Delft University of Technology

Delft Center for Systems and Control

Technical report 05-009

Development of advanced driver assistance systems with vehicle hardware-in-the-loop simulations*

O. Gietelink, J. Ploeg, B. De Schutter, and M. Verhaegen

If you want to cite this report, please use the following reference instead:
O. Gietelink, J. Ploeg, B. De Schutter, and M. Verhaegen, "Development of advanced driver assistance systems with vehicle hardware-in-the-loop simulations," *Vehicle System Dynamics*, vol. 44, no. 7, pp. 569–590, July 2006.

Delft Center for Systems and Control
Delft University of Technology
Mekelweg 2, 2628 CD Delft
The Netherlands
phone: +31-15-278.24.73 (secretary)
URL: `https://www.dcsc.tudelft.nl`

*This report can also be downloaded via `https://pub.deschutter.info/abs/05_009.html`

Preprint for Vehicle System Dynamics
Vol. 00, No. 00, Month 200x, 1–23

Development of advanced driver assistance systems with vehicle hardware-in-the-loop simulations

OLAF GIETELINK[†,‡,*], JEROEN PLOEG[†], BART DE SCHUTTER[‡] and MICHEL VERHAEGEN[‡]

[†]TNO Automotive, P.O. Box 756, 5700 AT Helmond, The Netherlands

[‡]Delft University of Technology, Delft Center for Systems and Control, Mekelweg 2, 2628 CD Delft, The Netherlands

(*Received 00 Month 200x; In final form 00 Month 200x*)

This paper presents a new method for the design and validation of advanced driver assistance systems (ADASs). With *vehicle hardware-in-the-loop* (VEHIL) simulations the development process, and more specifically the validation phase, of intelligent vehicles is carried out safer, cheaper, and more manageable. In the VEHIL laboratory a full-scale ADAS-equipped vehicle is set up in a hardware-in-the-loop simulation environment, where a chassis dynamometer is used to emulate the road interaction and robot vehicles to represent other traffic. In this controlled environment the performance and dependability of an ADAS is tested to great accuracy and reliability. The working principle and the added value of VEHIL are demonstrated with test results of an adaptive cruise control and a forward collision warning system. Based on the 'V' diagram, the position of VEHIL in the development process of ADASs is illustrated.

Keywords: advanced driver assistance systems, hardware-in-the-loop simulation, controller design and validation, adaptive cruise control, forward collision warning

1 Introduction

Every year in Europe alone, more than 40 000 casualties and 1.4 million injuries are caused by vehicle-related accidents [1]. Although advances in *passive safety*, as illustrated in figure 1, have made passenger cars ever safer, the safety potential of further improvements in passive safety features is limited. However, *active safety* systems like ABS [2] and ESP [3] offer possibilities for improving traffic safety by assisting the driver in his driving task. In addition, advanced driver assistance systems (ADASs) have the potential to significantly reduce the number of road accidents. An ADAS is a vehicle control system that uses environment sensors (e.g. radar, laser, vision) to improve driving comfort and traffic safety by assisting the driver in recognising and reacting to potentially dangerous traffic situations. Since an ADAS can even autonomously intervene, an ADAS-equipped vehicle is popularly referred to as an 'intelligent vehicle'. As explained in more detail in several surveys [4–7], the following types of intelligent vehicle systems can be distinguished:

- *Driver information* systems increase the driver's situation awareness, e.g. advanced route navigation systems [11].
- *Driver warning* systems actively warn the driver of a potential danger, e.g. lane departure warning, blind spot warning, and forward collision warning (FCW) systems [12]. This

[*]Corresponding author. Email: olaf.gietelink@tno.nl

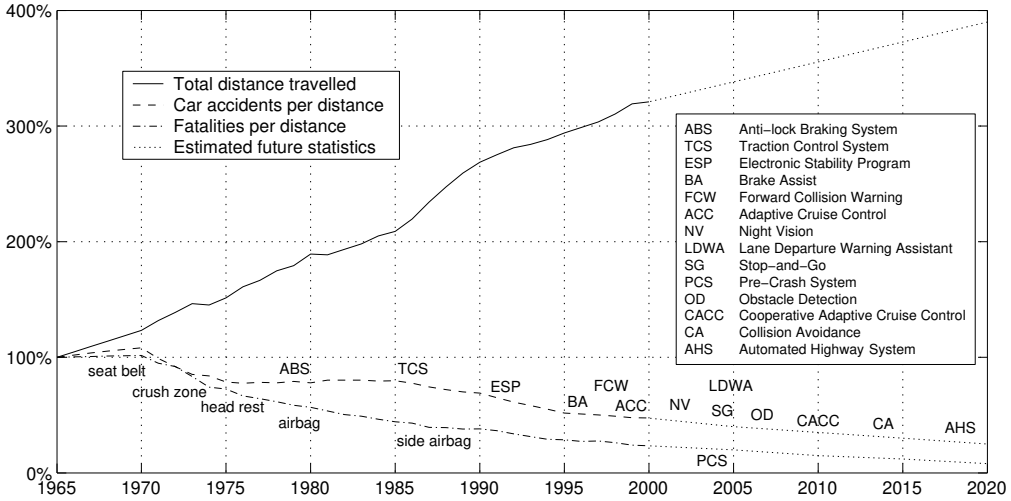


Figure 1. Total number of road accidents and fatalities per total distance travelled, normalised on 1965 data for the EU [1]. In addition, the graph shows when passive safety systems (which reduce fatalities in case of an accident) and active safety systems (which assist in avoiding an accident) have been introduced, as well as the expected safety potential of ADASs [8–10].

warning then allows the driver to take appropriate corrective actions in order to mitigate or completely avoid the event.

- *Intervening* systems provide active support to the driver, e.g. an adaptive cruise control (ACC) system [13]. ACC is a comfort system that maintains a set cruise control velocity, unless an environment sensor detects a slower vehicle ahead. The ACC then controls the vehicle to follow the slower vehicle at a safe distance, see figure 2. ACC is intended for speeds above 30 km/h, but is currently being extended to a stop-and-go application for automated longitudinal control in low-speed complex environments, such as traffic jams and urban areas [14].
- *Integrated passive and active safety systems*. In addition to passive safety systems that are activated *during* the crash, a pre-crash system can mitigate the crash severity by deploying active and passive safety measures *before* a collision occurs [15]. Pre-crash safety measures, such as brake assist and seat belt pre-tensioners, have recently been introduced on the market [16].
- *Fully automated* systems are the next step beyond driver assistance, and operate without a human driver in the control loop. Automated highway systems, using fully automated passenger cars, are expected to significantly benefit traffic safety and throughput, but are not considered for short-term introduction [17].

According to several surveys ADASs can prevent up to 40 % of traffic accidents, depending on the type of ADAS and the type of accident scenario [8–10]. Despite this safety potential, market penetration of ADASs has gone slow. Main challenges in this respect are customer acceptance and understanding of the added value, liability exposure, and regulatory issues [6, 18]. Drivers also expect an ADAS to meet high requirements in terms of (subjective) performance, reliability (low rate of false alarms), and safety (low rate of missed detections). Therefore, the ADAS must be tested for the wide variety of complex traffic situations that the system should be able to recognise and handle [19]. Unfortunately, exhaustive testing of an ADAS prototype is usually impossible due to constraints in costs and time-to-market. Not only the design, but especially the validation of ADASs, thus requires a growing effort in the development process. To address these issues, efficient

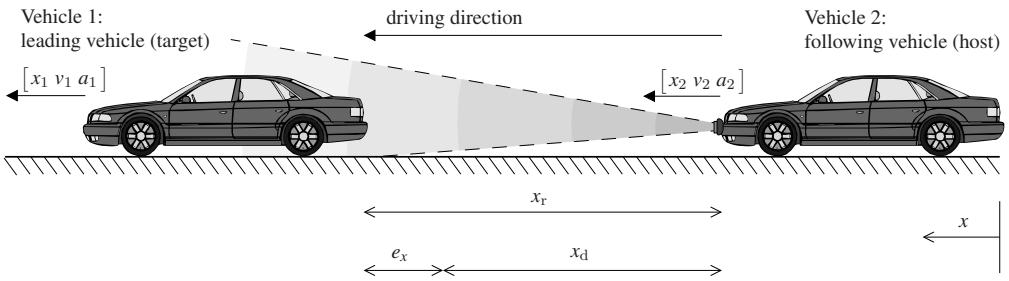


Figure 2. Schematic representation of an adaptive cruise control (ACC) system that controls vehicle 2 to follow the leading vehicle 1 with equal velocity at a desired safe distance x_d. Further defined are the position x, velocity v, and acceleration a of both vehicles, the relative velocity $v_\mathrm{r} = v_1 - v_2$, the clearance $x_\mathrm{r} = x_1 - x_2$ (neglecting the length of the vehicles), and the separation error $e_x = x_\mathrm{d} - x_\mathrm{r}$, all in longitudinal direction.

methods are required for the design of ADAS controllers and the validation of their safety and performance.

The objective of this paper is to present a new method for the development of ADASs that complements the existing development process. This method consists of vehicle hardware-in-the-loop (VEHIL) simulations that allow to efficiently and accurately test full-scale ADAS-equipped vehicles in an indoor laboratory environment.

The remainder of this paper is organised as follows. The problem statement is further defined in section 2 by reviewing the development process of ADASs and state-of-the-art test methods. In section 3 we then present the working principle and added value of the VEHIL concept, and discuss the position of the VEHIL laboratory in the ADAS development process. This is demonstrated in section 4, where VEHIL test results for ACC and FCW are presented. Finally, section 5 presents the conclusions, and discusses ongoing research activities. Lists of frequently used symbols and abbreviations are also included at the end of this paper.

2 Tools in the design and validation process

In the automotive industry the different phases in the development process of safety-critical control systems are often connected using the 'V' diagram, depicted in figure 3 [20]. The 'V' diagram uses a 'top-down' approach to *design* and a 'bottom-up' approach to *validation*, although in practice the process does not strictly follow all phases in this sequence and goes through several iteration loops. The 'V' diagram is frequently applied to the development process of mechatronic vehicle systems [21]. However, the various development phases for ADASs face some specific challenges.

2.1 *Challenges in the ADAS development process*

The ADAS development starts with a definition of the *functional requirements* in terms of the desired functions, driver comfort, and operational constraints. In addition, ADASs are *safety-critical* systems that require a high level of *dependability*, a term covering reliability, (fail-)safety, and fault-tolerance. Hazard and risk analyses are therefore performed to identify the *safety requirements*, usually in terms of the rate of false alarms (when an ADAS takes unnecessary action) and missed detections (when it fails to correctly detect a dangerous situation). State-of-the-art systems achieve a false alarm rate in the order of 10^{-5} per km, but this is still considered too high [12]. From the functional and safety requirements a

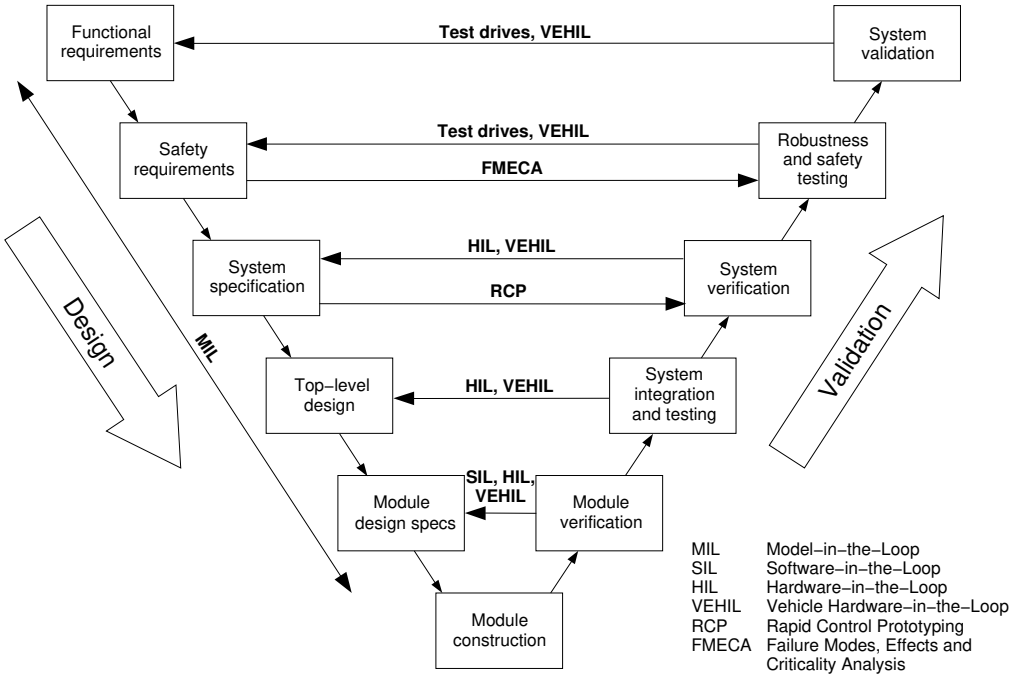


Figure 3. The 'V' diagram represents the sequential design and validation phases in the development of automotive safety-critical systems, including the use of various test tools in these phases.

system specification is produced to define the precise operation of the system. However, in practice requirements are often difficult to define and subject to ambiguity, which may lead to an incomplete or incorrect specification.

Subsequently, the system specification is used as the basis for the top-level design of the system architecture, followed by detailed module design (environment sensor, controller, actuator, driver interface). After implementation of the individual hardware and software modules, system integration takes place by assembling the complete system from its component modules. In every integration phase *verification* takes place to determine whether the output of a phase meets its specification, as illustrated by the horizontal arrows in figure 3. On component level this means testing the range, accuracy, and tracking capabilities of the environment sensor [22]. On a higher level, verification must assure that integration with other subsystems does not have any negative side-effect.

Since verification only confirms compliance with the specification, errors in the specification may result in a faulty product. It is therefore important to perform *validation* of the integrated system against its requirements, especially for type approval and certification purposes. Usually, the development process involves several iterations, where the results of verification and validation are used to modify the system specification and design, after which another test cycle takes place. Obviously, there is a need to reduce the number of iterations and speed up the process of verification and validation. Because of the need for fast, flexible and reproducible test results, various 'in-the-loop' simulation tools are increasingly being used for design and validation of ADAS controllers, as indicated in figure 3. After a review of these tools, the position of the new VEHIL simulation tool in this development process will be clarified in section 3.

2.2 *Model-in-the-loop simulations*

The initial design of the ADAS controller is supported by so-called *model-in-the-loop* (MIL) simulations, where the controller logic is simulated in closed-loop with models of vehicle dynamics, sensors, actuators, and the traffic environment. Unfortunately, current simulation tools lack the possibility for testing the complete ADAS in a reliable way with full integration of operating conditions, sensor characteristics, vehicle dynamics, and complex traffic scenarios. The new simulation concept PRESCAN was therefore developed in [23]. PRESCAN allows reliable MIL simulation of ADASs, using validated physical sensor models for radar, lidar, and camera vision in a virtual environment. The simulation of traffic scenarios is based on a multi-agent approach, as will be explained in section 3.

2.3 *Hardware-in-the-loop simulations*

When MIL simulations have provided sufficient results, software code can be compiled from the simulation model of the control system. The real code can then be verified with software-in-the-loop (SIL) simulations, where the remaining hardware components, vehicle dynamics, and environment are simulated in real-time.

Similar to testing the real software in a SIL simulation, the real hardware can be tested in a real-time hardware-in-the-loop (HIL) simulation. HIL simulations consist of a combination of *simulated* and *real* components, see figure 4. Alternatively, a real component can be *emulated*, i.e. replaced by an *artificial* component that has the same input and output characteristics. Ideally, every component should be unable to distinguish between real, simulated or emulated components that it is connected to in the closed-loop configuration. Therefore, HIL offers the flexibility of a simulation, where the use of real hardware offers a high level of reliability.

The main advantage of a HIL simulation is that it provides a repeatable laboratory environment for safe, flexible, and reliable controller validation. Controller performance and stability can be systematically tested without disturbances from other unrelated systems, and dependability can be tested by controlled injection of disturbances and faults. HIL also allows validation of the real hardware in an early development phase without the need for a prototype vehicle, since any missing vehicle components can be simulated. For these reasons, HIL simulations are more efficient and cheaper than test drives, and are extensively used for the development of vehicle control systems, such as ABS [24], engine control systems [25], and semi-active suspension systems [26]. ADASs can also be tested in several HIL configurations, as discussed next.

As indicated in figure 3, in an early stage *rapid control prototyping* is carried out with emulated control functions. This involves implementing a model of the desired controller in a prototype vehicle for the purpose of rapid proof-of-concept, controller testing, and parameter adjustments. Next, the hardware controller can be tested in a HIL simulation for its real-time behaviour [27]. This limited HIL setup can gradually be extended to include other modules, as the integration of the vehicle progresses. For instance, ADAS controllers can be tested in a HIL simulation with real actuators [27] and real sensors [28], where all other components are simulated. However, a complex interface between the *simulated* environment and the *real* sensor is necessary to generate a sensor signal. Yet another type of HIL simulation is a driving simulator, which creates an artificial environment for an 'in-the-loop' human driver [29]. Driving simulators are useful for subjective evaluation of the ADAS and for fine-tuning ADAS controller settings.

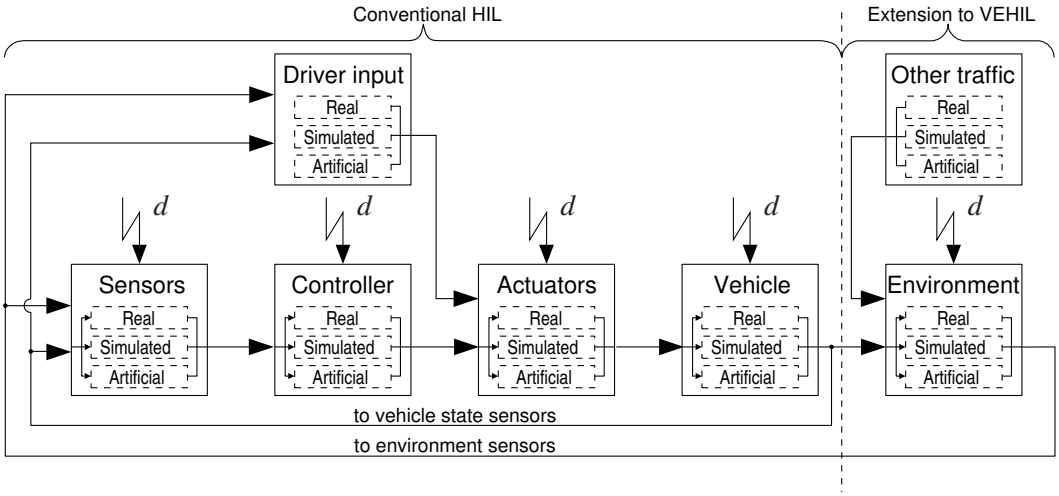


Figure 4. Possible configurations for HIL simulations, where parts of the system may be real, simulated or artificial. Feedback of signals from environment sensors and vehicle state sensors provides a closed-loop simulation. Additional disturbances d can be injected by the operator to test the system's dependability.

Finally, the complete system can be real, including sensor, controller, actuator and vehicle dynamics. This complete vehicle system is in interaction with the road surface (through its actuators), as well as with the traffic environment that is formed by other objects in the world (through its sensors). Since environment sensors should receive a real input, an *artificial* traffic environment must be created to test an ADAS-equipped vehicle in a HIL simulation. Up to now, no such HIL environment has been available for testing complete intelligent vehicles.

2.4 *Test drives*

Test drives with prototype vehicles are always the final link in the validation chain to evaluate the system's performance in the real world environment that it will finally be used in. However, the value of test drives for control system design is limited, because test results are hard to reproduce and often inaccurate, due to the lack of 'ground truth' knowledge on the exact state (e.g. obstacle position) of the vehicles involved in the test. In addition, these tests are often expensive, unsafe, time consuming, and heavily dependent on weather conditions [19, 27]. In the next section we therefore propose a solution to combine the advantages of HIL simulations with the representativeness of test drives, by extending the HIL environment from vehicle level to the traffic level, as indicated in figure 4.

3 Vehicle hardware-in-the-loop (VEHIL) simulations

To address the challenges mentioned in the previous section, we present a new method for the design and validation of intelligent vehicle systems: vehicle hardware-in-the-loop (VEHIL) simulations. VEHIL provides a solution for testing a full-scale intelligent vehicle in a HIL environment [30]. The VEHIL concept was first described in [31], patented in [32], and some preliminary test results have been presented in [33–36]. This paper presents the VEHIL working principle in more detail and discusses the added value and position in the ADAS development process based on new test results.

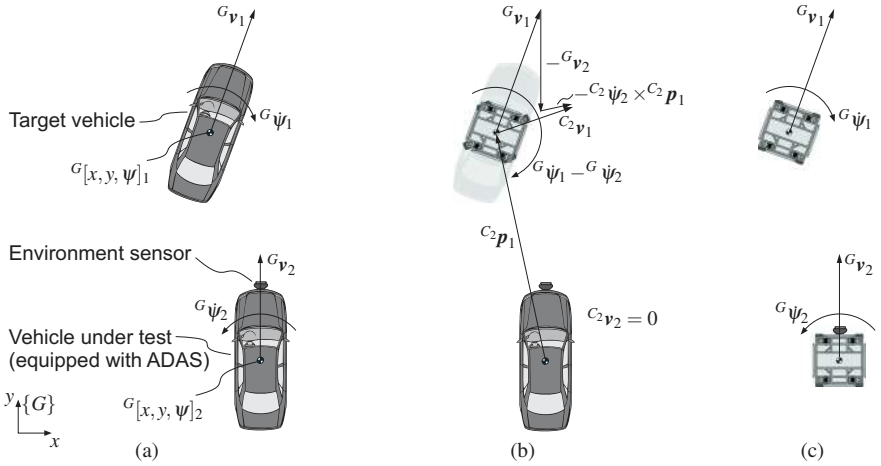


Figure 5. Transformation of coordinate frames: (a) absolute motion in real world; (b) relative motion in VEHIL; and (c) absolute motion with two moving bases in VEHIL.

3.1 *Working principle of the VEHIL simulation*

VEHIL constitutes a multi-agent simulator for intelligent vehicle systems, in which some of the simulated vehicles are replaced by real vehicles. These vehicles operate in an indoor laboratory that forms an artificial HIL environment for the intelligent vehicle. The environment sensors that are used in ADASs (radar, laser, vision), collect relative position data in the absolute traffic environment. VEHIL therefore makes a transformation from the *absolute* motion of the objects in a traffic scenario to *relative* motion between those objects, as illustrated in figures 5(a) and (b). Using only the relative motion between a fixed intelligent vehicle and target vehicles allows to have a controlled and space-efficient environment.

The software architecture of VEHIL is based on a multi-agent real-time simulator (MARS) [37], as illustrated in the lower-left part of figure 6. This multi-agent framework consists of a collection of autonomous *entities E* (vehicles, other road users, or any other dynamical component), each controlled by its own internal dynamics (e.g. a vehicle model, as discussed in section 3.2). An entity has an absolute state $\boldsymbol{x}$ in the global coordinate frame $\{G\}$, notated as

$$ {}^{G}\boldsymbol{x} = \left[\boldsymbol{p}^T\ \Phi^T\ \boldsymbol{v}^T\ \dot{\Phi}^T\ \boldsymbol{a}^T\ \ddot{\Phi}^T\right]^T, \tag{1} $$

where ${}^{G}\boldsymbol{p} = \left[x\ y\ z\right]^T$ represents the position and ${}^{G}\Phi = \left[\varphi\ \theta\ \psi\right]^T$ the orientation in Euler angles (roll, pitch, and yaw) of the entity. The corresponding velocity and acceleration components are represented by ${}^{G}\boldsymbol{v} = \left[\dot{x}\ \dot{y}\ \dot{z}\right]^T$, ${}^{G}\dot{\Phi} = \left[\dot{\varphi}\ \dot{\theta}\ \dot{\psi}\right]^T$, ${}^{G}\boldsymbol{a} = \left[\ddot{x}\ \ddot{y}\ \ddot{z}\right]^T$, and ${}^{G}\ddot{\Phi} = \left[\ddot{\varphi}\ \ddot{\theta}\ \ddot{\psi}\right]^T$.

Furthermore, a *virtual world* is defined that serves as a formal representation of the environment relevant to these entities. Entities are typically represented in the virtual world by *objects O* that interact with other objects (vehicles, bicyclists, pedestrians, infrastructure, traffic lights). Objects are not simulation models, but are merely the virtual representation of the simulation entities. A visual representation of this virtual world is shown in figure 7. After every integration time step of this multi-agent simulation, the internal dynamics of an entity (e.g. E_2, representing vehicle 2) result in a state $\boldsymbol{x}_2$ in its *local* coordinate frame

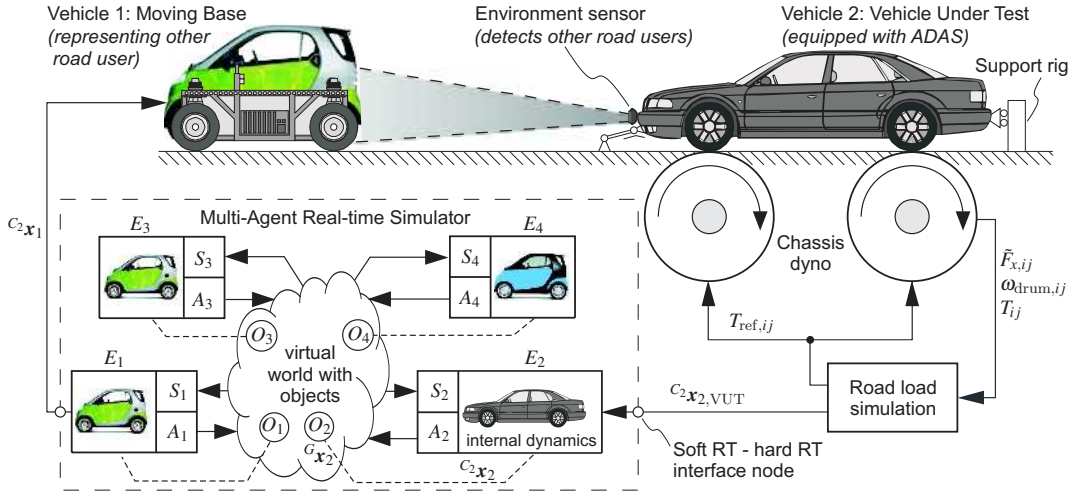


Figure 6. Schematic representation of the VEHIL closed-loop working principle. Every integration time step the simulation loop runs clockwise via the VUT, the chassis dyno, the MARS, and the MB, whose motion is detected by the VUT's sensor.

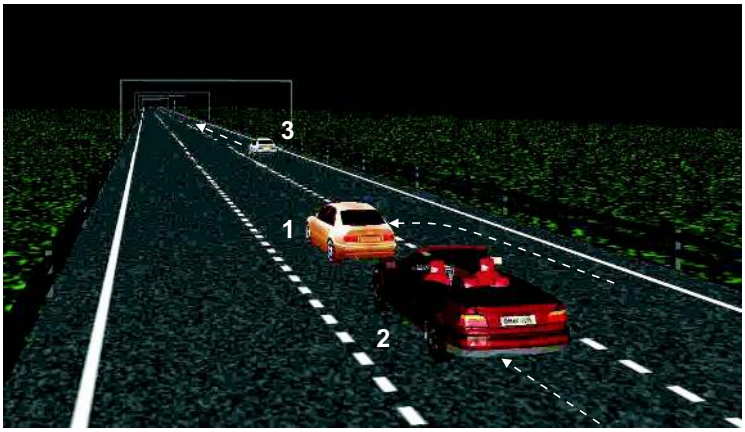


Figure 7. Visual representation of a cut-in scenario in the virtual world: the ACC-equipped vehicle (2) drives on the middle lane when suddenly a vehicle (1) cuts in from the right lane at a lower speed.

$\{C_2\}$, notated as ${}^{C_2}\boldsymbol{x}_2$. Through the link between the simulation entity E in its local frame and its virtual object O in the global frame, the entity updates the state ${}^{G}\boldsymbol{x}_2$ of the associated object O_2 in the *global* frame of the virtual world $\{G\}$. The link between entity and object is indicated by the dashed lines in figure 6.

An important feature of the MARS modelling concept is that an entity (e.g. a vehicle model) uses abstract sensors S and actuators A to interface with other objects in the virtual world. Through its abstract sensor S_2 the entity E_2 can collect information about the state ${}^{G}\boldsymbol{x}_1$ of another object O_1 (e.g. vehicle 1, associated with E_1) in the virtual world. Vice versa, the entity has an abstract actuator A_2 to change the state ${}^{G}\boldsymbol{x}_1$ of O_1. Note that these sensors and actuators are handled in an *abstract* way: they have no dynamics and data processing features. Instead they can be interpreted as queries and actions on the virtual world. Real sensors and actuators are model led as part of the entity's internal dynamics [23].

Using this simulation principle, the relative motion between vehicles 1 and 2 (entities E_1 and E_2) from the viewpoint of vehicle 2 is obtained by a coordinate transformation, where the state of vehicle 1 ${}^{G}\boldsymbol{x}_1$ is represented in the coordinate frame $\{C_2\}$ of vehicle 2, i.e. ${}^{C_2}\boldsymbol{x}_1$.

For the transformation to relative position and orientation, we then get

$$ {}^{C_2}\boldsymbol{p}_1 = {}^{C_2}_{G}\mathbf{R}({}^{G}\boldsymbol{p}_1 - {}^{G}\boldsymbol{p}_2) \tag{2} $$

$$ {}^{C_2}\boldsymbol{q}_1 = {}^{C_2}_{G}\boldsymbol{q}\,{}^{G}\boldsymbol{q}_1, \tag{3} $$

where ${}^{C_2}_{G}\mathbf{R}$ is the rotation matrix from frame $\{G\}$ to $\{C_2\}$ and $\boldsymbol{q}$ represents the orientation in Euler parameters [38]. If we neglect the vertical vehicle dynamics (z, φ, θ) and only consider relative motion in the horizontal plane (x, y, ψ) the coordinate transformation in (2) and (3) simplifies to

$$ {}^{C_2}\begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} \cos{}^{G}\psi_2 & \sin{}^{G}\psi_2 \\ -\sin{}^{G}\psi_2 & \cos{}^{G}\psi_2 \end{bmatrix} \left({}^{G}\begin{bmatrix} x_1 \\ y_1 \end{bmatrix} - {}^{G}\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} \right) \tag{4} $$

$$ {}^{C_2}\psi_1 = {}^{G}\psi_1 - {}^{G}\psi_2. \tag{5} $$

Please refer to figure 5 for a visual representation of this transformation. In a similar way the transformations to relative velocity $({}^{C_2}\boldsymbol{v}_1,\ {}^{C_2}\dot{\psi}_1)$ and relative acceleration $({}^{C_2}\boldsymbol{a}_1,\ {}^{C_2}\ddot{\psi}_1)$ are derived [38]. For brevity, these derivations are omitted here, since the use of non-constant transformation matrices and the hierarchical frame system becomes very complex.

The simulation is run by execution of entities on computing nodes, which are connected via a local area network. Each node has its own runtime environment, which also contains a representation of the virtual world. Entities communicate with this virtual world via its abstract sensors and actuators. The 'engine' of the entity simulation is an integrator (numerical solver), which invokes the entity's code (i.e. the vehicle model) in timely manner (synchronised with other entities in real-time). The implementation of the system architecture is Java based with time-critical parts in C/C++, but an interface is established to MATLAB/Simulink: C code compiled from Simulink models can be embedded into the runtime environment as entities. More details on this modelling concept and the runtime environment are described in [37].

3.2 *Vehicle modelling*

The multi-agent simulator provides the framework, in which *any* type of vehicle model can be simulated. The model complexity depends on the type of ADAS and the objective of the simulation. In case of an often used two-track model, the equations of motion are [39]:

$$ m^C\ddot{x} - m^C\dot{\psi}^C\dot{y} = \sum F_x - F_{\mathrm{air},x} - F_{\mathrm{grav},x} \tag{6} $$

$$ m^C\ddot{y} + m^C\dot{\psi}^C\dot{x} = \sum F_y \tag{7} $$

$$ J_z^C\ddot{\psi} = \sum M_z, \tag{8} $$

where m and J_z are the vehicle mass and inertia, $\sum F_x$, $\sum F_y$, and $\sum M_z$ are the combined tire forces and moments, and $F_{\mathrm{air},x}$ and $F_{\mathrm{grav},x}$ are the longitudinal components of the air and gravitational resistance forces.

These equations can be solved by numerical integration, using the following calculation sequence. The current velocities ${}^C\dot{x}$, ${}^C\dot{y}$ and yaw rate ${}^C\dot{\psi}$ in the vehicle frame $\{C\}$, the wheel angular velocities ω_{ij}, and the vertical tire forces $F_{z,ij}$ are used as initial conditions.

The velocity components at each wheel location are then

$$v_{\mathrm{long},ij} = {}^{C}\dot{x} - {}^{C}\dot{\psi} s_i \tag{9}$$

$$v_{\mathrm{lat},ij} = {}^{C}\dot{y} + {}^{C}\dot{\psi} l_i, \tag{10}$$

with $i \in \{1,2\}$ indicating the front and rear axle, and $j \in \{\mathrm{L},\mathrm{R}\}$ the left and right wheel, respectively, and l_i and s_i being the longitudinal and lateral distance from the vehicle centre of gravity. With a steer angle δ_{ij} the velocities in the wheel frame $v_{\mathrm{wheel},x,ij}$ and $v_{\mathrm{wheel},y,ij}$ are then

$$\begin{bmatrix} v_{\mathrm{wheel},x,ij} \\ v_{\mathrm{wheel},y,ij} \end{bmatrix} = \begin{bmatrix} \cos\delta_{ij} & \sin\delta_{ij} \\ -\sin\delta_{ij} & \cos\delta_{ij} \end{bmatrix} \begin{bmatrix} v_{\mathrm{long},ij} \\ v_{\mathrm{lat},ij} \end{bmatrix}. \tag{11}$$

For a tyre with radius r, the tyre slip angles κ and α are then given by

$$\kappa_{ij} = -\frac{v_{\mathrm{wheel},x,ij} - \omega_{ij} r}{v_{\mathrm{wheel},x,ij}} \tag{12}$$

$$\alpha_{ij} = \tan^{-1}\left(\frac{v_{\mathrm{wheel},y,ij}}{v_{\mathrm{wheel},x,ij}}\right). \tag{13}$$

For slip angles κ_{ij} and α_{ij}, camber angle γ_{ij}, and $F_{z,ij}$, the Magic Formula [39] gives

$$F_{x,ij}, F_{y,ij}, M_{z,ij} = \mathrm{MagicFormula}\left(\kappa_{ij}, \alpha_{ij}, \gamma_{ij}, F_{z,ij}\right). \tag{14}$$

These tyre forces and moments for each wheel are then transformed to the chassis frame by

$$\begin{bmatrix} F_{\mathrm{chassis},x,ij} \\ F_{\mathrm{chassis},y,ij} \end{bmatrix} = \begin{bmatrix} \cos\delta_{ij} & -\sin\delta_{ij} \\ \sin\delta_{ij} & \cos\delta_{ij} \end{bmatrix} \begin{bmatrix} F_{x,ij} \\ F_{y,ij} \end{bmatrix}. \tag{15}$$

Introducing a wheel inertia J_{wheel}, drive torque T_{drive}, brake torque T_{brake}, and rolling resistance coefficient f_{r}, the motion equation for the wheel is obtained as

$$J_{\mathrm{wheel}}\dot{\omega}_{ij} = T_{\mathrm{drive},ij} - T_{\mathrm{brake},ij} - F_{x,ij} r - F_{z,ij} f_{\mathrm{r}} r, \tag{16}$$

which gives ω_{ij} for the next integration time step. Equations (6)-(8) can then be solved by summing the chassis forces and moments:

$$\sum F_x = \sum_{i,j} F_{\mathrm{chassis},x,ij} \tag{17}$$

$$\sum F_y = \sum_{i,j} F_{\mathrm{chassis},y,ij} \tag{18}$$

$$\sum M_z = \sum_{i,j} \left(-F_{\mathrm{chassis},x,ij} s_i + F_{\mathrm{chassis},y,ij} l_i + M_{z,ij}\right). \tag{19}$$

The vertical tyre forces for the next integration time step can then be found by

$$F_{z,1L} = \frac{l_2}{2l}mg + \frac{\left(F_{\text{chassis},y,1L} + F_{\text{chassis},y,1R}\right)h_1}{2s_1} - \frac{h}{2l}m\ddot{x} \tag{20}$$

$$F_{z,1R} = \frac{l_2}{2l}mg - \frac{\left(F_{\text{chassis},y,1L} + F_{\text{chassis},y,1R}\right)h_1}{2s_1} - \frac{h}{2l}m\ddot{x} \tag{21}$$

$$F_{z,2L} = \frac{l_1}{2l}mg + \frac{\left(F_{\text{chassis},y,2L} + F_{\text{chassis},y,2R}\right)h_2}{2s_2} + \frac{h}{2l}m\ddot{x} \tag{22}$$

$$F_{z,2R} = \frac{l_2}{2l}mg - \frac{\left(F_{\text{chassis},y,2L} + F_{\text{chassis},y,2R}\right)h_2}{2s_2} + \frac{h}{2l}m\ddot{x}, \tag{23}$$

where the first term is the static load distribution, the second term the load transfer due to roll, and the third term the load transfer due to acceleration (neglecting suspension characteristics). Here, l is the wheelbase, h is the height of the centre of gravity, and h_i the height of the roll centre at the i-th axle.

The driver input ($T_{\text{drive}}(t)$, $T_{\text{brake}}(t)$, and $\delta_{ij}(t)$) to the vehicle model defines the traffic scenario that is simulated as a function of time t. The VEHIL scenario library contains a database of traffic scenarios, such as following, tailgating, cut-ins, lane changes, collisions, and near-miss scenarios, created from in-depth accident analysis [35]. The PRESCAN simulation tool, briefly mentioned in section 2.2, is used for scenario definition and simulation before the actual VEHIL test takes place, based on the same multi-agent approach. Alternatively, predefined trajectories (e.g. for benchmark and certification tests) or recorded test drives can be accurately reproduced in VEHIL.

3.3 *Substitution of the vehicle dynamics model by the vehicle under test*

With the ADAS-equipped vehicle and other road users modelled, the real-time simulation could run as a MIL simulation only, i.e. a PRESCAN simulation without hardware. However, (6)-(23) are usually not sufficient to accurately model the ADAS-equipped vehicle. In order to test a real intelligent vehicle in a HIL configuration, the vehicle model of entity E_2 is substituted by the real vehicle under test (VUT), hence the term 'vehicle hardware in-the-loop'. The ADAS-equipped VUT is therefore placed on a chassis dynamometer that provides a realistic load for the vehicle's actuators (engine, brake system) and sensors (e.g. wheel speed sensors).

The dynamic response of the chassis dyno, depicted in figure 8, to driving actions of the VUT must be representative of real road conditions, especially in terms of delay time and phase lag. The operating frequency of the MARS is 100 Hz, which means that the delay time is an acceptable 10 ms. The yaw response to steer input $\dot{\psi}/\delta$ and velocity response to throttle/brake input $\dot{x}/(T_{\text{drive}} + T_{\text{brake}})$ of a passenger vehicle typically show a bandwidth in the 1 Hz frequency range. This implies that the chassis dyno must at least have a bandwidth of about 5 Hz in order to minimise positioning phase lag. Furthermore, an emergency stop of a passenger vehicle can cause a maximum deceleration of around 10 m/s^2. Consequently, the chassis dyno must be able to achieve this as well.

These real-time requirements are met by a setup with four individual electric motor driven drums. The chassis dyno can fully simulate a vehicle mass between 500 and 3500 kg up to a maximum velocity of 250 km/h. The adjustable wheelbase accommodates a wide range

Figure 8. Vehicle under test with radar sensor at the front bumper, driving on the chassis dyno in VEHIL and supported by a rig at the front and back bumper.

Table 1. Specifications of the chassis dyno.

Wheelbase	1.8 – 4.0 m
Track width	1.2 – 2.4 m
Drum configuration	4-wheel independent drive
Total peak power	832 kW
Traction force	24 kN
Response time	< 10 ms
Drum diameter	1592 mm
Maximum velocity	250 km/h
Dynamic range passenger cars (500 – 3500 kg)	full dynamics, ± 10 m/s^2
Dynamic range commercial vehicles ($\leq$12 000 kg)	reduced dynamics

of vehicle types: apart from passenger vehicles, also trucks, buses, and other automated guided vehicles can be tested. Table 1 summarises the main specifications.

Note that the VUT itself replaces the vehicle model of (6)-(23). The chassis dyno only needs to emulate the tyre forces $F_{x,ij}$ that the VUT would encounter on the road. Each $F_{x,ij}$ is emulated by the drum inertia J_{drum} and the electric motor torque T_{ij} as

$$F_{x,ij} = \frac{c_0 + c_1\omega_{\mathrm{drum},ij} + c_2\omega_{\mathrm{drum},ij}^2 + J_{\mathrm{drum}}\dot{\omega}_{\mathrm{drum},ij} - T_{ij}}{r_{\mathrm{drum}}}, \tag{24}$$

where the first three terms in the numerator represent friction losses in the chassis dyno, $\omega_{\mathrm{drum},ij}$ is the measured drum speed, and r_{drum} the drum radius. From (24) the reference signals for the necessary motor torque are then calculated as

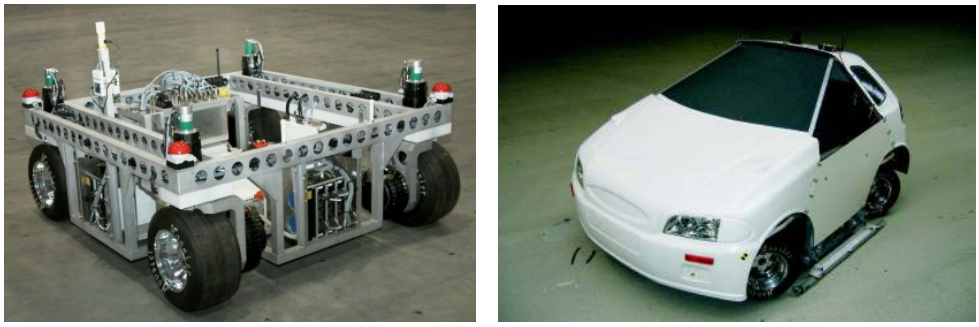
$$T_{\mathrm{ref},ij} = J_{\mathrm{drum}}\dot{\omega}_{\mathrm{drum},ij} + c_0 + c_1\omega_{\mathrm{drum},ij} + c_2\omega_{\mathrm{drum},ij}^2 - \tilde{F}_{x,ij}r_{\mathrm{drum}}, \tag{25}$$

where $\tilde{F}_{x,ij}$ are observer estimated tyre forces.

This setup also emulates the correct correlation between the individual drum speeds

$$\omega_{\mathrm{drum},ij} = \frac{v_{\mathrm{wheel},x,ij}}{r_{\mathrm{drum}}}, \tag{26}$$

to enable simulation of different wheel speeds when driving through curves, where $v_{\mathrm{wheel},x,ij}$ are calculated from (9)-(11). In addition, a special restraint system that keeps the



(a) (b)

Figure 9. Moving base: (a) without body and (b) with body.

vehicle on top of the drums allows realistic heave and pitch motions of the vehicle body, as shown in figures 8 and 10. This rig produces a realistic dynamic vertical load transfer between rear and front axle during braking and accelerating, in accordance with (20)-(23).

Finally, a road load simulation model estimates the VUT state vector ${}^{C_2}\boldsymbol{x}_{2,\mathrm{vut}}$ using the chassis dyno measurements and updates the state ${}^{G}\boldsymbol{x}_2$ of the associated object in the virtual world. No further interfacing between the real VUT and the simulation environment is necessary, such that the VUT can be tested as a black box system in a genuine HIL setup.

3.4 *Substitution of a simulated target by a moving base*

Similar to incorporation of the real VUT in a HIL simulation, surrounding road users can be represented by a so-called moving base (MB), depicted in figure 9(a). The MB is a 4-wheel driven, 4-wheel steered robot vehicle that responds to position commands of the MARS and emulates the motion ${}^{C_2}\boldsymbol{x}_1$ of other road users *relative* to the VUT, such that this motion is detected by the VUT's environment sensor. For this purpose, the soft real-time simulator (Ethernet network) and the hard real-time VUT and MBs (CAN bus) are linked through dedicated interfaces, indicated in figure 6. In order to carry out the desired relative manoeuvres, the MB must be able to perform motions that are not possible with a standard car (e.g. sideways), as illustrated by the resulting velocity vector ${}^{C_2}\boldsymbol{v}_1$ in figure 5(b). For this reason the individual wheels can be steered in a range of -350° to $+350^\circ$.

Like the chassis dyno, the MB should also have a control bandwidth of about 5 Hz in order to minimise positioning phase lag. In addition, the MB should be capable of accelerating with 10 m/s^2 in order to emulate the relative motion resulting from an emergency stop of the VUT. Finally, the top speed, which in view of the relative VEHIL world corresponds to the maximum relative velocity, should at least be equal to 50 km/h. This covers about 95 % of all collision scenarios [15].

These requirements are met by a vehicle platform equipped with independent all-wheel steering and all-wheel drive, using battery-powered DC servomotors. The trajectory controller of the MB realises the desired trajectory $\boldsymbol{x}_{\mathrm{mb,ref}}(t)$, defined by the relative motion ${}^{C_2}\boldsymbol{x}_1$ of the target vehicle in the horizontal plane. The only condition is that ${}^{C_2}\boldsymbol{x}_1$ fits within the dimensions of the VEHIL laboratory (200 m by 40 m) and meets the specifications of table 2. The MB controller determines the drive torques and steering angles so as to minimise the difference between the actual and desired MB position, such that a reproducible trajectory is achieved within an accuracy of 0.10 m. The MB navigation system uses a com-

Table 2. Specifications of a moving base.

Vehicle mass (including body)	650 kg
Wheelbase	1.4 m
Track width	1.4 m
Chassis configuration	4-wheel independent drive/steer from -350° to $+350^\circ$
Maximum velocity	50 km/h
Maximum longitudinal acceleration	10 m/s^2
Maximum longitudinal deceleration	-10 m/s^2
Acceleration from 0 to 50 km/h	2.1 s
Maximum centripetal acceleration	12 m/s^2
Installed power	52 kW
Battery pack	288 NiMH D-cells, 375 V, 100 kg

bination of magnet grid and odometry with a measurement accuracy of 0.04 m, resulting in a total positioning accuracy of 0.10 ± 0.04 m. Table 2 gives some further specifications. For more information on the design and control of the MB, the reader is referred to [40].

In order for the VUT to obtain realistic sensor data, the MB is equipped with a vehicle body that represents similar target characteristics as a real vehicle, see figure 9(b). Its radar cross section is similar to that of a standard passenger car, and the body has a similar shape and reflection properties for testing vision and laser systems.

Subsequently, the ADAS controller receives realistic input signals through its vehicle state sensors and environment sensors, and outputs command signals to the vehicle actuators (engine, brake) with a realistic actuator load, just as if the VUT was driving on the road. It must be emphasised that the actual MB motion in VEHIL is *not* known a priori, but is the real-time equivalent of the resulting relative motion between an autonomously simulated target vehicle and an ADAS-controlled VUT. For example, when the VUT makes an emergency stop with deceleration $a_{2,\mathrm{vut}}$, the MB accelerates forward with $a_{1,\mathrm{mb}} = -a_{2,\mathrm{vut}}$. In this way a *closed-loop* HIL simulation is obtained, such that the ADAS is validated in an artificial traffic environment, including real vehicle dynamics and real sensor input.

3.5 *Added value of VEHIL in the development process of ADASs*

By providing a world-wide unique HIL environment for intelligent vehicle systems, the VEHIL laboratory offers a number of distinct advantages:

- Tests are performed in a reproducible and flexible way with high accuracy, since the MBs are operated from a computer-controlled environment.
- The HIL setup allows precise and repeatable variation of test parameters to assess the influence of specific parameters and failure modes on the ADAS performance.
- Tests are safer, due to the absence of high absolute velocities. Furthermore, traffic scenarios are monitored by a supervisory safety system, which prevents any real collisions. This allows to test ADASs in safety-critical (and even pre-crash) scenarios.
- The costs of the validation process are reduced, because many tests are performed in a short time frame with a high success rate. The VUT can drive for hours and be continuously tested, which is not possible during test drives. Depending on the complexity of the scenarios, on average 15 tests per hour can be performed, including scenario compilation, trial runs, test execution, and data acquisition. A test cycle is therefore significantly faster than is possible with test drives [19].

Because of these advantages, VEHIL complements the existing development process of

ADASs in many phases of figure 3:

- Rapid control prototyping in VEHIL can help to define system specifications in an early development stage. In addition, based on safety-critical manoeuvres and fault injection, potential hazards can be analysed.
- The flexible transition from MIL simulation in PRESCAN to HIL simulation in VEHIL allows a model-based development of the controller. Critical scenarios that are identified with MIL simulations can be quickly uploaded in VEHIL for experimental testing. Test results can then be compared to the simulation results for model validation.
- On module level the ability to combine high position accuracy with high and accurate relative speeds makes VEHIL an efficient tool in verification and benchmarking of the exact performance of environment sensors (e.g. sensor calibration).
- On system level VEHIL especially facilitates the *functional* validation of the performance and dependability of complex black-box controllers against objective measures. Algorithm evaluation and fine-tuning can be done efficiently.
- For production sign-off and certification purposes the high reproducibility and ability to deal with safety-critical applications make VEHIL a strong tool.
- Finally, VEHIL facilitates the transition from simulations to outdoor test drives that are used to evaluate the real performance and dependability on the road. These test drives can be performed with a much higher confidence and less risk, when the ADAS has already been thoroughly tested in VEHIL.

3.6 *Representativeness of VEHIL*

A fundamental aspect of a HIL test environment is that it provides a *representative* testing environment. The input from the artificial VEHIL environment into the VUT must be representative for the actual driving conditions on the road. A restriction of the VEHIL simulation is that vehicle-based inertial sensors (accelerometers and yaw rate sensors) do not give a representative signal, since the VUT is held at a stationary position. Another restriction is that the chassis dyno does not produce lateral tyre forces in accordance with (13)-(14) during steering actions of the VUT, since α_{1j} equals δ_{1j}. However, the resulting relative lateral and yaw motion can still be correctly emulated, as shown in figure 5(b). On the road environment sensors can be perturbed by obstacles outside the relevant area (e.g. infrastructure elements outside the path of motion). Much of the effort in sensor post-processing is associated with filtering out these disturbances. In VEHIL these disturbances can be different from the real world or even absent, although the absence of these disturbances does not affect the basic operation of the ADAS.

To solve these issues, the HIL concept allows to feed the ADAS in real-time with a 'mixture' of real and virtual sensor signals. Any missing sensor signal can be generated from the real-time simulation of the vehicle model (6)-(23) in the MARS (the internal dynamics of entity E_2), which replaces the real sensor signal and is subsequently fed into the ADAS controller. In addition, this setup allows to inject additional signals that represent (infrastructure) disturbances or failure modes. Alternatively, inertial and environment sensors can be installed on an MB that executes a traffic scenario as if it was a standard road vehicle, while another MB represents a target vehicle, as shown in figure 5(c). This setup also allows to obtain a closing velocity of up to 100 km/h, when two MBs drive towards each other.

Due to the absence of a realistic driving environment, VEHIL is not intended to serve as a driving simulator, although it has potential to include driver interaction, as shown in section

4.2. VEHIL is therefore not meant to replace test drives, but focuses on reproducible and accurate testing of the ADAS performance and dependability before test drives take place. In addition, VEHIL tests are used for those scenarios that are too difficult or dangerous to perform on the road. We will therefore demonstrate the suitability and added value of VEHIL in the next section with test results.

4 VEHIL test results for ADAS applications

In cooperation with industrial parties, tests have been conducted for several vehicle types (trucks, cars), ADAS applications (ACC, stop-and-go, FCW, pre-crash systems, blind spot systems), and sensors (radar (pulse-Doppler, FSK, FMCW), vision (both mono and stereo), and lidar). Here we will discuss the test results for ACC and FCW. For clarity, the presented controllers are simpler than the actual implementation, since the focus is on the way they are tested.

4.1 *Adaptive cruise control system*

An ACC controller must be tested in a closed-loop experiment, since the ACC control actions affect the relative motion, which in turn is detected by the environment sensor. Apart from the vehicle itself, optionally a human driver can be included 'in-the-loop' to operate the ACC control lever and introduce disturbances. The prototype vehicle, depicted in figure 8 has been implemented with the feedback control law

$$a_{\mathrm{d}} = -k_1 e_x + k_2 v_r, \quad k_1,\, k_2 > 0, \tag{27}$$

to obtain a desired acceleration a_{d} that controls both e_x and v_{r} to zero. In order to achieve a natural following behaviour, the desired clearance is chosen as $x_{\mathrm{d}} = \max(v_2 t_{\mathrm{h}}, s_0)$ and the feedback gains are calculated by nonlinear functions $k_1 = f_1(v_2, x_{\mathrm{r}}, t_{\mathrm{h}}, s_0)$ and $k_2 = f_2(v_2, t_{\mathrm{h}})$, where s_0 is a distance safety margin and t_{h} is the driver-selected time gap (see figure 2 for a definition of the notation).

Control law (27) is tested for the traffic scenario of figure 10: the ACC-equipped vehicle 2 drives on the middle lane when suddenly another vehicle 1 cuts in from the right lane at a lower speed ($v_{\mathrm{r}} < 0$ and $e_x > 0$). This happens at $t = 22.9$ s, which can be seen from the range x_{r} and angle ϕ to the target in figure 11. As soon as the radar sensor on vehicle 2 detects the obstacle in its lane (i.e. $\phi \approx 0$), (27) gives $a_{\mathrm{d}} < 0$ and the ACC activates the brake system at $t = 25.3$ s. Vehicle 3 stays on the right lane and is used to test the ability of the radar to distinguish between important and irrelevant targets in the traffic environment (i.e. vehicle 3 should not be considered a target). On a test track it would be very difficult to safely and reproducibly carry out such a test with human drivers, but in VEHIL the scenario can be accurately reproduced. Especially note the transformation from absolute to relative motion, i.e. $\boldsymbol{v}_{\mathrm{mb}} = \boldsymbol{v}_1 - \boldsymbol{v}_2$.

The results also show that the MB has a maximum error ε of 0.10 m between desired and measured position, and a reproducibility within 0.01 m between consecutive test runs. The velocity error is usually smaller than 0.1 m/s. This dynamic accuracy is reached up to a bandwidth of 5 Hz and a velocity of 50 km/h, and is within the measurement noise of any automotive environment sensor. Similarly, the chassis dyno can be accurately controlled up to a bandwidth of 5 Hz.

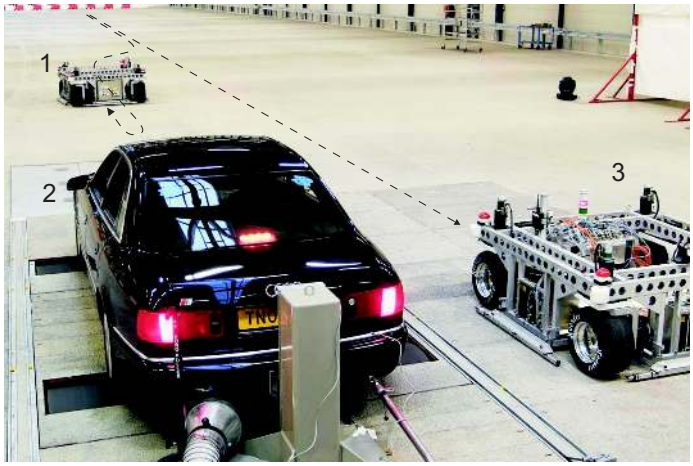


Figure 10. The cut-in scenario of figure 7 is reproduced in VEHIL to test the ACC system: (1) moving base no. 1 (MB1), representing the relative motion of vehicle 1 from the viewpoint of vehicle 2; (2) vehicle under test; (3) MB2, representing the other target vehicle 3.

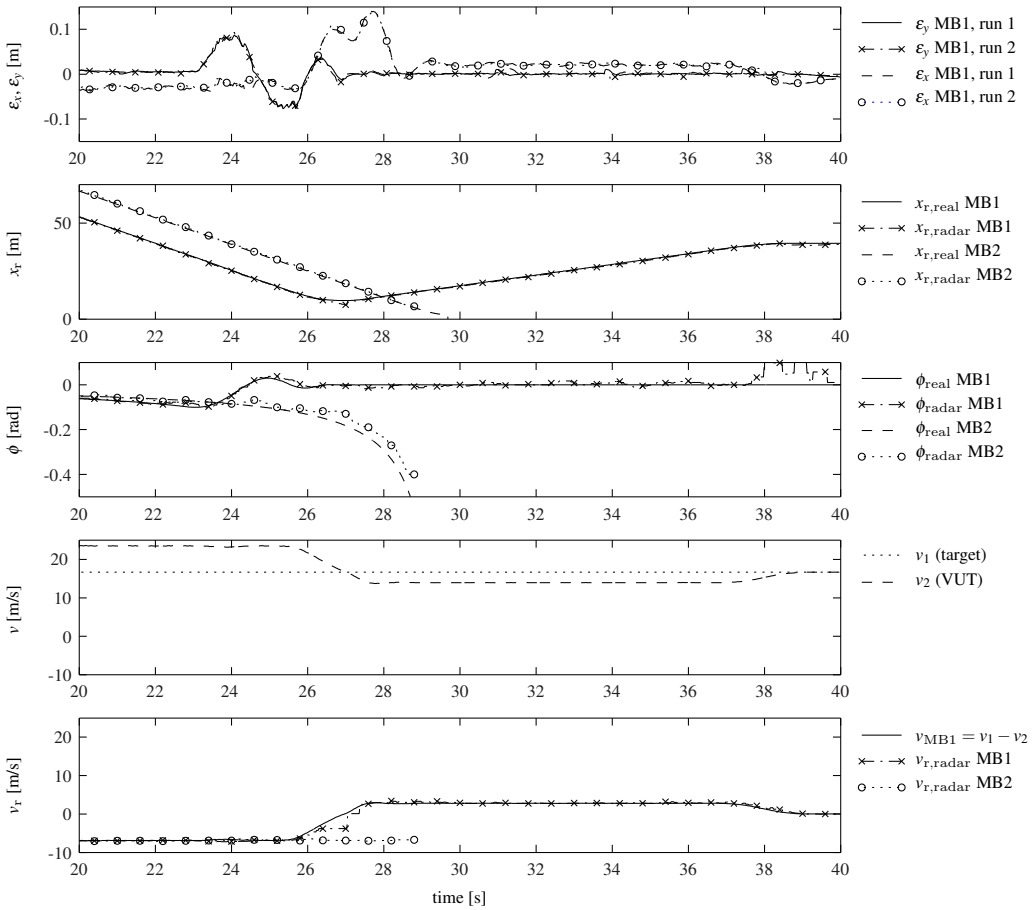


Figure 11. VEHIL test results for the ACC system.

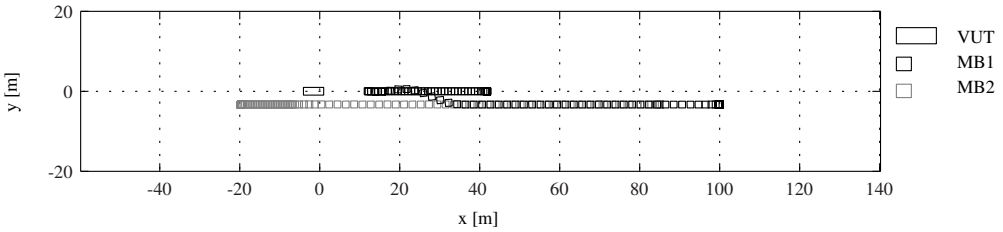


Figure 12. Trajectories of the moving bases in the VEHIL laboratory during the ACC test.

With these type of tests VEHIL has an added value in identifying the requirements and capabilities of an ACC system for safety-critical traffic scenarios in an early development stage. Using rapid control prototyping techniques, various control settings are efficiently tested for a variety of scenarios. When the effect of various traffic disturbances on control performance is known, controller parameters can be optimally tuned. In addition, in a later stage functional validation of the completed system to these requirements can be done unambiguously and efficiently.

4.2 *Forward collision warning system*

Testing an FCW system is more safety-critical than ACC, since a collision warning system is activated shortly before a collision is expected. A warning is issued when a threshold of maximum braking capability $a_{2,\min}$ is crossed by the required deceleration a_{d} to prevent a collision [41]. The algorithm takes into account whether an initially moving lead vehicle (i) stops prior to the following vehicle, or (ii) is still in motion when the host vehicle stops. Taking into account driver reaction time t_{r}, a_{d} is given by

$$a_{\mathrm{d}} = \begin{cases} \dfrac{a_1 v_2^2}{2a_1(t_{\mathrm{r}} v_2 - x_{\mathrm{r}} + s_0) + v_1^2} & \text{, case (i)} \\ \dfrac{a_1(x_{\mathrm{r}} - s_0) - \frac{1}{2} v_{\mathrm{r}}^2}{t_{\mathrm{r}}(\frac{1}{2} t_{\mathrm{r}} a_1 + v_{\mathrm{r}}) + x_{\mathrm{r}} - s_0} & \text{, case (ii),} \end{cases} \tag{28}$$

such that a collision is avoided by a safety margin s_0.

The truck, shown in figure 13, is equipped with a control law similar to (28) (including some nonlinear characteristics). In the simulated scenario, an inattentive truck driver slowly closes in with 25 m/s on another vehicle driving at 23 m/s (represented by the MB). After the preceding vehicle suddenly brakes at $t = 46.7$ s, a_{d} in (28) drops below $a_{2,\min}$ at $t = 49.2$ s, and subsequently the FCW system sends a collision warning to the driver. The corresponding test results in figure 14 show that, after a slight delay due to driver reaction time, the driver brakes at $t = 49.9$ s and avoids the collision.

In this way, optimum warning thresholds are defined by executing reproducible and safe experiments. Apart from objective parameter tuning, VEHIL also seems to have potential for subjective evaluation in addition to on-road tests. It can be verified whether the warnings, when given in defined critical situations, are adequate. Although the final subjective evaluation should be done on the road, VEHIL can be used for an initial evaluation. Furthermore, ongoing research focuses on validation of VEHIL test results with test drives and on linking VEHIL to a driving simulator [42].

Figure 13. Experimental setup of a VEHIL test with a truck, equipped with an FCW system.

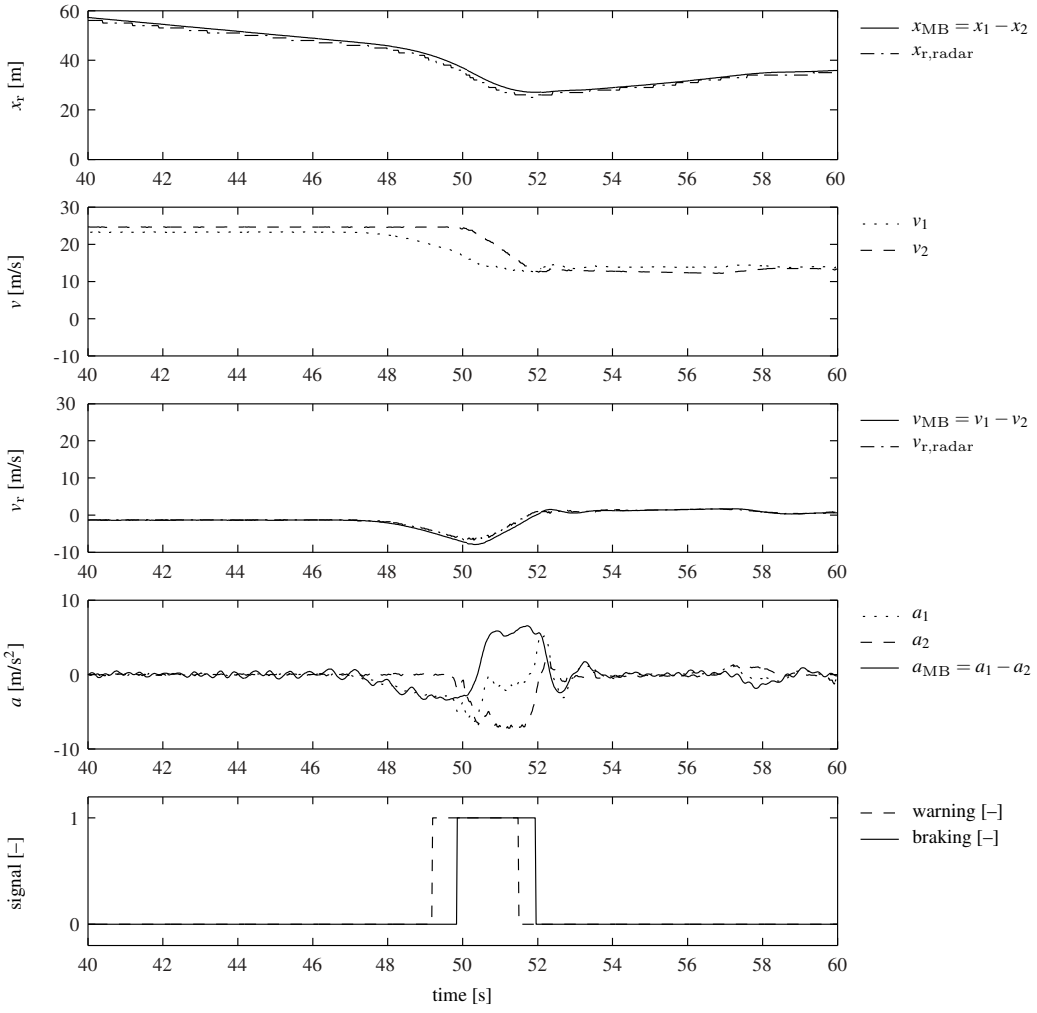


Figure 14. VEHIL test results for the FCW system in a closing situation with emergency brake manoeuvre.

5 Conclusions

This paper has presented the new VEHIL concept for testing ADASs, where a real intelligent vehicle is operated in a HIL environment. VEHIL is suitable for various types of ADASs: ACC, stop-and-go, FCW, pre-crash systems, blind spot systems, and fully autonomous vehicles. With test results for ACC and FCW it was demonstrated that VEHIL has an added value in several phases of the development process of an ADAS: sensor verification; rapid control prototyping; model validation; function level validation; fine-tuning of control algorithms; production sign-off tests; and preparation of test drives. For these purposes, VEHIL experiments are performed in an accurate, reproducible, and controllable way to create a representative test environment.

Furthermore, tests are performed more efficiently than with outdoor test drives, and test scenarios can be varied very easily, due to the connection to the underlying simulation environment. Subsequent test drives can then be performed with a much higher confidence in the system, since the ADAS has already been thoroughly tested in VEHIL. VEHIL is therefore *not* meant to replace MIL simulations and test drives, but to form an efficient link between them. Consequently, the number of iteration loops in the development process is reduced, saving on time and costs.

Ongoing research is focused on further optimisation of this development process by analysing the interactions between the different test methods: MIL simulations, VEHIL, and test drives. This will also include a comparison of the results obtained with test drives and VEHIL experiments, in order to further validate the effectiveness of VEHIL.

Acknowledgements

Research supported by TNO, TRAIL Research School, the Transport Research Centre Delft program “Towards Reliable Mobility”, and the European 6th Framework Network of Excellence “HYbrid CONtrol: Taming Heterogeneity and Complexity of Networked Embedded Systems (HYCON)”, contract number FP6-IST-511368.

Nomenclature

List of frequently used symbols

A	abstract actuator	–
$\boldsymbol{a}$	acceleration vector	–
a	acceleration	$\mathrm{m/s^2}$
C	local vehicle coordinate frame	–
E	entity	–
e_x	distance separation error	m
F	tyre force	N
G	global coordinate frame	–
h	height of vehicle centre of gravity	m
J	mass moment of inertia	$\mathrm{kg\,m^2}$
l	wheelbase	m
M_z	aligning torque	Nm
O	object in the virtual world	–
$\boldsymbol{p}$	position vector	–
$\mathbf{R}$	rotation matrix	–
r	radius	m
S	abstract sensor	–
s	track width	m
T	torque	Nm
t	time	s
t_h	time gap	s
t_r	reaction time	s
$\boldsymbol{v}$	velocity vector	–
v	velocity	m/s
v_r	relative velocity	m/s
$\boldsymbol{x}$	state vector	–
x	position in x-direction	m
x_r	relative position	m
y	position in y-direction	m
z	position in z-direction	m
α	tyre side slip angle	rad
γ	camber angle	rad
δ	steer angle	rad
ε	control error	m
θ	pitch angle	rad
κ	longitudinal slip angle	rad
Φ	orientation vector	–
φ	roll angle	rad
ϕ	angle to object	rad
ψ	yaw angle	rad
ω	angular velocity	rad/s

List of frequently used abbreviations

ACC	Adaptive Cruise Control
ADAS	Advanced Driver Assistance System
FCW	Forward Collision Warning
HIL	Hardware-In-the-Loop
MARS	Multi-Agent Real-time Simulator
MB	Moving Base
MIL	Model-In-the-Loop
VEHIL	VEhicle Hardware-In-the-Loop
VUT	Vehicle Under Test

References

[1] International Road Traffic Database (IRTAD). http://www.irtad.com/.
[2] J. Broughton and C. Baughan. The effectiveness of antilock braking systems in reducing accidents in Great Britain. *Accident Analysis and Prevention*, 34(3):347–355, May 2002.
[3] A. Lie, C. Tingvall, M. Krafft, and A. Kullgren. The effectiveness of ESP (Electronic Stability Program) in reducing real life accidents. *Traffic Injury Prevention*, 5(1):37–41, March 2004.
[4] S.E. Shladover. Review of the state of development of advanced vehicle control systems. *Vehicle System Dynamics*, 24(6–7):551–595, 1995.
[5] S. Tsugawa, M. Aoki, A. Hosaka, and K. Seki. A survey of present IVHS activities in Japan. *Control Engineering Practice*, 5(11):1591–1597, November 1997.
[6] A. Vahidi and A. Eskandarian. Research advances in intelligent collision avoidance and adaptive cruise control. *IEEE Trans. on Intelligent Transportation Systems*, 4(3):143–153, September 2003.
[7] R. Bishop. *Intelligent Vehicle Technology and Trends*. Artech House, Norwood, MA, USA, 2005.
[8] P.L. Zador, S.A. Krawchuk, and R.B. Voas. Automotive collision avoidance system (ACAS) program. Final Report DOT HS 809 080, National Highway Traffic Safety Administration, Washington, DC, USA, August 2000. http://www-nrd.nhtsa.dot.gov/departments/nrd-12/pubs_rev.html.
[9] H.M. Jagtman, V.A.W.J. Marchau, and T. Heijer. Current knowledge on safety impacts of Collision Avoidance Systems (CAS). In P.M. Herder and W.A.H. Thissen, editors, *Proc. of the 5th International Conference on Technology, Policy and Innovation*, Delft, June 26–29, 2001. paper nr. 1152. http://www.tbm.tudelft.nl/webstaf/ellenj/publications.htm.
[10] J. Golias, G. Yannis, and C. Antoniou. Classification of driver-assistance systems according to their impact on road safety and traffic efficiency. *Transport Reviews*, 22(2):179–196, April–June 2002.
[11] P.J.Th. Venhovens, J.H. Bernasch, J.P. Löwenau, H.G. Rieker, and M. Schraut. The application of advanced vehicle navigation in BMW driver assistance systems. *SAE Technical Paper Series*, 1999-01-0490, 1999.
[12] National Highway Traffic Safety Administration. Automotive collision avoidance system field operational test (ACAS FOT). Final Program Report DOT HS 809 866, Washington, DC, USA, May 2005. http://www-nrd.nhtsa.dot.gov/departments/nrd-12/pubs_rev.html.
[13] H. Winner, S. Witte, W. Uhler, and B. Lichtenberg. Adaptive cruise control system aspects and development trends. *SAE Technical Paper Series*, 961010, February 1996.
[14] P. Venhovens, K. Naab, and B. Adiprasito. Stop and go cruise control. *International Journal of Automotive Technology*, 1(2):61–69, 2000.
[15] R. Moritz. Pre-crash sensing - functional evolution based on short range radar sensor platform. *SAE Technical Paper Series*, 00IBECD-11, 2000.
[16] S. Tokoro, K. Moriizumi, T. Kawasaki, T. Nagao, K. Abe, and K. Fujita. Sensor fusion system for pre-crash safety system. In *Proc. of the IEEE Intelligent Vehicles Symposium (IV)*, pages 945–950, June, 14–17 2004.
[17] S.E. Shladover. Automated vehicles for highway operations (automated highway systems). *Proc. of the I MECH E Part I Journal of Systems & Control Engineering*, (I1):53–75, February 2005.
[18] S. Becker. ADAS: Market introduction scenarios and proper realisation. Deliverable D1, RESPONSE 2 – Advanced Driver Assistance Systems: From Introduction Scenarios towards a Code of Practice for Development and Testing, Cologne, Germany, January 21, 2004. http://response.adase2.net/.
[19] R.J. Kiefer, D.J. LeBlanc, M.D. Palmer, J. Salinger, R.K. Deering, and M.A. Shulman. Development and validation of functional definitions and evaluation procedures for collision warning/avoidance systems. Final Report DOT HS 808 964, National Highway Traffic Safety Administration, Washington, DC, USA, August 1999. http://www-nrd.nhtsa.dot.gov/departments/nrd-12/pubs_rev.html.
[20] N. Storey. *Safety-Critical Computer Systems*. Addison Wesley Longman Ltd., Essex, U.K., 1996.
[21] F. Mosnier and J. Bortolazzi. Prototyping car-embedded applications. In *Advances in Information Technologies: The Business Challenge*, pages 744–751. IOS Press, Amsterdam, The Netherlands, 1997.
[22] R. Abou-Jaoude. ACC radar sensor technology, test requirements, and test solutions. *IEEE Transactions on Intelligent Transportation Systems*, 4(3):115–122, September 2003.
[23] Z. Papp, K. Labibes, A.H.C. Thean, and M.G. van Elk. Multi-agent based HIL simulator with high fidelity virtual

sensors. In *Proc. of the IEEE Intelligent Vehicles Symposium (IV)*, pages 213–219, Columbus, OH, USA, June 9–11, 2003.

[24] J. Li, F. Yu, J.-W. Zhang, J.-Z. Feng, and H.-P. Zhao. The rapid development of a vehicle electronic control system and its application to an antilock braking system based on hardware-in-the-loop simulation. *Proc. Instn. Mech. Engrs. Part D: J. Automobile Engineering*, 216:95–105, 2002.

[25] R. Isermann, J. Schaffnit, and S. Sinsel. Hardware-in-the-loop simulation for the design and testing of engine-control systems. *Control Engineering Practice*, 7:643–653, 1999.

[26] R. Isermann. Diagnosis methods for electronic controlled vehicles. *Vehicle System Dynamics*, 36(2–3):77–117, 2001.

[27] K. Athanasas, C. Bonnet, H. Fritz, C. Scheidler, and G. Volk. VALSE – validation of safety-related driver assistance systems. In *Proc. of the IEEE Intelligent Vehicles Symposium (IV)*, pages 610–615, Columbus, OH, USA, June 9–11, 2003.

[28] K. Yi, M. Woo, S. Kim, and S. Lee. Study on a road-adaptive CW/CA algorithm for automobiles using HiL simulations. *JSME International Journal Series C – Mechanical Systems Machine Elements and Manufacturing*, 42(1):163–170, March 1999.

[29] G. Reymond, A. Heidet, M. Canry, and A. Kemeny. Validation of Renault's dynamic simulator for adaptive cruise control experiments. In *Proc. of the Driving Simulation Conference DSC'2000*, pages 181–192, Paris, France, September 6–8, 2000.

[30] `http://www.vehil.com/`.

[31] L. Verhoeff, D.J. Verburg, H.A. Lupker, and L.J.J. Kusters. VEHIL: A full-scale test methodology for intelligent transport systems and vehicles and subsystems. In *Proc. of the IEEE Intelligent Vehicles Symposium (IV)*, pages 369–375, Detroit, MI, USA, October 2000.

[32] L.J.J. Kusters, R.J.A. Kleuskens, D.J. Verburg, and A.C.M. van der Knaap. System for performing tests on intelligent road vehicles. US Patent US 2003/0183023 A1, October 2, 2003.

[33] D.J. Verburg, A.C.M. van der Knaap, and J. Ploeg. VEHIL, developing and testing intelligent vehicles. In *Proc. of the IEEE Intelligent Vehicles Symposium (IV)*, volume 2, pages 537–544, Versailles, France, June 17–21, 2002.

[34] O.J. Gietelink, J. Ploeg, B. De Schutter, and M. Verhaegen. VEHIL: Test facility for fault management testing of advanced driver assistance systems. In *Proc. of the 10th World Congress on Intelligent Transport Systems and Services (ITS)*, Madrid, Spain, November 16–20, 2003. Paper 2639.

[35] O.J. Gietelink, D.J. Verburg, K. Labibes, and A.F. Oostendorp. Pre-crash system validation with PRESCAN and VEHIL. In *Proc. of the IEEE Intelligent Vehicles Symposium (IV)*, pages 913–918, Parma, Italy, June 14–17, 2004.

[36] L.J.J. Kusters, O.J. Gietelink, J. van Hoof, and P.P.M. Lemmen. Evaluation of advanced driver assistance systems with the VEHIL test facility – experiences and future developments at TNO Automotive. In *Proc. of the 21th International VDI/VW Conference 'Integrated safety and driver assistance systems'*, Wolfsburg, Germany, October 27–29, 2004.

[37] Z. Papp, M. Dorrepaal, and D.J. Verburg. Distributed hardware-in-the-loop simulator for autonomous continuous dynamical systems with spatially constrained interactions. In *Proc. of the 11th IEEE/ACM International Workshop on Parallel and Distributed Real-Time Systems*, Nice, France, April 22–26, 2003.

[38] John Craig. *Introduction to Robotics, Mechanics and Control*. Addison-Wesley, September 2004.

[39] H. Pacejka. *Tyre and Vehicle Dynamics*. Butterworth-Heinemann, 2002.

[40] J. Ploeg, A.C.M. van der Knaap, and D.J. Verburg. ATS/AGV, design, implementation and evaluation of a high performance AGV. In *Proc. of the IEEE Intelligent Vehicles Symposium (IV)*, volume 1, pages 127–134, Versailles, France, June 18–20, 2002.

[41] P.L.J. Morsink and O.J. Gietelink. Preliminary design of an application for CBLC in the CarTALK2000 project: Safe, comfortable and efficient driving based upon inter-vehicle communication. In *Proc. of the e-Safety Conference*, Lyon, France, September 16–18, 2002.

[42] R. Bakker, J. Hogema, W. Huiskamp, and Z. Papp. IRVIN - Intelligent Road and Vehicle test INfrastructure. In *Proc. of the 8th International IEEE Conference on Intelligent Transportation Systems*, Vienna, Austria, 2005.