

# Volkswagen Group: Leveraging VIRES VTD to Design a Cooperative Driver Assistance System

---

By **Dr. Kai Franke**, Development Online Driver Assistance Systems, Volkswagen AG



**“A combination of ADTF, VTD, and OMNet++ allows us to do a host of experiments to test and validate cooperative driver assistance systems.”**



**Figure 1. VIRES VTD is an open platform for developing Advanced Driver Assistance Systems**

‘Cooperative Driving’ has attracted significant attention in recent years within automobiles. In order to increase the quality of signals, the availability and the perception range as well as to decrease the latency and the probability of total failure, advanced perception systems consisting of camera, radar and lidar systems with vehicle-to-vehicle (V2V) communication are required. Moreover, V2V communication enables advancements from individual to cooperative decision making. Advanced driver assistant systems (ADAS), which determine their behavior in due consideration of the definition of “cooperative behavior”, are capable of increasing the total utility of a group of cooperative vehicles. However, several technical issues have to be resolved on the way to deploying the first cooperative driver assistance systems (CDAS) on our public streets. For example, handling the misuse of the communication channel and

the consideration of unequipped vehicle are some of the key challenges for cooperative driving.

This article focuses on a test framework for CDAS, which can be leveraged to master the complexity of distributed driver assistance systems (DAS) during the development process. A combination of ADTF (the application prototyping framework within the Volkswagen group), VTD (Figure 1, a simulation tool-chain from VIRES GmbH) and OMNet++ (an open-source component-based network simulator) allows us to do a host of experiments to test and validate cooperative driver assistance systems.

## Simulation Framework

Since the development of CDAS requires at least two interacting vehicles, the implementation and the validation of the system necessitates a flexible test framework for connected vehicles. Figure 2 gives an overview of the proposed architecture used within this project (reference 1). The detailed description of interfaces and functionality follows hereinafter.

### A. Application

A simplified illustration of the CDAS is composed of a planner implemented in ADTF (Automotive Data and Time triggered Framework) and a controller for each involved vehicle. There are three relevant interfaces of the application. The first interface represents the environmental model and the current vehicle state provided by the simulation gateway. The second interface to the network enables the communication

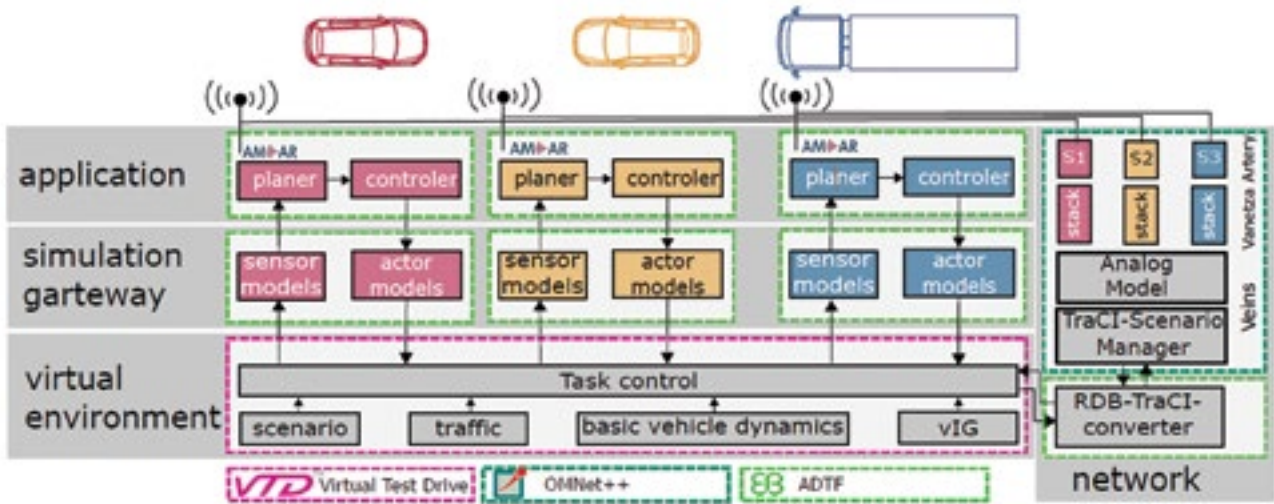


Figure 2. Overview of the VW simulation framework

between vehicles. The last interface to the simulation gateway realizes the controllability of the vehicle.

### B. Simulation Gateway

For each vehicle (here an example is shown for three vehicles in Figure 3), the simulation gateway fulfills among others two tasks: the modeling of the perception (environment and vehicle state), and the reaction to controller outputs. The interface for the vehicle state includes, but is not limited to, the velocity, the longitudinal and lateral acceleration, and the steering wheel angle.

### C. Virtual Environment

The software Virtual Test Drive (VTD) developed by VIRES provides the virtual environment we used.

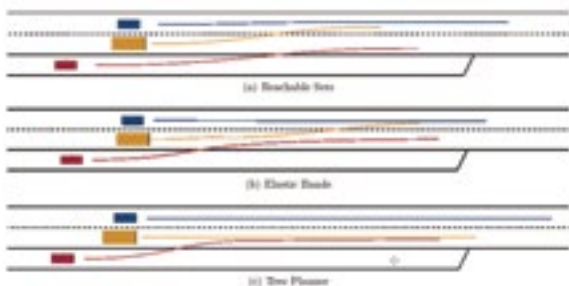


Figure 3. Results of the planning methods for the merging scenario of three vehicles

The central component is the task control coordinating additional modules with the help of the module manager. Additional modules are the scenario with roads and vehicle information, the traffic, the basic vehicle dynamics for internally controlled vehicles and the Image Generator (IG). The virtual environment transmits its information via Ethernet on the Real Time Data Bus (RDB) interface. Furthermore, the Simulation Control Protocol (SCP) interface provides a mechanism for operating the simulation.

### D. Network

The network simulation can emulate the communication of the application via for example, ETSI ITS G5. In order to simulate the signal damping, the analog model uses information about line of sight and distances between the communicating vehicles. The RDB interface and the map of VTD (\*.xodr format) contain the required information.

## Simulation Results

### A. Decentralized Decision Making

An example of a merging scenario on a highway is chosen to demonstrate the usability of the decentralized decision making (see Figure 3). The red vehicle wants to merge onto the highway, while the two lanes are blocked by a truck (yellow) and another vehicle (blue). The lane width amounts to three meters each.



Figure 4. Scenario for collision avoidance on oncoming traffic

Three different planning algorithms generate offers for the merging scenario. It can be seen that the planning methods (a) and (b) recommend a lane change for the truck, while method (c) makes the truck stay in its lane. Planner (b) starts the lane change later than planner (a). Planner (c) solves the conflict situation by accelerating the truck and merging maneuver of the red vehicle behind the truck. The diversity of the offers results from different discretizations and different evaluation criteria. In order to demonstrate the decentralized decision making process, a cost function based on a fuzzy logic is applied, which enables a continuous prioritization between comfort, driving enjoyment, efficiency, and safety. Table I illustrates the different preferences of each vehicle. The truck focuses on efficiency, the red vehicle prefers driving enjoyment, and the blue vehicle prioritizes comfort.

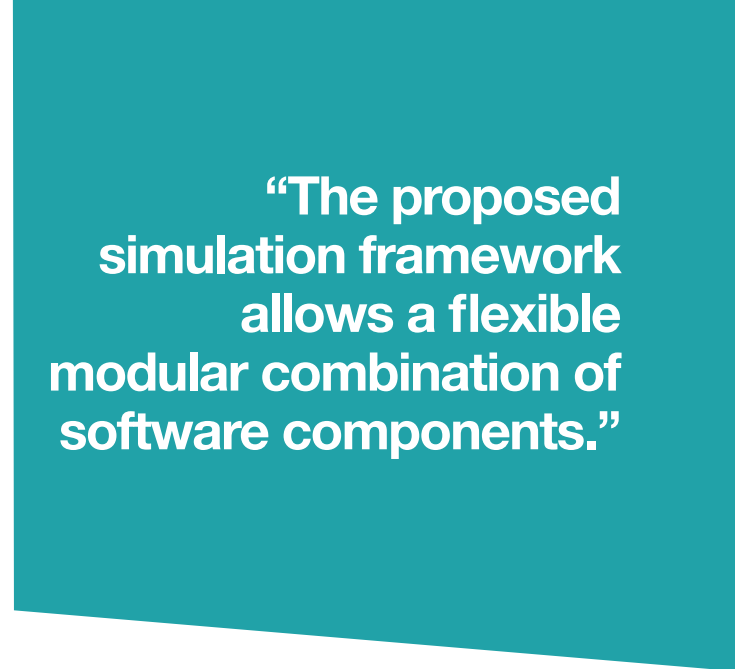
Each vehicle comes to a different evaluation or rating of the offers, because of the varying preferences. The varying preferences can be caused by different brands, different vehicle models (sedan, van or SUV), or by an online driver monitoring system. Table II shows the results of the evaluation of each plan by each vehicle and the result of the two proposed selection criteria. The selected solution (bold) represents the compromise of the solution options. Plan (c) is selected by the sum criterion and plan (a) is selected by the squared sum criterion.

## B. Closed Loop Simulation

The closed loop or hardware-in-the-loop simulation enables a study to evaluate the control error considering communication and calculation latencies

Table I. I PREFERENCES OF THE COOPERATIVE VEHICLE TO PARAMETRIZE THE COST FUNCTIONS

	red vehicle	yellow truck	blue vehicle
safety	1	1	1
comfort	0.2	0.5	1
efficiency	0.1	1	0.4
driving enjoyment	1	0	0.1



“The proposed simulation framework allows a flexible modular combination of software components.”

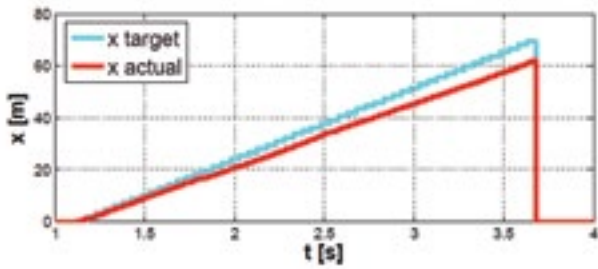
of planner, controller, and vehicle dynamics. An application for collision avoidance exemplarily demonstrates the closed loop performance (Figure 4).

As an initial scenario, a driver starts an overtaking maneuver on a rural road. The driver misjudges the situation and the danger of a collision with the oncoming traffic arises. An active safety system detects the danger and starts/triggers the cooperative maneuver planning. The detection criterion could also be the time to collision (TTC). The TTC is calculated as the quotient of distance and relative velocity. The calculated cooperative maneuver plan targets the completion of the overtaking maneuver of the red vehicle and a deceleration of the truck and the blue vehicle.

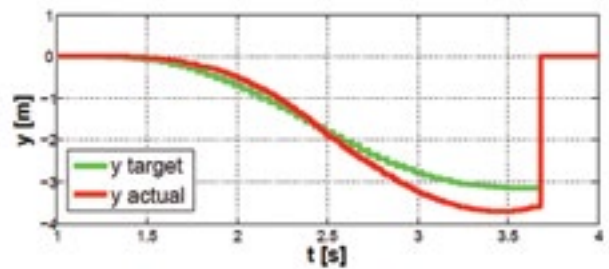
Figure 5 shows a flagging characteristic. The longitudinal controller has a linear increasing controller error. This is caused by a constant velocity error. A possible reason is that the longitudinal controller does not model the decelerating influence of a steering maneuver. In this case the vehicle decelerates stronger than planned. The lateral controller shows an overshooting. The vehicle stays with 50 cm maximum

Table II. INDIVIDUAL COSTS AND COMBINED COSTS FOR A COOPERATIVE MERGING SCENARIO

	plan (a)	plan (b)	plan (c)
red vehicle	0.59	0.85	0.80
yellow truck	0.20	0.41	0.13
blue vehicle	0.44	0.17	0.17
sum $k_{i,v}$	1.23	1.44	1.11
sum $k_{i,v}^2$	<b>0.44</b>	0.93	0.69



(a) longitudinal controller



(b) lateral controller

Figure 5. Results of closed loop simulation

controller error in a safe condition (stays on road, no collisions with obstacles). This error is caused by latencies and systematic errors in the feed forward controller. However, systematic errors, difference between vehicle dynamics model and inverted model in the feed forward controller, are made on purpose. A perfect vehicle dynamics model in the feed forward controller is impossible in reality, because of for example changing loads, changing wheel characteristics, and changing surface etc. Further controller adaption will be done with the help of real field test.

## Conclusions

A new CDAS (Cooperative Driver Assistance System) imposes new requirements on simulation methods. The high degree of connectivity and interaction of the

applications disable a development and later validation without considering the multi-directional influence. The proposed simulation framework allows a flexible modular combination of software components and considers modeling of perception, communication, and controlling of several vehicles in a virtual environment.

## Reference

1. "A Cooperative Driver Assistance System: Decentralization Process and Test Framework" by Kai Franke, Reza Balaghiasefi, Michael Düring, Hendrik-Jörn Günther, Proc. 7th Tagung Fahrerassistenzsysteme Conf., 2015.
2. Source: <https://pdfs.semanticscholar.org/6361/393b-8c4067f857bf68f8ea7b79588eb19aba.pdf>

