EVIDENT

Enabling VIrtual valiDation & vErificatioN for ADAS and AD features

Public report

Project within: FFI Electronics, Software and Communication
Author: Mari Eriksson, AstaZero
Co-Authors: Vijay Ramnath, Zeekr Technology Europe AB
Cecilia Hernqvist, AstaZero
Seif Bourogaa, AstaZero
Axel Tibbling, Einride Autonomous AB
Anders Andersson, VTI
Daniel Poveda, Chalmers (Revere)
M. Cagri Kaya, Chalmers
Samuel Scheidegger, Asymptotic AI
Hannes Lundberg, Aptiv
Mazen Mohamad, RISE
Beatriz Cabrero-Daniel, University of Gothenburg

Date 2025-01-31

‘

1. Table of Content

2. Summary

The EVIDENT project aims to address challenges in the automotive industry's validation and verification (V&V) processes for advanced driver assistance systems (ADAS) and autonomous driving (AD) features. Traditional V&V methods struggle to keep up with the increasing frequency of software updates. The project explores virtual validation strategies to complement or replace physical testing, thereby enhancing efficiency and safety assurance.

Automotive innovations are increasingly software-driven, necessitating frequent updates. Current validation processes heavily rely on physical testing, which is time-consuming and costly. The project focuses on how vehicle functionalities could be tested and validated in simulation models and what fidelity level that could be reached. By utilizing virtual environments, the project aims to proactively test software functions before deployment, ensuring accurate assessments of system performance in diverse scenarios.

The primary goal is to develop strategies that balance the realism of virtual test environments with practical implementation. Key research questions include:

- What level of realism is required for simulations to be credible for testing edge cases?
- How can virtual testing be integrated with real-world data to discover new edge cases?
- How can virtual testing ensure functional safety to satisfy regulatory bodies?

The project also seeks to establish metrics for comparing physical and virtual test results and to utilize open-source tools for broader industry use.

The project follows a structured approach:

1. **Gap Analysis**: Semi-structured interviews with industry experts were conducted to identify current best practices and challenges.
2. **Simulation Toolchain Assessment**: Each partner's simulation tools, and maturity levels were evaluated.
3. **Scenario Development**: Road network representations and test scenarios were developed using ASAM OpenDRIVE and OpenSCENARIO formats.
4. **Physical Testing**: Various scenarios were tested on the AstaZero proving ground using vehicles equipped with advanced sensors and emergency braking systems.
5. **Simulations**: Partners conducted virtual tests using the respective tool chains. The simulations aimed to replicate physical test conditions and gather comparable data.
6. **Data Comparison**: Physical and simulated test data were compared to evaluate fidelity levels and trustworthiness. Metrics such as time to collision (TTC), braking distances, and object detection errors were analysed.

Five key case studies were tested:

1. Automated Lane Keeping System (ALKS)
2. Car-to-Car Front Turn-Across-Path (CCFTap)
3. Car in Curve
4. S-Curve
5. Occluded Child

Each scenario focused on different aspects of vehicle dynamics, sensor performance, and emergency braking responses. For instance, the Occluded Child scenario tested automatic emergency braking when a child runs out from behind parked cars.

The project identified gaps between physical and simulated test results, such as differences in braking activations between physical test and simulation. It also highlighted the need for improving simulation tools' ability to replicate real-world vehicle behaviour accurately.

Key findings include:

- Virtual tests can be reliable but require tuning to achieve higher fidelity.
- Physical tests remain crucial for validating simulation models.
- Establishing standardized KPIs for virtual testing is essential to enhance credibility.

The project faced several challenges such as:

- Variability in sensor models across partners.
- Human factors introducing inconsistencies in physical tests.
- Limitations of existing simulation tools to accurately replicate real-world scenarios.

A comprehensive list of challenges was compiled to guide future research and development efforts.

EVIDENT successfully demonstrated the potential of virtual validation for ADAS and AD features. The project contributed to developing methodologies for comparing physical and virtual tests and provided insights into the requirements for credible virtual toolchains.

Future research is recommended to focus on refining simulation validation methods, improving data synchronization methods, and addressing identified challenges to make virtual validation a practical and reliable component of automotive software development.

3. Sammanfattning på svenska

EVIDENT-projektet syftar till att ta itu med utmaningar inom fordonsindustrins validerings- och verifieringsprocesser (V&V) för avancerade förarassistanssystem (ADAS) och funktioner för autonom körning (AD). Traditionella V&V-metoder kämpar med att hålla jämna steg med den ökande frekvensen av programuppdateringar. Projektet utforskar virtuella valideringsstrategier för att komplettera eller ersätta fysisk testning, vilket förbättrar effektiviteten och argumentationsbevis för funktionell säkerhet.

Fordonsutveckling blir alltmer programvarudriven, vilket kräver frekventa uppdateringar. Nuvarande valideringsprocesser är starkt beroende av fysisk testning, vilket är tidskrävande och kostsamt. Projektet fokuserar på hur fordonsfunktioner kan testas och valideras med simuleringsmodeller och vilken noggrannhetsnivå som kan uppnås. Genom att använda virtuella miljöer syftar projektet till att proaktivt testa programvarufunktioner innan de implementeras, vilket säkerställer noggranna bedömningar av systemets prestanda i olika scenarier.

Det primära målet är att utveckla strategier som balanserar realismen i virtuella testmiljöer med praktisk implementering. Viktiga forskningsfrågor inkluderar:

- Vilken nivå av realism krävs för att simuleringar ska vara trovärdiga för att testa olika kritiska scenarier?
- Hur kan virtuell testning integreras med verkliga testdata för att upptäcka nya kritiska användarfall?
- Hur kan virtuell testning säkerställa funktionell säkerhet för att tillmötesgå krav från tillsynsmyndigheter?

Projektet syftar också till att etablera mätvärden för att jämföra fysiska och virtuella testresultat och att testa hur open-source-verktyg bredare kan användas inom industrin.

Projektet följer en strukturerad metod:

1. **Gap-analys:** Semistrukturerade intervjuer med branschexperter genomfördes för att identifiera nuvarande praxis och utmaningar.
2. **Bedömning av simuleringsverktygskedjan:** Varje partners simuleringsverktyg och mognadsnivå utvärderades.
3. **Scenarieutveckling**: Representation av vägnät och testscenarier utvecklades med hjälp av ASAM OpenDRIVE och OpenSCENARIO-format.
4. **Fysisk testning**: Olika scenarier testades på AstaZeros testbana med fordon utrustade med avancerade sensorer och nödbromssystem.
5. **Simuleringar:** Partners genomförde virtuella tester med respektive verktygskedjor. Simuleringarna syftade till att replikera fysiska testförhållanden och samla jämförbara data.
6. **Jämförelse av data**: Fysiska och simulerade testdata jämfördes för att utvärdera noggrannhetsnivåer och trovärdighet. Mätvärden som tid till kollision (TTC), bromssträckor och avvikelser gällande objekt-detektion analyserades.

Fem viktiga fallstudier testades:
1. Automated Lane Keeping System (ALKS)
2. Car-to-Car Front Turn-Across-Path (CCFTap)
3. Car in Curve
4. S-Curve
5. Occluded Child

Varje scenario fokuserade på olika aspekter av fordonsdynamik, sensorprestanda och bromsfunktioner, AEB. Till exempel testade scenariot Occluded Child automatisk nödbromsning när ett barn springer ut bakom parkerade bilar.

Projektet identifierade gap mellan fysiska och simulerade testresultat, såsom skillnader i aktivering av nödbroms mellan fysisk och simulerad testning. Det betonade också behovet av att förbättra simuleringsverktygens förmåga att exakt replikera verkligt fordonsbeteende.

Viktiga upptäckter inkluderar:
- Virtuella tester kan vara pålitliga men kräver kalibrering för att uppnå högre noggrannhet.
- Fysiska tester förblir avgörande för att validera simuleringsmodeller.
- Att etablera standardiserade KPI:er för virtuell testning är viktigt för att öka trovärdigheten.

Projektet stötte på flera utmaningar såsom:
- Variabilitet i sensormodeller mellan partners.
- Mänskliga faktorer som introducerar osäkerhet i fysiska tester.
- Begränsningar av befintliga simuleringsmodeller för att exakt replikera verkliga scenarier.

En omfattande lista över utmaningar sammanställdes för att vägleda framtida forsknings- och utvecklingsinsatser.

EVIDENT demonstrerade framgångsrikt potentialen för virtuell validering av ADAS- och AD-funktioner. Projektet bidrog till att utveckla metoder för att jämföra fysiska och virtuella tester och gav insikter i kraven för trovärdiga virtuella verktygskedjor. Framtida forskning bör fokusera på att förfina valideringsmetoder för simulering, förbättra datasynkroniseringsmetoder och ta itu med identifierade utmaningar för att göra virtuell validering till en praktisk och pålitlig komponent i mjukvaruutveckling för fordon.

4. Background

Innovations in the automotive industry are increasingly driven by software. Many automotive manufacturers have prepared their vehicles to receive over-the-air (OTA) software updates. The increasing pace of innovation requires new software features to be rolled out more frequently than before.

Rapid software updates challenge traditional validation and verification (V&V) methods. Vehicles are equipped with more sensors and software, increasing the need for testing and accurate information about the surroundings. Traditional testing methods, like driving around to collect data, are insufficient. Alternative strategies include using data from real vehicles, testing in virtual environments, or validating on proving grounds.

To validate system functionality, “edge cases” - rare but challenging situations - are often used. Collecting this data from real vehicles has several issues: it requires many vehicles, the data is mainly useful in open-loop scenarios, and data collection is expensive and time-consuming.

A solution is to use synthetic data and simulation models, where scenarios can be performed, adjusted, and repeated in a controlled environment. The key is to develop validation processes and methods to measure their accuracy. To draw correct conclusions from simulations, their credibility must be determined. This requires methods to analyse and quantify the gap between simulation and reality, both for vehicles and the surrounding environment and traffic.

This project aims at exploring strategies which trades a certain degree of fidelity (how realistic the testing environment is) in a quantifiable way to complement or replace recordings from real traffic situations. This data could be captured by real sensors or by simulative components to turn retroactive V&V strategies, which are driven by real-world recordings only, into a proactive V&V strategy that is benefitting from previous recordings by systematically and plausibly extending them. This would allow for a systematic testing of an adjusted software feature before it is deployed to a vehicle system. Successfully identifying and evaluating such a strategy would allow all involved actors along the value chain of validating and verifying software-driven vehicle features, to more efficiently allocate the right testing resources, in the right quality, and with the expected degree of fidelity at the most pressing challenges driven by information from the field.

5. Purpose, research questions

The purpose of this research project is to explore and develop V&V (validation and verification) strategies that balance the reliability of test environments with the feasibility of implementation in a measurable way. This involves either complementing or replacing traditional data collection from real traffic situations with advanced simulations or a combination of both. To effectively achieve this, the project will conduct similar physical and simulated tests to quantify the gap between these two test environments. By transitioning from current reactive V&V strategies, which rely on real data retrospectively, to proactive strategies that encompass simulated data, the project aims to enable systematic testing of software functions in vehicle systems before deployment.

To guide this exploration, the project is structured around several critical research questions:

1. How "realistic" must the simulation or virtual test environment be for testing "edge cases" to be acceptable in the engineering process?
2. To what extent can simulation/virtual testing be used to evaluate the intended functionality?
3. What is the right approach to integrate virtual test elements with field data to reasonably and validly drive the discovery of new edge cases?
4. How can we take steps towards enabling the use of virtual testing to convincingly assure consumers and regulatory authorities of functional safety?

6. Method

The project method has been to follow a stepwise approach, described below.

6.1 Best practices of gap analysis

First step was to understand the common view and industry best practices of gap analysis between ADAS/AD simulation verification and real-world testing.

We designed semi-structured interviews to gather the current practices and challenges in the automotive industry for developing and troubleshooting simulations for V&V of AD features. The interviewees, selected domain experts predominantly from the automotive industry with at least five years of experience, shared their insights on how the fidelity gap is qualitatively and quantitatively evaluated as of today.

The interviews were transcribed and any information such as names or affiliations were removed to ensure anonymity of the interviewees. The raw transcript data was analysed to extract patterns and themes to qualitatively analyse the data using thematic analysis. This was then followed by a coding phase, in which segments of data were annotated with codes indicating specific ideas and concepts. Codes with overlapping patterns were then organized into potential themes that were rigorously reviewed by the authors in multiple rounds and refined to ensure they authentically represent the data.

To contrast the domain experts' input with recent insights and recommendations as suggested in scientific literature, we surveyed recent studies addressing the fidelity gap for simulations in the automotive industry, especially for AD and ADAS. Prior to the scientific literature review, preliminary searches aimed at both identifying existing systematic reviews and meta-analysis, as well as assessing the volume of potentially relevant studies. Based on these efforts, queries for manual and keyword automated searches were crafted. For the selection of primary studies, we queried four formal databases (i.e., ACM Digital Library, Springer, IEEE Xplore, and ScienceDirect) and one engine (i.e., Google Scholar) that indexes pre-print servers (e.g., arXiv).

6.2 Understand each partners simulation toolchain

The **second** step was to understand each partners simulation toolchain and maturity of simulation verification within workshop discussions and presentations in between partners.

6.3 Understand what type of complexity level project shall aim for

Third step was to understand what type of complexity the partners believe they could deliver simulations at and what **KPI’s** that could be comparable with physical tests. This was investigated in common workshops.

6.4 Develop road network representation and scenarios

Step **Four** was to develop road network representation and scenarios for the five chosen use cases that was distributed to the partners that perform physical and virtual tests.

The formats used for specifying the road network and defining scenarios have been ASAM OpenDRIVE and ASAM OpenSCENARIO. These are commonly known standard formats possible for usage with a wide range of software. The OpenDRIVE files were created in Mathworks RoadRunner and are based on a georeferenced point cloud of AstaZero proving ground, which in turn was developed with data gathered using lidar and camera sensors. The process followed is:

- Decide on available test tracks to use on AstaZero proving ground, suitable for the specific needs of the scenario.
- Scan the test tracks with positioning equipment, lidar and camera sensors to be able to create a georeferenced point-cloud of the environment.
- Use Mathworks Roadrunner to create a road network representation in ASAM OpenDRIVE format, based on the georeferenced pointcloud.
- Develop scenarios matching the chosen use cases in ASAM OpenScenario format, based on the road network representation.

6.5 Performing physical test

Fifth step was to perform physical test at test track AstaZero with at least two partners vehicles, one of which should use an open transparent tool chain.
The detailed method used is described per partner and use case:

Partner Aptiv

Test vehicles play a pivotal role in capturing real-world data under controlled and repeatable conditions. These vehicles are equipped with specialized hardware and instrumentation to record data from various scenarios relevant to the study. Aptiv contributed with a Volvo v40 test vehicle that performed two different scenarios used in the project. This test vehicle was equipped with mrr360 automotive corner radar sensors from Aptiv. These sensors have been carefully placed and calibrated. The vehicle was also equipped with a closed loop AEB system to trigger an emergency break when needed in the certain scenarios.

Partner AstaZero:

To perform the physical tests, possible test tracks was identified from the variety of tracks available at AstaZero proving ground, as well as necessary equipment needed to execute the selected use cases. AstaZero was selected to execute two of the five test scenarios, and the following stepwise approach was incorporated when preparing and performing the physical tests by AstaZero.

- Decide on necessary test equipment and sensors needed for succeeding with the defined tests.

- Physically prepare the test vehicle, by installing equipment for localisation and installing a forward-facing radar.
- Set up a test plan and perform risk analysis.
- Define interesting KPIs to be able to det what data that should be collected during test.
- Integrating sensor communication with the software controlling the vehicle.
- Integrate sensor data with AEB functionality to be able to apply emergency brake due to obstacles in front of the vehicle.
- Iterate on test setup to make sure that the planned scenario is possible to perform physically on the chosen track.
- Perform physical tests and record necessary data.

The vehicle used is a Ford Mondeo equipped with Dataspeed Inc drive-by-wire (DBW) system, which allows control of brake, throttle, steering and shifting of the vehicle. The test vehicle was also equipped with a forward-facing automotive FLR7 radar from Aptiv. The object list generated by the radar software is a key element in the software's decision making regarding if emergency brake should be applied or not during the execution of the chosen test scenarios.

To perform the test scenarios, AstaZeros inhouse developed test operating system called ATOS was used. It is an open-source ISO 22133-compliant and ROS2-based scenario execution engine. By using ATOS, it is possible to control, monitor and coordinate both physical and virtual vehicles and equipment, according to scenarios specified in ASAM OpenSCENARIO format.

The data gathered from the physical tests performed by AstaZero was shared with all project partners in Evident.

Partner Chalmers and Partner University of Gothenburg:

To perform the physical tests, and to select and install necessary equipment needed to execute the selected use cases. Chalmers (Revere) was selected to execute two of the five test scenarios, which were performed by following a structured approach described below:

- Identify the necessary test equipment and sensors needed for succeeding with the defined tests.
- Physically prepare the Volvo XC90 for tests by installing equipment for localisation and installing 4 forward-facing radars underneath the vehicle bumpers (x2 front, x2 rear).
- Integrate sensors communication with the software that enables data collection and vehicle control.
- Perform physical tests and record necessary data.

The vehicle used is a Volvo XC90 with computers operating with OpenDLV software, which is an open-source software that was developed by University of Gothenburg. OpenDLV allows data collection from multiple sources (sensors) and control of brake, throttle and steering, if needed and data synchronization, and data conversion.

The test platform was equipped with:

- Basler acA2040-35gc video cameras (x5)
- Velodyne HDL-32E LiDAR (x1)
- Applanix POSLV 220 IMU-42 GNSS/IMU (x1)
- smartmicro UMRR-96 Type 153 Radar (x4)
- Meinberg LANTIME M500 Time-synchronization device (x1)

The data gathered from the physical tests performed by Chalmers (Revere) was shared with all project partners in EVIDENT.

Partner Einride:

Prior to running the scenarios, the software intended to be used during the test was loaded and verified according to Einride’s safety checklist for the vehicle under test (VUT).

The setup of the scenarios was done by AstaZero which utilized line drawing robots to paint the pre-determined path on the testing grounds for the test cases performed on open asphalt flats where necessary.

Static objects were placed according to position in the OpenDRIVE files for the test cases, and in one test case where a pedestrian is present, the movement of the pedestrian platform was orchestrated by an ATOS server running at the AstaZero test track.

During the days of testing, before the respective tests were performed, it was ensured that the logging of the sensors and vehicle worked as intended. This was done to not accidentally lose any data from the test cases.

6.6 Performing simulation

Next step number **Sixth** is performing simulations of the use cases, including looping for simulation calibration of models as far as the partners could come within the project.
Detailed method used per partner:

Partner Aptiv:
Aptiv's toolchain was composed of different aspects of open-source components like OpenDRIVE and OpenSCENARIO, and the simulation itself was performed in IPG Carmaker. IPG Carmaker is a high-fidelity simulation platform used to execute the scenarios. It provided precise modelling of vehicle dynamics, traffic conditions, and environmental factors. The focus of this toolchain was to replicate the real-life drive. This implies that vehicle dynamics and driving parameters like velocity were to be a close replica of the physical test. A built-in high fidelity radar tracker and a low fidelity radar was used to replicate the physical sensor setup. The high-fidelity radar sensor analyses the signal-to-noise ratio (SNR) to detect obstacles. This sensor model incorporates object occlusion effects, antenna gain modelling, and propagation losses. The low fidelity radar mimics a physical sensor by simulating the transmission of electromagnetic waves within a digital environment. The setup was sufficient to simulate, and record needed data for comparisons with the physical test.

Partner AstaZero:
The simulation toolchain used by AstaZero has been developed entirely utilizing open-source tools, together with inhouse developed software. A key aspect of performing the simulation correlation was to be able to use the same OpenDRIVE and OpenSCENARIO files in the simulation as in the physical tests. These files are read and interpreted using Esmini basic scenario player, and the simulation is then executed in Carla simulator, which is an open-source autonomous driving simulator. The following approach was used to perform the simulations.

- Define architectural overview for the simulation tool chain.
- Integrate Esmini together with Carla simulator to be able to run and control the chosen scenarios.
- Iterate simulated scenario to verify behaviour of simulated agents compared to real world behaviour.
- Validate simulation tool chain by making sure the scenario behaves identical between each run.
- Set up logging in simulation to be able to extract the same necessary data as for the physical test.
- Integrate a functional mock-up unit (FMU) for the simulated radar sensor to receive object data in simulation.
- Perform test scenario simulation and record necessary data.

The vehicle under test is modelled in the simulation with a ready-to-use Carla vehicle asset, adapted to the measurements and characteristics of the real Ford Mondeo. The radar sensor is modelled using a functional mock-up unit (FMU), representing an ideal radar of mid fidelity being fed with some ground truth data about actors present in the simulation.

Partner Einride:

The simulation toolchain used by Einride is a proprietary simulation toolchain. The simulation models intended for use were tuned to acceptable levels for the means of this project. Parameters of the model were initially set to values derived from physical properties of the VUT as a starting point and then tuned against a set of logs pre–recorded. Since one of the aims of the project is to find suitable methods for evaluating fidelity level of the simulator, it was not relevant to create perfect environments and models.

The scenarios were defined and tuned based on ground truth data obtained from the physical tests (since OpenDRIVE and OpenSCENARIO were not fully supported in the toolchain). For the simulation of the test cases, looping was required to fine tune the environment and to verify that the behaviour of actors was agreeing with that of the physical tests.

The perception simulation models used are ray tracing based with high fidelity. The lidar models used generates rays in accordance with how the corresponding sensor functions physically. Similarly, the radar models are also based on ray tracing methods utilizing a high level of fidelity.

Partner VTI:

The Swedish National Road and Transport Research Institute (VTI) possess and maintain two moving base driving simulators and other smaller simulators. In this process, VTI has developed an in-house software for the simulator systems and as such, VTI tries to use as much standards as possible but also has quite some in-house developed solutions. From this perspective
The method has then been:

1. Parameterize the current vehicle model to be comparable to the vehicle that will be used on the test track.
2. Integrate given OpenDRIVE from AstaZero and extend the environment to make it more lifelike.
3. Integrate OpenSCENARIO description using ESMini from the test track scenarios.
4. Use a moving base simulator where the motion can be turned off to compare with and without motion.

In this way the same tests from the test track were run in the simulator as well.

Partner Zeekr:

The toolchain at Zeekr included software from VI-Grade i.e. VI-Worldsim which is extensively used in Driver in the loop (DiL) evaluations in their dynamic driving simulator and IPG Carmaker which is used by Hardware in the loop (HiL) teams.

The focus areas of the simulation methodology were the following:

1. Establish the necessary workflow to import AstaZero assets created in open-source formats into simulation tools such as VI-Grade VI-Worldsim and IPG Carmaker.
2. Evaluate the importance of base vehicle dynamic performance for the S-curve scenario in IPG Carmaker.
3. Evaluate the scenario import workflow and performance of the default ideal radar sensor in VI-Worldsim for the Euro NCAP CCFTap scenario.

6.7 Challenges

In step **Seven**, we organized a workshop to reflect on the challenges faced during the project and gather lessons learned.

A total of nine people attended, including two moderators. The aim was to identify key challenges, prioritize them, and analyse their nature in terms of complexity. The workshop began with a brainstorming session. Each participant listed the challenges they had encountered during the project, and then we discussed these as a group. After the discussion, similar challenges were grouped together to streamline the list and identify patterns.

Once the challenges were organized, participants voted on the ones they felt were most important. This helped highlight the areas that required more focus or further discussion.

In the final step, we asked participants to categorize the challenges based on two dimensions: the degree of agreement among the project partners regarding the way to address the challenge; and the degree of certainty and predictability about what results will be generated from the solution proposed for addressing them.

In order to gain more insights into the collected challenges, we sent out a survey to all project partners. They listed all identified challenges, and the following set of questions were asked for each challenge:

- Is this a challenge that you have encountered during EVIDENT and/or other projects?
- Tell us more about the challenge
- What was the degree of agreement among the participants regarding the challenge and the best way to address it?
- What solutions were proposed for addressing the challenge?

- What do you think is the degree of certainty about what results will be generated from the solutions proposed for addressing the challenge?
- What roadblocks are there to achieve this solution? E.g., lack of documentations, not containerizing, etc.

Additionally, we asked the respondents to select the challenges that should be the focus in upcoming projects, and we also asked if they wanted to add additional challenges.

6.8 Compare the physical and virtual test, fidelity level and trustworthiness

Step **Eight** is to compare the physical and virtual test and discuss the fidelity level and trustworthiness. This has been done both separately at each partner and in workshop between partners.
Detailed method used by each partner:

Partner Aptiv:

To compare collected data from simulations and physical tests the following methodology was used.

- Collected data was processed post simulation to align reference systems to account for vehicle heading and sensor mountings. Time synchronization of physical and simulated data logs to be able to correctly match results.
- Calculation of differences in the chosen KPIs.
- Comparisons between measurements in simulated and physical tests, analyze results and discuss differences between reality and simulation and causes for these.
- Gap analysis to identify and justify gaps in collected data.
- Retracing steps and tuning parameters to achieve lower differences between reality and simulation.

Partner AstaZero:

The following methodology was used when comparing the data gathered from both physical and simulated tests.

- Postprocess logs from both simulation and physical test to use the same reference system and measures.
- Time synchronisation of the logs and extract starting point for test case to be able to match physical run with simulated run.
- Calculate metrics for the chosen KPIs.
- Analyse and document differences and possible causes for these differences, such as vehicle characteristics and control models, environmental differences, and fidelity between physical sensor and simulated sensor model.
- Analyse processing and trigger times for emergency braking event between simulation and physical test.

- Reflect on acceptable gaps and differences in vehicle behaviour, without losing confidence in performing verification and validation in a simulated environment.

Partner Einride:

The general approach Einride has taken to compare the virtual and physical tests is the following:

- Pre-processing of recorded data consisting of interpolation, assigning common time vectors and setting the start, stop time of the time frame of interest for the respective scenario and filtering away redundant object detections.
- Create time series representative of all applicable physical runs for the scenario. This is done by taking the mean across each data channel for each time step.
- Deriving important KPIs from the data e.g. the distance to relevant obstacles.
- Calculate metrics which compare the simulations and physical testing.

To describe these steps in more depth, the pre-processing of the data begins at linear interpolation where applicable. This ensures that time steps are comparable between different recordings.

This is followed by identifying the time frames of interest for the different scenarios. For instance, the nominal driving is not interesting for the Occluded Child scenario, it is rather the time around when the hard braking occurs. The time frames of interest for the scenarios ran by Einride were defined as

- When throttle is first applied to when the VUT has come to a stop for S-Curve and Car in Curve
- An arbitrary set time before hard braking to full stop for the Occluded Child scenario

These times were automatically obtained and filtered for each recording. A common time vector was then applied to all recordings to ensure that the data channels are synced. The data from the physical testing were then merged into one representation by averaging over each data point for each time step. For each scenario the pre-processing then outputs one data collection representing simulation and one representing the physical runs of the scenarios, both sharing the same time vector making analysis simple.

When comparing simulation to physical tests, three main metrics were calculated to understand how they compare.

1. Normalized Root Mean Squared Error (NRMSE)
2. Normalized scalar difference
3. Normalized 90% and 95% quartile error

The NRMSE is defined as the RMSE divided by some relevant quantity aiming to normalize the RMSE to make different ranges of systems comparable. To highlight its usefulness, calculating the RMSE for the S-Curve scenario with 5 km/h and 50 km/h will most likely produce widely different values of the RMSE making them incomparable. However, normalizing by dividing the RMSE with the corresponding max velocity will bring them to a more comparable range.

The same normalization schema was applied for the 90% and 95% quartile errors, yielding results comparable between different scenario executions using different parameters (i.e. max velocity).

The normalization constant was chosen as followed

- Max velocity when calculating errors for velocities
- Standard deviation of the measured heading for the physical test
- Standard deviation of the measured steering angle for the physical test

These were chosen since the max value of an angle does not have any significance as opposed to higher velocities. The standard deviation instead highlights how much said value changes during a scenario execution.

Finally, the normalized scalar difference is defined as

$$\epsilon_a(b) = \frac{b - a}{a}$$

This quantity will be calculated in the context where underlying quantity is scalar, more specifically has it been utilized to compare obstacle detection distance errors. As a highlighting example consider the distance to an object when the VUT is at a complete stop. If the distance in simulation is smaller than the physical testing the resulting normalized scalar difference will be negative, and the value will be the fraction from the physical distance. This yields an intuitive understanding what the magnitude of the error is, and it makes it comparable between different scenario parameters such as velocity.

Partner VTI:

In short, the following steps were taken when comparing drivers in a driving simulator with drivers on a test track.

1. Collect and post-process data collected from test track
2. Collect data from drivers in the simulator
3. Compare KPIs and investigating if significant differences occur during different driving scenarios for the drivers in two settings
    a. Compare drivers in simulator with test track
    b. Compare drivers in a simulator with and without a moving base
4. Conclusions from the analysis of the different driving scenarios

Partner Zeekr:

The methodology used for correlation of simulation results to physical measurement results is outlined as follows:

1. Initial Simulation loop: Run first simulation iterations with digital assets in open-source format received from Asta Zero and extract KPI (key performance indicator) metrics from simulation data.

2. Measurement data processing: Identify and extract relevant sensor outputs for correlation. Resampling of signals where applicable. Data alignment and extraction of relevant sections of measurement logs for comparison with simulation KPI signals.
3. Comparison and visual illustration of relevant metrics to identify vehicle behaviour in both virtual and physical worlds and identify gaps.
4. Gap analysis and justification of gap for chosen KPI signals.
5. Sensitivity analysis of input parameters and evaluation of output to prove justification of gap and show improvement strategies to close any gaps.
6. Conclusions and lessons learned from correlation activity.

6.9 Credibility Assessment for Virtual Toolchains

Ninth step is to use the insight derived from the cases studies and literature studies to develop a draft method for credibility Assessment for Virtual Toolchains.

We started this activity by looking at relevant regulations, standards, and literature about credibility assessment of virtual toolchains. We identified the United Nations "New Assessment/Test Methods for Automated Driving (NATM) [33] Guidelines for Validating Automated Driving System (ADS)" [Henceforth referred to as NATM] to be the primary relevant document for our study. In particular, Annex III "Credibility assessment for using virtual toolchain in ADS validation" was used as the basis for our approach for virtual toolchain credibility assurance.
NATM requires that the ADS manufacturers produce a document structured according to the outline of Annex III to provide evidence of the credibility of virtual toolchains. This document would be used by an assessor. However, there is no description of how this document would be created. Moreover, NATM requires the ADS manufacturers to provide traceability between the provided documents and the corresponding parts of the toolchain and data, which can be a complicated task.
To tackle this, our approach suggests using assurance cases, which are bodies of arguments and evidence used to reason about a certain concern of a system (in our case credibility of virtual toolchains) [36]. Assurance cases have been used for a long time in various domains, e.g., automotive, to reason about safety and cybersecurity and have been proven to be a good approach in the literature. Moreover, they are explicitly required in various standards, e.g., ISO 26262 [34] and ISO/SAE 21434 [35].
Assurance cases can be expressed in different forms. However, the most common representation uses the Goal Structuring Notation (GSN) [36] and consists of the following elements:

- Top claim: this is the first claim that is made in an assurance case. It sets the abstraction level for the whole assurance case. For example, a credibility top claim can be on a tool level (The tool X is sufficiently credible for AD testing) or it could be on a tool chain level (The toolchain is sufficiently credible for AD testing).

- Claim: statement asserting a certain property, behaviour, or quality of the system/tool/component in question.
- Strategy: represents a reasoning approach or method to decompose a certain claim into sub-claims.
- Context: provides the background, assumptions, or conditions under which a claim, strategy, or evidence is valid, ensuring clarity and relevance to the argument.
- Assumption: statement taken to be true without direct evidence, serving as a foundational premise that supports the validity of claims, strategies, or evidence
- Evidence: tangible and verifiable information, such as test results, analyses, or expert judgments, that supports the truth of a claim.

The methodology used for this task is illustrated in Figure 6.9.1. We extract requirements and recommendations from NATM – Annex III for credibility assessment of virtual toolchain, and map these requirements to assurance case elements, e.g., claims, strategies, and evidence. We identify the following: (i) evidence that NATM explicitly requires, (ii) pre-requisites for the credibility assessment, and (iii) a set of claims that need to be justified, although NATM does not include any specific evidence for them.

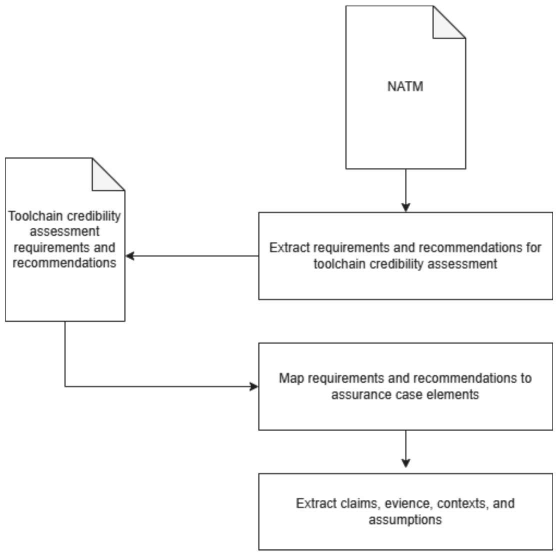

Figure 6-1: Methodology used for credibility risk assessment of virtual toolchains

7. Objective

The project aims to develop strategies which complements or replaces recordings from real traffic situations to augment already existing V&V strategies. As a result, concrete methods and metrics to determine the necessary "realism" of a virtual testing environment, herein referred to as fidelity for specific use-cases, will be explored. Utilising recordings from real-world driving and test track, concepts which enable a proactive testing in both an open loop and closed loop testing scenario, such as SiL, HiL and ViL setting will be prototyped. Possible open-source tools will be examined in order to generate results, which can be publicly shared. Proprietary tools not openly available and used in existing toolchains will also be under consideration. In both cases, standardised formats for tool communication will be utilised whenever possible as to create generic strategies.

The expected results can be noted more specifically as:

- Gap analysis presenting methodical need and requirements to solve problems related to real world-fueled virtual testing using simulation.
- Several case-studies carried out based on industry, consumer and regulatory need, where at least one case-study concerns openly available tools and equipment for full transparency.
- Methodologies how to determine or deal with metrics for determining degree of fidelity for a selected number of use-cases.
- A set of performed tests comparing virtual testing with physical testing on physical test track and real-world driving.
- Broader utilization of virtual testing influenced by real world data within OEM, Tier 1s and third-party V&V activities.
- The project aims for four scientific publications within the relevant area.

In the project we have realized that set ups for evaluating simulation of real traffic situations was too big of a step for all partners. The project had to focus on five simpler cases to be able to perform both tests on physical test track and ensure needed simulation models and virtual testing. This means that the project has identified:

- A draft methodical approach to perform a gap analysis between physical test case at test track and simulations.
- Performed five case-studies based on industry and EuroNCAP needs, all cases performed with at least one open-source tool chain for full transparency. One Use case had to be reduced to only one partner testing and only in simulation due to time and budget situation when project have had several issues.
  Due to the maturity of simulation usage and complexity to get high fidelity the case studies have been chosen to be EuroNCAP cases or Vehicle dynamic cases. This means that EVIDENT had to down prioritize the use of real traffic scenarios.
- In all but one case studies KPI's for comparison between simulation and reality have been developed to determine a method of fidelity comparison.
- Four of the test cases have been tested both on test track and in simulations by at least two different project partners. Recordings of data is done and used in the comparison.
- A much clearer view of what is needed to get a broader utilization of simulation is concluded and a list of challenges that needs to be solved is presented.
- The project has presented two papers in conferences and have four other papers in reviews for conferences coming up 2025.

This means that EVIDENT project has made a minor adjustment of the objectives and has a list of new research questions and objectives for coming projects.

8. State-of-Practice

Results from the landscaping of the industrial state-of-practice and current best-effort for gap analysis are presented here, and the themes arising from the reviewed white and grey literature will be used to structure the findings.

The development of autonomous driving technologies necessitates extensive testing in both simulated and real-world environments. While simulations offer a controlled and cost-effective means of testing, discrepancies—such as differences in lighting, textures, and material properties affecting sensor perceptions—create a "reality gap" between simulations and the real world [1]. Bridging this gap is essential to ensure that knowledge and strategies developed in simulations effectively transfer to real-world applications.

Current approaches to addressing the reality gap can be broadly categorized into two main strategies: transferring knowledge from simulation to reality (sim2real) and leveraging digital twins (DTs). The following summarizes these approaches based on recent literature surveys [1][2].

8.1 Sim2Real Techniques

Sim2real methods focus on transferring insights and learned behaviours from simulated environments to real-world applications. Key techniques include:

Curriculum Learning: Training models by progressively introducing more complex tasks allows systems to build upon simpler learned behaviours before tackling more complex scenarios [3]. In autonomous driving, this approach facilitates the revision of learned policies in simple examples before addressing more challenging situations [4][5].

Meta-Learning: Also known as "learning to learn," meta-learning focuses on creating algorithms that enable models to adapt quickly to new tasks or environments by leveraging prior experience [6]. In the context of autonomous driving, meta-learning helps vehicles adjust to changing tasks and environments more efficiently [7][8].

Knowledge Distillation: This technique involves transferring knowledge from a complex "teacher" model to a simpler "student" model to improve efficiency without significant loss in performance [9]. Applications in autonomous driving include improving trajectory prediction and object detection in LiDAR point clouds [10][11].

Robust Reinforcement Learning: Enhancing models to make reliable decisions in uncertain and dynamic environments is crucial for autonomous vehicles. Robust reinforcement learning aims to improve decision-making under such conditions by modelling adversarial agents and simulating perturbations [12][13][14].

Domain Randomization: Introducing variability in simulations helps models generalize better to real-world scenarios by exposing them to a wide range of simulated conditions [15]. This technique has been used to transfer learned driving strategies to real vehicles, considering complex road and high-speed driving situations [16][17].

Transfer Learning: Leveraging pre-trained models from related tasks accelerates learning in new domains [18]. In autonomous driving, transfer learning aids in transferring knowledge from simulation to reality, enhancing robustness and efficiency [19][20][21].

8.2 AR/MR Integration and Digital Twins

The integration of Augmented Reality (AR) and Mixed Reality (MR) technologies with digital twins further improves visualization and interaction [22][23]. These technologies enable users to interact with and control virtual models, enhancing testing and validation processes [24][25][26][27].
Digital twins involve creating detailed virtual replicas of physical systems, enriched with real-time sensor data and physical models [28]. In autonomous driving, DTs are used for multi-scale modelling of the environment and vehicles, enhancing simulation fidelity [29][30][31][32].

8.3 Challenges and Gaps

Despite these advancements, several challenges remain:

1. Lack of General Task-Independent Methods: Current solutions are often developed for specific scenarios, and their application to different tasks does not yield satisfactory results.
2. Need for Adequate Training Data: Gathering sufficient and high-quality real-world data is challenging. Assessing data quality in terms of completeness, correctness, diversity, and the inclusion of edge cases is crucial.
3. Comprehensive Digital Twin Methodologies: There is a lack of extensive methodologies for creating and assessing digital twins in autonomous driving.
4. Evaluation of DT Models and Algorithms: Separate DT models and algorithms exist without a comprehensive method to evaluate each of them.
5. Traceability Between Testing and Requirements: There's a need for transparent and seamless traceability between virtual testing and system requirements.
6. Criteria for Evaluating Virtual vs. Real Data: Establishing criteria to evaluate virtual data against real data is essential to tackle the fidelity gap.
7. Standardized KPIs: There's a demand for standardized Key Performance Indicators (KPIs) to support various testing activities.

9. Case Studies

The case studies investigated in EVIDENT encompass a mix of vehicle dynamics, sensors, and EuroNCAP-defined scenarios. The chosen scenarios are described below, along with related KPI measurements.

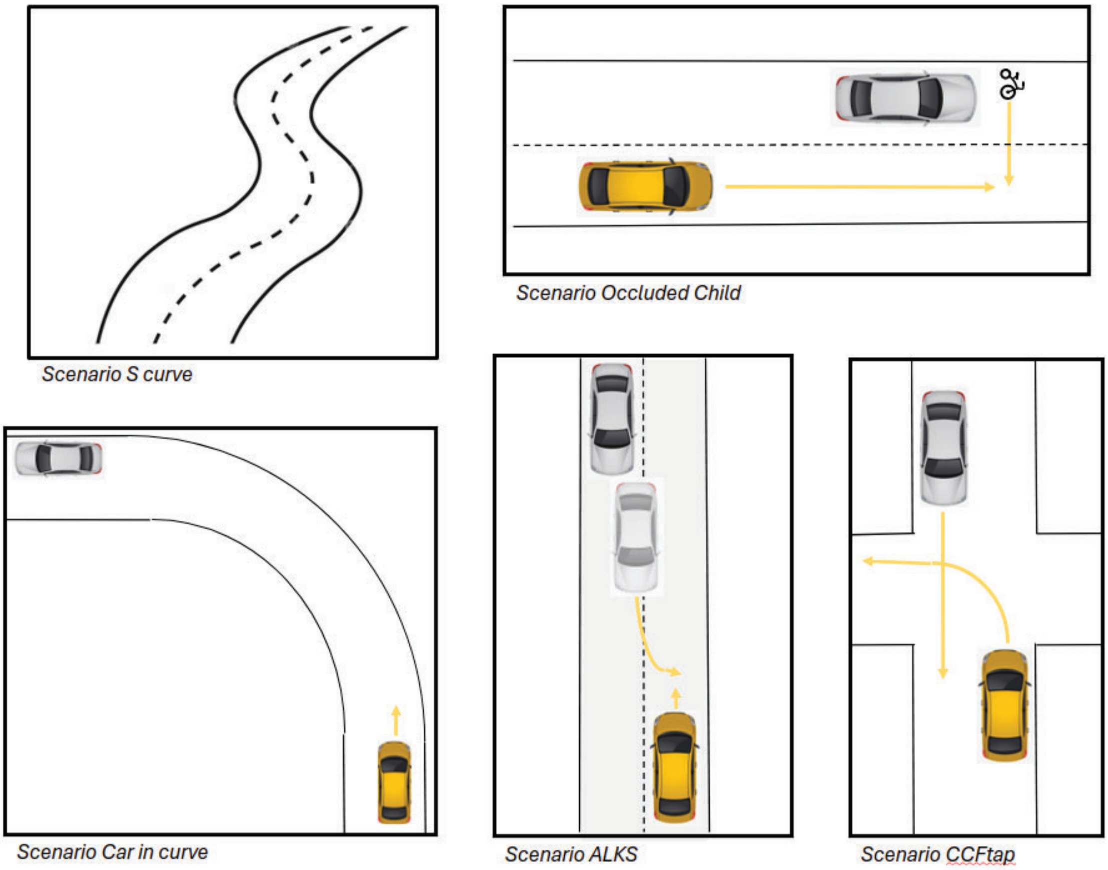

Figure 9-1: EVIDENT Case Studies

9.1 ALKS

ALKS ("Automated Lane Keeping System") according to UNECE R157 is defined as a system activated by the driver which keeps a vehicle in the lane by controlling lateral and longitudinal movements of the vehicle for extended periods without the need for further driver input. The intention of choosing this testcase was to evaluate the output data of different sensor models prior to collision. The focus here is not to evaluate vehicle response but to understand sensor model data in the "detection range" of the sensor models. Two scenario variants were defined:

1. A simplified drive-by of ego vehicle and GVT in their respective lanes at speeds defined by a test matrix below.
2. Wobble Scenario: GVT enters the lane of the ego vehicle and creates a situation where the ego needs to make decision on a minimum risk avoidance manoeuvre.

Since simulation models of different fidelity can be used for different purposes i.e. writing requirements, developing concept functions or component validation; one needs to understand the outputs from available sensor models. The ideal scenario would be to obtain the high-fidelity model from the sensor supplier: however, this is not always straightforward.

ALKS Test cases:

1. Drive by with constant offset:
   Ego at middle of its lane, GVT drives on the centre line of the road while staying in its own lane at 50km/h
2. Wobble scenario at 50km/h

ALKS KPI

Evaluate output of ideal radar sensor in VI-Worldsim simulation environment and compare it to radar output of physical sensor. Radar output in simulation is structured as follows:

- Object tag
- Yaw angle with respect to sensor in radians
- Distance with respect to sensor in meters
- Power of returned signal in dB
- Lat rate – rate of change of the range from sensor to object (m/s)
- Range rate – speed of object in direction perpendicular to the sensor (m/s)
- Status of the tracked object (0-new, 1-tracked)

9.2 CCFTap

Car-to-Car Front Turn-Across-Path (CCFTap) is a scenario in which the vehicle-under-test turns across the path of an oncoming vehicle traveling at a constant speed, which would result in a frontal collision between the vehicle-under-test and the oncoming vehicle, forcing the vehicle-under-test to perform automatic emergency braking to prevent it from occurring. CCFtap was tested at three different speeds:

1. 10 km/h
2. 15 km/h
3. 20 km/h

AstaZero collected the following data for each object in the scenario:

- x-position
- y-position
- longitudinal velocity
- lateral velocity
- longitudinal acceleration, filtered with a 12-pole phase less Butterworth filter with a cutoff frequency of *10 Hz*.
- lateral acceleration, filtered with a 12-pole phase less Butterworth filter with a cutoff frequency of *10 Hz*.
- automatic emergency braking (AEB) activation signal, **for the vehicle-under-test only.**

From this data, the following KPI's has been computed:

- euclidean distance
- longitudinal distance
- lateral distance
- relative longitudinal velocity
- relative lateral velocity
- time to collision
- braking distance, which is defined as the distance travelled by the vehicle-under-test from the time of automatic emergency braking signal activation until the vehicle-under-test comes to a complete standstill.
- AEB acceleration activation, which is a timestamp with is determined by identifying the last data point after AEB activation where the filtered acceleration signal is below $-1 m/s^2$ and then going back to the point in time where the acceleration first crossed $-0.3 m/s^2$.

This data and metrics were then compared across the physical and simulated runs of the same scenario in order to measure the difference between them.

9.3 Car in Curve

The Car in Curve scenario was aimed at evaluating obstacle detection range and nominal stopping behaviour. There were in total 4 different scenarios: 2 curve radii (15 and 30 meters) running at two different velocities (10 and 15 km/h). In each was a static NCAP balloon car placed in the middle of the turn with the intention of it not being seen initially, but then during the turn becoming detected.

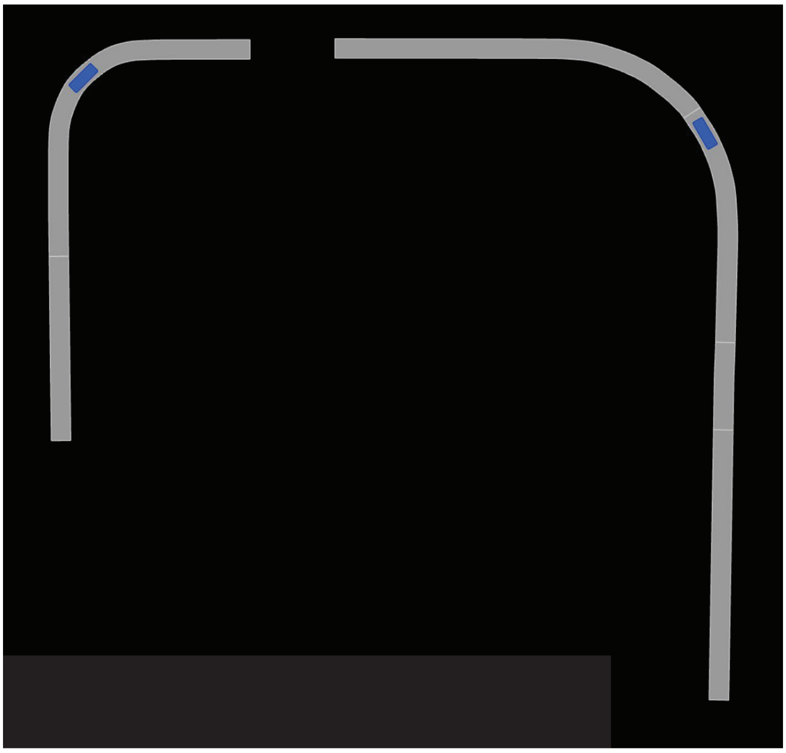

Figure 9-2: Graphical representation of the Car in Curve scenarios. The blue boxes highlight the pose of the static obstacle.

Amongst all the data Einride collected was the velocity, pose and obstacle detection data then used to derive the following KPIs:

- Euclidian distance at different points in time which are of interest
    - Error in first obstacle detection range
    - Error in stopping distance
- Velocity compared as time series
    - RMSE of the velocity from start to stop (normalized against average of physical data)
    - Normalized against the maximum velocity

9.4 S-curve

The S-Curve scenario was devised to compare nominal pure vehicle dynamics behaviour (no obstacles) where there is no influence of other sub-systems. There are in total three large curves followed closely after each other, each with a radius of 70 meters. The scenario was run in three different velocities

1. 10 km/h
2. 15 km/h
3. 25 km/h

Figure 9-3: Visualization of the S-Curve path at the test-track.

Amongst all the data Einride collected was the velocity, pose and obstacle detection data then used to derive RMSE and errors of the velocity and heading.

In addition to the above, Zeekr proposed the following speeds for measurement with the Chalmers Snowfox vehicle. The intention with the chosen speeds was to obtain measurement data within a "linear" lateral acceleration range and to avoid interferences from stability systems.

4. 30 km/h
5. 45 km/h
6. 50 km/h

9.5 Occluded Child

Occluded child is a scenario in where a child is occluded by two parked cars, and as the vehicle-under-test (VUT) is passing the cars the child runs out in front of the VUT, forcing it to perform automatic emergency braking to prevent a frontal collision with the child. There was a 50% overlap at the collision point along the longitudinal centreline of the VUT, and the scenario was run at three different speeds:

1. 15 km/h
2. 25 km/h
3. 30 km/h

AstaZero collected the following data for each object in the scenario:

- x-position
- y-position
- longitudinal velocity
- lateral velocity
- longitudinal acceleration, filtered with a 12-pole phaseless Butterworth filter with a cutoff frequency of *10 Hz.*
- lateral acceleration, filtered with a 12-pole phaseless Butterworth filter with a cutoff frequency of *10 Hz.*
- automatic emergency braking (AEB) activation signal, **for the vehicle-under-test only.**

From this data, the following KPI's has been computed:
- euclidean distance
- longitudinal distance
- lateral distance
- relative longitudinal velocity
- relative lateral velocity
- time to collision
- braking distance, which is defined as the distance travelled by the vehicle-under-test from the time of automatic emergency braking signal activation until the vehicle-under-test comes to a complete standstill.
- AEB acceleration activation, which is a timestamp with is determined by identifying the last data point after AEB activation where the filtered acceleration signal is below $-1 m/s^2$ and then going back to the point in time where the acceleration first crossed $-0.3 m/s^2$.

This data and metrics were then compared across the physical and simulated runs of the same scenario in order to measure the difference between them.

10. Simulation toolchains

In the project, we have utilized various simulation toolchains depending on each partner's setup. Below, we describe the different simulation toolchains used by each partner:

Partner Aptiv:

Aptiv simulation toolchain is based upon IPG Carmaker and uses it as a base for all simulations. IPG Carmaker support both OpenDRIVE road segments and OpenSCENARIO which makes integration of basic simulations possible. To be able to simulate radar performance two different methods were used, both based on IPG carmakers own radar models. These are the high fidelity (HiFi) Radar and the raw signal interference (RSI) radar. The high-fidelity radar works in a higher level where it detects objects and provides information about relative position, velocity acceleration and so on. While the RSI radar output data consist of detections at each time stamp of the simulation, providing cartesian coordinates for each detection, the power received, and velocity data for every point in the radar's detection cloud. However, the RSI radar output data is not directly available in IPG Carmaker. The sensor data is transferred via the RSDS (Raw Signal Data Stream) client on a TCP/IP connection. As two radars are running, two separate sensor clusters are configured, and each has the same socket number but a unique host port number for the TCP/IP data interface. The VUT is based on Carmakers own driver, which is a control system that replicates a real-life driver. Inputs to this controller is provided by the physical test, these inputs are velocity and steering angle.

Partner AstaZero:

AstaZero simulation toolchain is open-source, and combines the power of both Esmini simulator, which is a basic OpenSCENARIO player, and Carla, which is a simulator for autonomous driving research. The integration of these systems is written in Python 3.10 and uses Carlas Python API for communication with the simulation engine. The communication between the different internal nodes uses Robot Operating System 2 (ROS2). Below follows a high-level description of the different modules.

- Esmini adapter: Communicates with Esmini simulator to process scenario specification defined in OpenSCENARIO and road network description in OpenDRIVE.
- Carla API: Handles communication to and from Carla simulation engine.
- PID controller: Controller node for longitudinal and lateral control of an actor in the simulation.
- Radar FMU: Node handling the communication with the functional mock-up unit (FMU) radar sensor model of mid fidelity. Publishes an object list based on detections from the simulated radar.
- AEB module: Contains emergency brake logic. Receives object list from radar module and, if necessary, publishes AEB command, overruling the current trajectory. This module is identical in both simulation and physical toolchain.

- AEB command listener: Subscribes to AEB command signal on the ROS2 network and executes emergency braking if applicable.
- Logging publisher: Publishes all available signals onto the ROS2 network, to handle logging of data during runtime.

The data gathered from the tests are logged during runtime in a ROS2 bag file, making it possible to record all signals published on the ROS2 network. Each signal is timestamped, and the metadata provides information such as log start, log end and duration, and list of signals paired with number of samples as well as sampling rate for all signals. This format also allows for replaying the data afterwards. During postprocessing, the bag files are then converted into csv format for further processing. The following list displays the signals extracted after postprocessing.

- Timestamp Unix time [s]
- Simulation time [s]
- Radar TTC [s]
- TTC [s]
- Radar to target relative x position [m]
- Radar to target relative y position [m]
- Target local acceleration x [m/s²]
- Target local acceleration y [m/s²]
- Target local velocity x [m/s]
- Target local velocity y [m/s]
- Target position x [m]
- Target position y [m]
- Target pitch
- Target roll
- Target yaw
- VUT angular velocity x [rad/s]
- VUT angular velocity y [rad/s]
- VUT brake command [bool]
- VUT local acceleration x [m/s²]
- VUT local acceleration y [m/s²]
- VUT local velocity x [m/s]
- VUT local velocity y [m/s]
- VUT position x [m]
- VUT position y [m]
- VUT pitch
- VUT roll
- VUT yaw
- VUT steering angle [rad]
- VUT throttle position [%]

Table 10-1: Signal parameters logged

Partner Einride:

The simulator used by Einride was a proprietary simulation platform. It has scenario editing features, different ways to integrate AV stacks, sensor simulations and different ways to define actor behaviours. The specific simulator cannot be disclosed in this report.

All the data produced by sensors and the AV stack are defined using Protocol Buffers and are published on the host network. All network data is recorded and packed into a Packet Capture (PCAP) file which can be accessed and parsed to the relevant data channels when performing analysis post simulation. The exact same method is used when recording data during physical tests.

The running of scenarios was done with the stack in a closed loop with the simulator, as opposed to running it in open loop mode. Closed loop means that the AV software is reacting to the environment (obstacles for instance) just as it would on physical platforms. It was ensured that the same software was run in simulation as it was run on the physical test.

By performing simulations in this fashion, we ensure that we compare simulation to physical test platforms as a test bed for AV software, whereas if the simulations were to be run in open loop mode it would more be a correlation of the models themselves in an isolated setting. This would of course yield insight into how well the models perform, but it might not give the whole picture on how much trust can be put in tests performed in simulation.

Partner VTI:

The used driving simulator at VTI was Sim IV, which uses several different software systems. As such, focus was on the fidelity of the systems which strongly impact the driving dynamics and the road. These systems are the simulator motion control system, the environment and scenario handling, and the vehicle model. The scenarios that were tested at VTI were ALKS, Car in Curve, and S-curve and it was for these scenarios that the simulator toolchain was checked.

In this work the vehicle model used in the simulator was a Mathworks Simulink model which was compiled to an FMU and then run within the simulator system on the simulator kernel. The simulator kernel then used the driver input from the simulator cabin (a Volvo XC60 cabin) as input to the vehicle model. For this vehicle model to be a replica of the vehicle used on the test track, previous test track data combined with available data sheets was used for the parameterisation of the model. In the event of missing data reasonable values was applied to get an acceptable performance. The calculated dynamic states (velocities, accelerations and so on) from the vehicle model are then sent to the motion cueing algorithms for the moving base.

The Sim IV moving base system consists of a XY-table with a hexapod on top. The motion control software for the actuators is proprietary software from the supplier of the moving base system, but before the vehicle model signals are sent to this software system, they are processed through the motion cueing algorithms. These algorithms decide on how to represent the dynamic vehicle signals to the driver using both the XY-table and the hexapod. If the driver movements are known in advance it is possible to trim the motion cueing to be optimal but in general the drivers are free to drive how they want and as such the moving base system needs to be trimmed a bit restrictively so that if drivers do sudden or extreme manoeuvres the simulator will not hit its limits. This trimming is influenced by the scenario and the road, where for example the S-curve includes harsher curves and thus a tougher challenge to represent in a good way for the moving base system.

Continuing with the environment and the scenario description the test track was given in the OpenDRIVE format which is read by the simulator system in Sim IV. Software used to parse the road description is inhouse developed at VTI. Further, as the OpenDRIVE standard is a description focusing on the road it lacks parts of the surrounding environment around the road. For a driving simulator the surroundings are important, thus Mathworks Roadrunner was used together with further modifications in the Unreal Engine editor to build the complete simulator environment. For the scenarios the standard OpenSCENARIO was used. Here, VTI uses the open-source tool Esmini for reading and executing the scenarios.

Partner Zeekr:

Zeekr opted for a two-pronged approach for the toolchain selection and used IPG Carmaker which is used by Hardware in the loop teams and VI-Grade VI-Worldsim which is used by the Driver-in-Loop team. The S-curve and CCFTap scenario OpenDRIVE files were imported into VI-Worldsim via Mathworks Roadrunner, exported as “Unreal Engine fbx” file and packaged into the necessary format in VI-Worldsim Modkit which is an Unreal Engine Editor for VI-Worldsim. The output from the Unreal engine editor could then be imported into VI-Worldsim for further work such as adding vehicles, pedestrians, trajectories, etc.

IPG Carmaker was opted for the S-curve use case:
The reason for choosing IPG Carmaker for this scenario was the availability of a virtual vehicle model i.e. Volvo XC90 T6 AWD within IPG Carmaker databases. This model with updated vehicle parameters was deemed an acceptable representation of the reference vehicle i.e. Chalmers REVERE “Snowfox” Volvo XC90 T6 AWD. Further measurement of Snowfox was not in the scope of the project budget. Additionally, the release of Carmaker 12.0 which had the feature to import OpenDRIVE files directly made the workflow smooth. One had to add an extra “straight” road segment to setup the “route” which the vehicle model is meant to follow. Effort was not put into creating the surrounding trees and environment, this should however be strongly considered for the next project since it adds immersion to driving simulator sessions.

VI-Grade VI-Worldsim was opted for the CCFtap use case:
VI-Worldsim is the graphic engine driving the VI-Grade dynamic driving simulator at Zeekr Technology Europe. This tool was chosen for this use case to evaluate ease of importing assets created in OpenDRIVE format by external providers and to understand the output quality of the default radar sensor. Figure 10.2 displays the AstaZero FLXZone track after importing it into the VI-Grade environment.

Figure 10-1: Top Left: S-curve lane at Asta Zero Test track
Top Right: S-curve scenario in VI-Worldsim
Bottom Left: S-curve scenario in IPG Carmaker
Bottom Right: S-curve scenario in VI-Worldsim (top view)

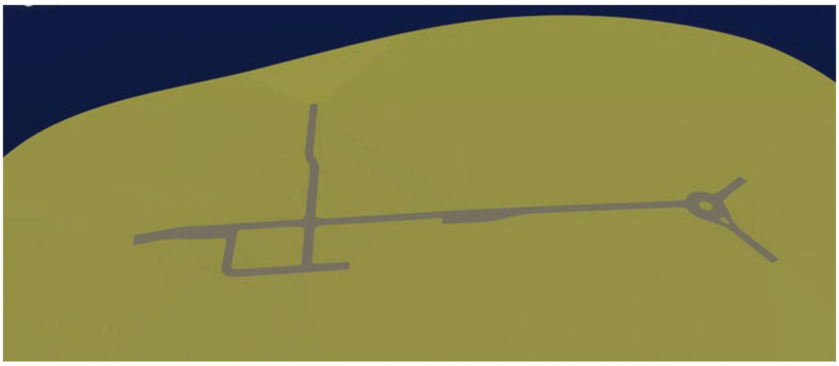

Figure 10-2: FLX Zone track in VI-Worldsim for CCF-Tap scenario

11. Physical test platform

The physical test platforms have slight differences between the cases and partners, which we describe here. First, we describe the base platforms used for each partner, then follows description of how the setup might vary for the specific use cases.

AstaZero base platform

The ego vehicle used for the use cases performed by AstaZero is a Ford Mondeo (model year 2019), equipped with Dataspeed Inc drive-by-wire (DBW) system, which allows control of brake, throttle, steering and shifting of the vehicle. The interface used by Dataspeed is ROS2. The test vehicle was also equipped with a forward-facing automotive radar, purchased from Aptiv, as well as an Inertial Navigation Systems (INS) unit from Oxford Technical Solutions Ltd (OxTS) for gathering position, orientation and motion data.

To perform the test scenarios, AstaZero test operating system ATOS was used. It is an open-source ISO 22133-compliant and ROS2-based scenario execution engine. By using ATOS, it is possible to control, monitor and coordinate both physical and virtual vehicles and equipment, according to scenarios specified in ASAM OpenSCENARIO format. Then OpenSCENARIO and OpenDRIVE files for a given scenario is handled in ATOS by a module integrating Esmini, which is a basic OpenSCENARIO player.

Below follows more in-depth description of the different systems.

Dataspeeds system provides some different modules for controlling the Ford Mondeo:

- Drive-by-wire modules: A ROS2 interface for sending control commands to the Mondeo’s CAN bus.
- Waypoint follower: Receives a trajectory to follow and calculates the necessary control commands to stay on track.
- OxTS driver: Module that connects to an OxTS device and outputs GNSS coordinates and odometry on the ROS2 network.

Around Dataspeeds system, we have developed our own modules for the **Mondeo platform**, also mainly communicating through ROS2:

- DBW communication: Passes through control commands from Dataspeed waypoint follower, unless an AEB command is received, then it will execute emergency braking instead.
- ROS2 Iso object: Connects to ATOS according to ISO 22133 and uses ROS2 to forward the trajectory to the waypoint follower module. Also sends start command to DBW and waypoint follower.
- Radar module: Receives object list data over UDP from the FLR7 radar and publishes onto the ROS2 network.

- AEB module: Contains emergency brake logic. Receives object list from radar module and, if necessary, publishes AEB command, overruling the current trajectory. This module is identical in both simulation and physical toolchain.

As previously stated, **ATOS** is used as test operating system, controlling both the Ford Mondeo, and in one test case, also the target platform. Here follows the relevant modules in ATOS:

- Esmini adapter: Communicates with Esmini simulator to process scenario specification defined in OpenSCENARIO and road network description in OpenDRIVE.
- Object control: Maintains test state and sends control signals to each test object, according to ISO 22133 protocol.
- OSI adapter: Translates ROS2 monitoring message into OSI data and then sends it over either TCP or UDP for all objects used in a test.
- Test control GUI: Test visualization and control panel communicating test state change requests.

The data gathered from the tests are logged during runtime in a ROS2 bag file, making it possible to record all signals published on the ROS2 network. Each signal is timestamped, and the metadata provides information such as log start, log end and duration, and list of signals paired with number of samples and sampling rate for all signals. This format also allows for replaying the data afterwards. During postprocessing, the bag files are then converted into csv format for further processing. The data logging for the test targets differs slightly depending on use case, which is described below for each scenario. But for both cases, the following list displays the signals extracted after postprocessing of the different logs.

- Timestamp Unix time [s]
- Altitude [m]
- Target local acceleration x [m/s²]
- Target local acceleration y [m/s²]
- Target local velocity x [m/s]
- Target local velocity y [m/s]
- Target position x [m]
- Target position y [m]
- Target pitch
- Target roll
- Target yaw
- VUT angular velocity x [rad/s]
- VUT angular velocity y [rad/s]
- VUT brake command [bool]
- VUT local acceleration x [m/s²]
- VUT local acceleration y [m/s²]
- VUT local velocity x [m/s]
- VUT local velocity y [m/s]
- VUT position x [m]
- VUT position y [m]
- VUT pitch
- VUT roll
- VUT yaw
- VUT steering angle [rad]
- VUT throttle position [%]

Table 11-1: Logged signal parameters

Aptiv base platform

Aptiv's base platform is built around a test vehicle, a 2014 Volvo V40, equipped with advanced sensor and processing systems. The vehicle is fitted with four corner radars of the model MRR 360 and logging equipment capable of recording and processing the collected data. This setup allows the vehicle to operate as an ego vehicle in a closed-loop system, where it gathers and processes data to enable advanced driver assistance features, such as Automatic Emergency Braking (AEB).

The ego vehicle includes two high-performance PCs, each with a specialized role. One PC is dedicated to logging radar and host data, ensuring comprehensive data collection during tests. The second PC is used to run Aptiv's radar tracker and implement features like AEB. This dual-PC configuration allows for simultaneous data collection and feature operation, providing both detailed radar data and the ego vehicle's host data for analysis.

In scenarios involving a target vehicle, control is managed using a robot provided by AstaZero, which ensures precise and repeatable movements. The AstaZero system also collects global reference data for both the ego and target vehicles. This reference data includes crucial parameters such as position and velocity, enabling accurate validation of the test scenarios and ensuring alignment with real-world conditions.

Einride base platform

Einride's physical testing platform was an electric truck outfitted with lidars, radars and localization sensors. It was driven by an autonomous driving software developed by Einride. The same platform was used across all scenarios where Einride participated. The physical platform is recording data in the exact same manner as done in simulation. This is described in section 10. Simulation Toolchains under Einride. This ensures consistency between the recordings from the two testing platforms (physical and simulation) during post processing and gap analysis of the data.

Chalmers base platform

Chalmers physical testing platform is a Volvo XC90 from the vehicle laboratory Chalmers Revere that has been instrumented with an OxTS RT3000 GNSS/IMU system to obtain a baseline reference of position, accelerations and speed. On-board vehicle data, such as steering wheel angle and brake request, is accessible via CAN for data logging. Additionally, the test platform counts with 5 exterior video cameras and 1 interior video camera, 4 automotive-grade radars underneath the bumpers (x2 front, x2 rear), and a rotary LiDAR. Radar and LiDAR data provides measurements between the ego vehicle and the target vehicle during the maneuver, meanwhile video cameras provide a visual description of the maneuver. The software for logging the vehicle data (OpenDLV) was contributed by the partner University of Gothenburg and executed on a computer and a time synchronizer device enables synchronized data collection of multiple data sources or sensors. OpenDLV was used as the data collection and post-processing software.

Case 1: ALKS was not tested physically due to issues with radar, budget constraints and accessibility to test track.

Case 2: CCFtap was tested by both AstaZero and APTIV. The test platforms are close to the same but how we steer the test vehicle (EGO) differs.

AstaZero test platform description

The physical test platform during CCFtap test case follows the description of AstaZero base platform. In addition to this, a target platform together with a EuroNCAP vehicle target (balloon car) was used, representing the target vehicle driving in opposite direction to the ego vehicle. The platform used is a Humanetics Ultra-flat overrun able Robot Platform (UFO). This type of platform is also controllable by ATOS, meaning that for CCFtap both scenario actors were controlled by the same system. Besides the logged ROS2 bag for each run, raw data files (RD files) were also extracted from the OxTS unit, to get localisation and orientation data for the target platform. These files where later converted to csv files, and then time synced towards the data from the ROS2 bag files, to be able to get a merged log file containing all the available data before further processing.

Aptiv test platform description

Testing with Aptiv's vehicles differs from how the EGO vehicle was controlled and how the start of the event was triggered. The main difference was the use of a drive-by-wire system, which AstaZero employed to control the EGO vehicle. In contrast, Aptiv's vehicle was driven manually, introducing the potential for human error in the tests.

Additionally, there were differences in the sensor setups between the vehicles. Aptiv's vehicle was equipped with four corner radars, whereas AstaZero's vehicle had a single front-facing radar sensor.

Case 3: Car in curve was physically tested by Einride and Chalmers. Since the target was stationary in this scenario, no robot platform was used. Instead, the target was made up of a EuroNCAP Global Vehicle Target (GVT) positioned at the end of the turn.

Einride test platform description

The test platform and data collection done by Einride for the Car in Curve scenario is described by the Einride's base testing platform section.

Chalmers test platform description

The test platform and data collection done by Chalmers for the Car in Curve scenario is described by the Chalmers' base testing platform.

Case 4: S-curve was physically tested by Einride and Chalmers. This scenario is a vehicle dynamics test without any other objects.

Einride test platform description

The test platform and data collection done by Einride for the S-Curve scenario is described by Einride's base platform setup described in this section.

Chalmers test platform description

The test platform and data collection done by Chalmers for the Car in Curve scenario is described by the Chalmers' base testing platform.

Case 5: Occluded child was tested live by Aptiv, AstaZero and Einride.

AstaZero test platform description

The physical test platform during Occluded child test case, follows the description of AstaZero base platform. In addition to this, a target platform was used together with a EuroNCAP Child Pedestrian Target. The platform used is an AB Dynamics LaunchPad. This platform is not possible to control with ATOS, therefore the procedure is a bit different compared to the CCFtap use case. Here, the relation between ego vehicle and platform is established by setting up a path file for the platform in AB Dynamics own software, and configuring a trigger point based on the position of the ego vehicle, meaning that when the ego vehicle passes a certain trajectory point, the LaunchPad platform will start executing its own path. This sync between the positions is done by calibrating the OxTS systems in both the ego vehicle and for the LaunchPad platform. After test, log files can be extracted from AB dynamics software, which are then converted to csv files and time synced towards the data from the ROS2 bag files.

Einride test platform description

The test platform and data collection done by Einride for the Occluded Child scenario is described by Einride's base platform setup described in this section, with the addition of a positioning reference system to obtain the distance to the VRU.

Aptiv test platform description

The test platform and data collection done by Aptiv for the Occluded Child scenario is described by Aptivs base platform setup, with the addition of a positioning reference system to obtain the distance to the VRU.

12. Results and gap analysis

Partner Aptiv:

Two scenarios were tested and simulated at different host speeds. The first test scenario simulated was an occluded child. The simulations overall showed a small gap between reality and the virtual environment. In the case of radar tracking, the simulated target's distance from the VUT had a root mean square error (RMSE) compared to Aptiv's physical tracker:

Case	RMSE (HiFi)	RMSE(RSI)
Host 15 km/h	0.23	0.21
Host 30 km/h	0.10	0.11

Table 12-1: Root mean square error – occluded child

Even though the measurements in the simulation were acceptable, the main difference between reality and the simulation was the AEB activation point. In both simulated cases, the AEB activated prematurely—0.6680 seconds early at 30 km/h and 0.5640 seconds early at 15 km/h. Additionally, the time from brake activation to a complete stop was significantly slower in the simulation, which can partly explain the premature AEB activation. Another notable difference was that the physical tracker detected the hidden child approximately 10 meters earlier than the simulation when traveling at 15 km/h.

The second scenario tested was Car-to-Car Turn across path. Similar results to the first scenario were observed until lateral movement was introduced. In this test, the target's velocity played a more significant role, and its tracking was successful. The table below shows the RMSE values for velocity and distance between Aptiv's tracker from the physical test and IPG Carmaker's high-fidelity tracker:

KPI	RMSE (10 km/h)	RMSE (20 km/h)
Distance	0.6988	0.6179
Velocity target	0.0593	0.0693

Table 12-2: Root mean square error – CCFTap

These values are within a reasonable margin and replicate the real test. However, the gap between reality and simulation appeared again at the point of collision. When the host vehicle turned, the simulation's tracking of the target's distance and velocity significantly worsened. Once again, the AEB activation was premature, occurring 0.2075 seconds early at 20 km/h and 0.3016 seconds early at 10 km/h. Furthermore, the turn in the simulation was much less aggressive in both cases, which influenced the radar tracking accuracy during this phase.

Both scenarios shared common issues with the host vehicle's behaviour in the simulations. In both cases, the vehicle's acceleration and deceleration were delayed compared to its physical counterpart.

Several identifiable issues in these simulations could be addressed to improve the results. One key problem observed in both tests was how IPG Carmaker handled the host vehicle. Instead of assigning the vehicle a set velocity and steering angle, their controller (IPG Driver) used these inputs as references, with the vehicle dynamics playing a significant role in determining the vehicle's behaviour. For instance, it was evident in these tests that the simulated vehicle lacked the braking capabilities of the physical vehicle. As a result, the simulated vehicle needed to begin braking earlier to avoid a collision.

Additionally, inconsistencies in the physical tests were present, as the host vehicle was not drive-by-wire. The human factor introduced variability in both velocity and lateral movement, making it harder to replicate the physical tests in the simulation accurately.

Partner AstaZero:

To ensure accurate comparison between the physical test and the simulated test, AstaZero synchronized both datasets using a Time to Collision (TTC) of 3 seconds as a reference point. This synchronization method allowed for direct comparison of events, with the zero second timestamp representing the 3-second TTC moment, and negative values showing the data leading up to this point.

In the Occluded Child scenario, while the longitudinal distance measurements showed good correlation, the AEB triggered slightly later in the simulation compared to the physical test:

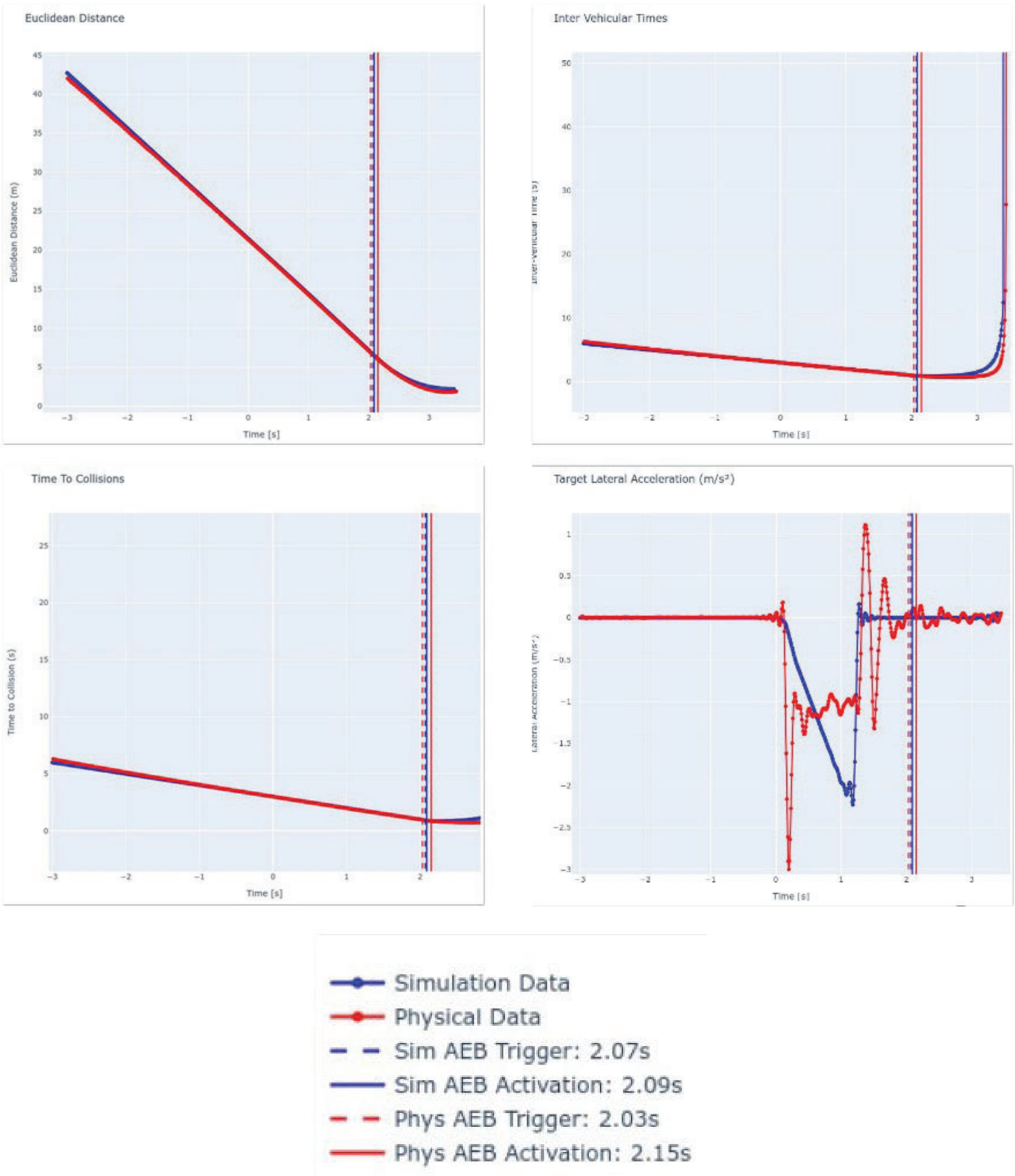

Figure 12-1: Plots for Occluded Child scenario, VUT driving 25 kph.

	Sim run	Mean Phys runs	
	Result	Result	Diff towards sim
Occluded child 15kph			
AEB triggered at TTC [s]	1,267	0,84	-0,427
Time from AEB trigger to activation [s]	0,02	0,15	0,13
braking distance [m]	1,9246	2,088	0,1634
Distance to target at standstill [m]	3,35	1,435	-1,915
Occluded child 25kph			
AEB triggered at TTC [s]	0,92	0,9855	0,0655
Time from AEB trigger to activation [s]	0,02	0,135	0,115
braking distance [m]	4,425	5,3985	0,9735
Distance to target at standstill [m]	2,16	1,5425	-0,6175
Occluded child 30kph			
AEB triggered at TTC [s]	0,93	0,86	-0,07
Time from AEB trigger to activation [s]	0	0,16	0,16
braking distance [m]	6,175	6,678	0,503
Distance to target at standstill [m]	1,71	0,56	-1,15

Figure 12-2: Key measurements for Occluded Child case.

This disparity was attributed to differences in target positioning and movement speeds, with the simulated target moving faster and starting from a different lateral position compared to the physical test.

The second scenario tested was Car-to-Car Front Turn across path (CCFTap). This scenario revealed several inconsistencies between the simulation and physical tests:

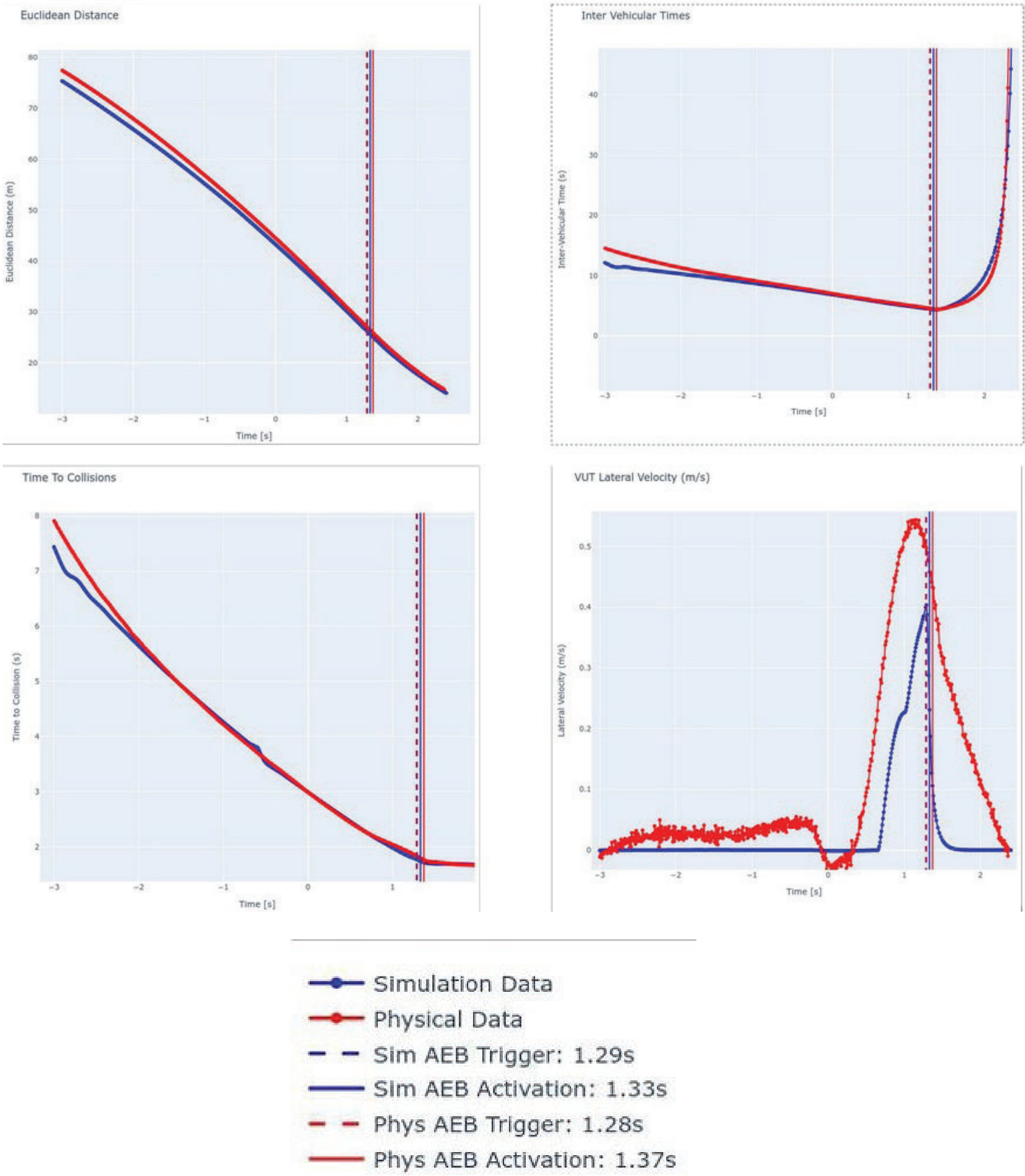

Figure 12-3: Plots for CCFtap scenario, VUT driving 20 kph.

	Sim run	Mean phys runs	
	Result	Result	Diff towards sim
CCFtap 10kph			
AEB triggered at TTC [s]	2,18	3,099	0,919
Time from AEB trigger to activation [s]	0,04	0,108	0,068
braking distance [m]	1,017	0,9958	-0,0212
Distance to target at standstill [m]	17,4	27,196	9,796
CCFtap 15kph			
AEB triggered at TTC [s]	1,93	2,44575	0,51575
Time from AEB trigger to activation [s]	0,04	0,115	0,075
braking distance [m]	1,889	2,074	0,185
Distance to target at standstill [m]	15,5	21,935	6,435
CCFtap 20kph			
AEB triggered at TTC [s]	1,76	1,929	0,169
Time from AEB trigger to activation [s]	0,04	0,1	0,06
braking distance [m]	2,987	3,4082	0,4212
Distance to target at standstill [m]	13,44	15,634	2,194

Table 12-3: Key measurements for CCFtap case.

The initial time to collision measurements didn't align well, though they improved as the tests progressed with higher speeds. Notable differences were observed in the vehicle trajectories, with the physical VUT beginning its turn earlier than the simulated version, likely due to different approaches to trajectory following. The simulation also showed stability issues, with the target controller exhibiting oscillation and rapid acceleration before stabilizing again.

The analysis identified multiple areas requiring improvement in the simulation-based testing methodology. A significant limitation was the lack of comprehensive data about parked car positions and dimensions in the occluded child scenario, making it difficult to determine why the child was detected earlier in the lateral direction during physical tests. Questions arose about whether this was due to incorrect vehicle dimensions in the simulation, differences in VUT offset, or variations in radar processing times between physical and simulated systems.

The analysis highlighted several technical challenges that need addressing:

1. The controllers used in the simulation were found to be inadequate and unstable, particularly evident in the CCFTap scenario, where unexplained acceleration changes occurred.

2. The physical tests also showed unusual data patterns, such as unexpected positive lateral velocity in the CCFTAP scenario, raising questions about data noise and the influence of the target balloon car on the robot platform.

3. Additionally, consistent positioning at the start of each physical test proved challenging, introducing an element of uncertainty into the results.

Despite these various discrepancies and technical challenges, the AEB system demonstrated effective functionality across all test scenarios. While there are evident gaps between physical and simulated testing in terms of physics modelling, timing delays, and overall behaviour, the lack of representative models in the simulation makes it difficult to fully quantify and explain these differences. This suggests that future work should focus on developing more accurate simulation models that better represent real-world conditions and behaviours.

Partner Einride:
Einride participated in three scenarios with both physical and virtual testing activity: S-Curve, Car in Curve and Occluded Child.

Beginning with discussing the gap analysis from the S-Curve scenario. By comparing the error in heading as well as the velocity one can obtain a sense of how the simulation and the physical results compare. Looking at the figures below it is generally clear that the gap increases for higher velocities.

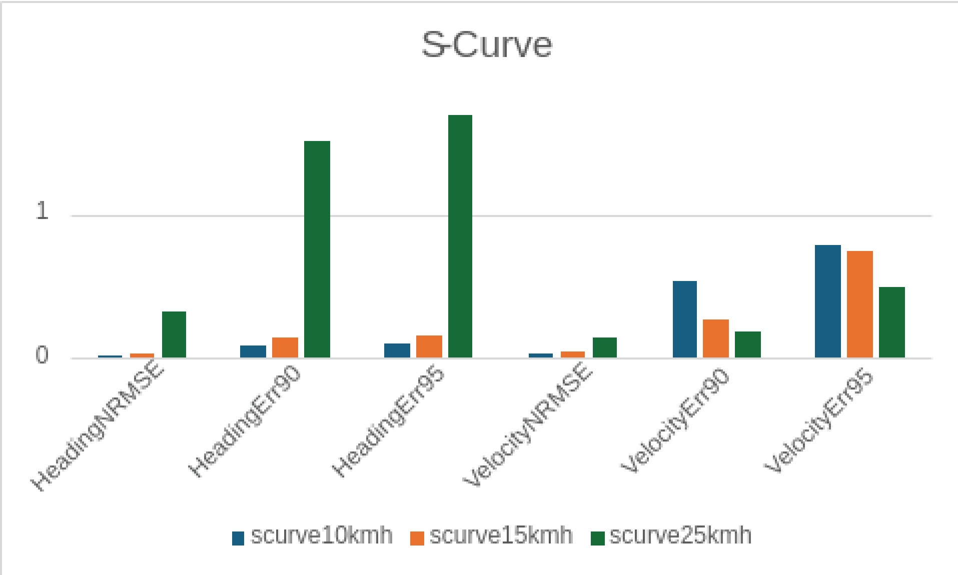

Figure 12-4: Einride gap analysis results for the S-Curve scenario. The time frame for the underlying quantities is chosen from when the VUT starts driving to when it stops.

With the predictable nature of the scenario with regards to lateral errors and with the heading consistently differing between simulation and physical testing. It insinuates that one of the simulation or the physical test was driving slower than the other. From analysing the drive logs, it was apparent that the simulation is generally slightly slower than the physical test; especially for the 25 km/h scenario. Combined with the normalized error results does this highlight a gap in the longitudinal model.

Meanwhile, the largest errors we see (90 and 95% quantile errors) have a decrease with the velocity. Since the error is normalized against the maximal velocity of the physical runs it reveals that the upper quantile errors are quite consistent and do not change with velocity.

Continuing to the Car in Curve case which results can be seen in figure 12.7. The gap analysis shows both how the longitudinal model acts when decelerating and how the initial detections compare against each other between simulation and physical testing.

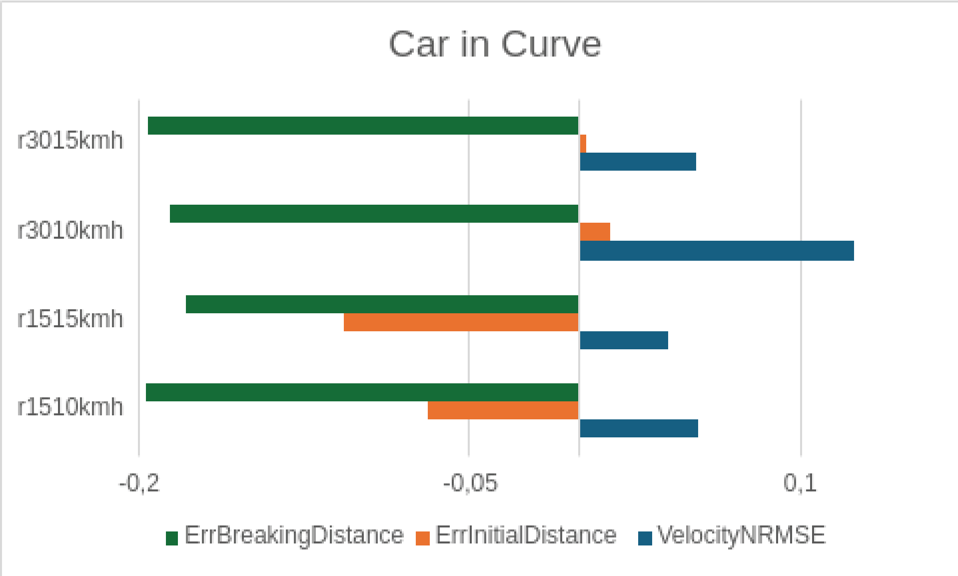

Figure 12-5: Einride gap analysis results for the Car in Curve scenario. The time frame for the underlying quantities is chosen from when the VUT starts driving to when it stops.

The distance errors are calculated by taking the differential between the obtained quantity in simulation against the physical test, then it is divided by the latter. This means that negative values show when the distance is shorter in simulation compared to the corresponding physical test.

Results reveal that the error for initial detection distances are quite small. With increasing velocity does the initial detection distance decrease compared to the physical tests. Also, the detected distance in simulation for the R15 scenarios decreased around 10%. Error in velocity stayed quite even, and did not change much excluding the R30 10km/h case where the error doubled compared to the other runs.

Finally, the Occluded Child scenario. The KPIs used to analyse Occluded child highlighted in the figure are the same as for the analysis of the Car in Curve scenario results, with addition of error at initial braking. It should also be noted that in none of the scenarios were there a collision with the VRU; neither in the physical nor in the simulation test. The velocity NRMSE was calculated from the window in time starting at 1.5 seconds before initial braking to 1.5 seconds after standstill. This time was chosen arbitrarily with the intent to capture the most essential parts of a safety stop. The results can be found in figure 12-6.

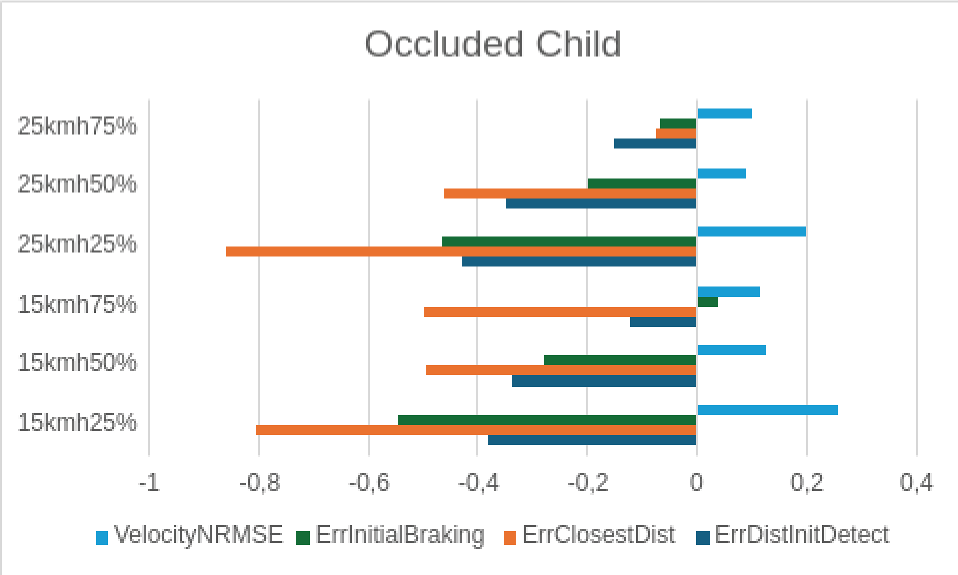

Figure 12-6: Einride gap analysis results for the Occluded Child scenario. The distances are recorded at the relevant times, and the Velocity NRMSE is calculated for the results show that with increased difficulty of the scenario (i.e. lower percentage of projected impact) the gap does increase between simulation and physical testing.

Firstly, the value of the error becomes larger when the projected point of impact (impact fraction) is decreased to 25%, while it always is negative. Consequently, the VUT in simulation consistently stops closer to the VRU and with decreasing impact fractions does the gap grow larger.

The outlier here is the 25km/h 25% case, which in theory would have the largest error but instead have the smallest closest distance error. From the nature of the scenario the VUT will stop closer and closer to the target as the impact fraction is decreased. So, it makes sense that the closest distance error is smaller here, because the margins become smaller.

Further supporting this result is the fact that the RMSE of the velocity increases with lower impact fraction. Since the error quantity is calculated in the time frame of interest to safety-region, a larger error here means that the velocity profiles during braking differ more.

Another aspect the results highlight is the difficulty of setting up the syncing of when the VRU should enter the road (because the simulations are run in closed loop configuration). Because the error at initial detection changes similarly from 25% to 75% between the two velocities it has seemingly little dependence on the velocity itself; it depends mostly on the impact distance. The negative value of this quantity shows that the VRU in simulation enters from behind the occluding stationary vehicle later than on the physical test.

Partner VTI:

At VTI the scenarios ALKS, Car in Curve and S-curve were driven in the simulator, both with and without motion for each driver. ALKS was driven once for each participant as either wobbly or straight car. Car in Curve was driven in speeds of 15km/h and 25 km/h for the two curves. The S-curve was driven in speeds of 30 km/h, 40km/h, 50km/h, and 60 km/h. In the simulator a recurring section was used to teleport drivers in between scenario so that drivers could drive all scenarios in sequence with balanced order. After the first run the same run was repeated but with motion on if it was turned off the first time or vice versa. This means that each driver in a compact way drives through all the scenarios and both runs (with and without motion) could be completed within one hour. An example of a simulator run for one driver was then: S-curve in 30 km/h, Car in Curve right curve in 15 km/h, S-curve in 40 km/h, Car in Curve right curve in 25 km/h, S-curve in 60 km/h, Car in Curve left curve in 25, S-curve in 50 km/h, S-curve in 30 km/h, Car in Curve left curve in 15, S-curve in 40 km/h, S-curve in 60 km/h, S-curve in 50 km/h, ALKS.

During the simulator study a total of 34 participants were recruited. With the drive consisting of many turns and speed changes a large amount of participant experienced sickness. In total 10 participants had to stop before completing the driving, thus only 24 participants were used for the data analysis.

But before we continue to present data from the scenarios it is good to be aware of the vehicle model parameterization. With some parts in the tyre model missing data, collected data from other tyre studies was used to give a more reasonable performance. The difference in dynamic response is showed in Figure 9.5-5 .

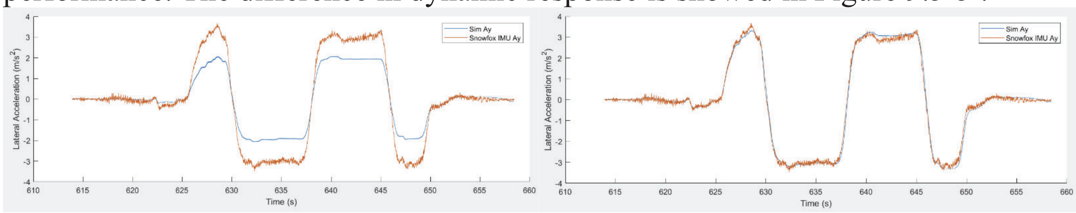

Figure 12-7: Vehicle model lateral acceleration during the S-curve. The left figure shows before and right figure shows after the tyre parameters modification.

With adjusted parameters we see that the lateral acceleration follows the logged data from test track. This was a good enough fit. For more accurate parameters it would be desirable to measure the tyres on the test vehicle in a tyre testing facility.

ALKS
In this scenario the oncoming car is sliding over into the lane of the simulator driver. An alert driver might notice that something is weird early on while some drivers react late. How drivers handled these situations are shown in figure 12-8and figure 12-9.

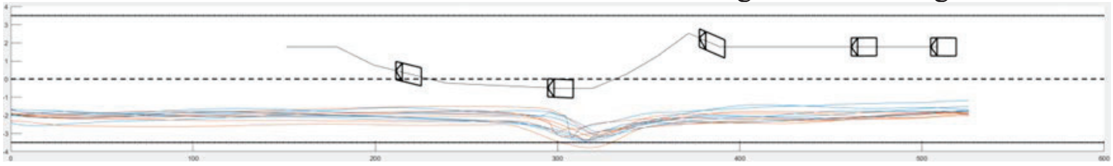

Figure 12-8: Simulator driver trajectories when meeting the wobbly ALKS vehicle.

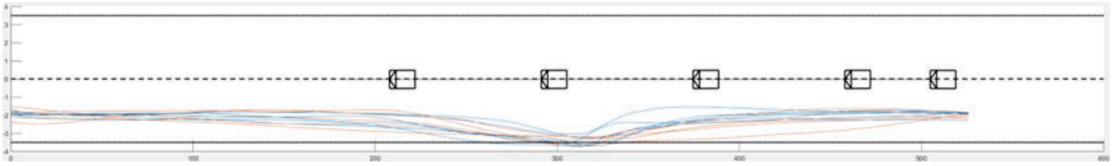

Figure 12-9: Simulator driver trajectories when meeting the straight ALKS vehicle.

As can be seen in these figures drivers laterally moves the vehicle to avoid collision. Investigating the influence of the moving base we look at the distance the driver moves the vehicle laterally. How this distance is calculated is shown in figure 12-10.

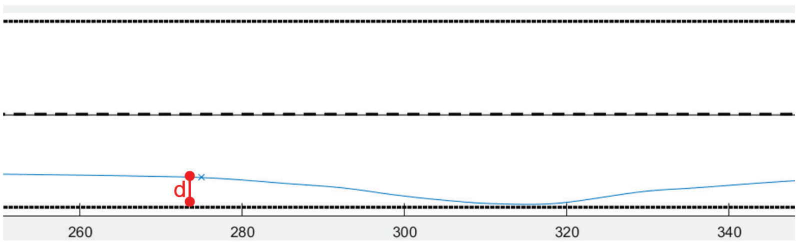

Figure 12-10: Lateral distance d a driver moves to avoid collision with an oncoming vehicle.

Using this distance the displacements for the drivers in the simulator are shown in . figure 12-11.

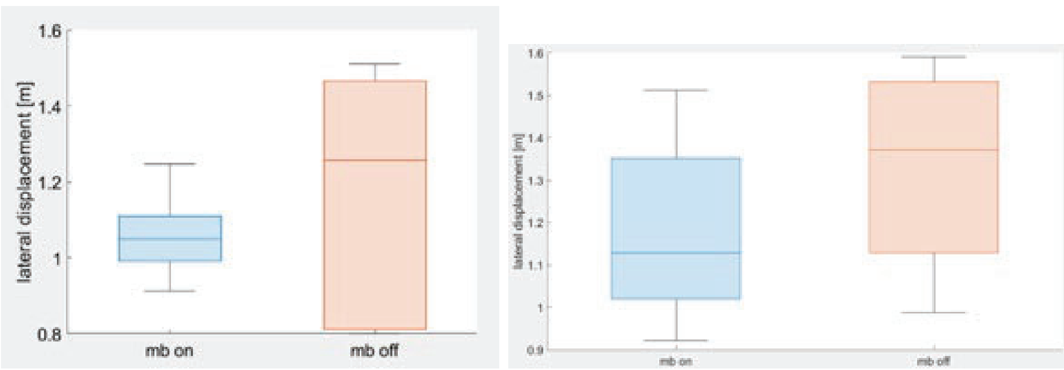

Figure 12-11: Difference in lateral displacement between moving base on/off for wobbly car (left figure) and straight car (right figure).

Here we see the moving base help drivers control their vehicle when something unexpected occurs. This is significant in the case of the wobbly car as this situation is more surprising compared to the straight car. As this scenario wasn't run on the test track, we couldn't perform any analysis on how an autonomous vehicle with radar manoeuvring would differ from driver behaviour.

Car in Curve

As the scenarios in the simulator was run in sequence the drivers were not instructed to specifically stop behind the standing still vehicle in the Car in Curve scenario but was instructed to drive as they normally do. As such, there were only two occasions where a drivers came to a halt, which are marked with an x in figure 12-12. Figure 12-12 also shows all the driver trajectories from the simulator drivers.

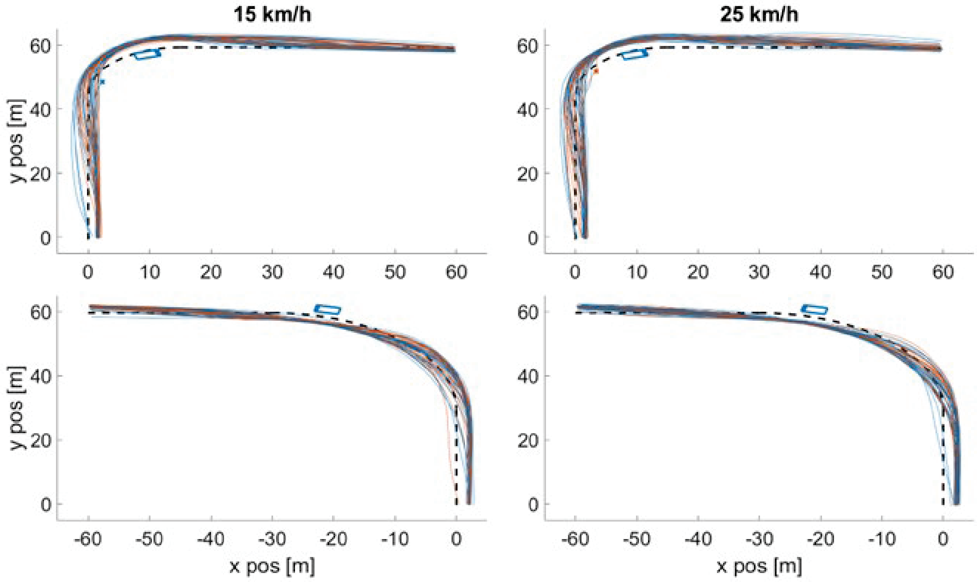

Figure 12-12: Simulator driver trajectories in the Car in Curve scenario. One blue x in the top left figure and one red x in the top right figure marks where a driver came to a complete stop.

Here it can be seen that there was a large span of different strategies of how and when to overtake the car. Some drivers start to break a little while others start to turn over to the opposing lane. To further look into this, we calculated the first instance when a driver takes action to either brake or turn. The distance from when a driver decides to act to the standing still car is then showed in figure 12-13.

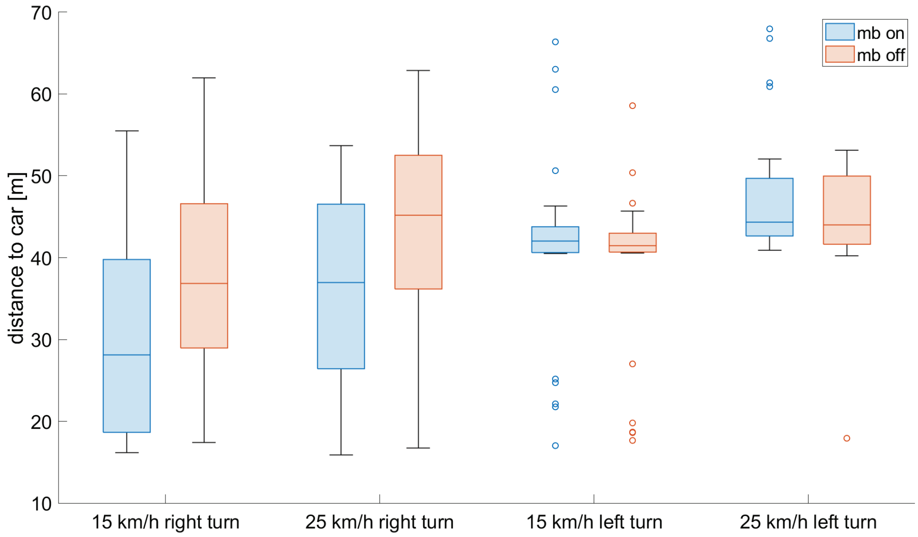

Figure 12-13: Distance between when a driver acts to the position of the standing still car for the different curves and speeds.

Here it can be seen that for the left turn the behaviour is similar with and without a moving base while for the narrower right turn there is a bigger difference in how close to the standing still car drivers choose to overtake. This would suggest that drivers with a moving base simulator chose to overtake closer to the car ahead.

At the test track the driver drove close to the standing still car and then stopped. After that the driver reversed away from the standing still car and continued to make a turn back to the starting position by overtaking the vehicle on the inside due to the layout of the road. As such, if an autonomous vehicle would stop at such a close and safe distance from the car this would probably be too close to what drivers would like and most drivers would expect the autonomous vehicle to overtake the standing still car.

S-curve

From the test track data there were issues with the GPS signal. As we want to compare the KPIs with respect to vehicle position we calculated a simplified odometer value for the test track vehicle. Using this odometer, we present the different KPIs for the 24 simulator drivers with the test track driver in 30 km/h in figure 12-14.

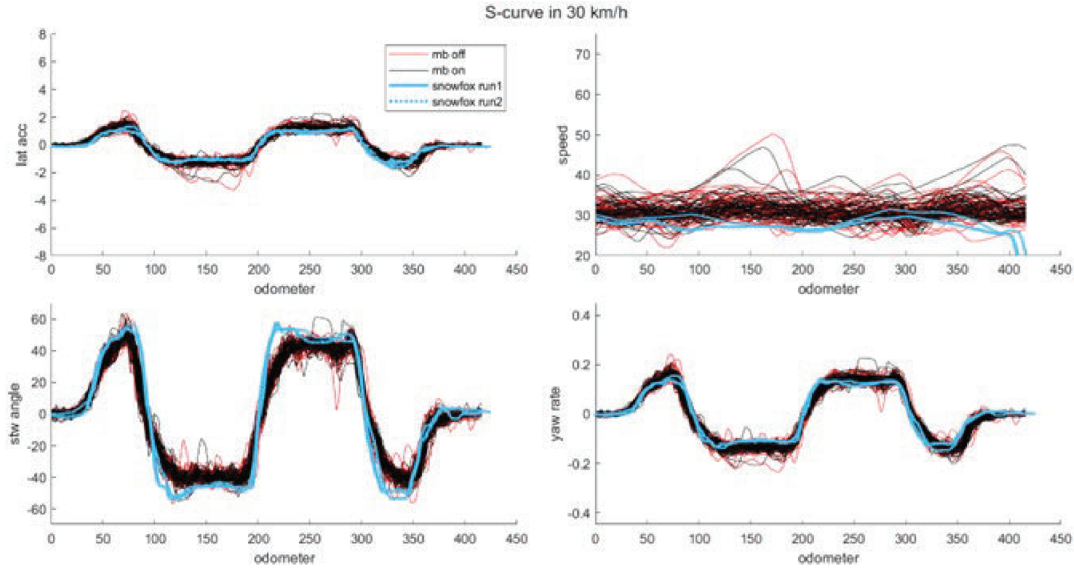

Figure 12-14: All the simulator and test track drives for the S-curve in 30 km/h.

In figure 12-14, we see that drivers in a simulator drove like the driver on test track. From the speed we can see that the test track driver drove slightly slower than the simulator driver, but this difference is within normal individual variation. For the lateral acceleration and the yaw rate there is a similar motion and for the steering wheel angle there is faster movement in the test track vehicle. If this is due to the simplified odometer or if the test track vehicle needs more steering to follow the S-curve could be further investigated although for such an analysis it would be good to re-run the test track test with GPS to get more accurate data. When looking at the speeds 40km/h, 50km/h and 60 km/h we see the same pattern and conclude that drivers in the simulator drivers as the test track drivers where differences between simulator or test track data is depending on the vehicle model or different drivers.

To further investigate differences in the simulator data with and without motion we look at the sustained curves in the s-curve. Here the first sustained curve is a right turn which then changes into a sustained left turn. Looking into these curves we show the KPIs for each of the different speeds and curves, see figure 12-15.

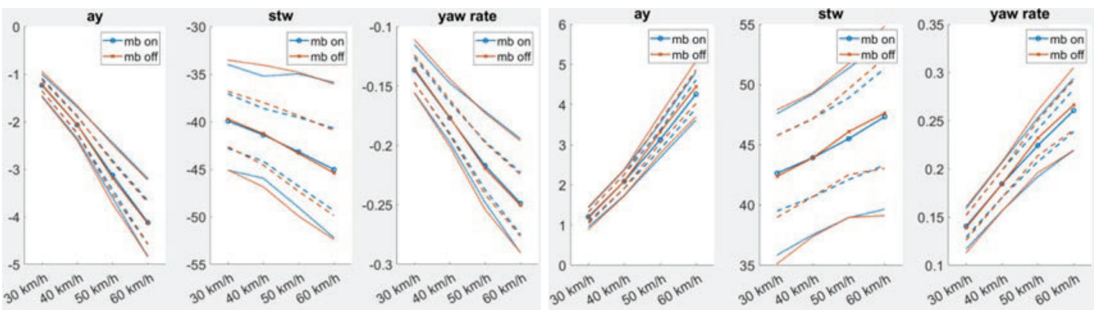

Figure 12-15: The KPIs lateral acceleration, steering wheel angle and yaw rate for all the S-curves in the simulator.

In these curves we can see that the difference between moving base on or off are small. We can further look at the position in the “lane” which in a similar way show only small differences, see figure 12-16.

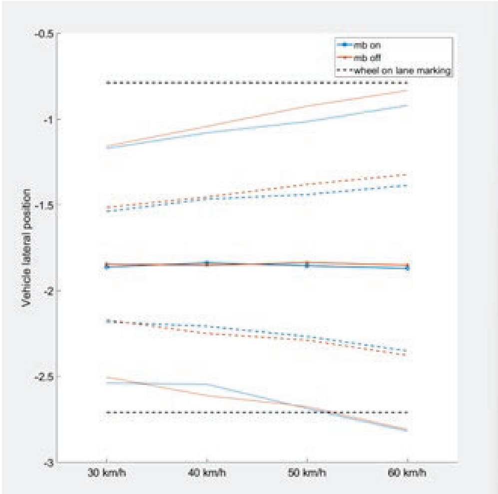

Figure 12-16: Difference in lateral position in the lane for the simulator drivers in the S-curve.

Lastly, we also present the percentage of drivers who drove outside the road in the S-curve in figure 12-17 for the different speeds.

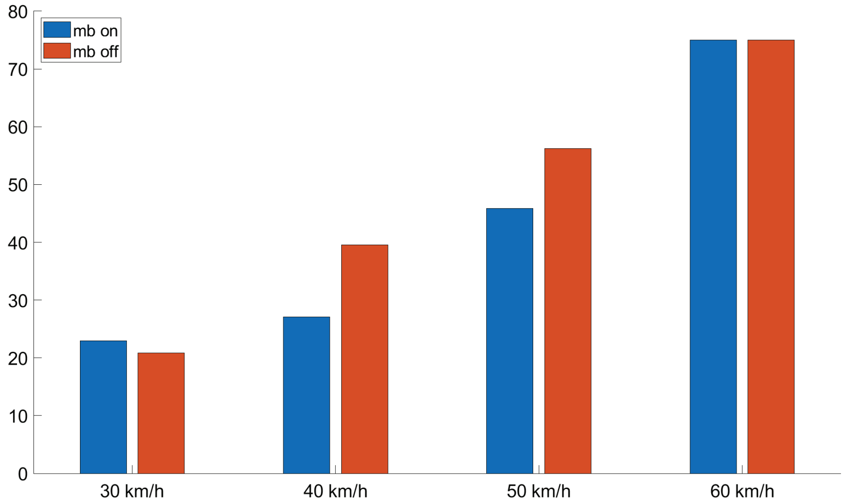

Figure 12-17: Percentage of simulator drivers that drove outside the lane in the S-curve in the different speeds.

Here we see that the moving base has its benefits in the range of 40-50 km/h. For slower speeds it is easy to maintain the position in the S-curve and for higher speeds it is difficult regardless of the moving base system. In summary, the observed benefits of a moving base system in this strictly defined scenario are small. It is worth noting that such small differences also could appear from people feeling sick, thus making it difficult to isolate only the effects from the moving base system. For this, further studies are needed.

Partner Zeekr:

The S-curve simulations were correlated against measurement data of the Chalmers REVERE Snowfox vehicle at AstaZero test track driven by a human driver. The results for 30 km/h correlation are shown and discussed below. Correlation for the remaining speeds show a similar trend.

S CURVE 30 kph	Sim	MEAS 1	MEAS 5	Unit
v_max	30.01	30.36	31.59	km/h
lat_acc_max	1.26	1.24	1.45	ms-2
lat_acc_min	-1.23	-1.45	-1.78	ms^{-2}
yawrate_max	8.64	9.19	10.00	deg/s
yaw_rate_min	-8.41	-6.84	-7.89	deg/s
SWA_max	42.02	57.49	56.60	deg
SWA_min	-40.88	-53.24	-53.29	deg
SWT_max	3.21	2.18	2.26	Nm
SWT_min	-3.13	-2.46	-2.62	Nm

Table 12-4: Chalmers REVERE S-Curve comparison

The figure 12-18: S-Curve plots, below shows that it is naturally easier to maintain a steady target velocity within simulation environments compared to test track measurements done by a human driver. There is a velocity reduction in measurement signals as the curves are being navigated. One can notice generally higher steering wheel angle peaks (both positive and negative) in the measurement signal.

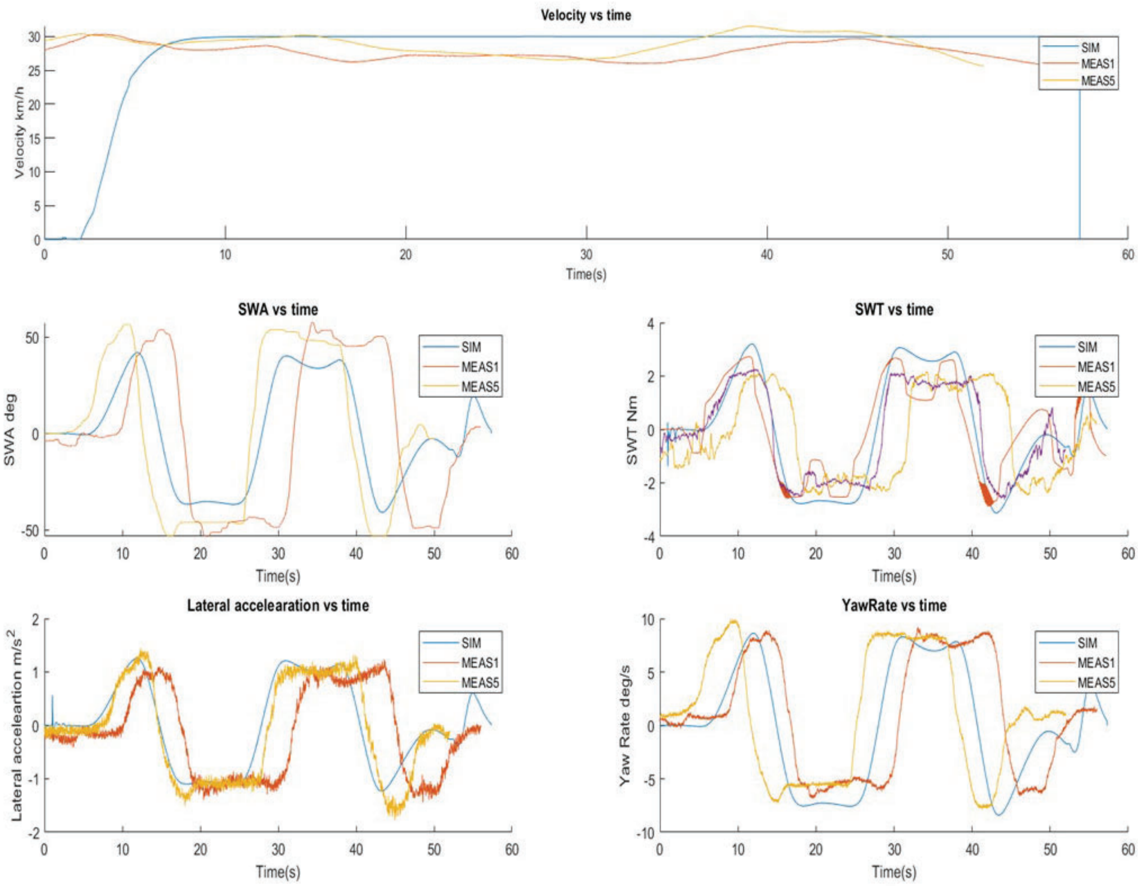

Figure 12-18: S-Curve plots

Despite the variability in the input signals, the comparison of lateral acceleration and yaw rate peak values differed by approximately 18% which is acceptable considering detailed measurements of the reference vehicle was not possible. Tire data, kinematic data, weight distribution data to name a few were left unmodified and the default model parameters in IPG Carmaker were used. The Steering wheel Torque (SWT) peaks showed anomalous behaviour (delta to measurement ~ 50%) and a sensitivity analysis was conducted to determine the root cause

Sensitivity analysis.

Sensitivity analysis was conducted to understand the root cause for the higher steering wheel torque in simulations. The lack of a steering assist or power-steering model was found to be the main contributor. The simulation loop was repeated with a Pfeffer power steering model available in IPG Carmaker and that resulted in a better correlation with measurement data as shown in the steering wheel plot at 30km/h below. The trend was similar for the other chosen speeds. This suggests the need for vehicle models with accurate representation of subsystem dynamics, which could play a vital role in simulation driven decision making in critical or emergency scenarios. Detailed tuning of the power steering parameters was not part of the project scope.

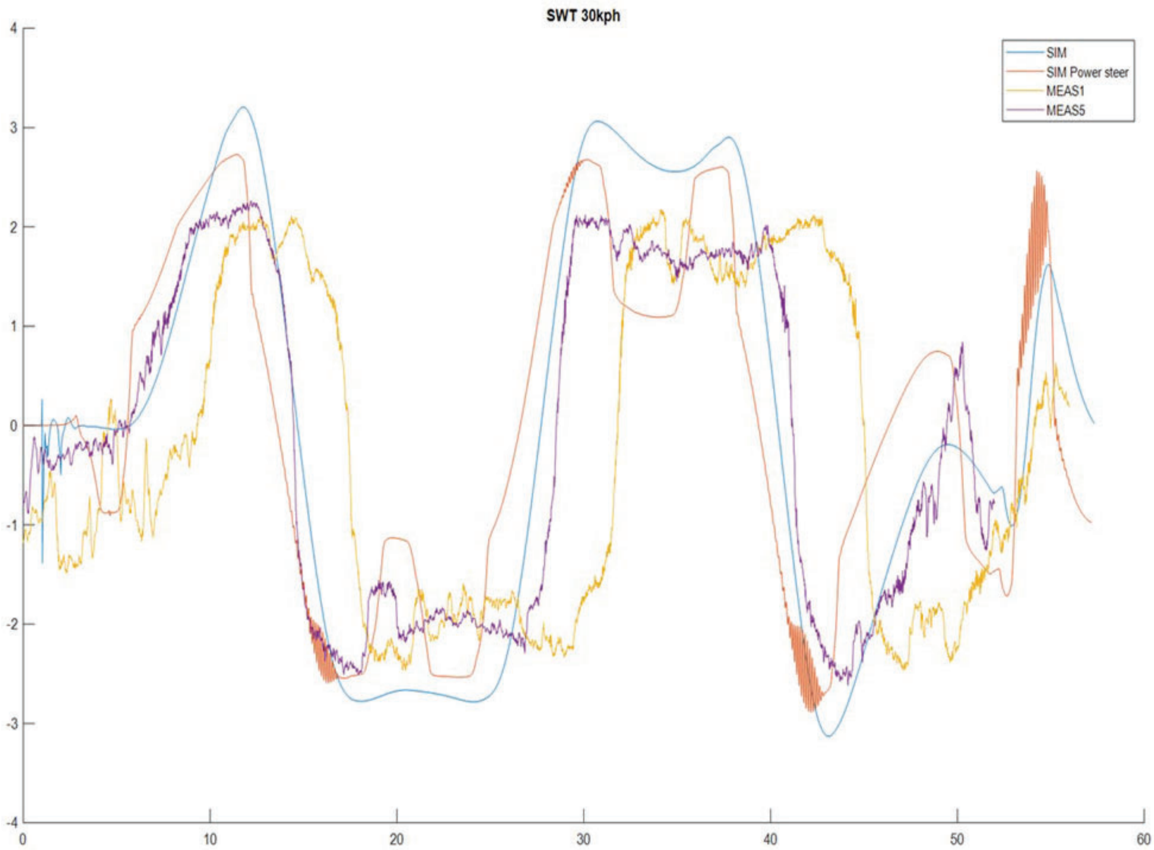

Figure 12-19: Steering wheel torque

It is also important to have access to visual data such as plots and animations for insightful understanding of system behaviour and quick debugging instead of merely comparing numerical metrics. There are associated risks that can arise when scaling simulations: a simulation error or a false positive (simulation test case represented as "passed" despite having failed) could go easily unnoticed when comparing just numerical metrics while evaluating and approving system performance.

For the CCFTap scenario, the simulations were correlated against measurement data from a Ford Mondeo vehicle measurement done by AstaZero. The simulation model of the Aptiv radar that was used in the measurement was not available due to a lack of legal/IP agreements for model sharing. Due to this, it was decided to evaluate existing radar model within VI-Worldsim. AEB activation was not included in the scope of this correlation due to the lack of detailed information regarding the brake system and tire parameters. It was highly likely that accurate modelling of these components would be needed for a detailed correlation. The red vertical line in the, Figure 12-7: CCFTap plots, below is the first instance of brake signal at 100% demand.

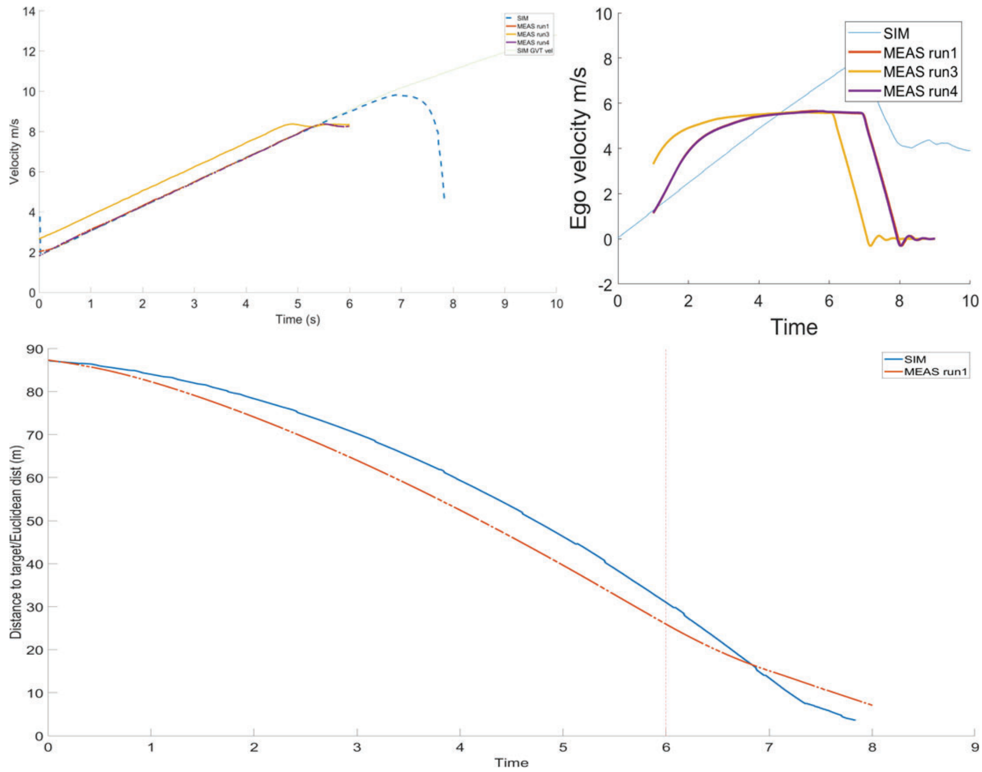

Figure 12-20: CCFTap plots

The velocity output of the radar model in simulation closely matched the velocity of the target vehicle. The velocity of the target vehicle in simulation was tuned with aggression factors, speed multipliers and the behaviour were matched with the measurement target vehicle. This was a straightforward process since the target vehicle velocity was changing in a linear fashion.

The distance to target correlation had an offset as seen in the plot and this is likely attributed to the ego velocity behaviour in simulation not matching the measurement velocity. It is possible to connect the VI-Worldsim simulation to the vehicle dynamic solvers available in VI-CarRealTime to enable better control of the ego vehicle in simulation, but this was not part of the project scope due to time constraints. The ego velocity behaviour tuning was only possible in a linear fashion in the Worldsim editor, but it is very likely that the offset can be improved once the ego velocity of the measurement can be mirrored in the simulation model.

Although these default simulation models cannot be used for final validation and calibration, they can easily be used for concept development, requirement setting and better understanding of product performance which will play a crucial role in writing good technical requirements and enable productive communication and decisions to suppliers. This will establish an efficient flow of communication across different parties from an early stage in the development process.

13. Argument-based Credibility Assessment for Virtual Toolchains

NATM does not explicitly identify evidence to provide to relevant authorities. However, during our analysis, we encountered several points in NATM that discuss some artifacts that can be considered evidence in an assurance case to fulfil other requirements in NATM:

- A process to identify and evaluate the individual's competence and skills is established.
- A process for training competent personnel to perform Models & Simulation (M&S)-related duties is established.
- Basis for the ADS manufacturer's confidence in the Experience and Expertise of the individual/team that uses the simulation to execute virtual testing with the purpose of validating the ADS.
- Demonstration of applying the principles of Management Systems through best practice or standard, with regard to the competence of M&S organization and the individuals in that organization.
- Results from exploitation of redundant information.
- Report showing that numerical methods do not introduce flaws in the virtual models.
- Known range of input parameters and their combinations that do not lead to unstable or unrealistic behaviour.
- Report of critical parameters influencing the simulation output by means of sensitivity analysis techniques.
- Robust calibration procedures that are adopted to identify and calibrate the most critical parameters.
- Validation reports showing that KPIs are met.
- Report showing that accuracy requirements are under a provided threshold.

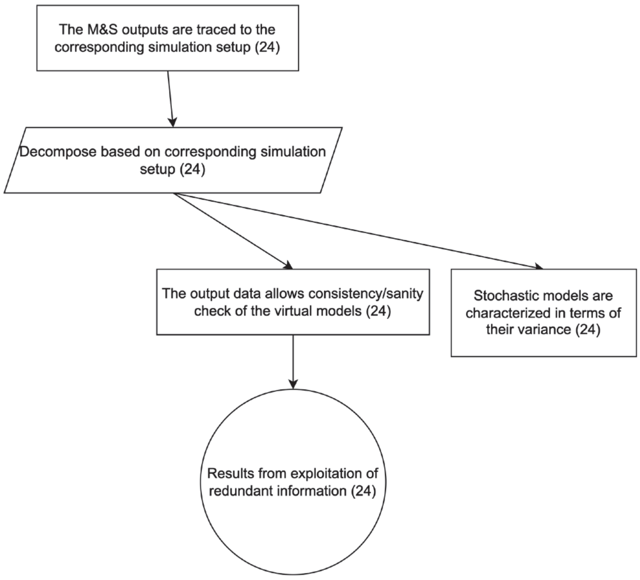

Figure 13-1: An excerpt from the assurance case for virtual toolchain credibility assessment.

These evidences, however, are not enough and do not cover all of the claims that need to be made to in an argument-based credibility assessment following NATM, i.e., there are multiple claims that need to be made in order to assure the credibility of virtual toolchains for AD functions based on Annex III of NATM lists the following (followed by the number of the point in NATM).

As part of the analysis of the NATM, we observed that many of the items that can be considered as a requirement, are not accompanied by a concrete way to fulfil them. Thus, in an NATM-based credibility assurance case, additional evidence would need to be proposed in order to guarantee the fulfilment of said requirements. For instance, in the case of point 24 in the NATM Annex III, the need to trace the M&S out to the corresponding simulation setup is presented as a claim that is broken down into sub-claims for which NATM sometimes fails to specify concrete evidence. Figure 13.1 shows two of these sub-claims, out of which only one is accompanied by potential evidence that can justify it.

Throughout our analysis, we found that NATM expects some documentation and artifacts to take place for the credibility assessment. These could be mapped to context notes in proposed assurance cases. Examples of these contextual notes that are explicitly listed in NATM include:

- Operational Design Domain (ODD) description.
- Description of the scope and limitations of the models and tools in the toolchain, and their uncertainty sources.
- Definition of acceptable accuracy as per the relevant standards.

From this analysis we conclude that using assurance cases to structure the requirements for credibility assessment of virtual toolchains can prove to be beneficial. Future efforts could therefore involve providing a template that any industrial organization working with virtual toolchains for virtual testing can use to prove that their virtual toolchains are credible and usable for AD V&V.

Additionally, future work is needed to make sure that provided evidence for a certain claim sufficiently justify the claim. This will help build confidence in the credibility of both the provided evidence and the virtual toolchain as a whole.

14. Challenges identified with in the EVIDENT project

In this chapter, we present the results from the study we conducted to elicit and classify the challenges that the EVIDENT partners have encountered. Our study resulted in 17 different challenges classified based on the partners' feedback on whether the challenge is: (i) Already solved, (ii) Next frontier, and (iii) Long-term future.

Already solved:

- Simulation models lack correlation with real sensor data.
- Limited and shared access to specific physical test tracks requires frequent changes in simulation scenarios.
- Ensuring consistent timestamping and synchronization of simulated data with the PT (Physical Twin)
- Lot of [human] resources have been changed during the project.
- Sensor failures or malfunctions leading to missing data
- Lack of clear data requirements makes it difficult to plan the use of sensors.

Next frontier:

1. Verifying automotive functionalities through simulations is hindered by differences between physical and simulated "ideal" sensors, and data gathering limitations.
2. Integrating different proprietary simulation, data formats, and platforms requires significant effort, despite industry standards.
3. Parsing complex simulation data into manageable formats for model fidelity and correlation analysis
4. Verifying automotive functionality through simulations requires determining "fidelity gap" and establishing meaningful metrics to verify system performance
5. Simulation fidelity varies greatly depending on project scope and phase (within V model), which was not specified beforehand
6. Never defined whether to simulate an autonomous driving system or a human driver for verification purposes
7. Lack of clear and detailed sensor specifications makes it difficult to develop accurate simulations.
8. Unexplained distortions occurred in physical sensors' log data

Three of the challenges were classified as long-term future challenges by the project partners. One of them is the difficulty of balancing fidelity between simulation and reality to accurately represent full system behaviour in a trust-worthy and comprehensible way for easy debugging. A related challenge was also described as long-term by the participants: ensuring that test cases and simulation results are statistically significant and representative of real-world scenarios. This is surprising given the large body of work in the academic literature covering these challenges.

Moreover, sharing simulation models across organizations poses, according to the surveyed partners, legal and intellectual property concerns. This seems to be a long-term challenge, even though it is not a technical one.

15. Evident projects contribution to the FFI Objectives and sub-program.

Here we describe how the project correlates with FFI 2021 objectives:

Overarching 2021 FFI objective to meet:	**In the application we had the intention to addresses below goals**	**The project has by this argumentation reach the goal**
"increasing the Swedish capacity for research and innovation, thereby ensuring competitiveness and jobs in the field of vehicle industry"	...by conducting systematic research into a topic area that currently challenges the established automotive OEMs and Tier-1 suppliers.	The project has found that all partners have unknown fidelity gaps between physical testing of a vehicle and performing same test in virtual environment with simulation models. This challenges the OEMs and Tier-1 to improve their virtual chain but also give them a draft method to improve even more. This improves Swedish automotive competitiveness.
"promoting cooperation between industry, universities and higher education institutions"	...by setting up an academic/industrial research project supervised by renowned practitioners and researchers in the domain of ADAS/AD development and V&V for automotive function. The unique and competitive edge for this proposal's consortium is that our team covers several OEMs (CEVT, Einride), SME (Asymptotic.AI) with state-of-the-art methods and tools for cloud-enabled AI/ML methods, VTI to provide long experience for virtual testing-based V&V for automotive features, and three strong academic partners (Chalmers, RISE and University of Gothenburg) representing years-long academic research for automotive features.	In the way the project has been conducted with a lot of collaboration in between the different partners for example Astazero have delivered the open road and Open scenario to all partners. Data from physical drives from both AstaZero and Chalmers Revere have been used as comparison for both VTI and Zeekrs studies. RISE, Chalmers, and University of Gothenburg have conducted workshops to gather best practices and challenges discussions in between partners. An open discussion has also been performed during final presentation at SAFER event.

"promoting cooperation between different OEM"	...by including CEVT and Einride each focusing on different vehicular platforms for different operational design domains and hence, representing such cross-collaboration in an exceptional project setup.	By sharing problems and solutions in between all partners openly we have had the opportunity to boost collaboration between OEM's and SME Asymptotic AI, including technical workshops with Zeekr (formerly CEVT) and Aptiv to refine high-fidelity simulation tools and align on scenario generation requirements for autonomous vehicle validation.
"promoting the participation of small and medium-sized companies"	...by including the SME Asymptotic.AI and putting their leading experience and toolchain in cloud-enabled AI/ML at the core of WP-4. In this WP Asymptotic will contribute towards establishing a cloud-enabled and automatic way to feed virtual testing environments with relevant scenarios from reality. By doing this, one can push to edge case situations in the virtual world to increase the degree of fidelity.	Asymptotic AI has developed tools to generate high-fidelity simulation scenarios from real-world data. Their automated annotation system accurately labels multi-sensor inputs (camera, lidar, radar) with per-point instance IDs and bounding boxes. By addressing challenges like data association, object tracking over time, and handling occlusions, these contributions ensure the creation of realistic, reusable scenarios. These tools have been validated using datasets such as those from AstaZero, supporting reliable validation of autonomous vehicle systems.

Table 15-1: The overarching FFI objectives and how EVIDENT project addresses the goals.

Here we describe how the project correlates with the 5 different sub-program areas:

Sub program areas:	**In the application we had the intention to addresses at least three areas**	**The project has by this argumentation reach the goal addressed four areas.**
Architecture	The strategies developed within this project will enable industry to develop software and hardware more effectively, as testing using virtual elements can be incorporated much earlier during the development process.	By using the draft methods and knowledge developed with in the EVIDENT project the industry could accelerate the use of virtual testing with improved fidelity. We have also described the architecture of several virtual test chains, and the physical test chain used in EVIDENT.
Intelligent and Reliable Systems	Similarly, this consortium includes leading expertise covering today's needs for training and validating neural networks (NNs) using these virtual approaches.	By understanding the gap between ideal sensor testing in virtual environments and physical sensor testing at proving ground we have be able to highlight the challenges and needs of virtual models that have correct sensor characteristics to avoid ideal sensors.
Human-Machine Interaction		EVIDENT have also investigated the differences in how a human drive these test cases in different level of simulators and compared with real drives on test track.
Verification and Validation.	Furthermore, with the aim of utilising virtual testing as means of verification and validation of vehicle functions, it contributes also to the sub-program area Verification and Validation.	The complete project focusing on how we could get trustworthy verification within simulation and how to get necessary fidelity level for each functional test level in the simulations. Gap analyses are performed, and draft method are presented.
Elektronik för ”Gröna, säkra, autonoma och uppkopplade funktioner”		By focusing on moving testing from test track with increased fidelity we hope to accelerate the virtual testing and reduce energy consuming physical test to only validation. We still see that the goal is a little bit further on in the future, but we have found several pieces that take us closer and more research questions that needs to be investigated.

Table 15-2: The Sub-program priority areas and how EVIDENT project addresses the goals

16. Dissemination and publications

16.1 Dissemination

How are the project results planned to be used and disseminated?	Mark with X	Comment
Increase knowledge in the field	x	The EVIDENT project significantly contributed to advancing knowledge in the field of virtual validation for ADAS/AD features. By addressing the fidelity gap between physical and virtual testing and proposing a credibility assessment framework for simulation toolchains, the project provides valuable insights for academia and industry. The project has also resulted in several publications and conference presentations, disseminating findings to a broader audience.
Be passed on to other advanced technological development projects	x	The project's findings on gap analysis and virtual validation methodologies have been shared with partners involved in related technological projects, particularly those focusing on autonomous systems, simulation environments, and safety assurance. The open-source tools and standardized formats (OpenDRIVE, OpenSCENARIO) used in EVIDENT are also likely to be reused and refined in future projects.
Be passed on to product development projects	x	The project's findings on gap analysis and virtual validation methodologies have been shared with partners involved in related technological projects, particularly those focusing on autonomous systems, simulation environments, and safety assurance. The open-source tools and standardized formats (OpenDRIVE, OpenSCENARIO) used in EVIDENT are also likely to be reused and refined in future projects.
Introduced on the market		While the project itself did not introduce a commercial product, it laid the groundwork for future market introductions by providing tools, methods, and best practices for virtual validation. The open-source tools and credibility assessment framework have the potential to be integrated into commercial simulation platforms or testing services in the near future.
Used in investigations / regulatory / licensing / political decisions	x	The project aligns with the United Nations NATM guidelines and has contributed to discussions on the credibility of virtual toolchains for ADAS/AD system validation. The draft method for credibility assessment could influence regulatory bodies by providing a structured approach to evaluating virtual testing, which is critical for future regulatory and licensing frameworks for autonomous vehicles.

Table 16-1: Dissemination table

16.2 Publications

EVIDENT project results are or will be presented in several papers and conferences. Below is a list of publications and presentations.

Publication name	Author	Conference Date	Status at report release	Link
Digital Twins for Early Verification and Validation of Autonomous Driving Features: Open-source Tools and Standard Formats	• Beatriz Cabrero-Daniel • Ahmed Yasser Abdelkarim • Axel Broberg	IEEE 2-5 June 2024	Presented Released	https://ieeexplore.ieee.org/abstract/document/10588808
Digital Twin-based Simulated Automotive Radar for Virtual Testing	• Silian Karadag	IPG Apply Innovate 11-12 September 2024	Presented Released	Thesis-Digital Twin-Based Simulated Automotive Radar for Virtual Testing
A Practitioners Perspective on Fidelity of Simulations for V&V of Automated Driving	• M. Cagri Kaya • Ali Nouri • Beatriz Cabrero-Daniel • Christian Berger • Mazen Mohamad	36th IEEE Intelligent Vehicles Symposium	Submission deadline of 1 February 2025 Notification date of March 30th	
Challenges	• Beatriz Cabrero Daniel • Mari Eriksson • Mazen Mohamad	36th IEEE Intelligent Vehicles Symposium	Submission deadline of 1 February 2025 Notification date of March 30th	
Argument-based Credibility Assessment for Virtual Toolchains	• Mazen Mohamad • Beatriz Cabrero Daniel	DSN 2025 – Industry track		
Driving behaviour comparison between data from proving ground and simulation	• Anders Andersson • Maytheewat Aramrattana	Transportforum 2025 15-16 Jan 2025	Presented	

Table 16-2: EVIDENT Publications

17. Conclusions and future research

The EVIDENT project has successfully introduced a draft methodological approach for conducting gap analysis between physical and simulation tests. This project has provided a clearer understanding of the necessary steps to achieve broader utilization of simulations in comprehensive vehicle testing and validation, including functional safety proof, EuroNCAP tests, and type approval.

The project has made several clear observations in our pursuit of high fidelity:

- **Control Over Simulation Models**:
    - Full control over all simulation models is essential. This includes understanding how each simulation model impacts test results and determining the appropriate test level and extent for each simulation test results.
    - Each simulation model must be correlated with reality across various scenarios, and it is recommended to perform a sensitivity analysis to establish key parameters which can influence overall result

- **Environment for Gap Analyses:**
    - The virtual test environment needs to be well-designed and incorporate all active and inactive objects (host and target), as well as disturbing elements.
    - We need to know the validity and fidelity of the environment, including the surrounding environment as a disturbing factor in the test.
    - The project has proven that external annotations on real-world sensor data can be used as input for simulations.
    - The project also sees the possibility of using annotation comparisons between the physical and virtual environments as a tool for assessing environment fidelity. Some work is still needed to improve size and heading estimation for stationary and distant objects through machine learning models.

- **Sensor Fusion:**
    - The project could not validate the sensor fusion performance, due to most of the partners are working with ideal radar sensors from simulation manufacturers. The ideal sensors are tuned against physical radar used in the vehicle. However, these sensors still fall short of accurately representing real-world conditions at this level.

- **Vehicle Dynamics and Control:**
    - Vehicle dynamics and control significantly impact the outcomes. If the dynamics in the simulation differ even slightly from those in real tests, uncertainties are introduced, reducing the credibility of the simulation.
    - Factors such as brake system response, steering models, and tires can affect the end result.
    - As the vehicle project progresses and control systems are fine-tuned, vehicle and control system models must align with controller calibration

stages. This should increase simulation model fidelity and reduce the correlation gap

- **Physical Test Procedure:**
  - To achieve accurate simulation model gap analysis, we need to ensure reproducible physical test cases.
  - The project recommends using a virtual driver/control robot in the initial investigation loops to create similar simulation scenarios and reduced variability.
  - Choosing scenarios and KPIs should be done with consideration of the physical environment and possible limitations, to minimize the risk of needing to change scenarios due to physical reproducibility issues.

- **Virtual Test Procedure:**
  - The virtual test procedure with physical drivers in a simulation needs to have clear instructions, an extended environment, and possibly some looping to achieve very similar trajectories. This is because humans may find ways that were not anticipated in the scenario build.
  - It is very important to be able to replicate all timings between physical and virtual drive, in the occluded child we could see that this could be tricky and be a source of error; however, eventually, it was possible to solve.

- **Test Matrix:**
  - To perform high-fidelity tests of level 2 and 3 vehicles with human drivers, the test matrix needs to incorporate the variability of different drivers with statistical significance.
  - Even physical systems have imperfections and tolerances, so a collection of test scenarios and repeatable tests is needed to fully understand the behaviours that should be replicated in the simulations.

- **Decision Matrix and Positioning Algorithms:**
  - Decision matrix and positioning algorithms can be tested by all partners with varying fidelity due to different gaps within simulation models. The project has identified issues in vehicle dynamics, speed models, AEB, acceleration, and longitudinal and lateral positioning.
  - Despite these issues, we have proven that high fidelity can be achieved within simulations.

- **Collaboration:**
  - The project has demonstrated that it is entirely possible to collaborate using test and environment descriptions specified by OpenDRIVE and OpenSCENARIO formats, across several toolchains without impacting any partner's intellectual property rights. The project has even used the same physical test objects and shared sensor data when the data is not sensitive.
  - To enhance collaboration and the potential to reuse data, further work on standardization is recommended.

- **Credibility:**
  - Throughout the project, it has become increasingly evident that there is a significant need for a framework to assess the credibility of simulation models and the complete toolchain.

A key takeaway is that physical testing remains crucial for analysing and understanding vehicle dynamics, as well as evaluating how sensors and their corresponding mathematical models operate in real-world conditions.

In this project, we did not account for weather conditions and illumination, which are important factors for fully understanding the gap between physical and virtual testing. This requires further studies.

This research has made significant strides towards developing a clear method for evaluating simulation models against physical systems. Each partner has identified several potential improvement points in the simulation models, and while many issues have been resolved, some still persist.

Further studies are needed to provide guidance on the validity and fidelity levels required for different purposes when using simulations and virtual environments as tools or verification proof. The studies need to ensure a statistically verified test step up and analysed to ensure deviations are correctly included in the recommended fidelity thresholds. Establishing a common framework for performing credibility assessments of simulation chains and environments will support the use of simulations in regulatory testing, type approval, and certifications such as EuroNCAP. But will also give the trust for simulations needed to kick off the broad usage and possibility to replace physical testing.

18. Participating parties and contact persons

Partner	Partner Contribution	Contact person	Role	Logo
AstaZero AB	Coordinator, Physical test facility, Test vehicle provider Simulations	Mari Eriksson	Project manager	
Aptiv	Test vehicle provider Simulations	Henrik Clasen	Steering group	
Asymptotic	Environment Annotations	Samuel Scheidegger	Steering group	
Zeekr	Simulations	Alejandro Morate	Steering group	
Chalmers Revere	Researcher Test vehicle provider	Anna K Carlsson	Steering group	
GU	Researcher	Christian Berger	Steering group	
Einride	Test vehicle provider Simulations	Andreas Allström	Steering group	
RISE	Researcher	Mazen Mohamad	Steering group	
VTI	Researcher Simulations	Magnus Eek	Steering group	

Table 18-1: Partner contacts

19. References

1. [1] S. Shah, D. Dey, C. Lovett, A. Kapoor, Airsim: High-fidelity visual and physical simulation for autonomous vehicles, in: Field and Service Robotics: Results of the 11th International Conference, Springer, 2018, pp. 621–635.
2. [2] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, V. Koltun, Carla: An open urban driving simulator, in: Conference on robot learning, PMLR, 2017, pp. 1–16.
3. [3] Y. Bengio, J. Louradour, R. Collobert, J. Weston, Curriculum learning, in: Proceedings of the 26th annual international conference on machine learning, 2009, pp. 41–48.
4. [4] P. Soviany, R. T. Ionescu, P. Rota, N. Sebe, Curriculum self-paced learning for cross-domain object detection, Computer Vision and Image Understanding 204 (2021) 103166.
5. [5] C. Florensa, D. Held, M. Wulfmeier, M. Zhang, P. Abbeel, Reverse curriculum generation for reinforcement learning, in: Conference on robot learning, PMLR, 2017, pp. 482–495.
6. [6] J. Schmidhuber, Evolutionary principles in self-referential learning, or on learning how to learn: the meta-meta-... hook, Ph.D. thesis, Technische Universität München (1987).
7. [7] Y. Jaafra, A. Deruyver, J. L. Laurent, M. S. Naceur, Context-aware autonomous driving using meta-reinforcement learning, in: 2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA), IEEE, 2019, pp. 450–455.
8. [8] A. Kar, A. Prakash, M.-Y. Liu, E. Cameracci, J. Yuan, M. Rusiniak, D. Acuna, A. Torralba, S. Fidler, Meta-sim: Learning to generate synthetic datasets, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 4551–4560.
9. [9] G. Hinton, O. Vinyals, J. Dean, Distilling the knowledge in a neural network, arXiv preprint arXiv:1503.02531 (2015).
10. [10] M. R. U. Saputra, P. P. De Gusmao, Y. Almalioglu, A. Markham, N. Trigoni, Distilling knowledge from a deep pose regressor network, in: Proceedings of the IEEE/CVF international conference on computer vision, 2019, pp. 263–272.
11. [11] L. Zhang, R. Dong, H.-S. Tai, K. Ma, Pointdistiller: structured knowledge distillation towards efficient and compact 3d detection, arXiv preprint arXiv:2205.11098 (2022).
12. [12] J. Morimoto, K. Doya, Robust reinforcement learning, Neural computation 17 (2) (2005) 335–359.
13. [13] L. Pinto, J. Davidson, R. Sukthankar, A. Gupta, Robust adversarial reinforcement learning, in: International Conference on Machine Learning, PMLR, 2017, pp. 2817–2826.
14. [14] X. He, B. Lou, H. Yang, C. Lv, Robust decision making for autonomous vehicles at highway on-ramps: A constrained adversarial reinforcement learning approach, IEEE Transactions on Intelligent Transportation Systems (2022).
15. [15] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba and P. Abbeel, "Domain randomization for transferring deep neural networks from simulation to the real world," 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 2017, pp. 23-30, doi: 10.1109/IROS.2017.8202133.
16. [16] G. D. Kontes, D. D. Scherer, T. Nisslbeck, J. Fischer, C. Mutschler, High-speed collision avoidance using deep reinforcement learning and domain randomization for autonomous vehicles, in: 2020 IEEE 23rd international conference on Intelligent Transportation Systems (ITSC), IEEE, 2020, pp. 1–8.

17. [17] S. Pouyanfar, M. Saleem, N. George, S.-C. Chen, Roads: Randomization for obstacle avoidance and driving in simulation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, 2019, pp. 0–0.
18. [18] K. Weiss, T. M. Khoshgoftaar, D. Wang, A survey of transfer learning, Journal of Big data 3 (1) (2016) 1–40.
19. [19] D. Isele, A. Cosgun, Transferring autonomous driving knowledge on simulated and real intersections, arXiv preprint arXiv:1712.01106 (2017).
20. [20] J. Kim, C. Park, End-to-end ego lane estimation based on sequential transfer learning for self-driving cars, in: Proceedings of the IEEE conference on computer vision and pattern recognition workshops, 2017, pp. 30–38.
21. [21] S. Sharma, J. E. Ball, B. Tang, D. W. Carruth, M. Doude, M. A. Islam, Semantic segmentation with transfer learning for off-road autonomous driving, Sensors 19 (11) (2019) 2577.
22. [22] R. T. Azuma, A survey of augmented reality, Presence: teleoperators & virtual environments 6 (4) (1997) 355–385.
23. [23] E. Costanza, A. Kunz, M. Fjeld, Mixed reality: A survey, Springer, 2009.
24. [24] P. Lindemann, T.-Y. Lee, G. Rigoll, An explanatory windshield display interface with augmented reality elements for urban autonomous driving, in: 2018 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct), IEEE, 2018, pp. 36–37.
25. [25] S. Su, X. Zeng, S. Song, M. Lin, H. Dai, W. Yang, C. Hu, Positioning accuracy improvement of automated guided vehicles based on a novel magnetic tracking approach, IEEE Intelligent Transportation Systems Magazine 12 (4) (2018) 138–148.
26. [26] R. Moezzi, D. Krcmarik, H. Bahri, J. Hlava, Autonomous vehicle control based on hololens technology and raspberry pi platform: An educational perspective, IFAC-PapersOnLine 52 (27) (2019) 80–85.
27. [27] X. Tu, J. Autiosalo, A. Jadid, K. Tammi, G. Klinker, A mixed reality interface for a digital twin based crane, Applied Sciences 11 (20) (2021) 9480.
28. [28] B. Yu, C. Chen, J. Tang, S. Liu, J.-L. Gaudiot, Autonomous vehicles digital twin: A practical paradigm for autonomous driving system development, Computer 55 (9) (2022) 26–34.
29. [29] H. Xiong, Z. Wang, G. Wu, Y. Pan, Design and implementation of digital twin-assisted simulation method for autonomous vehicle in carfollowing scenario, Journal of Sensors 2022 (2022).
30. [30] A. Niaz, M. U. Shoukat, Y. Jia, S. Khan, F. Niaz, M. U. Raza, Autonomous driving test method based on digital twin: A survey, in: 2021 International Conference on Computing, Electronic and Electrical Engineering (ICE Cube), IEEE, 2021, pp. 1–7.
31. [31] K. Voogd, J. P. Allamaa, J. Alonso-Mora, T. D. Son, Reinforcement learning from simulation to real world autonomous driving using digital twin, arXiv preprint arXiv:2211.14874 (2022).
32. [32] Z. Hu, S. Lou, Y. Xing, X. Wang, D. Cao, C. Lv, Review and perspectives on driver digital twin and its enabling technologies for intelligent vehicles, IEEE Transactions on Intelligent Vehicles (2022).

33. [33] W. P. on Automated/Autonomous and C. Vehicles, “New Assessment/Test Method for Automated Driving (NATM) Guidelines for Validating Automated Driving System (ADS),” Economic Commission for Europe (ECE), Apr. 6, 2023, Accessed: 2024-10-21. [Online]. Available: WP.29-2023-44e.pdf.
34. [34] International Organization for Standardization, “ISO 26262 Road vehicles Functional safety, 1st Edition,” Geneva, Switzerland, 2011.
35. [35] International Organization for Standardization and Society of Automotive Engineers International, “Road vehicles — cybersecurity engineering,” ISO/SAE International, Vernier, Geneva, CH, Standard ISO/SAE 21434:2021, Aug. 2021, p. 81. [Online]. Available: https ://www.iso.org/standard/70918.html
36. [36] M. Mohamad, J.-P. Steghöfer, and R. Scandariato, “Security assurance cases—state of the art of an emerging approach,” Empirical software engineering, vol. 26, no. 4, p. 70, 2021.