BUILDING TRUST IN VIRTUAL TESTING OF IN-VEHICLE ROAD SAFETY MEASURES: IDENTIFICATION OF EXPECTABLE INTER-TOOL VARIATIONS

Peter Wimmer
Virtual Vehicle Research GmbH
Austria

Henri Chajmowicz
LAB PSA/Renault
France

Jeroen Uittenbogaard
TNO
The Netherlands

Leonid Tulin
TÜV Süd Czech
Czech Republic

Lars Schories
ZF Friedrichshafen AG
Germany

Paper Number 26-104

Abstract

Virtual testing has emerged as a cornerstone in the development of in-vehicle road safety technologies. We therefore need to improve trust in and comparability of virtual testing campaign results. A key factor in comparing virtual test results is knowledge about unavoidable variations due to using different simulation tools. First steps have been made into this direction in a previous study [1] by specifying, performing and evaluating a round-robin study. The objective of this work is to further isolate tool-inherent variation sources. The research question addressed here can be summarized by: "Which steps in a tool comparison for virtual testing of active vehicle safety are required to minimize the variations and how much remaining variation can be expected?"

We extended the existing round-robin specification to eliminate known sources of unwanted variation and to gain insight into remaining tool-inherent variations. The new process consists of three steps: Unit tests aimed at assessing critical models of the simulation chain, baseline simulations to gain insights into variations (and minimize them) before adding any treatment (i.e., the safety measure under investigation), and treatment simulations to obtain the variations to be expected in safety impact assessment.

Every participant in this round-robin study performed simulations for all three steps using their preferred simulation tool. Results were then compared to gain insight into avoidable and unavoidable variations and the specification adapted accordingly until only unavoidable (i.e., tool-inherent) variations remained. As a result, the variations of this second round-robin study are much smaller than in the first version.

This study clearly shows that round-robin tests can help build trust in virtual testing and that precise specifications of virtual tests can help to improve comparability between test runs executed with different tools. However, a certain amount of variation remains due to tool-inherent features that cannot be changed by users.

Keywords

Active safety; driver assistance systems; safety performance assessment; virtual testing; harmonization

INTRODUCTION

As the automotive industry accelerates towards automated and software-defined vehicles, virtual testing has emerged and is a cornerstone in the development and validation of in-vehicle road safety technologies as also described in [2]. Advanced Driver Assistance Systems (ADAS) and other safety-critical functions are already evaluated in simulated environments, enabling the exploration of complex, rare, and hazardous situations that would be impractical or unsafe to replicate in real-world testing as done in [3], [4], [5], [6], [7], and [8]. The growing reliance on virtual testing also raises a critical question on the trust one can put in simulation tools and processes.

The first, important step is the establishment and application of a comprehensible, reliable, transparent, and accepted methodology for prospective assessment of traffic safety for vehicle-integrated active safety technologies by means of virtual simulation like the activities in P.E.A.R.S. [9,10], that also resulted in an ISO technical report [11] and an ISO technical specification [12].

Despite this theoretical common ground and widespread adoption of simulation-based validation, the ecosystem of virtual testing tools remains fragmented. Each tool varies in its modelling assumptions, sensor fidelity, scenario interpretation, and physics engines, similar findings have been made also by [13]. These differences can lead to inter-tool variations - discrepancies in outcomes when the same scenario is evaluated across multiple platforms. Such variations pose a challenge to the credibility and comparability of simulation results, especially in regulatory and safety assurance contexts where consistency and reproducibility are key like in the upcoming Euro NCAP virtual testing protocol [14]. This paper addresses the urgent need to increase trust in virtual testing by systematically identifying and characterizing expectable inter-tool variations.

Recent efforts to harmonize simulation-based safety assessments have already revealed critical sources of variation that impact the reliability of virtual testing: With the overall goal to harmonize prospective effectiveness assessment of active safety systems, we already presented a first study [1] where the specific objective was to identify and evaluate sources of variation in virtual precrash simulations and to suggest topics for harmonization resulting in increased comparability and thus trustworthiness of virtual simulation-based prospective effective-ness assessment.

Assessments of the safety performance of an advanced driver assistance system were performed using an array of state-of-the-art virtual simulation tools on a set of standard test cases – the CATS car-to-cyclist accident configuration dataset was chosen for the purpose. The results were analyzed to examine reasons for deviations to identify and assess aspects that needed to be harmonized and standardized. Deviations between results calculated by independent engineering teams using their own tools should be minimized if the research question is precisely formulated regarding input data, models, and postprocessing steps.

Two groups of sources of variations were identified; one group (mostly related to the implementation of the system under test) can be eliminated by using a more accurately formulated research question, whereas the other group highlights further harmonization needs because it addresses specific differences in simulation tool setups. The main sources of variation found were:

- Test specification-related variations:
    - TTC calculation variation
        - Major source of variation - further harmonization needed.
        - Can be minimized through more comprehensive specifications – e.g., standard distance and TTC calculation methods.
- Simulation tool-related variations:
    - Vehicle dynamics - Variations in hit-point location
        - Can have considerable influence on avoidance.
        - The “baseline” hit-point locations must be met with a specific tolerance prior to simulating the AEB test cases.
    - Vehicle dynamics - differences in the braking behavior
    - Simulation controls – collision flag not refreshed at the same pace as vehicle dynamics

The work we present here is based on these findings and aims to provide a foundation for harmonized validation practices and improved transparency in simulation-based safety assessments. Our work contributes to the broader goal of establishing simulation as a reliable and auditable pillar in the safety lifecycle of modern vehicles. Its objective is to address the findings of the previous study and to further improve the tool comparison method to isolate tool-inherent sources of variation even further. The research question of this paper can therefore be summarized by “Which steps in a tool comparison for virtual testing of active vehicle safety are required to minimize the variations and how much remaining variation can be expected?”

METHODS

Based on the outcomes of previous work, we extended the existing specification for the round-robin to eliminate known sources of unwanted variation and to gain insight into remaining tool-inherent variations in test results. In the previous study we identified variations related to definition interpretations, e.g., TTC and distances. By performing unit tests, e.g., for collision detection logic, these disparities can be minimized. Variations related to input data were a source of disparity as well. By performing baseline simulations (i.e., simulations without the safety measure under investigation implemented) we gained insights into variations to be expected between the

different tools and how to minimize them (to reduce the variations before going to the next step). Finally, after optimizing these two types of tests, we performed treatment simulations (i.e., with the safety measure under investigation implemented) to investigate if the results are closer together.

Process Overview

The outcomes of the previous round-robin showed clearly that more steps are required to understand and assess variations between simulation tools. Thus, we took several actions:

- Extension of the test specification by describing how TTC should be calculated to ensure that this source of variation is eliminated/minimized.
- Addition of unit-tests to get additional insights into tool-inherent variations by isolating effects of sub-parts of the overall simulation model.
- Simulation and analysis of baseline cases prior to performing the treatment simulations to learn more on variations in hit-point locations and how far these could be reduced.

Including these updates, the entire process now looks like this:

1. Execution and analysis of the unit tests (sensor, brakes, collision detection)
2. Execution and analysis of two types of baseline simulations (non-optimized, optimized)
3. Execution and analysis/comparison of treatment simulations

Details of these steps follow in the next sub-chapters. The left side of **Figure 1** shows the flow of actions in an AEB system simulation. Each of the blocks can introduce variation. The right side gives details on the respective possible sources of variation and in which step of the current study these are addressed.

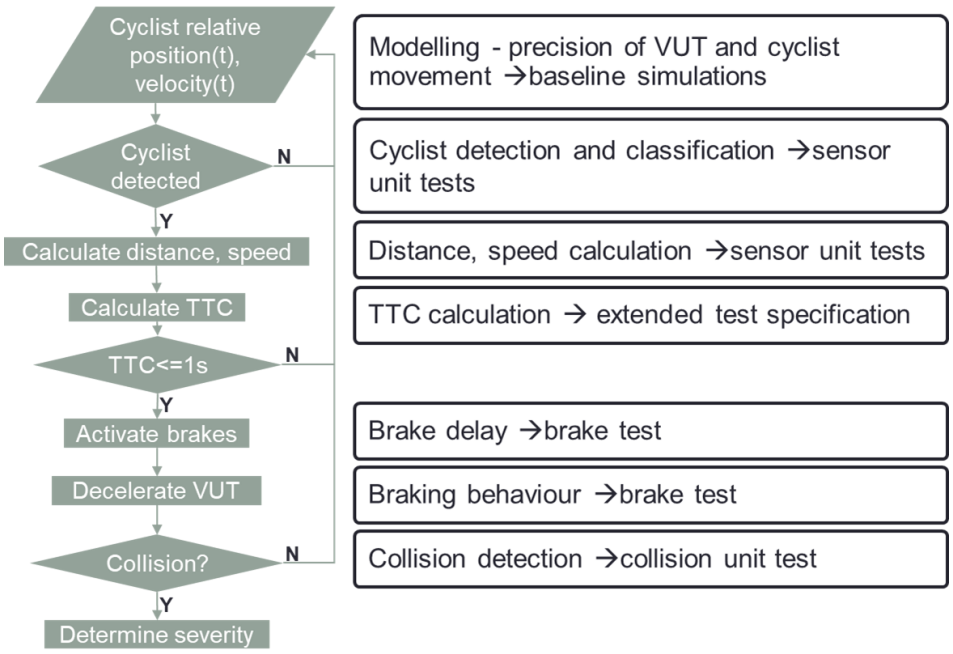


Figure 1. Flow of actions of an AEB system simulation, sources of variation and where they are addressed.

Unit Tests

We decided after our first round-robin study to set up unit tests that aim at assessing critical steps or models present in the workflow. The reason for adding these unit tests is the fact that it is easier to analyze smaller models and see where variations, if any, come from and try to reduce them instead of looking at the global outcome and have a backwards search for where variations may come from amongst the complete set of models involved in the workflow. Therefore, "simple" tests were designed to challenge specific parts of the workflow:

- sensor logic
- braking capacities of the vehicle dynamics model
- collision detection.

The **sensor unit tests** are supposed to give insight into variations in target detection and classification as well as relative distance, speed, and angle calculation variations. We therefore set a simplified cyclist shape to enter the

field of view (FoV) in challenging configurations, such as at a flat angle or from behind an obstacle. An overview of t tests is shown in **Figure 2**: The leftmost plot shows the tests without view-blocking obstruction (1, 2, 3, 6, 7), the center plot test 4, and the rightmost plot test 5. The arrows indicate the respective cyclists' direction of travel. The sensor parameters can be found in Appendix A, **Table A 1**.

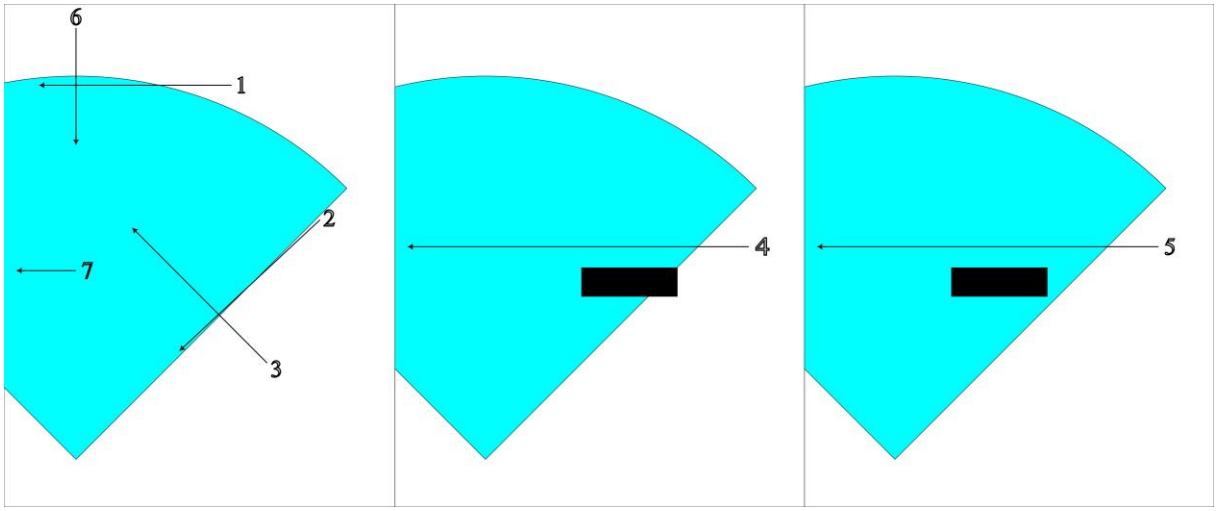


Figure 2. Overview of the sensor unit tests. Sensor FoV in cyan, view-blocking obstruction in black.

We used two types of results for comparison. The first type of results are event-based points in time. The points in time of interest are:

- Time of full detection: The point in time where the cyclist becomes fully visible for the first time.
- Time of classification: The point in time where the cyclist is detected, and the acquisition time has elapsed.

We compared all participants' results for these points in time and all sensor unit tests to the respective analytical results. These can be calculated using the information given in the test description (geometries, initial positions, and speeds).

The second type of results are time dependent outputs and their deviation from the analytical solution. The time-dependent outputs of interest are:

- Relative distance
- Relative speed
- Relative angle

For analysis, we plotted these results with markings at the time of classification.

In the **braking unit test**, we assessed the braking behavior of the vehicle dynamics models by setting a simple braking instruction and analyzing the speed / acceleration outcome profile. The initial speed of the VuT is set to be 60 km/h, the brake actuation should follow the instructions given in Appendix A, "Actuator". Here we also used discrete values and time-dependent outputs. The discrete values of interest are the time to stop and the distance to stop. Again, we calculated range, minimum and maximum of all participants' results. Regarding time-dependent outputs, we used the VuT's position, longitudinal speed, and acceleration to calculate corridors and median lines.

In the **collision detection unit tests**, shapes, ranging from the simplest half plane to a complex, multiple edged cyclist-like shape were placed in front of the advancing VuT to assess collision detection. **Figure** 3 shows an overview of the tests, the VuT is colored blue, the collision shapes are black hatched.

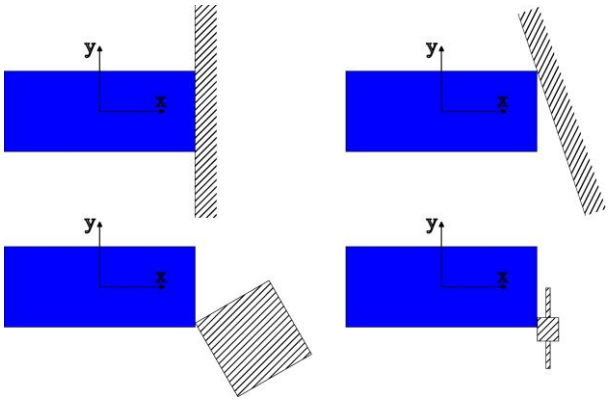


Figure 3. Overview of the four collision detection unit test setups.

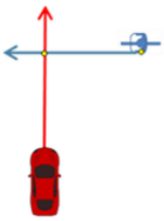
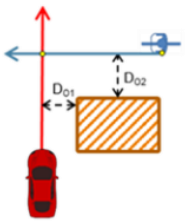
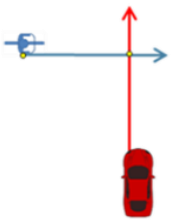
We evaluated these tests by using the point in time at which the collision was detected and the position in the direction of travel of the VuT at this point in time. We set these results for each participant and test in relation to the analytical solution and calculated range, minimum and maximum values of the resulting deviations from the analytical solution.

Baseline Simulations

In the context of road safety simulation, the baseline [15] serves as a reference against which safety measures or technologies are evaluated. Within this study, baseline simulation results represent the initial output of each simulation tool, providing a foundation for assessing tool performance prior to the implementation of any safety interventions.

We used the test cases defined in [16] for our study. The test matrix consists of four test cases, CVNB (Car to VRU Nearside Bicyclist), CVNBO (Car to VRU Nearside Bicyclist Obstructed), CVBF (Car to VRU Far side Bicyclist), CVLB (Car to VRU Longitudinal Bicyclist), see **Table** 1.

Table 1.
Baseline simulation test cases as defined in [16].

	CVNB	**CVNBO**	**CVFB**	**CVLB**
VuT speed	20–60 km/h	10–40 km/h	20–60 km/h	20–60 km/h
Cyclist speed	15 km/h	10 km/h	20 km/h	15 km/h
Obstruction	Without	With D1=3.55m, D2=4.80m	Without	Without
Hit-point location on VuT front	50%	50%	25%	50%
				

The velocity of the VuT is varied between 10 – 60 km/h. The velocity of the cyclist is a constant speed of 10, 15 or 20 km/h depending on the test case. One test case (CVNBO) also involves view-blocking obstruction. Different hit-point locations on the VuT front are considered. The shape of the VuT front and the cyclist are defined for the simulation. Details on the test case descriptions (hip-point locations, geometries etc.) can be found in Appendix A.

These baseline results are critical for identifying discrepancies between simulation tools. By analyzing differences at this early stage, inconsistencies can be addressed before proceeding to treatment simulations. To

facilitate this comparison, we defined a set of simulation outputs including VuT and cyclist position, velocity, and time data.

As shown in [1], tool specific simulation controllers and configurations can lead to differences in simulation results in the baseline simulation. We see this as an issue as differences in baseline simulation results can make comparisons of treatment results impossible as the differences seen in the treatment simulation results are a combination of at least two sources - the differences in the baseline simulation results combined with a different treatment. Without further knowledge of the difference already existing in the baseline, it is thus impossible to isolate the influence (and thus the potential safety benefit) of the treatment. To further investigate these differences, we have performed two types of baseline simulations. "**Non-optimized baseline simulations"** where the starting conditions have been fixed (the initial x and y positions of both VuT and cyclist) and "**optimized baseline simulations"** where the participants modified the starting conditions such that the end conditions of the simulation are met as closely as possible (collision point defined by the x and y positions of VuT and cyclist at the time of collision). The first type of baseline simulation gives insight into how using solely identical starting conditions for all simulation tools (and then let the simulation develop itself) influences the situation at the time of collision in terms of relative positions, speeds, timing, etc.. The second type aims at having a baseline simulation as best aligned as possible at the point in time where it is most relevant: the time of collision.

It is important to note that differences may still arise due to the definition of reference systems within individual tools. To mitigate this, a unified reference system has been selected and consistently applied across all simulations.

For the **non-optimized baseline simulations**, we assessed the precision of end positions and speeds, as well as deviations of time-dependent outputs (VuT speeds and positions) from the trajectories given by the baseline test case descriptions. For the **optimized baseline simulations**, we assessed the precision of the end positions and speeds (i.e., how much the actual hit-point locations are off from the defined one).

Treatment Simulations

To see the full picture, an AEB system is added to the baseline simulations. The AEB system consists of three parts: sensor, algorithm, and actuator. We considered to slightly different sensor setups compared to the sensor unit tests: "Medium FoV" and "Large FoV". The details of these setups can be found in Appendix A, **Table A 1**.

We used a TTC-based algorithm for brake initiation with a threshold of TTC=1s. As the previous study revealed different interpretations of how to calculate TTC, we added a description of how to calculate TTC in the round-robin specification. **Figure A 5** in Appendix A depicts this description.

The braking that is applied once the algorithm has fired, is described as follows:

- Actuator delay: 0.2 s
- Time to maximum deceleration (Build-up time): 0.4 s
- Deceleration increase over time is linear until the maximum value is achieved.
- Maximum deceleration: 1.0 g

The tire-road friction coefficient is assumed to be 0.9, therefore it is not possible to gain a deceleration above 0.9 g. We analyzed these types of simulation output:

Time independent discrete values: Here we distinguish between two types of results: binary results ("cyclist classified yes/no", "AEB fired yes/no", "collision avoided yes/no") and numerical values (speed reduction). For the binary results we calculated the respective rates per participant: the "Classification rate" contains the percentages of test cases where the cyclist was successfully detected and classified before collision. A value of 74% means that in 25 cases of the 34 test cases (9xCVNB, 7xCVNBO, 9xCVFB, 9xCVLB) the cyclist was detected and classified. "Firing rate" and Avoidance rate" follow the same logic. Firing (i.e., triggering of the brake actuation) occurs if the cyclist is detected and classified and the TTC value reaches the given threshold of 1s. If no firing occurs, a collision with an impact speed same as the initial speed is the result. In case of firing, either the braking is successful, and the collision can be avoided (these cases will be used to determine the avoidance rate) or a collision occurs with reduced impact speed (mitigation). All resulting speed reductions (0km/h in case of non-firing, initial speed in case of avoidance, reduced impact speed in case of mitigation) are used to calculate the average speed reduction.

Conformity rates: Conformity rates can be calculated based on the binary events "classified yes/no", "firing yes/no", and "avoidance yes/no". A conformity rate of 100% would indicate that all participants produced the same binary results for a given event (e.g., classification=yes) for a specific test case (e.g., CVNB60). This would mean perfect agreement between all participants on this result. On the other end, a conformity rate of 0%

would indicate that one half of the participants produced a “yes” for a binary result, while the other half produced a “no”. In that case, maximum disagreement between the participants on this result occurs – a very undesirable result.

Tools Used

We used different tools to perform the tasks outlined above; brief tool descriptions are as follows:

PERFSIM (LAB Tool) is an in-house MATLAB-Simulink tool aimed at assessing ADAS effectiveness by simulating their effects in actual collision situations with VRUs. It uses an open “point-mass” model of the vehicle dynamics, to which user-defined environment and sensor/actuator blocks can be added.

TNO implemented its own simulation tool in a MATLAB/Simulink environment to assess AEB feasibility and quantify its effectiveness in CATS project test cases [17]. Vehicle dynamics are modelled through a simple, open “point-mass” model of the vehicle and added sensor/actuator models.

TÜV SÜD used a MATLAB/Simulink-based simulation tool to evaluate driver assistance systems. The tool combines a simple vehicle-dynamic model with sensor, actuator and environment block models enabling study of the system with the required level of accuracy.

ViF Safety-Toolchain is developed by Virtual Vehicle. It is based on the AVL co-simulation platform Model.Connect that allows coupling of sub-models (each with its own solver and level of complexity) in a global framework aimed at simulating active, passive, and integrated vehicle safety systems. In the context of this study, the VuT is modelled using an extended version of the nonlinear single-track model from the CommonRoad vehicle model collection [18] with a Pacejka magic formula tire model, along with actuator/sensor blocks and static environment producing view-blocking obstructions. More details can be found in [19]

The **ZF Tool** is based on IPG CarMaker and enables effectiveness assessments of ADAS in full-vehicle context. The tool makes use of IPG’s detailed 3D multibody vehicle model, including thermomechanical tire models, combined with sensor/actuator block models.

RESULTS

Every participant in this round-robin study performed simulations for all three steps described above using their preferred simulation tool. We then compared the results to gain qualitative and quantitative insights into avoidable and unavoidable variations and adapted the specification accordingly until only unavoidable (i.e., tool-inherent) variations remained.

Unit Tests

Sensor unit test results

Table 2 shows the individual deviations of the simulation results from the analytical solutions and the resulting range for classification time in all 7 tests and full detection time for test 5. The reason for reporting full detection time for test 5 only is that for all other tests full detection time is 0.2s before classification time as, once the cyclist is fully in the field of view, it stays there for the remainder of the test. In test 5, it disappears after less than 0.2s after being fully in the field of view for the first time behind the obstacle and thus is not classified. This happens once it reappears behind the obstacle. Therefore, in this test both values are of interest. The full set of results can be found in Appendix B.

Table 2.
Individual deviations and resulting range of classification time and full detection time (only for test 5) results from analytical solution. Outliers are marked in bold

	Test 1: classification time [s]	**Test 2: classification time [s]**	**Test 3: classification time [s]**	**Test 4: classification time [s]**	**Test 5: full detection time [s]**	**Test 5: classification time [s]**	**Test 6: classification time [s]**	**Test 7: classification time [s]**
Sorted results	0.016	0	0.001	0	0.001	0.001	0.001	0.030
	0.003	0	0.001	0	0	0	0	0.001
	0.001	0	0.001	-0.007	0	-0.006	0.000	0.001
	-0.477	-0.002	0.000	**-0.106**	0	**-0.141**	**-0.177**	0
	-0.487	**-1.053**	**-0.192**	**-0.404**	**-0.150**	**-0.177**	**-0.487**	0
Range	0.503	1.053	0.193	0.404	0.151	0.178	0.488	0.030

Figure 4 shows as an example for time-dependent results these results for test 5 (distance, angle, and speed error). The crosses mark the points in time where the cyclist was classified. The lines represent the individual simulation results. The respective plots of all other tests can be found in Appendix B.

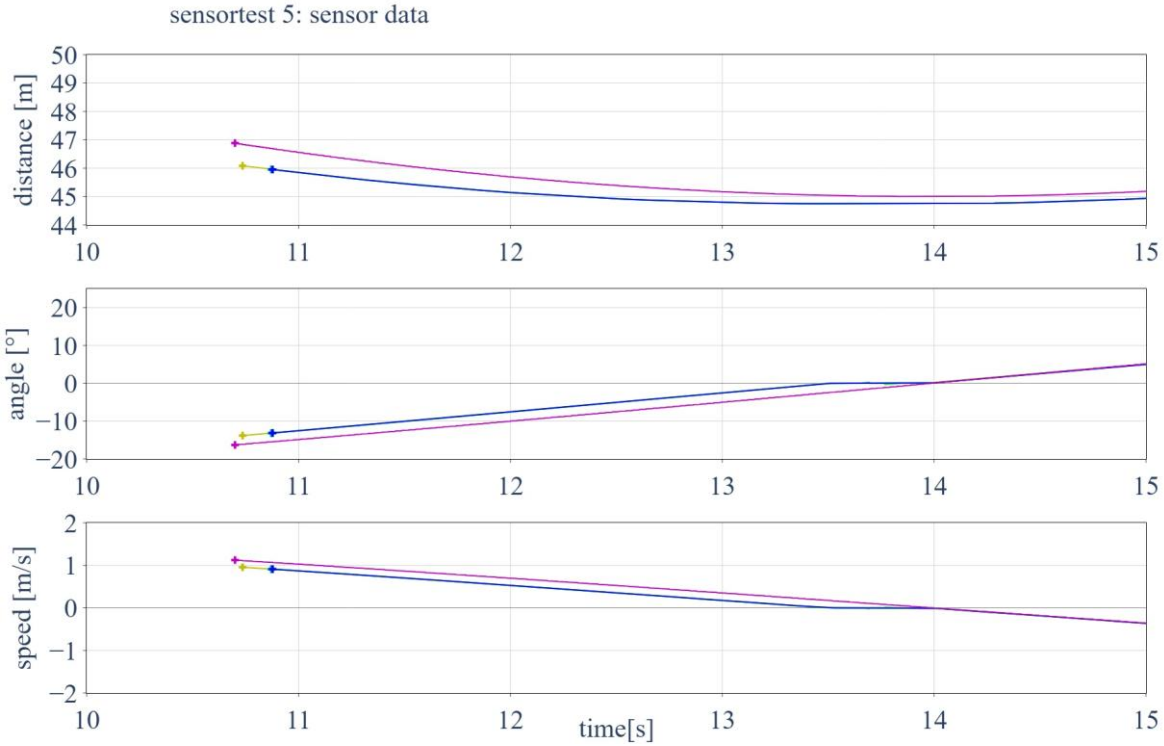


Figure 4. Time-dependent outputs for test 5. The cross marks the time of classification.

Brake test results

The time to stop in the brake test ranges from 2.222s to 2.255s (range of 0.033s) among all participants, the distance to top ranges from 21.702m to 22.016m (range of 0.314m). **Figure 5** shows the time series results for distance, speed, and acceleration for the brake test. The corridor spans the results of all participants. The time series were synchronized by setting t=0 at the time where braking was initiated. The actual deceleration starts 0.2s later due to the implemented brake delay.

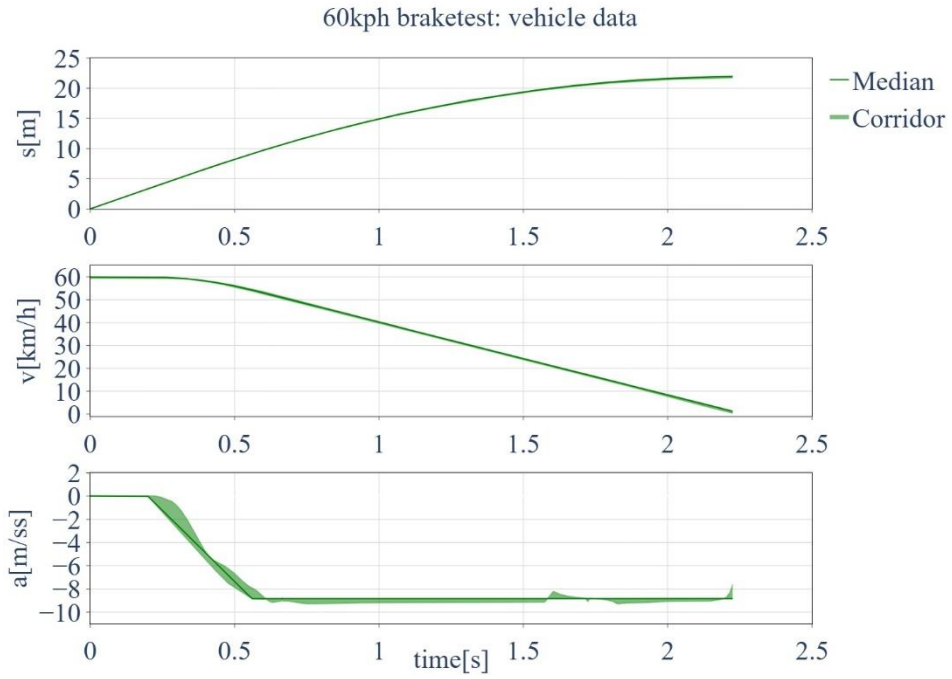


Figure 5. Median and corridor plots of VuT travelled distance, speed, and acceleration for the brake test.

Collision detection unit test results

Table 3 shows the minimum, maximum, and median values of the variations of VuT position at time of collision and the time of collision relative to the analytical solution for all four collision detection unit tests.

Table 3.
Minimum, maximum, range, and median values of VuT position in direction of travel at collision (in [m], relative to analytical solution) and the time at collision (in [s], relative to analytical solution)

		Half Plane (n°1)	**Half Plane (n°2)**	**Square Shape**	**Bicycle Shape**
VuT position at collision deviation [m]	Min	0	0	0	0
	Max	0.010	0.007	0.010	0.010
	Median	0.002	0.003	0.006	0.002
Time at collision deviation [s]	Min	0	0	0	0
	Max	0.009	0.008	0.010	0.010
	Median	0.001	0	0.001	0

Baseline Simulations

Non-optimized baseline simulation results

Table 4 shows the summarized results (minimum, maximum, range) for the non-optimized baseline simulation runs. The values of interest are difference in speed (V_{delta} in the table) of both VuT and cyclist as well as the differences in position (y/x_{delta} in the table) at the time of collision from the speeds and positions given by the respective test case description.

"x" is the VuT's direction of travel in all test cases (CVNB, CVNBO, CVFB, and CVLB) and the cyclist's direction of travel in the CVLB test case. In all other test cases, the cyclist moves in the "y" direction.

The most relevant values are the "cyclist y_{delta}" values for CVNB, CVNBO, CVFB (marked as bold in the table) as they give insight on how much the cyclist's actual hit-point location is off from the defined one. The aim of the subsequent step – optimizing the baseline simulations – is to minimize these deviations as much as possible.

Table 4.
Summarized results for the non-optimized baseline simulation runs.

	V_{delta} to reference speed at collision [km/h]		VUT x_{delta} to reference position at collision [m]		Cyclist y/x_{delta} to reference position at collision [m]	
	VuT, all test cases	**Cyclist, all test cases**	**CVNB, CVNBO, CVFB**	**CVLB**	**CVNB, CVNBO, CVFB (y_{delta})**	**CVLB (x_{delta})**
Min	-0.001	0	-0.001	0	**-0.366**	0
Max	0.036	0	0.017	4.572	**0.275**	3.662
Range	0.037	0	0.018	4.572	**0.641**	3.662

Figure 6 shows the deviations for VuT speed and longitudinal position of all simulation runs from the respective given trajectories and speeds. The corridor spans the range between minimum and maximum deviation per timestep. In addition to corridor and median, the simulation run producing the minimum cyclist y_{delta} value (-0.366m) is plotted (named "min/max" cyclist y_{delta}). The respective test case is "CVFB20" where both VuT and the cyclist travel at 20km/h. The maximum cyclist yd_{elta} value occurs in the "CVNB20" test case where the VuT also travels at 20km/h. As minimum and maximum y_{delta} value are produced by the same simulation tool, the respective VuT deviation curves are identical and thus only one of both (the minimum) result is plotted.

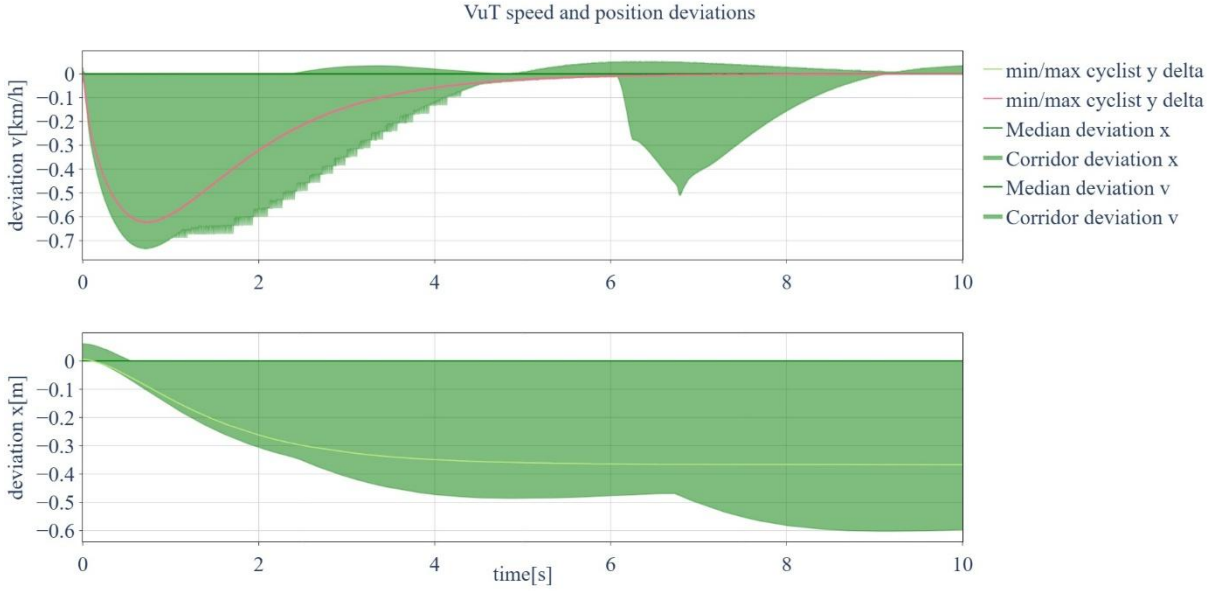


Figure 6. Deviations of VuT longitudinal positions and speeds.

The cyclist position and speed deviations for all participants and simulation runs are negligible (position deviations below ±500µm and speed deviations below ±0.003km/h) and are thus not plotted here. The reason for this can be found in the fact that in all simulation tools participating in this study the cyclist is moved according to the given position over time in the test case description and no dynamic model was used like for the VuT.

Optimized baseline results

Table 5 shows the summarized results (minimum, maximum, range) for the optimized baseline simulation runs. The values of interest are difference in speed (V_{delta} in the table) of both VuT and cyclist as well as the differences in position (y/x_{delta} in the table) at the time of collision. "x" is the VuT's direction of travel in all test cases (CVNB, CVNBO, CVFB, and CVLB) and the cyclist's direction of travel in the CVLB test case. In all other test cases, the cyclist moves in the "y" direction.

Again, as mentioned above, the most relevant values are the "cyclist y_{delta}" values for CVNB, CVNBO, CVFB (marked as bold in the table) as they give insight on how much the cyclist's actual hit-point location is off from the defined one.

Table 5.
Summarized results for the optimized baseline simulation runs.

	V_{delta} to reference speed at collision [km/h]		VUT x_{delta} to reference position at collision [m]		Cyclist y/x_{delta} to reference position at collision [m]	
	VuT, all test cases	Cyclist, all test cases	CVNB, CVNBO, CVFB	CVLB	CVNB, CVNBO, CVFB (y_{delta})	CVLB (x_{delta})
Min	-0.028	-0.024	0	-0.66	**-0.004**	-0.66
Max	0.028	0.012	0.017	0.01	**0.006**	0
Range	0.056	0.036	0.017	0.67	**0.01**	0.66

Treatment simulation results

Table 6 shows the summarized treatment simulation results (minimum, maximum and range) for the "medium FoV" setup, the "large FoV" setup and both setups together.

Table 6.
Summarized treatment simulation results.

Medium (2 x 30º)	**Classification rate**	**Firing rate**	**Avoidance rate**	**Average speed reduction [km/h]**
Min	74%	68%	41%	11.8
Max	76%	68%	44%	11.9
Range	**3%**	**0%**	**3%**	**0.1**
Large (2 x 60º)				
Min	100%	100%	74%	26.2
Max	100%	100%	76%	28.2
Range	**0%**	**0%**	**3%**	**2.0**
All sensor setups				
Min	87%	84%	57%	19.0
Max	88%	84%	60%	20.0
Range	**1%**	**0%**	**3%**	**1.1**

Table 7 shows the summarized conformity rates for the two sensor setups individually, and for the full set (both setups, i.e., 68 cases). A value of 91% means that in 31 of 34 cases the conformity rate was 100%.

Most of the individual (per event and per test case) conformity rates are 100%., only one detection event and 6 avoidance events resulted in a conformity rate of 60% (4 "yes", 1 "no").

Table 7.
Conformity rates for medium only, large only and all sensor setups.

conformity rate	classified	fired	no collision
Medium (2 x 30º)	97%	100%	91%
Large (2 x 60º)	100%	100%	91%
All setups	99%	100%	91%

DISCUSSION

Unit Tests

Sensor unit tests

The results in **Table 2** show considerable ranges for each of the tests apart from test 7 where the cyclist already starts in the FoV. In most cases, this might not be an issue, but when classification is close to firing TTC, this could produce variations due to differences in time of firing.

The reason for these variations lies in differences in how the event "cyclist is fully in FoV" is determined in the tools. Three different approaches were used by the participants:

- **"Top-view based approach "**: This is the most common approach. It checks whether the whole outline of the cyclist is within the sensor cone, see also **Figure 7**, right side.
- **"Ray-based approach"**: Here rays are cast from the sensor's origin spanning the whole opening angle range. If all intersection points between cyclist geometry and ray array are within the given sensor range, the cyclist is assumed to be fully in view, see also **Figure 7**, left side.
- **"Camera model approach"**: This model cannot exactly be set-up as geometric sensor models. It counts the number of pixels of an object and if this is above a given threshold, the object is perceived as "cyclist is fully in FoV".

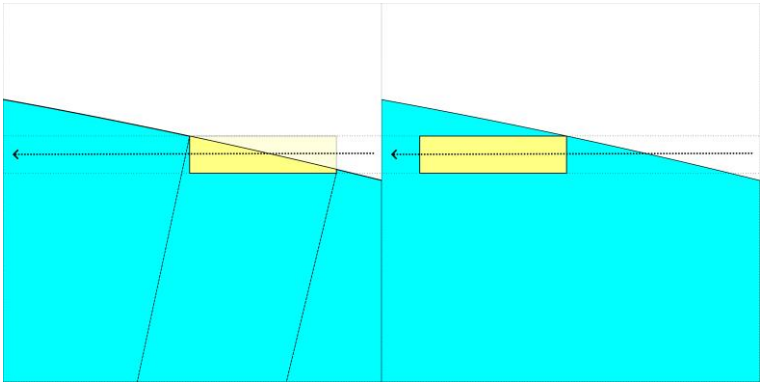


Figure 7. Position of the cyclist (yellow) at the point in time where it is fully in FoV (cyan). Left side for ray-based approach, right side for top-view approach.

Table 2 also lists all individual deviations from the analytical solution. We have several outliers. Most of them are result of camera model and ray-based approaches as they use differing approaches for the „cyclist is fully in FoV" criterion than the top-view approach (which we also used for the reference as this is also the criterion given in the sensor specification (Appendix A). This especially applies to tests 1-3 and 6. We have two smaller outliers in the view-blocking obstruction tests for a top-view approach which are related to how view-blocking obstructions are handled. The rest of the values lie between -0.007 and 0.03 seconds.

Another sensor model inherent source of variation is the way the relative distance to the cyclist is determined. Most of the used tools take the closest point of the cyclist geometry to determine the distance while one tool uses the cyclist geometry's reference point which is located on the longitudinal centerline 1.04 m behind the front line. This can be seen in **Figure 4**, top plot. Variations in the relative distance calculation have a direct impact on TTC calculation (see Equation 1)

$$TTC = \frac{d_{rel}}{v_{rel}} \qquad \text{Equation (1)}$$

More details on the TTC calculation method can be found in Appendix A, "Algorithm".

Brake test

The results show variations, both in time (0.033 s) and distance (0.314 m). When looking at the acceleration time series corridor, variations – especially while build-up – can be seen. Three out of five participants use point-mass models which produce ideal deceleration behavior (i.e., perfect replication of the given deceleration profile) These results also form the median curve. The upper and lower bounds of the corridor stem from more complex vehicle dynamics models.

The resulting variation is a direct consequence of the usage of different vehicle dynamics models and thus a tool-related variation which cannot be eliminated by more strict specifications.

Collision detection unit test

Regarding the VuT position at time of collision, all values lie in the range of 0.01 m. This is the distance travelled in one timestep (1ms) at the given speed of 10 m/s. We consider the range of one timestep as acceptable variation as everything below this value can be attributed to small numerical variations which cannot be influenced.

Regarding the time values, larger variations up to 10 timesteps of one ms can be seen. This value is much larger than the variation in timesteps calculated from distance variations above. This can be explained by the fact that here also variation in VuT speed before the collision plays a role – an unwanted source of variation and uncertainty for this test. We thus consider the distance results to be the relevant ones.

Baseline Simulation

Non-optimized baseline simulation runs

The VUT x_{delta} (CVNB, CVNBO, CVFB) minimum and maximum values in **Table 4** are in the expected range. Like in the collision detection unit test results, the errors are smaller than or equal to the travelled distance in one timestep.

The cyclist y_{delta} for CVNB, CVNBO, CVFB test cases is a direct consequence of VuT longitudinal position error as shown in **Figure 6**, bottom plot. This should be explained using the min/max cyclist y_{delta} curve shown in this **Figure 6**: The deviation in x at 10s (this is the point in time where the collision should occur according to the given trajectories) is -0.367m, i.e. the VuT is 0.367m behind the given position and arrives too late at the intended point of collision. As the VuT travels at 20km/h in positive x-direction in this test case (CVFB20), the respective time needed for the VuT to reach the collision position is 0.367m * 20/3.6m/s=0.066s. The cyclist travels at 20km/h in negative y-direction in the CVFB test case. Thus, the cyclist would cover 0.367m in the given time of 0.066s which is 0.001m away from the respective cyclist delta y value (-0.366m). The minus is a result of the travel direction (negative y) indicating that the cyclist travelled too far compared to the given trajectory before collision occurs. A similar argumentation explains the maximum value of cyclist y_{delta} (0.275, coming from CVNB20). In that case, the cyclist travels at 15km/h in positive y-direction, the VuT again at 20km/h in positive x-direction, resulting in a covered distance by the cyclist of 0.275m in 0.066s.

The positional deviations in the CVLB test case appear large but have no impact on the precision of the crash configuration as in this category both VuT and cyclist travel in the same direction.

The top plot in **Figure 6** shows the source of deviation from the given crash configuration: Variations in VuT speed result in Vut longitudinal position deviations and thus crash configuration deviations. In the worst case, these crash configuration variations could result in having a collision or not even when applying an identical treatment measure as in one case it might be a close avoidance because the cyclist has travelled further than defined in the given test case description and thus has already left the VuT corridor when the VuT is at the position of intended collision. In the other case it might be a collision as intended by the test case description.

More complex vehicle dynamics models require a speed controller and cannot keep the given speed during the whole simulated time. Especially at the start of the simulation, such deviations occur. This is a source of variation that cannot directly be eliminated, as it is tool inherent. However, mitigation strategies exist like modifying the VuT's initial position, using position control instead of speed control or increasing the simulation time so that the VuT is already at the required speed when reaching the starting position of the trajectory given by the test case description.

Optimized baseline simulation runs

The results of the optimized baseline simulation runs show clearly the improvement that can be achieved by adapting the initial conditions so that the crash configurations given in the test case descriptions are met as closely as possible. We were able to reduce the extreme values of the cyclist's lateral position deviations from (-0.366 / +0.275) to (-0.004 / +0.006) which is in the range of one timestep. This result ensures that the basis for the treatment simulations of all participants is the same and thus eliminates an unwanted source of variation in collision avoidance and speed reduction by braking.

Treatment Simulation

The treatment simulation results in **Table 6** show low variations of the discrete values (classification- detection- and avoidance rate) with a maximal range of 3% (avoidance rate for all sensor setups and classification rate for medium sensor setup). The same holds for the average speed reduction where the maximum range is 2 km/h (large sensor setup, 28.2 - 26.2 km/h). These values are much lower than in the previous round-robin study [1] where the range for avoidance rate was 36% and for average speed reduction 9.1 km/h for all sensor setups. The substantial reduction can be attributed to the steps made before the actual treatment simulations:

- extension of the test specification by describing how TTC should be calculated to ensure that this source of variation is eliminated/minimized.
- the unit tests which helped us understand and reduce tool-inherent source of variations.

- the baseline simulations where the optimization step helped in reducing the variations in baseline crash configuration which has a crucial impact on the variations in the treatment simulation results.

LIMITATIONS

The study focused on a limited set of test cases with straight line driving only, thus it was not possible to investigate lateral dynamics and steering control as source of variation for test cases with curved trajectories or safety measures that include steering like autonomous evasive steering (AES) systems.

Apart from the simplified unit tests there is no ground truth that the simulation results could be compared to, thus only relative comparisons between tools can be made. Idealized mathematical solutions can be calculated for the treatment simulations, but they do not reflect the real-world performance of vehicles equipped with such systems. Real-world tests on the other hand show variations even if the exact same test is repeated due to changes in conditions that cannot be fully controlled like temperature, changes in lighting conditions etc.

Most of the tools used are in-house developments, only one commercial tool was included. It might be of common interest to extend the list of tools to tools with a wider user group, such as commercial and open source tools like openPASS [20] or Carla [21] – additional participants a very welcome.

CONCLUSIONS AND FUTURE WORK

This study extends the work done in our previous study. First, we added unit tests because it is easier to analyze smaller models instead of looking at the global outcome and have a backwards search for where variations may come from. The sensor unit tests revealed the biggest variations, both in detection/classification but also in distance calculation. The first variation only becomes relevant in situations where time of detection/classification is close to the trigger time of the safety system, the second one is more critical, as it has direct impact on TTC calculation and thus triggering of the safety system. This is a tool-inherent source of variation which cannot be eliminated in most cases. Variation in braking was detected stemming from the application of different models for vehicle dynamics. This is also a tool-inherent variation that cannot be eliminated. We found collision detection to be well in line amongst partners, as the collision flags were refreshed at the same pace (or at the same timesteps) as the vehicle dynamics.

Then we also had a closer look at the baseline simulations to quantify and minimize respective variations. We found relevant variations in relative positions at the time the collision if initial conditions (positions, speeds) given by the test case specifications were used as starting conditions for the simulations. We identified speed control and complex vehicle dynamics models as source of the resulting variations as these models need time to reach a stable state. Point mass models can replicate the given test case specifications much better but are also further away from real vehicle behavior. This is also a tool-inherent source of variation, but the variation can be minimized by adapting the initial positions to compensate for the errors made by non-constant speed. This was done in the second loop of baseline simulations where all partners optimized their initial conditions with the aim to minimize the variation in relative positions at time of collision. By this step we were able to reduce this variation to the range of one timestep – the same range as obtained in the collision detection unit tests.

Finally, we performed treatment simulations including all improvements made in the previous steps. As a result, the variations in treatment performance were much smaller than in the previous study, that did not include the steps described above.

This study clearly shows that a precise specification of a virtual test and preparation steps before performing the actual treatment simulations can help to improve comparability between test runs executed with different tools. However, a certain amount of variation remains due to tool-inherent features that usually cannot be easily changed by users. The most relevant sources we found are sensor modelling (mainly distance calculation but also detection/classification) and the vehicle dynamics model (braking behavior but also speed control). While the first only has impact on the treatment simulation results, the second could also impact the baseline simulation results and thus make comparison between studies even more difficult.

Round-robin tests can help build trust in virtual testing as they make results generated by different organizations using different simulation tools more comparable. The more precise and unambiguous the specification is, the less variation can be expected due to different interpretations of the specification. However, it must be stated that round-robin tests as described here do not aim to provide a ground truth but rather give insights on variations to be expected when using different tools in virtual testing.

Possible next steps are extending the method to other, more complex use-cases and inclusion of more participants and tools. Anybody interested is kindly invited to contact the authors. Another follow up idea is to set up an online tool where users can compare the outcomes of their virtual testing campaign to anonymized results of other participants.

ACKNOWLEDGEMENTS

We would like to thank Harald Kolk for setting up the base of the sensor unit tests and for providing the script to generate the respective analytical solutions.

This project has received funding from the European Union under grant agreement No. 101075068. The authors acknowledge all partners involved in V4SAFETY and especially in WP5for their contributions to the content presented in this paper. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Climate, Infrastructure and Environment Executive Agency (CINEA). Neither the European Union nor the granting authority can be held responsible for them.

REFERENCES

[1] Wimmer, P., Düring, M., Chajmowicz, H., Granum, F., King, J., Kolk, H., et al., "Toward harmonizing prospective effectiveness assessment for road safety: Comparing tools in standard test case simulations," *Traffic Injury Prevention* 20, no. sup1 (2019): S139–S145, doi:10.1080/15389588.2019.1616086.

[2] Thorn, E., Knisley, V., Auchter, J., "Advancing Verification and Validation for ADAS and ADS," *SAE Technical Papers* (2025): , doi:10.4271/2025-01-8052.

[3] Kolk, H., Kirschbichler, S., Tomasch, E., Hoschopf, H., Luttenberger, P., Sinz, W., "Prospective Evaluation of the Collision Severity L7e Vehicles Considering a Collision Mitigation System," *Transportation Research Procedia* 14, no. (2016): 3877–3885, doi:10.1016/J.TRPRO.2016.05.474.

[4] Scanlon, J.M., Sherony, R., Gabler, H.C., "Injury mitigation estimates for an intersection driver assistance system in straight crossing path crashes in the United States," *Traffic Injury Prevention* 18, no. (2017): S9–S17, doi:10.1080/15389588.2017.1300257.

[5] Sander, U., "Opportunities and limitations for intersection collision intervention—A study of real world 'left turn across path' accidents," *Accident Analysis & Prevention* 99, no. (2017): 342–355, doi:10.1016/J.AAP.2016.12.011.

[6] Chajmowicz, H., Saadé, J., Cuny, S., "Prospective assessment of the effectiveness of autonomous emergency braking in car-to-cyclist accidents in France," *Traffic Injury Prevention* 20, no. sup2 (2019): S20–S25, doi:10.1080/15389588.2019.1679797.

[7] Saadé, J., Chajmowicz, H., Cuny, S., "Prospective evaluation of the effectiveness of autonomous emergency braking systems in increasing pedestrian road safety in France," Conference proceedings International Research Council on the Biomechanics of Injury, IRCOBI, 2019.

[8] Saadé, J., Chajmowicz, H., Cuny, S., "Prospective Evaluation of Autonomous Emergency Braking systems Performance in Car-to-Powered Two-Wheelers Accidents in France," Conference proceedings International Research Council on the Biomechanics of Injury, IRCOBI, 2022.

[9] Page, Y., Fahrenkrog, F., Fiorentino, A., Gwehenberger, J., Helmer, T., Lindman, M., et al., "A comprehensive and harmonized method for assessing the effectiveness of advanced driver Assistance Systems by virtual simulation: The P.E.A.R.S. initiative," Proceedings of the 24th International Technical Conference on the Enhanced Safety of Vehicles (ESV), 2015, http://www-esv.nhtsa.dot.gov/Proceedings/24/isv7/main.htm.

[10] Alvarez, S., Page, Y., Sander, U., Fahrenkrog, F., Helmer, T., Jung, O., et al., "Prospective Effectiveness Assessment of ADAS and Active Safety Systems via Virtual Simulation: A Review of the Current Practices," 25th International Technical Conference on the Enhanced Safety of Vehicles (ESV), 2017, http://www-esv.nhtsa.dot.gov/Proceedings/25/25ESV-000346.pdf.

[11] Standardisation, I.O. for, "ISO/TR 21934-1:2021 - Road vehicles — Prospective safety performance

assessment of pre-crash technology by virtual simulation — Part 1: State-of-the-art and general method overview," 2021.

[12] Standardisation, I.O. for, "ISO/TS 21934-2:2024 - Road vehicles — Prospective safety performance assessment of pre-crash technology by virtual simulation — Part 2: Guidelines and requirements for application," 2024.

[13] Singh, H., Midlam-Mohler, S., Tulpule, P., "Simulation Based Virtual Testing for Safety of ADAS Algorithms - Case Studies," *SAE Technical Papers* 2021 (2021): , doi:10.4271/2021-01-0114.

[14] Euro NCAP, "Safe Driving & Crash Avoidance Virtual Testing," 2025.

[15] Wimmer, P., Camp, O. den, Weber, H., Chajmowicz, H., Wagner, M., Lorente Mallada, J., et al., "Harmonized approaches for baseline creation in prospective safety performance assessment of driving automation systems," *27th International Technical Conference on the Enhanced Safety of Vehicles (ESV)* (2023): 1–15.

[16] Op den Camp, O., Montfort, S. van, Uittenbogaard, J., Welten, J., "Cyclist target and test setup for evaluation of cyclist-autonomous emergency braking," *International Journal of Automotive Technology* 18, no. 6 (2017): 1085–1097, doi:10.1007/S12239-017-0106-5/METRICS.

[17] Gelder, E. de, Fusco, M., "AEB Simulation Tool, TNO 2015 R11642," 2015.

[18] Althoff, M., Koschi, M., Manzinger, S., "CommonRoad: Composable benchmarks for motion planning on roads," IEEE Intelligent Vehicles Symposium, Proceedings, 2017, doi:10.1109/IVS.2017.7995802.

[19] Kolk, H., Wimmer, P., Lorente Mallada, J., "A Baseline Definition and Result Weighting Method for Prospective Effectiveness Assessment of ADAS for VRU Protection," Proceedings of the FISITA - Technology and Mobility Conference Europe 2023, 2023, doi:10.46720/fwc2023-sca-003.

[20] Dobberstein, J., Bakker, J., Wang, L., Vogt, T., Düring, M., Stark, L., et al., "The Eclipse working group openPASS - an open source approach to safety impact assessment via simulation," 25th International Technical Conference on the Enhanced Safety of Vehicles (ESV), 2017.

[21] Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A., Koltun, V., " CARLA: An Open Urban Driving Simulator," Proceedings of the 1st Annual Conference on Robot Learning, 2017.

APPENDIX A – ADDITIONAL DEFINITIONS FOR THE ROUND-ROBIN STUDY

VuT Geometry

All geometry definitions are given in a 2-dimensional top view, as for the round-robin the third dimension (height) has not been considered. A generic vehicle geometry is used. The vehicle has a length of 4.5 m, and a width of 1.9 m. Special attention is given to the vehicle front structure, which is rounded and linearly approximated between six points, see **Figure A 1**.

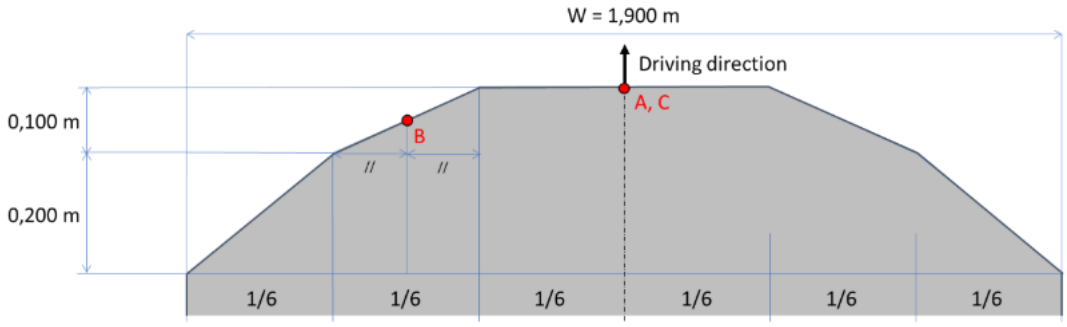


Figure A 1. Vehicle rounded front structure geometry definition.

Cyclist Geometry

The geometry of the cyclist including the bicycle is considered constant for each test. Since the round-robin does not cover detailed sensor modelling and simulation, it is possible to consider a geometry that is simplified as an object based on three rectangles (**Figure A 2**). The first rectangle describes the front wheel of the bicycle (width 0.10 m, length 0.61 m), followed by the middle rectangle describing the cyclist on the bicycle (width 0.50 m, length 0.70 m) and the rearmost rectangle describing the bicycle's rear wheel (width 0.10 m, length 0.64 m). The zero-reference point for the cyclist ("O" in **Figure A 2**) is located at the bike's crank shaft.

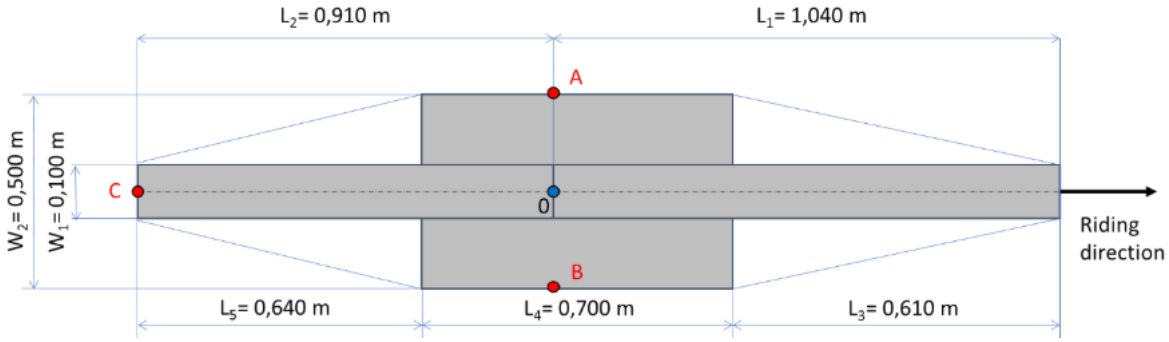


Figure A 2. Cyclist target geometry definition.

Test Case Descriptions and Collision Configurations

In the **CVNB** test case, the VuT travels forward in a straight line towards a cyclist crossing its path cycling (in a straight path) from the near side. The cyclist's zero reference point (point "A" in **Figure A 2**) must be at 50% of the VuT width (point "A" in **Figure A 1**) at time of collision as depicted in **Figure A 3** (left side). In this test case, no view-blocking obstruction is present and there is a direct line of sight between the VuT and the cyclist for at least four seconds before the possible collision. During these four seconds, the cyclist speed and direction do not change; neither does the direction and speed of the VuT apart from possible AEB activation.

In the **CVNBO** test case, a view-blocking obstruction is located at DO1 = 3.55 m from the center line of the VuT and DO2 = 4.80 m from the left side of the cyclist. The obstruction completely blocks the view, so no single part of the cyclist can be seen from the perspective of the VuT during the complete approach, until the cyclist appears from behind the obstruction. The relative positions of VuT and cyclist at time of collision identical to the ones in the CVNB test case (depicted in **Figure A 3**, left side).

CVFB describes a test case in which the cyclist approaches from the far side. In this test case, no view-blocking obstruction is present. The cyclist's zero reference point (point "B" in **Figure A 2**) must be at 25% of the VuT width (point "B" in **Figure A 1**) at time of collision as depicted in **Figure A 3** (right side). Please note that in

this case the “B” points of VuT (see **Figure A 1**) and cyclist (see **Figure A 2**) cannot be in the same location due to geometrical/collision constraints and only should have the same position in the VuT’s lateral direction.

The fourth test case **CVLB** considers a sheer longitudinal configuration in which the VuT travels forward towards a cyclist that is moving in the same direction in front of the VuT. The cyclist’s zero reference point (point “C” in **Figure A 2**) must be at 50% of the VuT width (point “A” in **Figure A 1**) at time of collision as depicted in **Figure A 3** (center). This configuration is common when cyclists swerve in front of a car in the inner city.

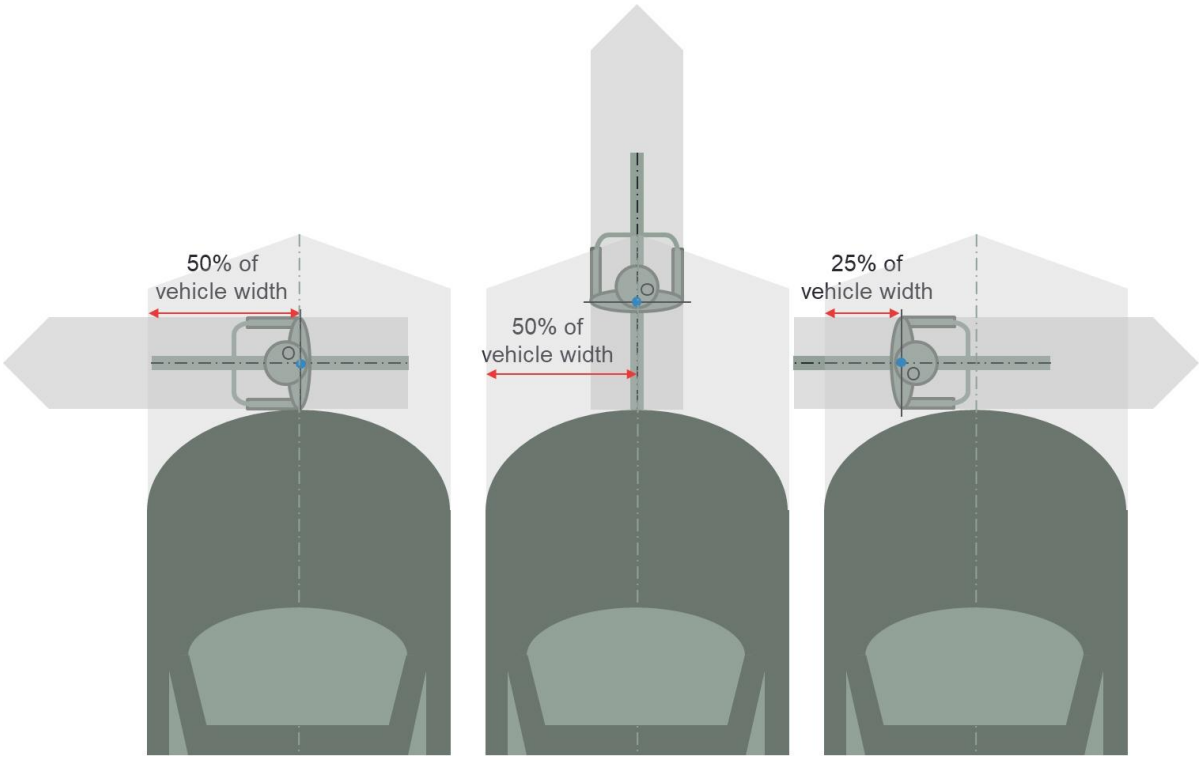


Figure A 3. VuT and cyclist relative position at time of collision for the CVNB and CVNBO (left side), CVLB (middle) and CVFB (right side) test cases (without braking action).

AEB System

Sensor

The evaluation of the sensor is done in 2D, as most of the current simulation methods do not consider heights. **Table A 1** shows an overview of the sensor parameters that will be used for simulation.

Table A 1. Definition of sensor parameters.

Sensor	Sensor unit tests	"Medium FoV"	“Large FoV”
Opening Angle	2 x 45° (90°)	2 x 30° (60°)	2 x 60° (120°)
Range	80m	80 m	80 m
Mount (longitudinal)	Front		
Mount (lateral)	Mid		
Object detection	100 % in view for classification		
Acquisition time	0.2 s		

The opening angle is defined as shown in **Figure A 4**. It is defined as the angle between the left and right outer boundaries of the FoV. For the study, three different opening angles (60°, 90°, 120°) are considered. The range of the sensor is defined as the longitudinal distance between the mounting point of the sensor at the vehicle and

the most forward point of the sensor FoV. The range is defined as 80 m for all two sensors. The detection area of the sensor is defined as circle sector with range as radius and opening angle to define the angular extent of the sector. The mount-point of the sensor is defined as the point where the sensor is attached to the vehicle. For all sensors, the mount-point is in the middle of the front of the vehicle, with the symmetry axis parallel to the vehicle symmetry axis. The object, i.e., the cyclist, is detected as soon as all corners of the cyclist (bicycle plus cyclist) are within the field of view. That means that 100 % of the area of the cyclist needs to overlap with the field of view (from a top view perspective). The point in time when the cyclist comes completely into the field of view for the first time during the approach of the car and the cyclist is defined as the time of detection ($t_{detection}$). The Time-To-Collision (TTC) value at which the cyclist comes completely into the field of view for the first time is defined as $TTC_{detection}$. After the cyclist is detected, the cyclist needs to stay in the field of view for 200 ms to be classified. After these 200 ms, the object is detected, classified and a notification will be sent to the algorithm, that there is a cyclist. This value is defined as $TTC_{classification}$.

If it happens that the cyclist is not any longer 100% fully in the field of view, the cyclist is not detected any longer and by that also classification is lost.

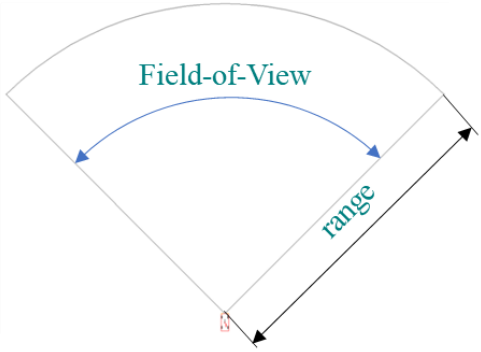


Figure A 4. Definition of sensor shape, angles, range, and mount point.

Algorithm

A TTC based algorithm calculates in every time step the time-to-collision based on the assumption, that both collision partners, cyclist and car, move on with their current speed and direction. The calculation of TTC should be performed using this equation:

$$TTC = \frac{d_{rel}}{v_{rel}}$$

d_{rel} and v_{rel} are explained in **Figure A 5**.

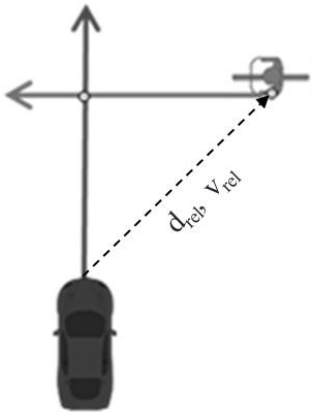


Figure A 5. Input values for TTC calculation: d_{rel} is the current minimal distance from sensor location to cyclist profile, v_{rel} is the time derivative of d_{rel}.

As the simulation does not consider any sensor errors, inaccuracies, or delays, the estimated TTC equals the real TTC in every time step. The algorithm triggers the actuator at a TTC of 1 s (under the condition that the cyclist is classified at this point in time). Once the algorithm has triggered the actuator, the vehicle will continue braking

until standstill, even if the trigger condition is not met any more (e.g., because the TTC rises above the threshold due to the braking action or because the classification is lost again)

Actuator

The simulated actuator is limited to a brake, which is simplified by prescribing the deceleration profile upon actuation according to the following parameters:

- Max deceleration: 1.0 g
- Time to max. deceleration (Build-up time): 0.4 s
- Deceleration increase over time is linear until the maximum value is achieved.
- Actuator delay: 0.2 s

Figure A 6 shows the deceleration profile over time. The blue lines represent the required deceleration. After the trigger of the algorithm, the brake actuator has a delay of 200 ms, in which no deceleration is built-up. This delay represents the time required for brake pressure build-up and movement of the brake pads to the brake disc. Within the next 400ms, the deceleration is built up to a maximum deceleration of 1.0 g. This equals a brake gradient of 24.525 m/s3 (see blue solid lines). As the tire-road friction coefficient is assumed to be 0.9, it is not possible to gain a deceleration above 0.9 g. Therefore, the realistic maximum deceleration is 0.9 g and will be reached within 360 ms with the same gradient of 24.525 m/s3 (see red dotted lines).

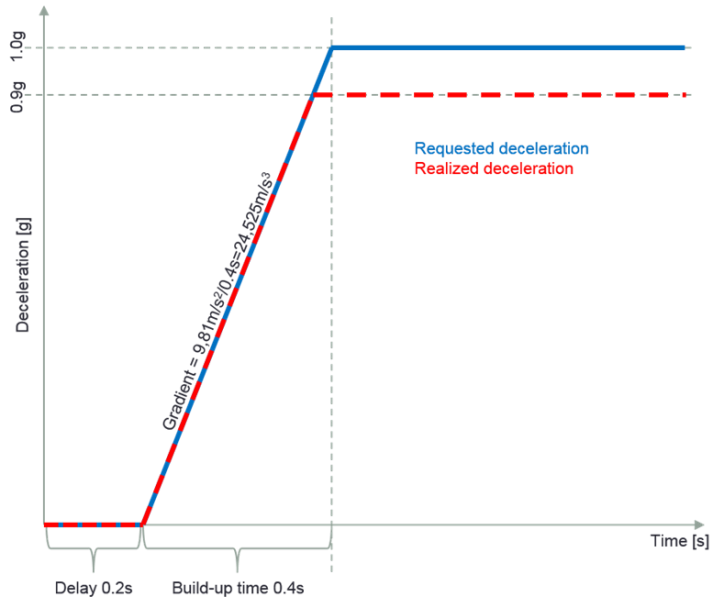


Figure A 6. Actuator deceleration profile.

APPENDIX B - ADDITIONAL RESULT PLOTS AND TABLES OF THE ROUND-ROBIN STUDY

Table B 1 shows the minimum, maximum, range, and median of absolute values and deviations of the simulation results from the analytical solutions for full detection and classification time in all 7 tests.

Table B 1. Minimum, maximum, range, and median values of partial, full detection and classification time (in s, absolute and relative to analytical solutions) for sensor unit tests.

		absolute		relative to analytical solution	
		Full detection time [s]	**Classification time [s]**	**Full detection time [s]**	**Classification time [s]**
Test 1	Min	3.830	4.030	-0.487	-0.487
	Max	4.320	4.533	0.003	0.016
	Range	0.490	0.503	0.490	0.503
	Median	4.317	4.518	-0.238	0.001
Test 2	Min	6.030	6.230	-1.053	-1.053
	Max	7.083	7.283	0.000	0.000
	Range	1.053	1.053	1.053	1.053
	Median	7.083	7.283	-0.001	0.000
Test 3	Min	3.570	3.771	-0.193	-0.192
	Max	3.764	3.964	0.001	0.001
	Range	0.194	0.193	0.194	0.193
	Median	3.763	3.964	0.000	0.001
Test 4	Min	7.980	8.180	-0.404	-0.404
	Max	8.384	8.584	0.000	0.000
	Range	0.404	0.404	0.404	0.404
	Median	8.384	8.581	-0.053	-0.007
Test 5	Min	2.640	10.700	-0.150	-0.177
	Max	2.791	10.878	0.001	0.001
	Range	0.151	0.178	0.151	0.178
	Median	2.790	10.874	0.000	-0.006
Test 6	Min	2.241	2.441	-0.487	-0.487
	Max	2.728	2.929	0.000	0.001
	Range	0.487	0.488	0.487	0.488
	Median	2.728	2.928	-0.089	0.000
Test 7	Min	0.000	0.200	0.000	0.000
	Max	0.030	0.230	0.030	0.030
	Range	0.030	0.030	0.030	0.030
	Median	0.000	0.201	0.001	0.001

Figure B 1 to **Figure B 7** show the time-dependent results for tests 1-7 (distance, angle, and speed error). The crosses mark the points in time where the cyclist was classified. The lines represent the individual simulation results.

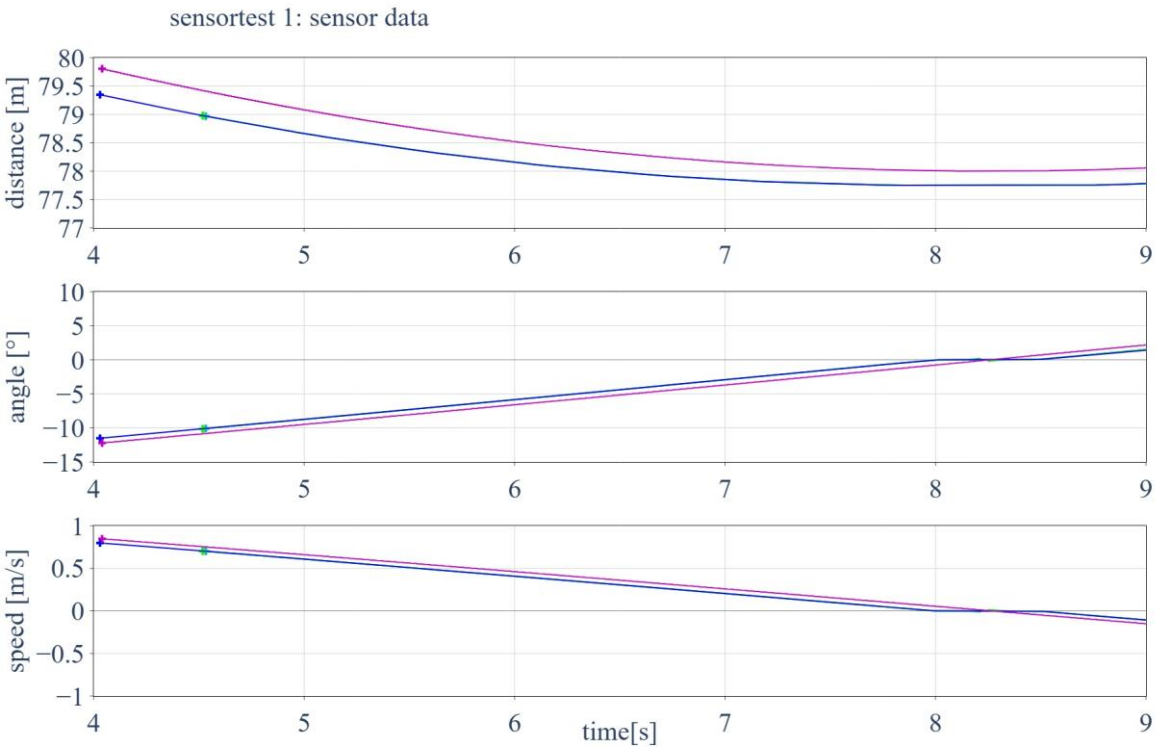


Figure B 1. Time-dependent outputs for test 1. The cross marks the time of classification.

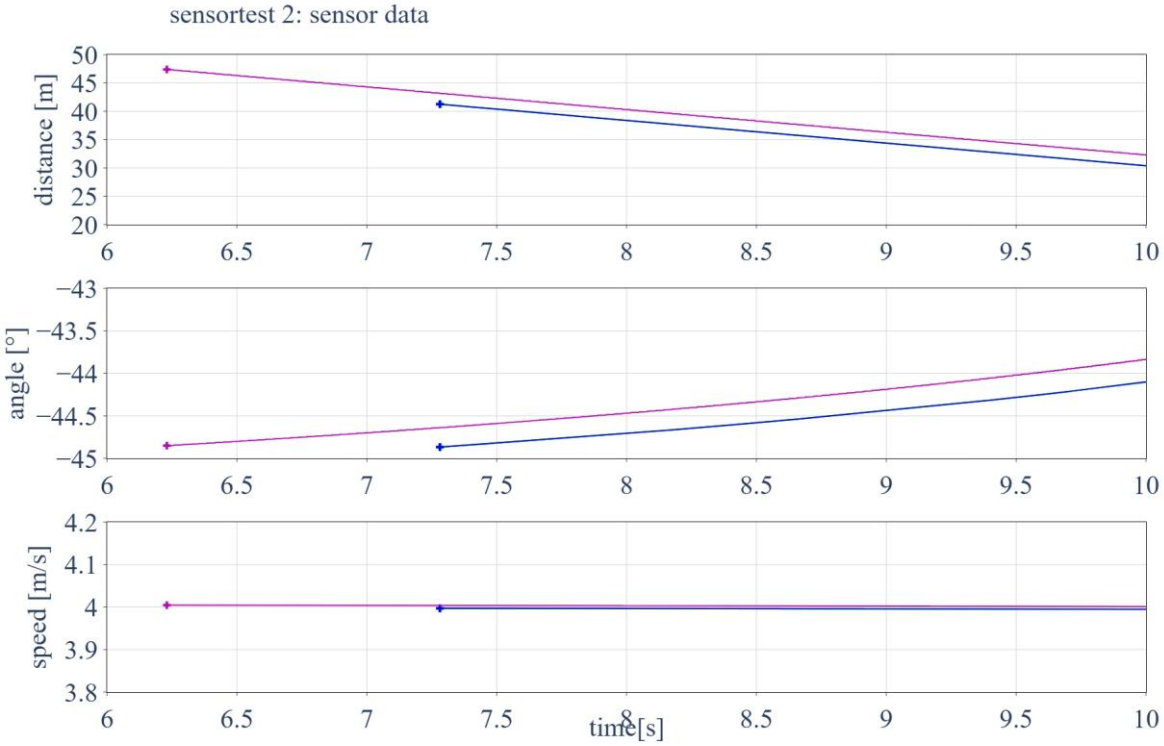


Figure B 2. Time-dependent outputs for test 2. The cross marks the time of classification.

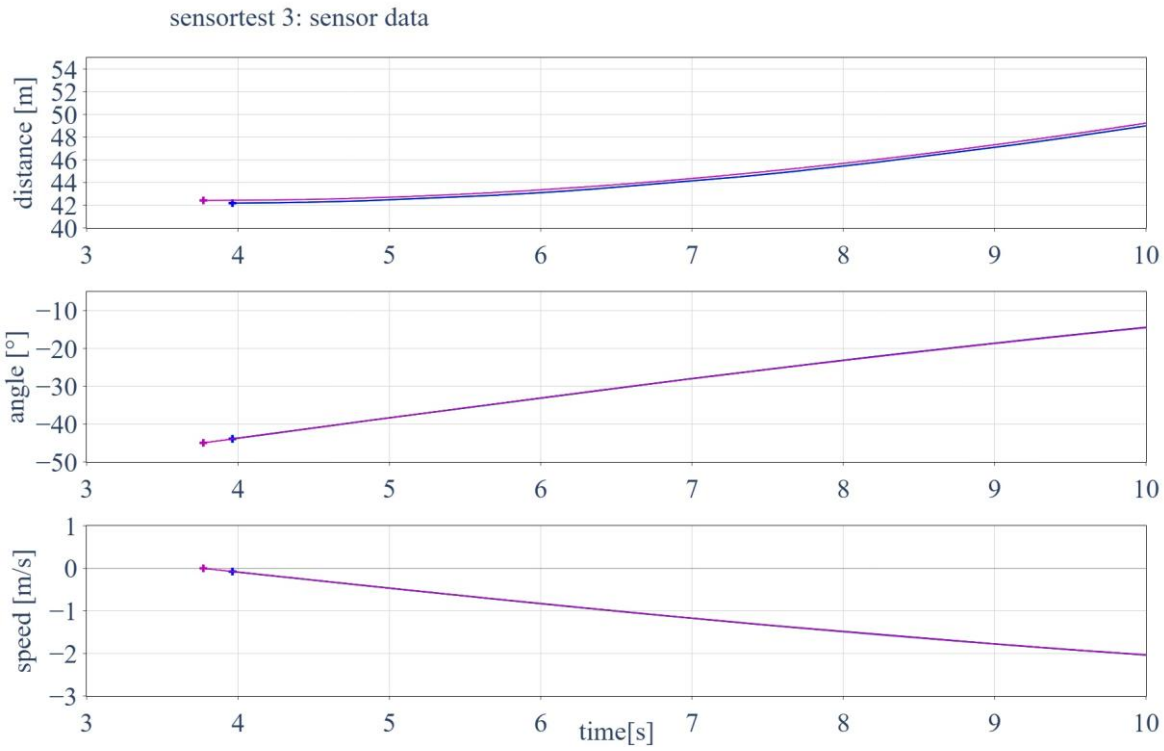


Figure B 3. Time-dependent outputs for test 3. The cross marks the time of classification.

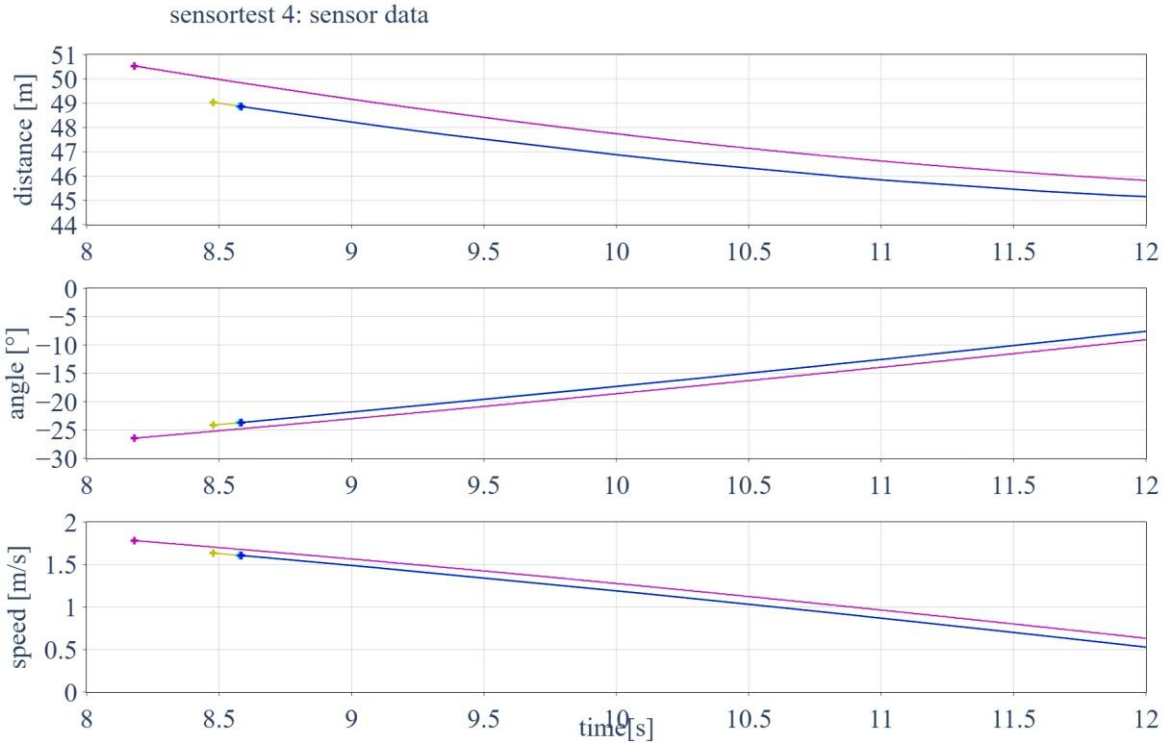


Figure B 4. Time-dependent outputs for test 4. The cross marks the time of classification.

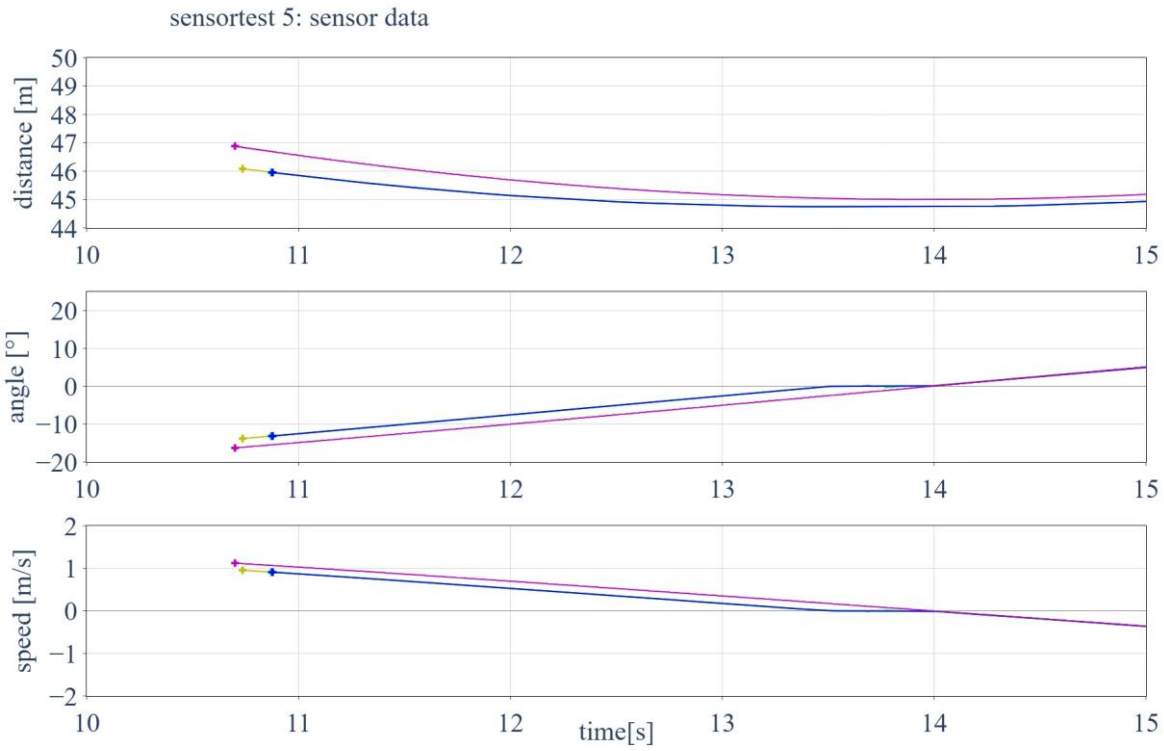


Figure B 5. Time-dependent outputs for test 5. The cross marks the time of classification.

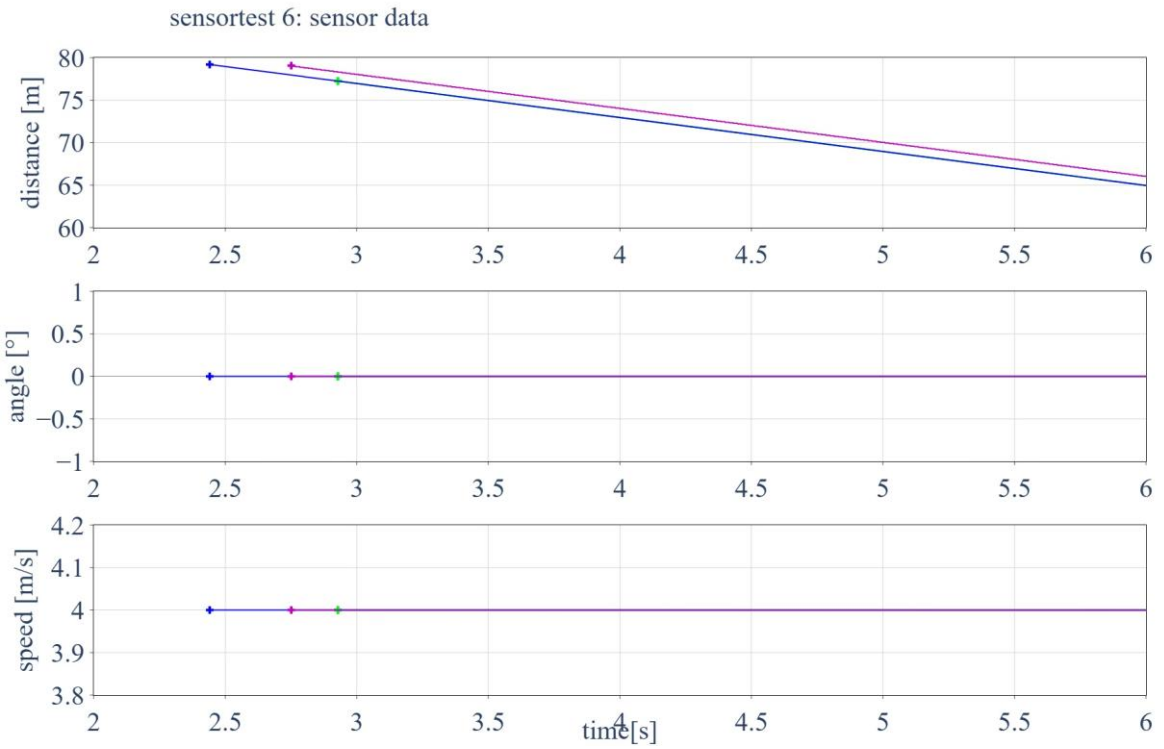


Figure B 6. Time-dependent outputs for test 6. The cross marks the time of classification.

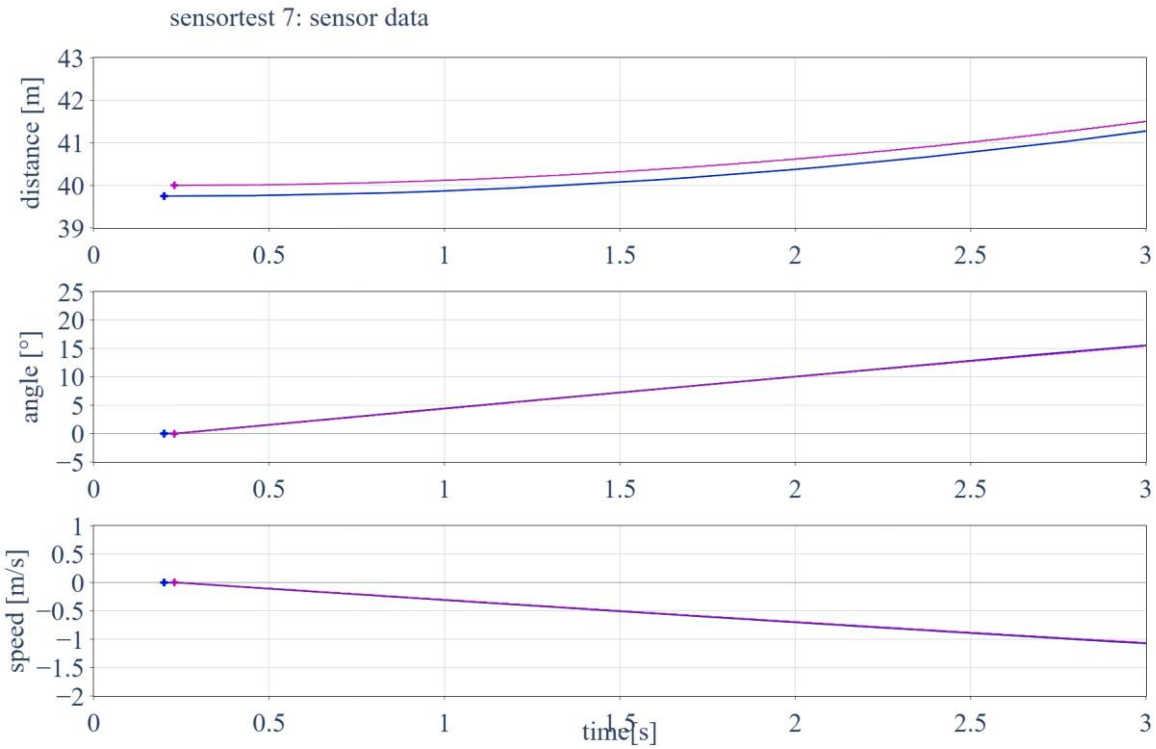


Figure B 7. Time- dependent outputs for test 7. The cross marks the time of classification.