is the first stage, focusing on the validation of an algorithm without respect to timing constraints or bus systems. The tests can be executed before a hardware prototype exists.

Second stage is Hardware-In-The-Loop (HIL) where the final hardware platform (Electronic Control Unit, ECU) is connected to a simulator, which mimics the behavior of a real car. Thus, the whole hardware together with the electrical interfaces and software is fully tested.

Due to autonomous driving more and more functions and ECUs are integrated in a car. Hence without the use of virtual test methods, testing will become a barely manageable task.

Sensor Interfaces

A crucial element of ADAS (Advanced Driver Assistance Systems) and autonomous driving (AD) is a reliable and robust environmental perception by means of camera, RADAR, and LIDAR sensors. Autonomous cars are going to integrate up to two dozen of those sensors. For development and validation of sensor based algorithms it is necessary to generate synthetic sensor data for the driving simulation. Different options exist to inject sensor data within a sensor-based ECU (c.f. Figure 1). These insertion options are explained in further detail based upon a camera sensor. However, the same principle is applicable to RADAR and LIDAR.

'Over-The-Air' (OTA) stimulation is the classical approach for injecting output of a driving simulation into a camera-based ECU (c.f. Figure 1, Option 4). Therefore, the environment simulation is rendered by a 3D engine and visualized by a monitor. This is used as an input for the camera ECU by placing it in front of a monitor. This is a fair approach if the visualization quality is close enough to reality. The advantages are that the complete processing chain is tested and the optics can be analyzed as well. Limitations of this approach exist specifically with respect to the dynamic range (HDR) and a multi lens setups such in a stereo or triple lens camera. These drawbacks require additional methods to inject synthetic data into ECUs to validate and develop automatic driving algorithms.

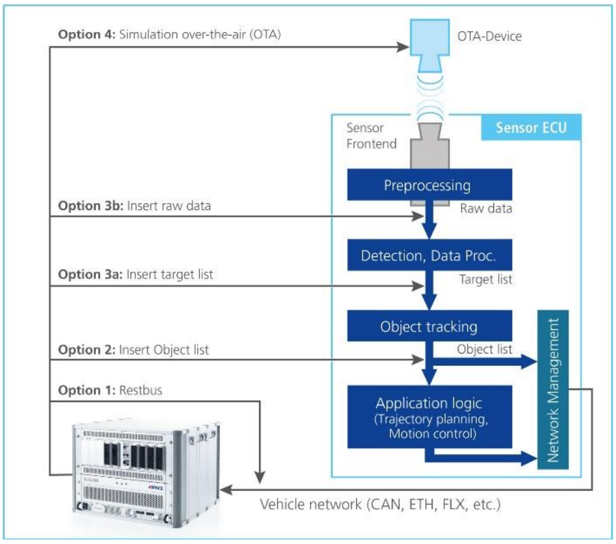

Figure 1. Validation of a sensor ECU can be done via different options of abstraction.

Removing the camera's optics and imager chip allows for the injection of sensor raw data (in form of ADC data) by the driving simulator (c.f. Figure 1, Option 3b). This extends the simulation capabilities since the raw data can be generated for each sensor setup individually. This includes infrared and stereo camera approaches, fisheye lenses as well as HDR cameras.

Another testing approach is the injection of target lists into the ECU (c.f. Figure 1, Option 3a). In terms of cameras, this is for example a detected target right before the tracking algorithm.

The remaining Options 1 and 2 are sensor independent. The Ground Truth information of the driving simulation can be used to provide detected object lists (c.f. Figure 1, Option 2), which can be injected into the ECUs Application Logic. Within Option 1, the whole ECU is bypassed since the HIL simulator is publishing results from the simulation directly to the automotive bus system. This is a cost efficient solution to test the application logic like trajectory planning or motion control of the car.

Sensor simulation requires a high performance PC for both HIL and SIL. In the SIL use case it is needed to calculate sensor data faster than real-time to test millions of test kilometers as fast as possible. However, in the HIL use case new data samples have to arrive within hard real-time constraints.

Scenario-Based Testing

As mentioned before, functions for autonomous driving have to be test-driven by means of hundreds of millions of kilometers. Here, it's not only a question of testing the ADAS and AD algorithm itself, but also taking the formerly introduced different sensor levels into account.

To overcome this challenge, the process is needed to be split up into pieces. The overall ADAS and AD workflow can be classified as "see", "think" and "act", i.e. recognizing the environment, coming to the right decision and then control the system.

These abstract levels of classification can be mapped onto concrete real representatives, like the different sensor types as camera, RADAR and LIDAR systems. The ADAS and AD algorithm can be seen as the "brain" of the process that results in a motion planning for the vehicle under test. To control the system in longitudinal and lateral directions, the actuator side, like brake and steering systems, come into play.

Test scenarios within the ADAS and AD field focus not only on the pure application or central algorithm. Rather, the perception and processing functions of the sensors have to be taken into the loop as well. There are different options for virtual test levels, which can be separated into open- and closed-loop tests from an abstract perspective. In open-loop tests, the device under test (DUT) can be stimulated with data taken from real measurements in form of data replay. The benefit of this test variant is that the DUT gets real inputs with all effects, like environmental conditions and real existing sceneries. The disadvantage is, that the ego vehicle, that is equipped with the ADAS/AD algorithm, cannot freely move in the virtual environment, as the stimulus data is defined for a fixed trajectory only. To overcome this limitation, a closed-loop and scenario-based test methodology can be established (c.f. Figure 2).

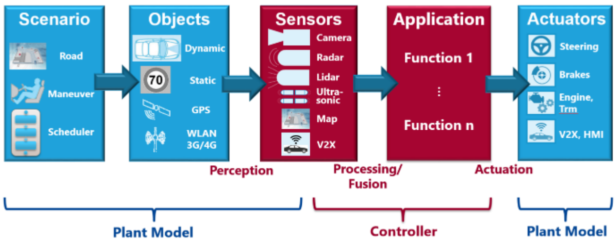

Figure 2. ADAS/AD process

The sensors are stimulated by means of virtual objects, for example dynamics objects (e.g., other vehicles), or static ones (e.g., traffic signs, trees, buildings). In addition, for eHorizon and V2X applications, GPS signals as well as Wifi and mobile communication signals have to be emulated. The core elements from the scenario itself are complex road networks, the maneuver definition for the ego vehicle, and the trajectory planning. The latter includes the scheduler, for the other participants, like moving fellows.

To close the loop and take the algorithm output signals into account, the different actuators have to be modelled as well. Thus, active steer assists, automatic emergency braking, economic driving and V2X communication can be tested and allow for an arbitrary number of variants. Thus, not only functional tests are possible, but complete validation routines are covered to ensure safety critical functions in a wide range of scenarios.

For an efficient and easy-to-adopt scenario-based test process, the underlying plant model must be highly flexible and executable on different platforms. As one example, the open and modular model structure of the dSPACE Automotive Simulation Models (ASM) fulfil these requirements. Scenarios can be setup through graphical user interfaces, like dSPACE ModelDesk, and fully be automated to cover a wide range of variants. Via compiling for different platforms, the continuous reuse for SIL and HIL applications can be guaranteed, as well.

Sensor Simulation

As mentioned before, one core component to be tested and validated is the sensor side, especially the related perception, image processing and object detection. Depending on the available interfaces (c.f. Figure 1) and test focus, the sensor simulation can be divided into different levels (c.f. Figure 3).

Starting with the Ground Truth (GT) sensor models that run on the CPU, the sensor object list is generated and can be input directly into the ADAS and AD algorithm (Option 2). To cover all kinds of real sensors, like RADAR and LIDAR systems, generic sensor model are used, like the traffic module from the ASM tool suite. These models can run in hard real-time, e.g. for HIL use-cases, are scalable to a huge amount of sensors and inject all data synchronously to the different sensor types.

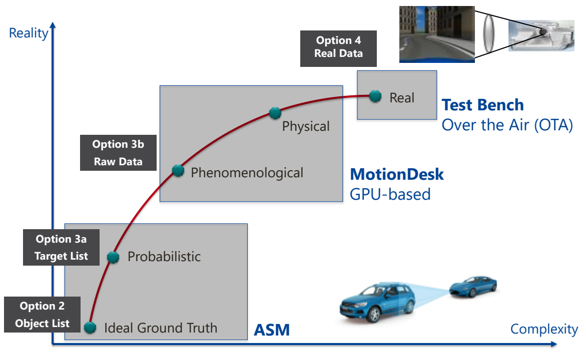

Figure 3. Sensor Simulation Levels

Based on the ideal object list, realistic effects, like environmental conditions as fog and rain, and multiple targets, can be superimposed. These probabilistic sensor models can be used to address the target list interface (Option 3a).

dSPACE MotionDesk can be used to visualize and animate the ASM ground truth data. This animated scenery is used for raw data (Option 3b) and Over-The-Air' stimulation (Option 4).

To meet the high data rates and challeging synchronization requirements of raw data stimulation, the dSPACE Environment Sensor Interface Unit (ESI Unit) can be used. This FPGA-based Unit uses the GPU-based data as an input and injects the raw data via various sensor-specific interfaces.

Software in the Loop (SIL)

Testing a sensor-based ECU can be executed on a common PC. Although timing effects based on bus systems are not taken into account, this approach should not be neglected. This chapter looks at the benefits of a SIL toolchain.

Validation of software for autonomous driving reaches a high complexity. A lot of different scenarios have to be tested virtually before bringing it on a real road. In fact, millions of test kilometers have to be driven for a proper validation [Kru16]. Neither real test drives nor HIL systems are suitable for this, since testing in real time would take far too long. Especially within a development process with daily updates, testing in real time is not possible. Within the SIL tool chain, the software that should actually be executed on the target hardware is simulated on a standard PC. The software can be integrated within the offline simulator dSPACE VEOS.

To test the chain of effects as early as possible, raw sensor data can be generated on a GPU and injected into the algorithm. However, both target lists and object lists can be generated, depending on the level of validation of the software. Of course there is no possibility to test the sensor over the air because the actual sensor frontend is not available (c.f. Figure 1, Option 4).

Hardware in the Loop (HIL)

This chapter presents two HIL testing methods as examples for the toolchain discussed in this work. For raw data injection (Option 3b), a camera test system is presented whereas a RADAR testbed explains the over the air (Option 4) with closed loop simulation.

In case of a faulty HIL test run, the test setup will be saved and can be replayed on the HIL testbed for further analyzation. Due to the unified dSPACE toolchain for SIL and HIL testing, the test can be easily replayed by the algorithm developer using the dSPACE SIL platform on his own PC.

Camera Raw Data Injection

Figure 4 shows an example setup for closed-loop HIL simulation and raw data injection (Option 3b) for a camera sensor. The setup described in this chapter is used to inject data into the first digital interface of the camera processing chain. This means that the camera image sensor (including the lens) is replaced and simulated by the presented HIL environment (c.f. Figure 5). All components of the simulation setup are optimized for a low end-to-end latency so the control algorithms within the ECU can be tested properly.

The state of the driving simulation is sent to the Environment Sensor Simulation PC. This PC runs dSPACE MotionDesk and uses a powerful GPU to render the complete scenery including the vehicle under test, the environment, weather conditions and other cars. The different sensors of the car can be configured in detail like, e.g., the mounting position, resolution, the simulated lens, and the field of view.

The simulated output of all sensors is merged into one DisplayPort or HDMI output and sent to the Environment Sensor Interface Unit (ESI Unit). This FPGA-based platform converts the DisplayPort data into the format required by the camera ECU. Typically, the ECUs require a dedicated raw data format (e.g., Bayer pattern) or color encoding (e.g., YUV). In addition, the ESI Unit guarantees synchronization of all sensors.

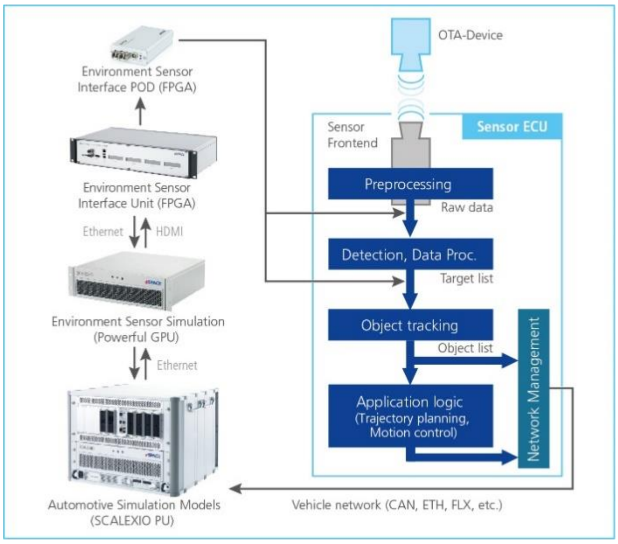

Figure 4. Closed-loop camera HIL simulation setup for Option 3b

For the transfer of sensor raw data in a car no interface standard has been established till now. Thus a test system for sensor raw data injection has to support a high number of sensor interfaces. The ESI Unit is highly modular and supports all relevant interfaces. Certain interfaces can be realized via plug-in modules (long range interfaces, e.g., Ethernet, TI FPD-Link III, Maxim GMSL). For short range interfaces like MIPI CSI2 or HiSPI, an optional plug-on device (ESI-POD) is used. The ESI-POD can be directly attached to the component under test.

Independent of the sensor interface, the ESI Unit has to simulate the I2C control channel that is used by the ECU to configure the imager chip (c.f. Figure 5).

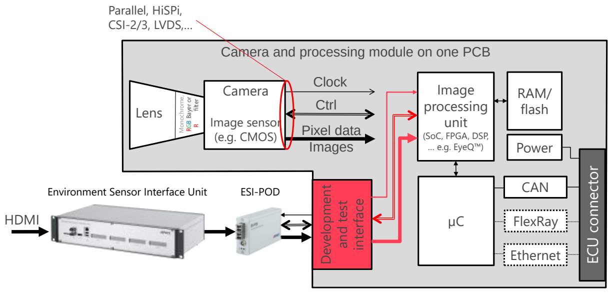

Figure 5. Camera raw data injection (Option 3b) for a short range video interface

In addition to the presented camera raw data setup, the integration of LIDAR and RADAR sensors is under development.

Conclusion

This work presents a powerful integrated and unified toolchain for SIL and HIL testing of autonomous vehicles. Different options for sensor testing in terms of injection points of the sensor ECU are described. The options can be combined for different sensors. An example setup may combine a RADAR tested over the air (Option 4) and a camera raw data injection (Option 3b). The toolchain allows for high productivity for the validation of sensor based ECUs during all stages of the development and testing process.

Acknowledgment

This work has been partly funded by the BMBF project SafeMove.

References

[ISO11] **ISO Road vehicles -- Functional safety. (ISO 26262)**, *ISO*, Geneva, Switzerland (2011).

[Bir13] Birch, John, et al. **Safety cases and their role in ISO 26262 functional safety assessment**, *International Conference on Computer Safety, Reliability, and Security*. Springer, Berlin, Heidelberg, 2013.

[Kru16] Krügel, K. and Jochheim, J. **Wie das Hochautomatisierte Fahren die Absicherung von Steuergeräten ändert,** *AutoTest Technical Conference on 'Test of Hardware and Software in Automotive Development'*, 2016.

[Gra17] Graf, S. **OTA RADAR Test for Autonomous Driving,** *20th European Microwave Week (EuMW)*, Nürnberg, Germany, 2017.

[Gra18] Graf, S. and Rožmann M. **Closed-Loop and Real-Time Testing of RADAR Sensors For Autonomous Driving Based On A HiL System,** *International Conference on Microwaves for Intelligent Mobility (ICMIM)*, Munich, Germany, 2018.