ROADVIEW

Robust Automated Driving in Extreme Weather

Project No. 101069576

Deliverable 7.2

Relevant metrics to assert the validity of the simulation-assisted test methods

WP 7 – X-in-the-Loop test environment

Authors	Sébastien Liandrat (CE), Maytheewat Aramrattana (VTI), Abhijeet Behera (VTI)
Lead participant	Sébastien Liandrat (CE)
Delivery date	31 August 2024
Dissemination level	Public
Type	Report

Version 1.0

Revision history

Co-funded by the European Union

Innovate UK

Co-funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or European Climate, Infrastructure and Environment Executive Agency (CINEA). Neither the European Union nor the granting authority can be held responsible for them. Project grant no. 101069576.

UK participants in this project are co-funded by Innovate UK under contract no.10045139.
Swiss participants in this project are co-funded by the Swiss State Secretariat for Education, Research and Innovation (SERI) under contract no. 22.00123.

Author(s)	Description	Date
Sébastien Liandrat (CE) Maytheewat Aramrattana (VTI) Abhijeet Behera (VTI)	Draft deliverable	12/08/2024
Anders Thorsén (RISE)	Revision 1	19/08/2024
Maikol Funk Drechsler (THI)	Revision 2	22/08/2024
Sébastien Liandrat (CE) Maytheewat Aramrattana (VTI)	Revision 3	27/08/2024
Maytheewat Aramrattana (VTI) Maikol Funk Drechsler (THI)	Final Version	29/08/2024
Alexandra Furio (accelCH), Mario Ceccarelli (accelCH)	Formatting and formal check	31/08/2024

Contents

List of Figures

List of Tables

Partner short names

THI	Technische Hochschule Ingolstadt
VTI	Swedish National Road and Transport Research Institute
CE	Centre d'études et d'expertise sur les risques, l'environnement, la mobilité et l'aménagement
RISE	RISE Research Institutes of Sweden AB
accelCH	accelopment Schweiz AG

Abbreviations

1D-TTC	One-dimensional Time-to-collision
2D-TTC	Two-dimensional Time-to-collision
CLP	Cumulative Lateral Position
D	Deliverable
DCE	Distance of Closest Encounter
DMM	Dynamic Motion Model
DRAC	Deceleration Rate to Avoid Crash
EP	Environment Perception
FN	False Negative
FP	False Positive
FPPI	False Positive Per Image
HEU	Horizon Europe
HiL	Hardware-in-the-Loop
IoU	Intersection over Union
MOTA	Multiple Object Tracking Accuracy
MOTP	Multiple Object Tracking Precision
MP	Motion Planning
MTTC	Modified Time-to-Collision
PET	Post-Encroachment Time
PG	Proving Ground
RSS	Responsibility Sensitive Safety
SiL	Software-in-the-Loop
SSM	Surrogate Safety Measures
T	Task
TET	Time Exposed Time-to-collision
TG	Time gap

TIT	Time Integrated Time-to-collision
TN	True Negative
TP	True Positive
TTC	Time-to-collision
TTCE	Time-to-Closest Encounter
UC	Use Case
ViL	Vehicle-in-the-Loop
VRU	Vulnerable Road User
WP	Work Package
XiL	X-in-the-Loop

Executive summary

The primary goal of the ROADVIEW project is to enhance the performance of perception and decision-making subsystems for connected automated vehicles (CAVs) under challenging weather conditions such as rain, fog, and snow. Achieving this is crucial for the widespread adoption of CAVs. Throughout the project, various subsystems have already been improved and evaluated across different work packages (WP) such as WP3, WP5, and WP6.

This deliverable (D) aims to present a methodology for assessing the overall system performance when subjected to the test cases defined in D7.1. Each test case will be executed in multiple environments, including proving ground (PG) as well as X-in-the-Loop (XiL) simulation environments such as Software-in-the-Loop (SiL), Vehicle-in-the-Loop (ViL), and Hardware-in-the-Loop (HiL). The system's performance will be evaluated in each of these environments.

Additionally, this deliverable seeks to explore approaches for assessing the discrepancies in system performance across these different test environments.

Objectives

The goal of this deliverable is to define and develop a methodology for evaluating the performance of the system-under-test. This involves comparing the system's performance in virtual and real-world testing. Specifically, the task is to accurately describe the metrics and their application across the various test cases outlined in D7.1.

It also provides guidelines for interpreting the results of the test cases, suggesting thresholds for evaluating the performance of the system-under-test during the test cases and identifies areas for improvement.

Methodology and implementation

The methodology involves reviewing current research literature and selecting basics metrics to be applied in the ROADVIEW context throughout the various defined test cases and use cases. These metrics are then aggregated at the test case level using several methods adapted to the test cases that will be implemented in the ROADVIEW project.

Finally, a multi-level approach is employed to facilitate comparison between the results obtained from numerical simulations and those from the PG. The aim is to quantify the discrepancies between the simulation and the PG results and to provide insights into whether the system-under-test can be validated.

Outcomes

The results proposed by this task are a list of metrics enabling the system to be evaluated in the test-case configurations defined in D7.1. This deliverable proposes a methodology for selecting thresholds and evaluating the performance of the system under test. Criteria for validating the simulation in relation to the real cases are also proposed.

Next steps

The next steps will be to carry out the tests defined in Task (T) 7.1 and apply the methodology described in this report in order to assess the performance of the test methods and ROADVIEW system-under-test in T7.4 and T7.5 respectively.

1 Introduction

D7.2 presents results from T7.2, which is related to defining metrics for assessing the performance of the ROADVIEW systems with respect to test cases. Defined in D7.1, the test cases were developed on the basis of the use cases defined in WP2 and introduced various parameters that vary from one test case to another, including traffic and weather conditions. In particular, traffic density can vary from a rural area to a dense urban area, while test cases can take place in clear weather, rain, fog or snow.

The environmental perception module of the ROADVIEW system was developed in WP5, while the decision planning and control module was developed in WP6. These modules were evaluated individually in their respective work packages. The primary objective of this task is to define metrics for evaluating the performance of the ROADVIEW system as a whole. The primary objective for this task is to define metrics to evaluate the performance of perception and control modules in the ROADVIEW project. The second objective is to propose a methodology to support the validation of XiL test methods compared to the proving ground (PG).

The performance of the ROADVIEW system will be evaluated on the basis of both controlled test tracks (i.e., on PG) and X-in-the-loop (XiL) tests. XiL tests in the ROADVIEW project comprehend the Software-in-the-Loop (SiL), which includes the interaction of developed software and simulation in a development machine, HiL, which include also the final hardware to be integrated in the vehicle and finally Vehicle-in-the-Loop (ViL), which comprehend the complete vehicle, which drives in an empty road and is stimulated by the virtual environment. XiL tests are vitally important and have become an integral part of the validation process for an autonomous vehicle, as it is not possible to cover the large number of kilometres required for in-depth tests in the real world. In the literature, figures of the order of a billion kilometres are sometimes quoted, which is not feasible in real-life conditions [1] [2].

All the test cases will therefore be carried out both on the proving ground and in the simulations, integrating the various parameters introduced with test case definitions (traffic, weather conditions, etc.) in a similar way. More specifically, the simulation of adverse weather will be based on the models developed in WP3. Figure 1.1 shows how this task interacts with the rest of the ROADVIEW project.

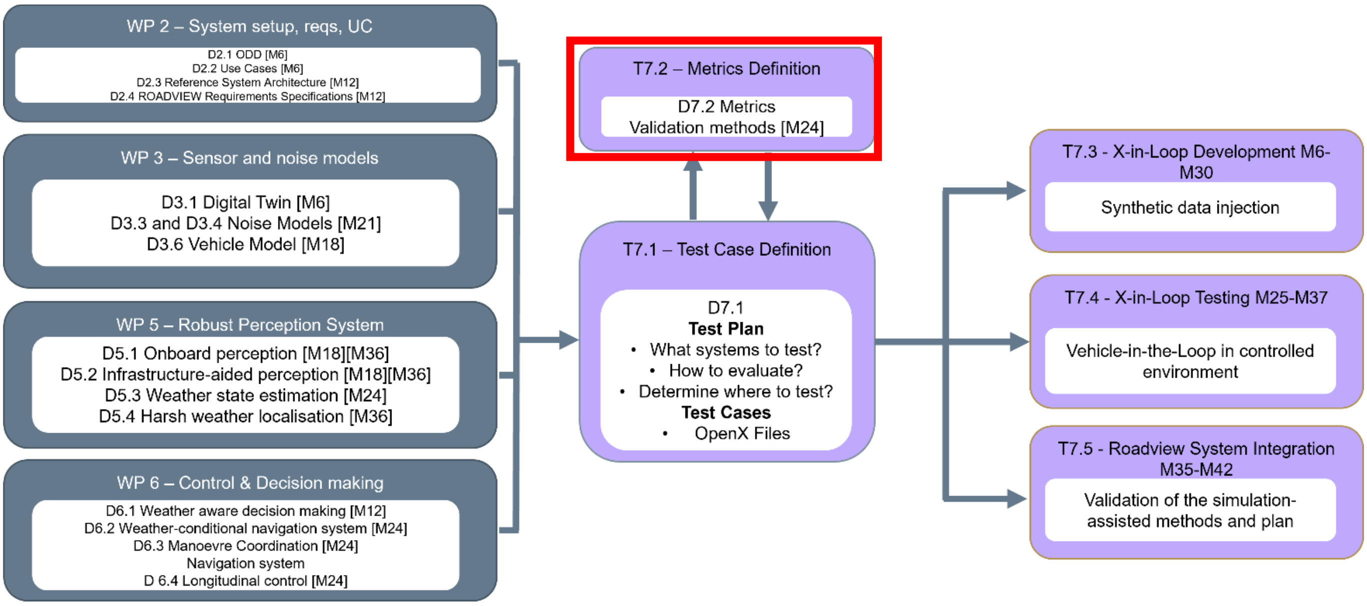


Figure 1.1 Scope of task 7.2 (highlighted in red) with respect to other tasks in ROADVIEW project

The aim is also to compare results between the physical tests on PG and the simulation. Therefore, this report also describes a methodology for the validation of simulation-assisted methods in comparison to PG reference values.

In a scenario-based testing approach considered in the ROADVIEW project, test cases will be executed in several test environments such as simulation and PG. This deliverable suggests metrics that should be used to evaluate environment perception and motion planning systems after each test execution.

This report is outlined as follows. Chapter 2 revisits the definitions of metrics in a general context and sets out the proposed methodology for comparing the results obtained in different test configurations, both real and simulated. The test cases defined in deliverable 7.1 and which will be carried out are briefly described in Chapter 3. The basic metrics, calculated at each time t, used in this task are then described in detail in Chapter 4. We then describe the specific metrics used to evaluate the performance of the perception system (Chapter 5) and the movement planning system (Chapter 6). These test case metrics are obtained by aggregating the basic metrics for each defined test case and/or use case. The 4-level methodology used to compare simulation results with real-world data is discussed in Chapter 7. Example application of the suggested methodology is then presented in Chapter 8. Lastly, Chapter 9 concludes our findings.

2 Background and definitions

This section recalls some general definitions frequently found in the literature, and where necessary, these are also specified in the context of this task. In particular, the notions of metrics, aggregation, perception of the environment and motion planning are presented.

Secondly, the context of this task and the means and methodologies proposed to achieve its objectives are presented.

2.1 General definitions

As they are often used in the research literature, the following concepts are defined and presented below: metrics, aggregation, environment perception metrics and motion planning metrics. These definitions will be used in the context of this deliverable.

More importantly, the notions of basic metrics and test case metrics, defined in the context of this deliverable, are introduced.

2.1.1 Metric

In general, a metric is a mathematical function that returns a value for a given scenario S and time t. This value is typically a real number but can also be infinite under certain boundary conditions [3]:

$$\begin{cases} S \times \mathbb{R}^{+} \rightarrow \mathcal{O} \subseteq \mathbb{R} \cup \{-\infty, +\infty\} \\ (S, t) \rightarrow \mathrm{metric}(S, t) \end{cases} \quad (1)$$

The principle of a metric, as the name suggests, is to measure something—be it a phenomenon, behaviour, or any interaction involving one or more actors within the given scenario. This measurement is context-specific, which is why the time variable t is an essential input. Metrics enable the translation of a complex space, encompassing both context and time, into a single value. This simplification facilitates easier comparisons between values, making it significantly more manageable to evaluate and analyse the scenario.

Metrics usually correspond to physical quantities such as time, distance, or energy, or they may represent rates of occurrence, success, or failure.

A metric can be associated with a specific parameter such as a threshold, however, a metric is considered as better if it is not linked to anything. This threshold can be adapted to the scenario: for example, according to adverse weather conditions, a breaking parameter can be adjusted if it’s raining and slippery.

In the context of task 7.2 of the ROADVIEW project, the metrics defined for any time t are called "basic metrics".

2.1.2 Aggregation

The definition of a metric in the context of evaluating autonomous vehicle performance is not singular. Given the importance of assessing behaviour over the entirety of a scenario, many metrics incorporate an aggregation method. This method consolidates various data points into a single result for a given scenario.

Aggregation allows for a comprehensive evaluation, summarising complex interactions and behaviours into a unique value that can be readily interpreted and compared across different scenarios. For instance, it might be required to analyse the evolution of a metric over a specific period or during a particular event, in which case we talk about aggregation over time [3].

It may also be a question of evaluating a global behaviour relative to several actors, for example, the detection of a group of objects and people in the vehicle's environment, in which case we talk about aggregation over actors.

It should be noted that in the literature, certain metrics are predefined for an entire scenario and, in a way, intrinsically incorporate the aggregation stage described above. They are not time-dependent and therefore do not really correspond to the definition given above, however, it is possible to adapt the definition considering the start of the scenario t_0. Then the metric would be:

$$Metric(S,t) = \int_{t_0}^{t} sub_{metric}(S,t)\, dt \qquad (2)$$

For example, aggregations are generally mathematical operations: minimum, maximum, average, median, sum, p-quantile, integration over time, etc.

In the context of task 7.2 of the ROADVIEW project, the metrics aggregated at the level of a test-case, which corresponds to a specific scenario, are called test-case metrics.

2.1.3 Environmental perception and motion planning metrics

Existing metrics generally fall into two categories: environment perception and motion planning [4]. Environment perception involves assessing the state of surrounding (dynamic) obstacles, including their position, velocity, acceleration, and classification/type. Object detection metrics deal with the direct processing of data received by sensors. Cameras are typically employed for detecting and tracking objects (obstacles) in Advanced Driver Assistance Systems. Additionally, data from other sensors, such as point cloud data from LiDARs, can also be utilised for object detection and tracking.

The general principle of detecting an object from a sensor is to detect an area where the sensor measurement is particularly contrasty. For a camera, this means the presence of a characteristic shape in the image; for lidar, it will also be a shape, but in 3 dimensions. When we talk about contrast, it can be a change in colour identified between the environment and the object, a change in reflectance, or even a change in the dynamics of the scene (it can be easier to identify a moving object). Algorithms can then be used to match this shape to an object. final information. The quality of the detection is then generally assessed according to the accuracy of the perception system compared to the ground truth: is the object detected in the right place, is it detected when it should be detected, are false non-existent objects sometimes detected? Finally, the notion of tracking is essentially the notion of detection aggregated over time: if an object is detected constantly throughout its journey, then it is correctly tracked. Tracking becomes more difficult when there are multiple objects and it is necessary to maintain the identity of each object even when they cross paths

In contrast, motion planning metrics involve more complex decision-making processes [5]. Motion planning involves determining the future states of a vehicle at the trajectory level and planning its manoeuvres. In the context of ROADVIEW, motion planning metrics are used to assess the safety (and sometimes efficiency) performance of the motion planning module. Behaviours of the motion planning system—such as acceleration, position, speed, etc.—are taken into consideration when calculating the motion planning metrics.

Additionally, there are metrics related to driving comfort, which aim to minimise instances of excessively high lateral or longitudinal acceleration (such as evasive manoeuvres performed too late). However, it should be noted that these comfort-related metrics are not examined in the ROADVIEW project.

2.2 Compare the performance of the PG and the simulation

One of the aims of this deliverable is to propose a methodology for comparing simulation performance with that of real tests. The notion of simulation to reality gap is introduced as well as a methodology to quantify it.

2.2.1 Challenges in assessing the simulation to reality gap

The purpose of a simulation is to reproduce a phenomenon or situation observed in the real world. Then, the ideal case of a perfect simulation would correspond to the situation where all the physical quantities of the actors in a virtual scenario are strictly identical to those of the real scenario it is trying to reproduce. However, a perfect simulation is unfeasible due to technological and knowledge limitations, in this way, the simulation always represents a simplification of the real phenomena.

As a reminder, the challenge of simulation, already presented in the introduction, is to be able to test the behaviour of an automated vehicle in a large number of situations and over a large number of kilometres. The aim of these tests is to confirm that the vehicle-under-test behaves in a way that is adapted to the real world and is acceptable from a safety point of view. Meeting this challenge does not necessarily mean that the simulation must be perfectly identical to the real entity; it is sufficient that it achieves a performance identical to reality in terms of the criteria used to assess the vehicle's behaviour.

In the context of the ROADVIEW project, the evaluation of the system's performance is based on the performance obtained in the test cases, whether for real or simulated cases, but also on the simulation's ability to reproduce the behaviour obtained in reality.

This is what is known as the simulation-to-reality gap. As mentioned above, the measurement of this gap necessarily relies at least indirectly on the quantification of the difference between the various physical quantities that characterise a real situation and the simulated situation, namely the position of the objects, their speed and their acceleration.

In practice, the evaluation of this gap can also be based on the comparison of the performances obtained on the classic metrics used on automated vehicles.

T7.2 of the ROADVIEW project consists precisely in defining a methodology that will make it possible to compare the system's performance in real-life situations and numerical simulation. The objective is therefore not simply to calculate metrics to evaluate the behaviour of autonomous vehicles. The aim is to compare the results obtained with these metrics between the real situation and the simulation.

2.2.2 Methodology for evaluating the simulation to reality gap

The various steps are briefly described below, without specifying the metrics used. These are described in detail in the following sections.

- Step 1: selection of several basic metrics, calculated for each time t, related to object detection and motion planning. These metrics must be computable both in the real world and in simulations (HiL, SiL and ViL).
- Step 2: Several second-level metrics, known as test case metrics, are defined. These metrics are based on the basic metrics defined and calculated in step 1. Several test case metrics can be based on the same basic metric. Some test case metrics may not be applicable to all test cases, or at least their calculation may not be relevant. If metrics have additional input parameters, these may vary from test case to test case. Then, using a different process including at least a time aggregation method, each test case metric is calculated for all possible test cases. At this stage, there is a result for each test case metric and for each test case, whether for PG or simulation (except for the irrelevant metrics/test cases mentioned above).

- Step 3: The simulation-to-reality gap is assessed by comparing the results obtained by the system on real and simulated cases. This comparison is divided into 4 levels, from the most basic to the most advanced. The most basic compares only the success or failure of the test-case, while the most advanced compares directly the curves of the main physical quantities.

The purpose of these 4 levels is to compare real tests on PG with tests in simulation and to help identify the differences. In addition, an order of application is proposed, starting with object detection and moving on to motion planning. This order remains a suggestion and needs to be adjusted according to the test cases.

First stage: object detection

- Level 0: Success criteria for object detection in test cases
- Level 1: Direct comparison of test-case metric results
- Level 2: Comparison of an overall score calculated on the basis of the various test-case metrics
- Level 3: Comparison of detected object trajectories between simulation and real case

Note that we might encounter a situation where the test case is a failure (for example, an object is detected incorrectly), but the simulation is validated regarding level 0 because it also failed to detect the object. The same reasoning applies to the test-case metrics: the result obtained by the vehicle on the metrics assessing the quality of object detection may be poor, but if it is identical between the simulation and reality, this means that the simulation is performing well in its ability to reproduce reality.

The succession of these levels of metrics makes it possible to identify the point at which the simulation deviates from reality.

Level 3 is considered the ultimate level, since it implies that the main physical quantities of the detected object are identical between simulation and reality.

This is also an important starting point for the second stage of motion planning. For the simulation and reality to react in the same way, the environment must itself be detected in the same way.

Second step: Motion planning

- Level 0: Success criteria related to motion planning in test cases
- Level 1: Direct comparison of test-case metric results
- Level 2: Comparison of an overall score calculated on the basis of the various test-case metrics
- Level 3: Comparison of ego vehicle position and velocity curves between simulation and real case

2.2.3 Global scheme of the system assessment methodology

The Figure 2.1 illustrates the methodology proposed in this deliverable for validating test cases and evaluating the simulation to reality gap. The upper section of the diagram represents the scenarios tested, referred to as "test cases" within the context of ROADVIEW, which are derived from the use cases defined in Deliverable D2.2. In summary, these test cases vary different parameters such as traffic and weather conditions within the use cases. The test cases must be conducted both in real-world settings and XiL simulation environments.

The middle section of the diagram focuses on the calculation of metrics. This process begins with synchronising all the results, followed by calculating the basic metrics, and finally computing the test case metrics. Various parameters influence the calculation of these metrics. The test cases are validated based on the results obtained from these test case metrics.

Finally, the lower section of the diagram presents the evaluation of the simulation to reality gap, which is assessed through a four-level framework. This ranges from Level 0, the simplest, to Level 3, where the simulation most closely aligns with reality.

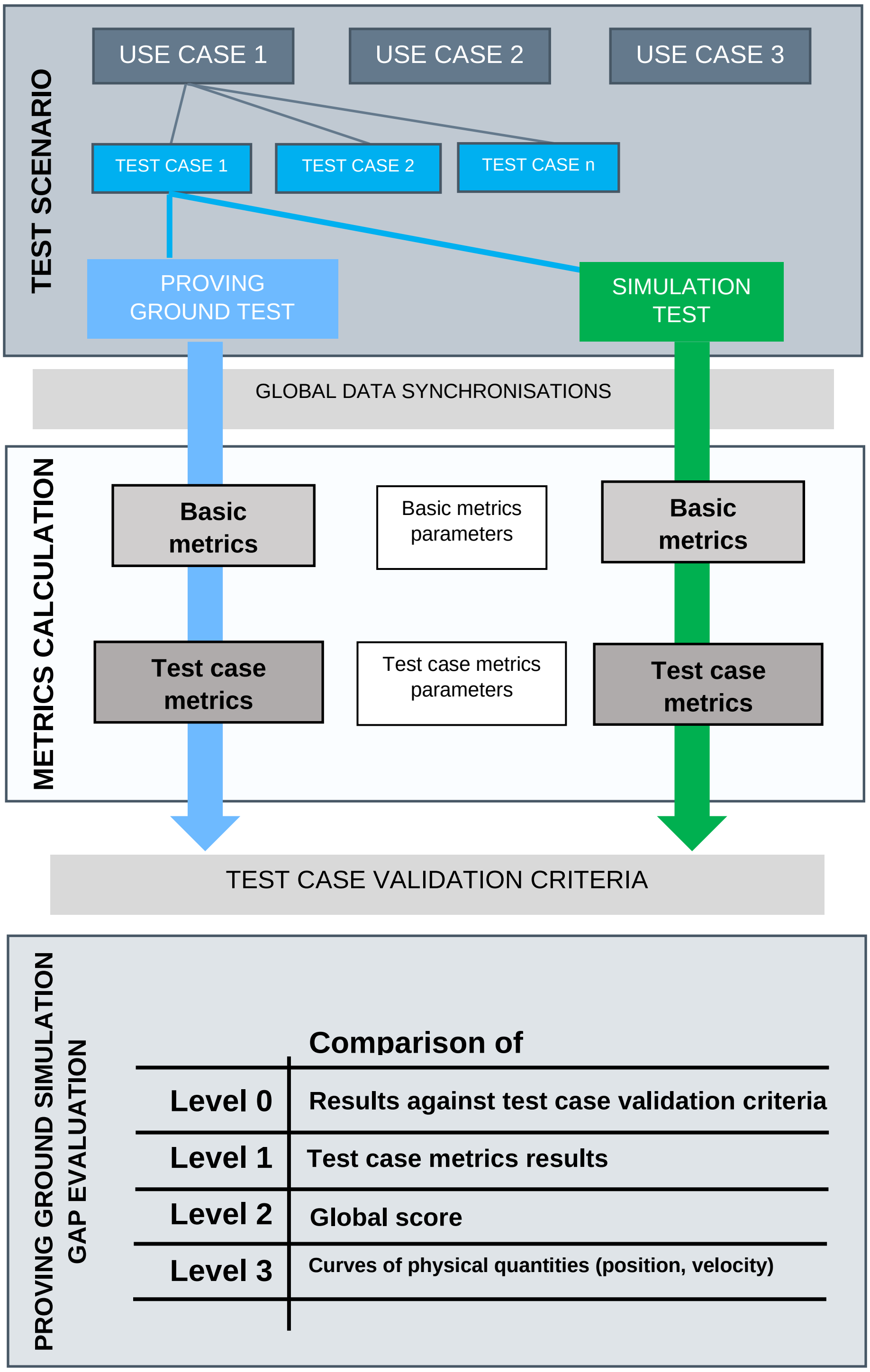


Figure 2.1 Summary diagram of the methodology proposed in D7.2

3 Test-cases description

Test cases in ROADVIEW are derived from the set of five use cases (UC) presented in D2.2 according to the test plan proposed in D7.1. This section intends to provide a brief overview on the test cases and use cases as a background for the readers.

3.1 Use-cases

The ROADVIEW use cases are summarised in Table 1 and Figure 3.1. Based on these use cases, test cases will be derived based on the approach presented in D7.1. Different parameters will be considered such as a variety of weather conditions, including fog, rain, and snow, as well as different road surface states such as wet and icy. This approach ensures comprehensive evaluation of vehicle performance and safety across diverse environmental scenarios.

Additionally, the test cases will be executed in different traffic environments, encompassing rural, urban, and intercity settings. This will allow for the assessment of the vehicle's behaviour and adaptability in varying traffic densities and patterns, providing a thorough understanding of its performance in real-world driving conditions.

Table 1 List of use cases defined in D2.2.

Use case ID	Use case name	Short description
UC1	Driving straight	Ego vehicle travels on the ego lane with or without a leading road user
UC2	Coming to a full stop	Ego vehicle approaches an obstacle and stops in the ego lane
UC3	Interacting with vulnerable road users (VRU)	Ego vehicle approaches VRU (which may be crossing in front of it)
UC4	Entering a lane	Ego vehicle enters a lane (including from a road shoulder and on-ramp)
UC5	Exiting a lane	Ego vehicle exits a lane (including to a road shoulder and off-ramp)

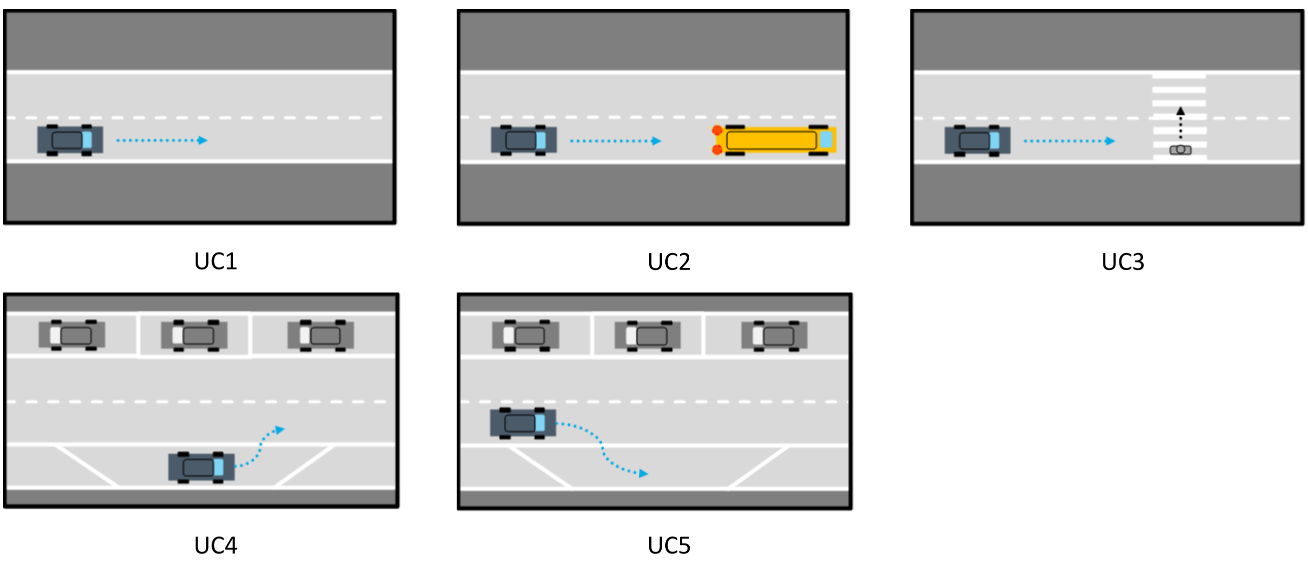


Figure 3.1 Illustration of the five use-cases defined in D2.1

Metrics for assessing safety of motion planning are typically dependent on the type of manoeuvre(s) that are being executed under test case. Generally, there is a clear distinction made in the literature between longitudinal and lateral manoeuvres. Furthermore, car-following and intersection situations are typically assessed differently. The use cases are then categorised as presented in Table 2 based on their respective manoeuvre and situation(s).

Table 2 List of manoeuvres and situations in different ROADVIEW use cases

Use case	Ego vehicle manoeuvres	Situation
UC1	Longitudinal	Car-following & lane-following
UC2	Longitudinal	Car-following
UC3	Longitudinal	Intersection
UC4	Longitudinal and lateral	Lane-changing
UC5		

UC1 and UC2 contain a car-following situation, which refers to a situation as depicted in Figure 4.2. If there is no leading vehicle, UC1 could also include the lane-following manoeuvre. In this context, the ego vehicle (the vehicle-under-test) will be regarded as a following vehicle and the goal is to avoid a collision with a leading vehicle (or other road users on its trajectory). When considering a conflict between a vehicle and a VRU (i.e., UC3), the conflict can be split into three different phases according to [8]: conflict phase, passing phase, and stopping phase. These phases represent different categories of test cases for UC3, and thus we will use different metrics to assess tests in different phases. Lastly, UC4 and UC5 can be categorised as lane-changing situations, where the ego vehicle performs both longitudinal and lateral manoeuvres. Similar to the car-following case, we only consider cases when the ego vehicle is the following vehicle.

3.2 Adverse weather and traffic conditions

For the various test-cases, a number of parameters vary, including weather conditions, traffic density, speed and the position of the various participants (including the ego vehicle).

The weather can vary from clear, rainy, foggy and snowy. Each modality can be represented with varying intensity, to a greater or lesser extent. It is important to note that weather conditions can affect both perception of the environment (reduced visibility) and motion planning (changes in road grip when wet or icy). In particular, visibility conditions can have an impact on the moment (or rather the distance) at which an object should be detected, and therefore on the classification of a non-detected object as "False Negative" or "True negative".

Traffic can be light or heavy, and will also have an impact on the metrics, with a variable number of detectable objects, but also a variable number of objects to take into account when driving.

Other parameters vary depending on the test case. For example, for UC3, the VRU may be waiting on the side of the road or crossing.

4 Basic metrics

As a reminder, in the context of task 7.2 of the ROADVIEW project, the metrics calculated for any time t are called basic metrics (more details are given in 2.1.1).

Basic metrics are metrics that are calculated at each time step (t). These are the lowest level metrics that are later aggregated to evaluate gap between PG and simulation according to the methodology described in Chapter 7. Some of these basic metrics can be used to evaluate the performance of the system-under-test as described in Chapter 5 and 6.

4.1 Environment perception metrics

For a vehicle, perceiving its environment means knowing the state of the objects around it. This includes position, speed, acceleration and type of object. In practice, if it is capable of knowing the position of an object as a function of time, it can deduce its speed and acceleration. However, it should be borne in mind that, as these measurements are indirect, their uncertainty depends very much on the time step between two successive positions.

Cameras are the most commonly used sensors for object detection, but they are very often combined with other sensors, and the object is detected and tracked by merging the data.

The method used almost systematically is the Intersection over Union (IoU) method [6].

4.1.1 Intersection over Union

The principle of Intersection over Union (IoU) is to compare two boxes, one ideally surrounding the detected object and the other corresponding to the detection estimated by the sensors combined with the detection algorithms [4]. This box can be either 2-dimensional, in the case of image-based detection, in which case two rectangles are compared. It can also be 3-dimensional, in which case two rectangular parallelepipeds are compared.

As part of the ROADVIEW project, the 3D intersection over the union was used. In practice, each box is given by the position of its centre, its 3 dimensions (length, width, height) and its orientation around the vertical axis. Its coordinates are given in the frame of reference of the sensor (or sensors) used for detection, so it is necessary to align the position of the detected object in a global frame of reference and to take into account the position of the sensor in this frame of reference. As the sensor is positioned in the vehicle, it is generally mobile in the global reference frame. The methodology for this recalibration is described in Chapter 8.1.

The intersection of the two 3D boxes, if it exists, is itself a volume. The IoU, also known as the Jaccard index, is a widely used measure for evaluating object detection. It consists of calculating the ratio between the volume of the intersection of the two boxes and the volume of the union of the two boxes [7] [8].

In the case where the 2 boxes are completely disjoint, the result is 0, in the case where the 2 boxes are completely superimposed, the result is 1.

$$IoU\ (t) = \frac{|D \cap G|\ (t)}{|D \cup G|\ (t)} \tag{3}$$

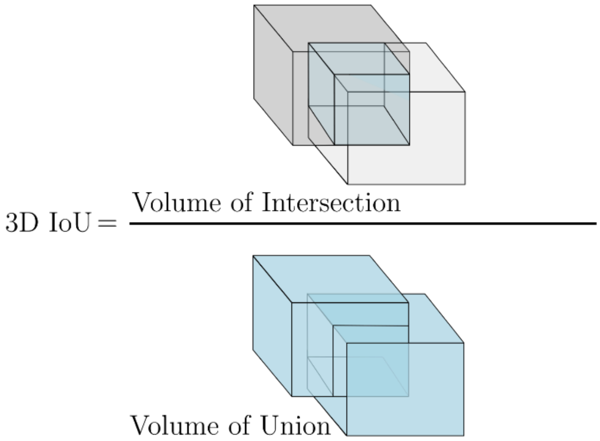


Figure 4.1 Diagram showing the principle of Intersection over Union in the 3-dimensional case (adapted from [8])

4.1.2 True/False Positive/Negative situations

Calculating the IoU is the first step, after which it is necessary to check whether or not this value is above a predefined threshold. When the IoU exceeds this value, the object is considered detected, otherwise it is not.

Usually, this threshold is given as between 0.35 and 0.5 (note that the order of magnitude of the threshold is very much affected by the choice of operating in 2D or 3D). They can be also adjusted according to the object class to be detected.

Finally, it is necessary to determine whether the detection is a true positive, a false positive, a true negative or a false negative. In other words, is an object detected when there really is an object and is nothing detected when there is nothing to detect. To be precise, there are exactly four possibilities:

- TP = True Positive: when the algorithm detects the object correctly (IoU > threshold)
- TN = True Negative: no objects detected and no objects to be detected
- FP = False Positive: when the algorithm detects a non-existent object (or detects an existing object at the wrong place)
- FN = False Negative: when the algorithm misses the detection of an existing object

In order to determine when an object should be detected or not, it may be necessary to determine whether an object should be visible or not. This parameter can be either a distance from which the object should be detected, or an instant in the current scenario (for example, if a VRU suddenly appears). A brief analysis of the impact of this parameter is given in section 4.4.1.

In the usual case, human perception is sometimes used as a reference, but depending on the sensor and the situation this may not make sense.

In the context of the ROADVIEW project, this parameter is particularly important, as the various test cases can vary according to weather conditions. These affect visibility, and therefore object detection (an object in fog is generally perceived at a closer distance). The assumed performance of sensors in adverse weather conditions should be taken into account when defining this parameter for the various test cases.

4.2 Motion planning metrics

Several safety metrics can be applied to evaluate the safety of connected and automated vehicles. One approach is to consider surrogate safety measures (SSM) as reviewed in [9] [10] [11]. According to [10], SSM can be categorised into three main categories: i) time-based; ii) energy-based; and iii) deceleration-based. The time-based SSM indicates the risk of a collision in terms of time proximity, while the energy-based SSM consider the severity of an interaction. Lastly, the deceleration-based approaches measure how much deceleration is required to avoid a potential collision.

Despite being threshold-sensitive, the time-based approach is mainly considered in this deliverable due their feasible applicability to simulation and PG testing. The energy-based and deceleration-based approaches often require precise knowledge (or at least a strong assumption) about the vehicle under test and other traffic participants in terms of acceleration/deceleration capabilities. This information could be difficult to obtain in real world test scenarios such as tests on PG.

4.2.1 Distance

Distance between a following vehicle and a leading vehicle ($D_{f,l}$ in Figure 4.2) is one of the most basic metrics that can be used to evaluate safety of a vehicle. For instance, following the Responsibility-Sensitive Safety (RSS) model, a minimum safe longitudinal distance when following another vehicle can be calculated [12]. This can then be used to evaluate safety of car-following situation.

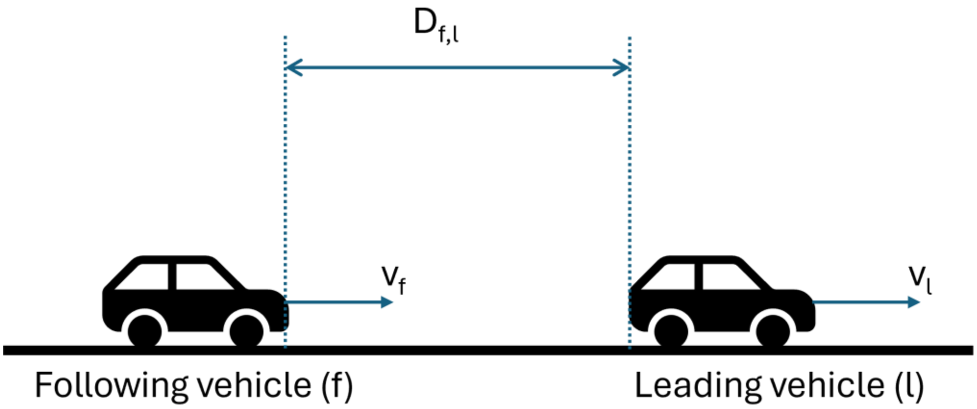


Figure 4.2 Car-following situation where a following vehicle follows a leading vehicle

4.2.2 Time Gap

Given an information about velocity of the following vehicle, as an addition to a distance, a time gap (TG) between a following vehicle (f) and a leading vehicle (l) can be calculated as:

$$TG_{f,l}(t) = \frac{D_{f,l}(t)}{V_f(t)} \tag{4}$$

Where,

$D_{f,l}$ denotes longitudinal distance in metres between the following vehicle and the leading vehicle

V_f denotes velocity of the following vehicle in m/s

In other words, time gap is a temporal distance between the following vehicle and its leader. This can be a good metric from both safety and efficiency perspectives. For instance, maintaining a stable constant time-gap is one of commonly used speed control policy for autonomous vehicles in car-following situation, especially in platooning.

4.2.3 Time To Collision

Time-to-collision (TTC), introduced by Hayward in 1972 [13], is one of the most commonly and frequently used SSM [9]. Given a car-following situation, where two vehicles are following each other, TTC is defined as the time remaining until collision if both vehicles maintain their current speed and heading. TTC is calculated at each moment in time (t) and for each follower-leader pair. Consider a car-following situation depicted in Figure 4.2, TTC between the following vehicle (f) and the leading vehicle (l) can be described mathematically as:

$$TTC_{f,l}(t) = \frac{D_{f,l}(t)}{V_f(t) - V_l(t)} \;\; \forall\, V_f(t) > V_l(t) \tag{5}$$

Where,

$D_{f,l}$ denotes longitudinal distance in metres between the following vehicle and the leading vehicle

V_f denotes velocity of the following vehicle in m/s

V_l denotes velocity of the leading vehicle in m/s

Furthermore, it should be noted that TTC only applies to the case where the following vehicle has higher velocity than its leader, otherwise TTC is considered to be infinite (i.e., there is no risk of collision). In such case, distance or TG can potentially be used instead of TTC to assess risk in car-following situations.

Researchers have suggested extensions and improvements to TTC over the years. For instance, Modified TTC (MTTC) [14], which takes acceleration of both the following vehicle and the leading vehicle into account instead of a constant acceleration that TTC assumes. However, MTTC requires both accelerations to be known, which is not always the case in real-world or PG test. Therefore, we do not consider MTTC as a suitable safety metric for the ROADVIEW project. Nevertheless, some extensions of TTC will be used in this project with respect to each test case, as described further in Chapter 6.

In the literature, we also find the notion of Time To Closest Encounter (TTCE) [15], which is very close to TTC since it is a generalisation of the latter for any chosen Distance of Closest Encounter (DCE). If DCE is equals to zero (i.e., there is a collision), then TTCE is equal to TTC.

In some cases, it may be useful to know the time before two actors will find themselves at a given distance if they both maintain their Dynamic Motion Model (DMM). DMM refers to the dynamic behaviour assumed for an object. The most common assumption is that the object will maintain a constant velocity. A more advanced version extends this assumption and considers that the acceleration will be constant. However, it is possible to imagine more complex DMMs that incorporate changes in the acceleration and trajectory of an object.

4.2.4 Deceleration Rate to Avoid Crash

The Deceleration Rate to Avoid a Crash (DRAC) corresponds to the minimum (or maximum) acceleration necessary to avoid a collision between two actors. In general, it corresponds to the minimum acceleration (which will be negative then) for a following vehicle V_f to avoid colliding with a leading vehicle V_l.It is necessary to assume a DMM for both vehicles, usually the metric is then calculated assuming the leading vehicle will have constant velocity. Using the notation in Figure 4.2, DRAC of the following vehicle can be calculated as:

$$DRAC_f(t) = \frac{(V_f(t) - V_l(t)^2}{2D_{f,l}(t)} \tag{6}$$

4.3 Variables needed for basic metrics calculation

To calculate the basic metrics at each time t, several system parameters and quantities must be known at each time. These parameters have to be known for PG and XiL simulation. The time step between successive points should be as small as possible, in ROADVIEW it was defined as a requirement that the algorithms and sensors should deliver data at a minimum frequency of 10 Hz. In ROADVIEW, it was defined as a requirement that the algorithms and sensors should deliver data at 10 Hz. This requirement will also be used in the metrics evaluation.

The sections 4.3.1 and 4.3.2 present a list of the essential variables that need to be known for any given time t in order to calculate the basic metrics defined above. Section 4.3.3 presents the variables that are also often necessary for calculating common metrics relating to the evaluation of the performance of automated vehicles but are not strictly speaking essential for the work of this task.

The ego vehicle is the vehicle whose performance we intend to evaluate, whether in terms environment perception or motion planning. All the other actors that are important in the sense that they should be minimally detected are labelled 'object'.

4.3.1 Variables needed for environment perception

The list of essential parameters to calculate environment perception basic metrics is given in the Table 3.

Table 3 Time-dependent variables needed for environmental perception.

Time-dependant variables	
Notation	Meaning
$\boldsymbol{p_{ego}}$	Position of ego vehicle (3 dimensions, x, y and z)
$\boldsymbol{v_{ego}}$	Velocity of ego vehicle (3 dimensions, x, y and z)
$\boldsymbol{a_{ego}}$	Acceleration of ego vehicle (3 dimensions, x, y and z)
$\boldsymbol{\psi_{ego}}$	Yaw angle of ego vehicle (orientation angle)
$\boldsymbol{\varphi_{ego}}$	Steering angle of ego vehicle
$\boldsymbol{p_{object}}$	Position of object (3 dimensions, x, y and z)
$\boldsymbol{v_{object}}$	Velocity of object (3 dimensions, x, y and z)
$\boldsymbol{a_{object}}$	Acceleration of object (3 dimensions, x, y and z)
$\boldsymbol{Orientation_{object}}$	Roll, pitch, yaw of the object

The position and orientation of the sensors should also be known in order to enable the space synchronisation between the different sensors and the ego vehicle. Furthermore the coordinate system used by each need to be considered, since for example positioning system use the SAE system with Z pointing up and cameras detection have the Z axle pointing to the depth of the image.

4.3.2 Variables needed for motion planning

The time variables used for the basic metrics related to motion planning are the same as those described for environment perception. Although we are interested in the dynamics of the ego vehicle, we generally evaluate its behaviour in relation to that of the other actors (for example a lead vehicle or a Vulnerable Road User (VRU)).

To assess the position of the ego vehicle relative to the road (cumulative lateral position basic metric), however, it is necessary to know the exact position of the road, in particular the middle of the lane.

4.3.3 Other variables often needed for metrics calculation

Other variables are often used to calculate metrics. Although these are not necessary for the metrics considered in this report, it is still worth knowing them in order to consider the subsequent calculation of other metrics. These variables are listed in Table 4.

Table 4 Variables often needed for metrics calculation.

Notation	Meaning
m_i	Mass of actor i
$a_{i,min}$	min(t) Minimal available acceleration of actor i
$a_{i,max}$	max(t) Maximal available acceleration of actor i
j_i	jerk of actor i
u_i	Control inputs of actor i
Θ_i	Steering rate of actor i
ω_i	Yaw rate of actor i
F_{idxy}	Tire forces of actor i with direction d for tire (x, y)
$c_{i\alpha f}$	Front tire cornering stiffness of actor i
$c_{i\alpha r}$	Rear tire cornering stiffness of actori
l_{if}	Distance from front axle to center of gravity of actor i
l_{ir}	Distance from rear axle to center of gravity of actor i
L	Distance from front to rear axle
I_{iz}	Moment of inertia of actor i

4.4 Basic metrics application and parameters

This section provides some details to bear in mind when calculating the metrics. Section 4.4.1specifies the impact of a parameter on the application of the detection metric and section 4.4.2 points out the importance of temporal resolution for motion planning metrics.

4.4.1 Distance (or instant) for first detection

The distance at which an object should start to be detected is an important parameter for qualifying the status of detections as 'FN' or 'TN' when the object is not detected.

This distance depends on the sensor but is also strongly influenced by weather conditions, which can vary from scenario to scenario in the context of the ROADVIEW project, it is important to be vigilant when choosing its value.

The two graphs from Figure 4.3 show the curves for the different detection statuses (TP, FP, TN, FN) as a function of time and for two different distances (100m and 130m) at which the object should be detected. To make it easier to analyse visually, the categorical variable, which can take four values (TN, FN, TP, FP), was transformed into a quantitative variable with values of 0.2, 0.4, 0.6 and 0.8 respectively. The wider blue curve identifies when the object should be detected (i.e. when the object is closer than the detection distance parameter).

The detection status as a function of time can be seen to differ from graph to graph, in other words for a given instant the curve can be one colour or another. It should be noted that the detection performance is exactly the same and that the only variation is the choice of the value of the distance at which the object should be detected. This choice nevertheless has an impact on the analysis of detection performance and may have an impact on the metrics.

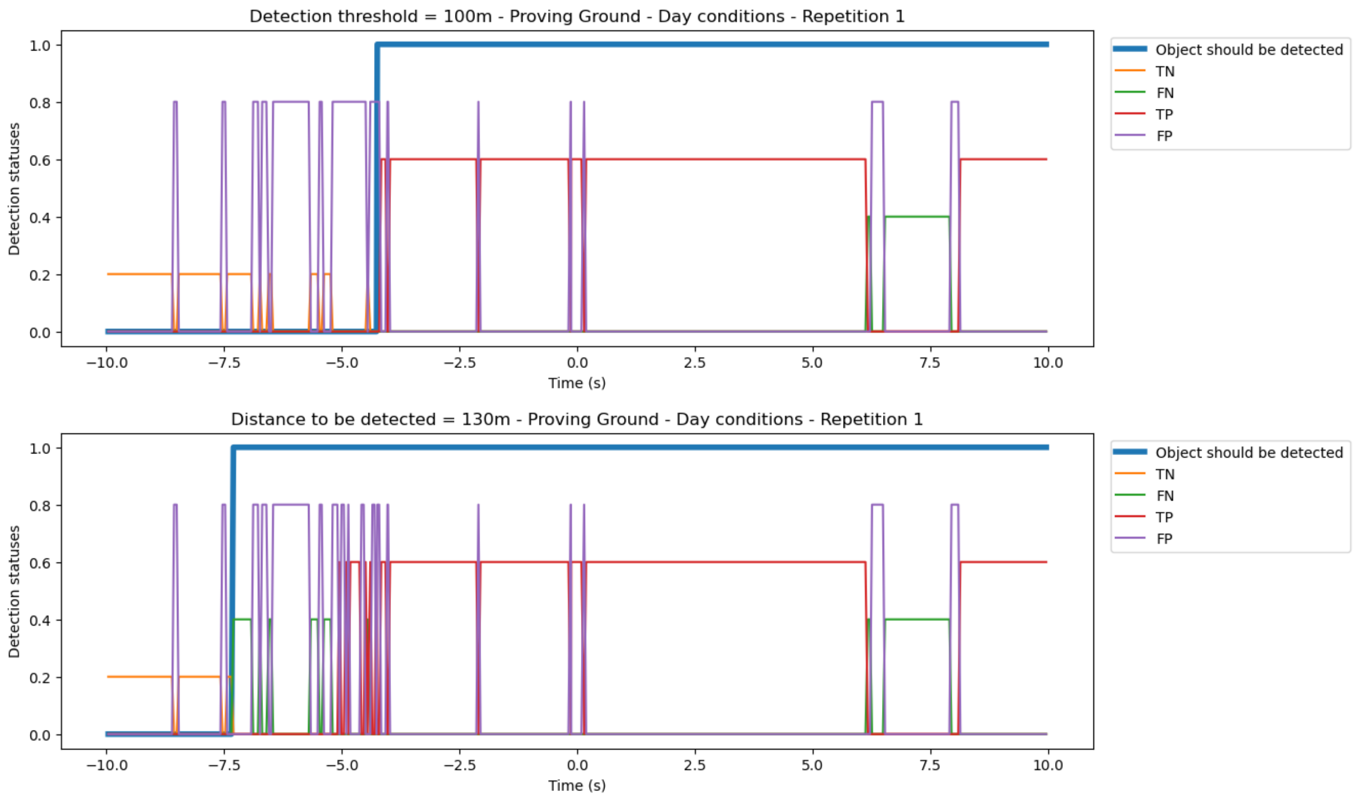


Figure 4.3 Object detection results when the detection threshold is 100 (top graph) or 130 meters (bottom graph)

4.4.2 TTC and DRAC measurement

Metrics such as TTC or DRAC must be calculated from data with the best possible resolution, i.e. with a minimum time step. As an example, in the Figure 4.4 it is possible to evaluate that the velocity curves of the two actors (ego vehicle and target) are of poor quality and very noisy in the simulated environment. This has a direct impact on the results obtained for the metrics, showing the limitations of the simulation in the example.

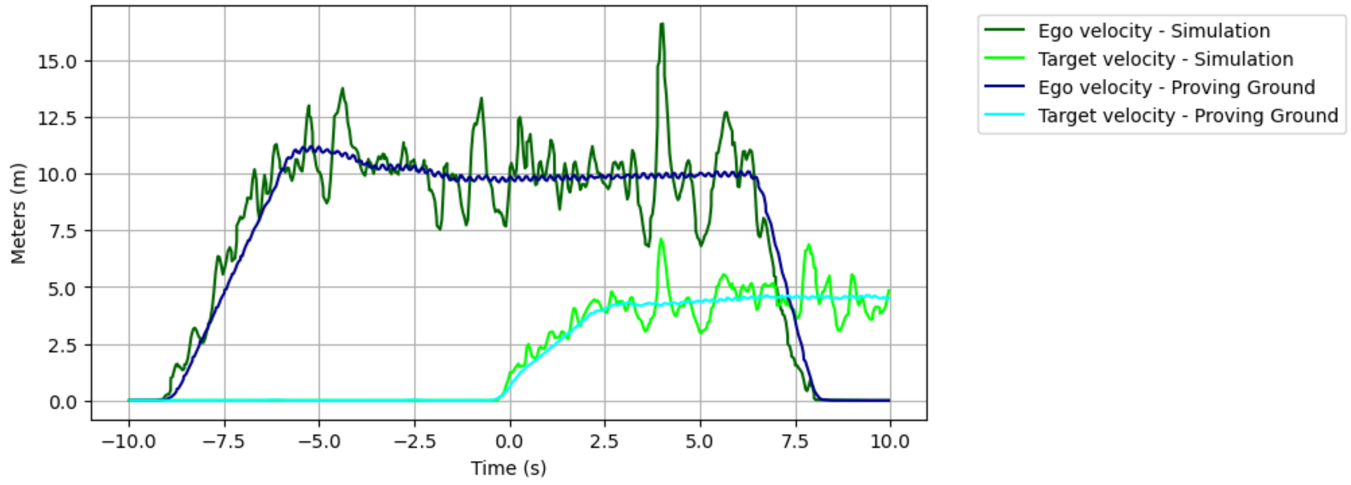


Figure 4.4 Velocity curves for ego vehicle and a given target

5 Test-cases metrics for environment perception

In this part are described the aggregation methods used in order to obtain a result at the test-case for the environment perception metrics.

5.1 Object detection – First detection

This measurement corresponds to the first moment when an object is detected. It is expressed in seconds and can be defined as the moment when the first "True Positive" for the object in question occurs.

This definition could be revised for a case where the first "True Positive" is isolated and followed by numerous "False Negatives" (the object should be detected but no longer is). In this case, we could impose an uninterrupted successive number or period of "True positives".

It should also be noted that this parameter can be linked to the choice of scenario start time t_0, although the start time should have no impact on the performance or otherwise of a detection system.

$$t_{first\ detection} = minimum\{t \geq t_0 | IoU(t) > threshold\} \quad (7)$$

5.2 Object detection – Sensitivity

The sensitivity (s) also called recall (r) is the ratio of true positive instances to the actual number of positive instances. This metric is classic and is suitable when false negative is of high importance [4]:

$$r = \frac{TP}{TP + FN} = \frac{"objects\ seen\ correctly"}{"objects\ it\ should\ have\ seen"} \quad (8)$$

5.3 Object detection – Confidence

The confidence (c) also called precision (p) is described as the ratio of true positive instances to the predicted number of positive instances. This metric is useful when false positive instances are important [4]:

$$p = \frac{TP}{TP + FP} = \frac{"objects\ seen\ correctly"}{"objects\ seen"} \quad (9)$$

5.4 Object detection - F1-Score

F1-score is described as the harmonic mean of sensitivity and confidence defined above [4].

It is obtained as:

$$F_1 - score = \frac{2 \times r \times p}{r + p} \tag{10}$$

To be noted, it exists also Jaccard distance but it's related to F1-score and not used here.

5.5 False Positive Per Image

The number of false positives per image (FPPI) is the average number of false alarms per image. This measurement is useful for identifying a detection threshold that is too low, i.e. the system too easily detects an object that does not exist. It is given by the average of False Positive per frame [16], given by:

$$FPPI = \frac{FP}{nFrames} \tag{11}$$

Where nFrames denotes the total number of video frames

5.6 Multiple Object Tracking Accuracy

The test-case metrics described above indicate performance in terms of detection, but less so in terms of tracking. Other metrics were introduced by the CLEAR (Classification of Events, Activities and Relationships) study, such as the Multiple Object Tracking Accuracy (MOTA) described below and the Multiple Object Tracking Precision (MOTP) described in the next section [[17] [18] [4] [19].

For MOTA, the equation is as follows:

$$MOTA = 1 - \frac{\sum_{\forall t}[FN_t + FP_t + e_t]}{\sum_{\forall t} g_t} \tag{12}$$

Where e is the association error, it happens when 2 detected objects are swapped in the detection algorithm (for example when their paths cross) and g is the actual number of objects that should be detected.

5.7 Multiple Object Tracking Precision

This measure essentially provides information about the accuracy of the location. It is therefore a question of assessing how close the estimated bounding box is to the true bounding box. References about this metric are basically the same as for MOTA and are not given here. The equation is as follows:

$$MOTA = 1 - \frac{\sum_{\forall t}\sum_{\forall t} IoU_{i,t}}{\sum_{\forall t} TP_t} \tag{13}$$

Note that this is one of the few metrics that use the IoU value directly.

5.8 Metrics application and thresholds

This chapter provides guidelines for the application of metrics. Section 5.8.1 specifies the context in which they may be calculated. In 5.8.2, validation criteria are proposed.

5.8.1 Metrics application according to test cases configuration

Table 5 simply reminds us that the MOTA and MOTP metrics are more appropriate when there are several objects to be detected and tracked. It is therefore proposed to calculate them only for scenarios with several objects.

On the other hand, the metric corresponding to the instant of first detection seems more relevant for a situation with a single object. In the context of ROADVIEW, this metric seems particularly relevant for assessing the impact of adverse weather on the first detection distance.

Table 5 Summary of motion planning metrics to apply with respect to situations

Metrics	All Use Cases	
	A single object	**Several objects**
First detection	X	
Sensitivity	X	X
Confidence	X	X
F1-Score	X	X
FPPI	X	X
MOTA		X
MOTP		X

5.8.2 Test-case metrics criteria

Test case validation is based on compliance with the criteria for test case metrics. It is difficult to propose firm and absolute criteria, as they seem to depend so much on the context. Values are proposed on the basis of references [18] [20], but will need to be adjusted according to the configuration of the scenarios. These values seem both achievable and sufficient to ensure acceptable results in terms of detection and tracking.

Table 6 Suggested validation criteria for each object detection test-case metrics

Metrics	Validation criteria
First detection	> 50 metres
Sensitivity	> 0.8
Confidence	> 0.8
F1-Score	> 0.8
FPPI	< 0.08
MOTA	> 0.8
MOTP	> 0.8

6 Test-cases metrics for motion planning

Considering the five use cases defined in D2.2, they involve different manoeuvres and situations as summarised in Table 2. A few of motion planning performance metrics considered in this deliverable can be applied in more than one situation. Therefore, the metrics will first be introduced individually, followed by their application with respect to the ROADVIEW use cases.

6.1 Cumulative Lateral Position

The cumulative lateral position (CLP) metric proposed below was introduced by [21] in 2022. It uses a single metric to assess both the handling of a vehicle (average of its position in relation to the centre of the road) and driving stability (oscillation of the position in relation to the centre of the lane). Usually, the standard deviation of the lateral position and the mean lateral position were looked at instead.

$$CLP\ (m) = \int_0^L Latpos(s) \cdot ds/L \tag{14}$$

where $Latpos(s)$ is the function of lateral position at each position s; s is the curvilinear abscissa; and L is the length of the complete curve for which we are calculating the metric.

Note that this metric is not defined for any instant t, but rather for a linear distance travelled (i.e. between two instants t). It will therefore be calculated directly for the entire test case (after selection of a clear begin and end).

6.2 Minimum distance

In case the ego vehicle stops to avoid a collision with an obstacle (e.g., in UC2), the distance between the ego vehicle and the obstacle is a good indicator of how safe the stopping manoeuvre was. Moreover, the distance at point when the ego vehicle starts braking could be another good indicator for this case.

In the UC1 (i.e., driving straight) context, risky situations are likely to occur during a car-following situation, where a leading vehicle could change its speed abruptly (e.g., due to emergency braking or cut-in). Consider a car-following situation illustrated in Figure 4.2, and let

- r denote a response time by the following vehicle (i.e., the follower);
- $accel_{f,max}$ denote the maximum acceleration of the follower during the response time;
- $decel_{f,min}$ denotes the follower's deceleration during the response time after the acceleration; and

- $decel_{l,max}$ denotes the maximum braking performed by the leading vehicle.

A minimum safe following distance according to the RSS principle [12], can be defined as:

$$d_{min} = \left[v_f r + \frac{1}{2} accel_{f,max} r^2 + \frac{\left(v_f + r \cdot accel_{f,max}\right)^2}{2 \cdot decel_{f,min}} - \frac{v_l^2}{2 \cdot decel_{l,max}} \right]_+ , \quad (15)$$

$$where\ [x]_+ := \max\{x, 0\}$$

In the context of automated driving system, a response time could be estimated in a precise way. If we consider an ideal autonomous vehicle, where the response time could be zero, the equation above can be simplified as follows.

$$d_{min} = \left[\frac{(v_f)^2}{2 \cdot decel_{f,min}} - \frac{v_l^2}{2 \cdot decel_{l,max}} \right]_+ , \ where\ [x]_+ := \max\{x, 0\} \quad (16)$$

However, the parameter $decel_{l,max}$ is still difficult to obtain in real-world, where acceleration and deceleration capabilities of other vehicles are often unknown, especially if those vehicles are not connected. This is one of the major drawbacks of this principle.

Alternatively, we suggest that the desired safety distance could be derived from considering ego vehicle's braking capacity together with time gap. Reusing the notation above, assuming that a desired deceleration $decel_{f,min}$ of the follower for braking is known (and constant throughout the manoeuvre). One can derive the time required for the follower to stop (t_{stop}) with the deceleration by calculating:

$$t_{stop} = \frac{v_f}{decel_{f,min}} \quad (17)$$

Consequently, if the following vehicle keeps the time gap larger than t_{stop}, then we can ensure that the following vehicle could stop before a potential collision. Furthermore, a safe following distance (d_{safe}) can be derived from the time gap equation, which results in:

$$d_{safe} = V_f(t) \cdot t_{stop} \quad (18)$$

Lastly, DCE (Distance-of-Closest-Encounter) as defined in [15] could potentially be used to assess the performance in the context of UC1 and UC2 as well. DCE is the minimum distance over a time horizon between two road users. All these distance metrics will be considered for UC1, UC2, and UC3 test cases, where applicable and relevant.

6.3 Time To Collision - minimum

Minimum value of TTC (TTC_{min}) during a test case is also a good indicator describing the worst-case scenario that has occurred during the test case. Indeed, the lowest value reached in the context of two vehicles approaching each other (for example, a first vehicle following a second) can be considered an indicator of the severity of the encounter. A low TTC value indicates a close proximity to a collision. Depending on the context, different acceptable TTC minimum are considered in the literature [9] [10].

As a reminder, in section 4.2.3, the notion of TTCE proposed in [15] was introduced, generalising that of TTC to the case where the minimum distance between the two actors concerned is not zero (i.e., it considers a non-collision cases). but at a distance to closest encounter (DCE) is used when calculating TTCE. Indeed, minimum TTCE could be another good safety indicator, which is similar to minimum TTC but also consider a non-collision case. As a reminder, TTCE is defined as the time to an encounter between two road users with the shortest distance over a time horizon. The shortest distance is named DCE (Distance-of-Closest-Encounter) in [15]. In case DCE = 0 (i.e., a collision will occur), then TCCE = TTC. It is then possible to generalise the TTC_{min} metric test case with the TTCE.

The minimum TTC and related approach are suitable to be applied in the context of UC1, UC2, and UC3.

6.4 Time Exposed Time-to-Collision

Assuming a threshold value TTC ($TTC_{critical}$), where a TTC under $TTC_{critical}$ is considered unsafe, Time Exposed Time-to-Collision (TET) [22] is the amount of time spent under the threshold value. In other words, it is the amount of time that a road user is exposed to a risky situation. An example of how TET is calculated is depicted in Figure 6.1 below. Mathematically, it can be expressed as:

$$TET_i = \sum_{t=0}^{T} f_i(t) \cdot \Delta t$$
$$f_i(t) = \begin{cases} 1, & \forall \ 0 \leq TTC_i(t) \leq TTC_{critical} \\ 0, & else \end{cases} \tag{19}$$

Where T represents total time in seconds and Δt represents a time-step used for analysis in seconds.

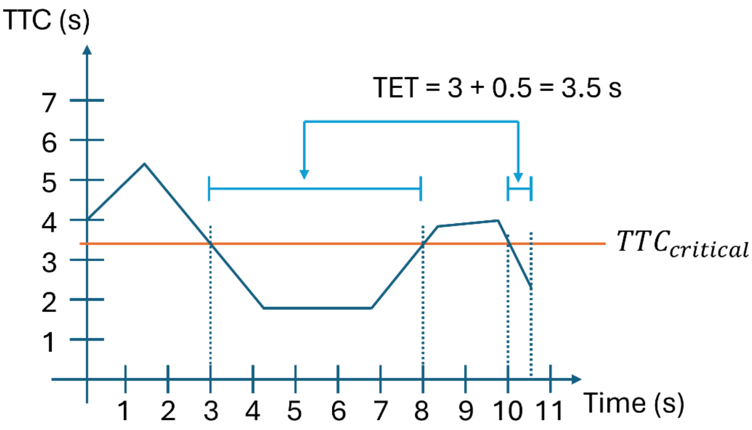


Figure 6.1 An example of TET calculation

TET is a good indicator of the duration in which the vehicle-under-test is exposed to a critical situation. In the context of the ROADVIEW project, TET could be used in UC1, UC2 and UC3. The challenge of applying TET is to determine the value of $TTC_{critical}$. One way to determine the critical TTC value is to combine with the other approach such as using a $TTC_{critical}$ value equal to $TTC_{min} + 1$ for TET calculation. This will allow testers to assess how long has the vehicle been exposed to the dangerous situation.

6.5 Time Integrated TTC

A drawback of TET is that the "criticality" aspect is not considered. That is, a value of TTC much lower than the $TTC_{critical}$ has the same effect on TET with a TTC value just below the critical value. Therefore, Time Integrated TTC (TIT) [22] is proposed to address this challenge. TIT is derived based on TET; it represents a measure of the integral of the TTC-profile during the time it is below the threshold. An example is illustrated in Figure 6.2.

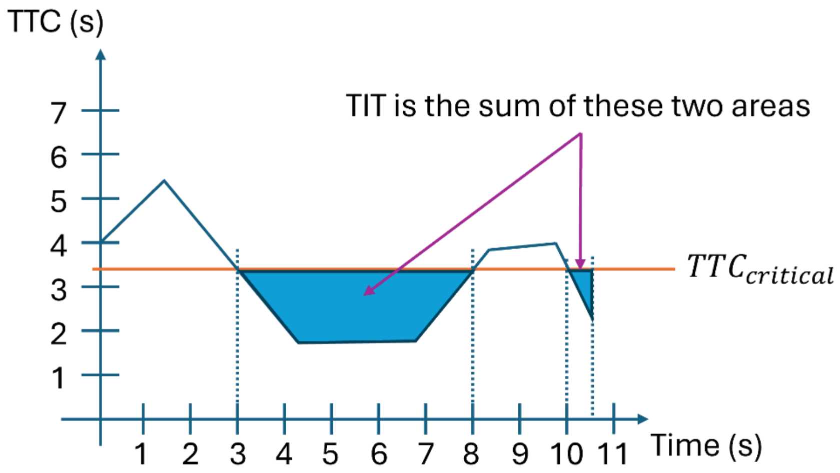


Figure 6.2 An example of TIT calculation

TIT can be calculated using the following formula:

$$TIT_i = \int_0^T (TTC_{critical} - TTC_i(t))\, dt \;\; \forall \;\; 0 \leq TTC_i(t) \leq TTC_{critical} \qquad (20)$$

Where T represents the total time in seconds.

One of the drawbacks of TIT is that it is difficult to interpret alone as it is an area with a unit of s^2. For instance, a very critical situation where a vehicle is exposed to a TTC value of 3 seconds below the critical value for 1 second would yield the same TIT as another situation where a vehicle is exposed to a TTC value of 0.5 seconds below the critical value for 6 seconds. Therefore, TIT is often used together with TET. Both indicators could be used in the test cases for UC1, UC2 and UC3.

6.6 Post Encroachment Time

Given a defined conflict area, post-encroachment time (PET) is the time measured from the moment that a road user leaves a conflict area until another road user enters the conflict area. PET is first proposed by Allen et al. in 1978 [23]. This is a temporal indicator of how closely a collision has been avoided for a certain conflict area. The process for calculating the PET is illustrated in Figure 6.3.

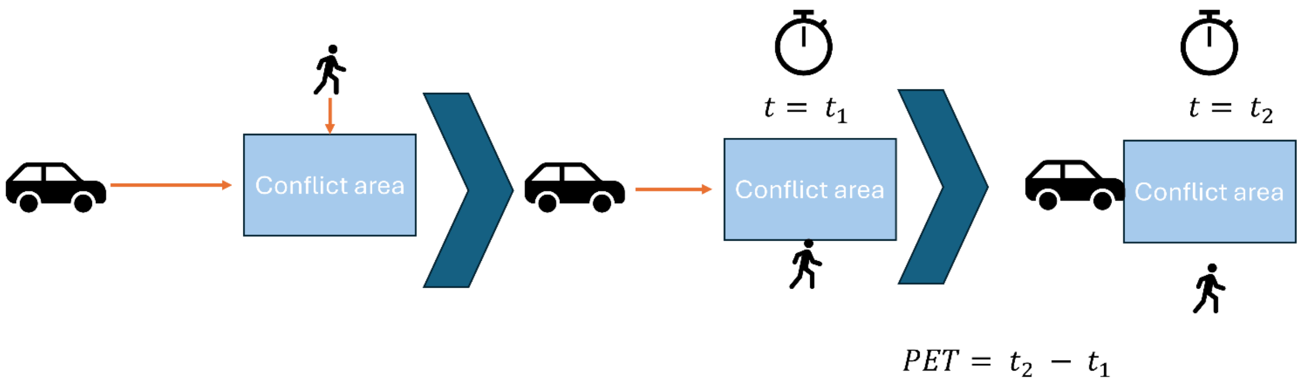


Figure 6.3 Illustration of post-encroachment time (PET) calculation process

PET is a suitable metric for evaluating intersecting conflict, where two road users' trajectories intersect with each other. Therefore, PET is suitable for UC3 test cases, where PET can be applied for all phases of UC3 (i.e., conflict, passing, and stopping). The PET would then indicate how safe the interaction was. A high value of PET could indicate that there is a sufficiently high amount of time between the potential conflict and thus indicate a safe interaction.

6.7 DRAC - maximum

The DRAC metric corresponds to the acceleration required (generally deceleration) to avoid a collision, and its minimum within a scenario therefore corresponds to the moment when it was necessary to brake most sharply to avoid the collision.

It is interesting to have this information because it is possible to estimate a maximum accessible value for a given configuration (type of vehicle, road conditions, etc.) and therefore to check by simulation whether the braking required has at some point exceeded the maximum accessible braking [24].

6.8 Time Gap - minimum

In a car-following situation, TTC is sometimes not available due to the leading vehicle having a higher speed than its follower. However, there are still collision risks associated with the distance between the vehicles. Using distance alone might not be a good indicator to assess the situation due to the lack of speed information. For instance, 30 meters is a relatively large distance when driving at 30 km/h (about 8 m/s); the vehicle has more than 3 seconds to come to a stop. That is not the case if the ego vehicle is driving at 60 km/h, where there will be less than 2 seconds to react.

Therefore, the time gap can be used as an alternative to assess the overall performance of the vehicle-under-test. In the context of ROADVIEW use cases, this can be applied to UC1, UC2, and UC3. A minimum time gap can be utilIsed in a similar way to TTC. Furthermore, average time gap over the test period could indicate the stability of the following behaviour (given car-following situations).

6.9 Two-dimensional Time-to-Collision - minimum

Traditional Time-to-Collision (TTC), as described in the previous section, primarily focuses on the relative motion along a single axis, usually the longitudinal axis, making it effective for identifying rear-end collisions. However, real-world traffic scenarios often involve lateral movements that can lead to sideswipe collisions before a rear-end impact occurs. This is particularly common during lane changes, where a vehicle may move into the path of another vehicle in an adjacent lane. In such cases, the traditional one-dimensional (1D) TTC metric, which only considers longitudinal motion, is insufficient for predicting potential lateral collisions. Therefore, both longitudinal and lateral dimensions have to be considered for lane-changing situations. These kinds of scenarios are more common among articulated vehicles where the likelihood of sideswipe collision is more than rear-end collision.

Figure 6.4 illustrates the two collisions, a rear-end collision and a sideswipe collision, between a tractor-semitrailer combination and a car. It is clear from the figure that a sideswipe collision happens before the rear-end collision. Consequently, 1D-TTC is non-reliable in these types of scenarios. Therefore, we suggest that two-dimensional (2D) TTC [25] should be considered.

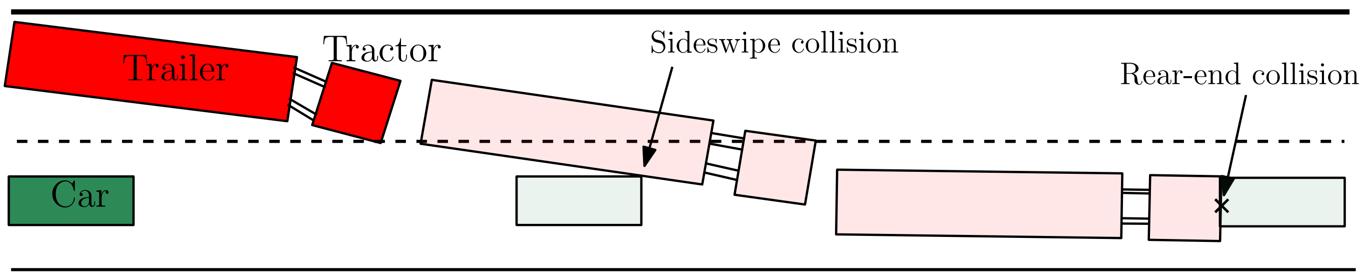


Figure 6.4 Collision between ego vehicle (tractor – semitrailer) and a car

Previous research such as **[25]** proposed 2D-TTC metric ($\boldsymbol{TTC_{2D}}$) for cars which considers the motion of the vehicles in both longitudinal and lateral directions and predicts collision risk in both these directions. The expressions used for computing the metric read as:

$$TTC_{lon} = \begin{cases} \dfrac{s_{0lon} - l}{v_{lon} - v_{0lon}} : s_{0lon} > l \cap v_{lon} > v_{0lon} \cap s_{0lat} - (v_{lat} - v_{0lat}).\dfrac{s_{0lon} - l}{v_{lon} - v_{0lon}} < w \\ \infty : else \end{cases} \tag{21}$$

$$TTC_{lon} = \begin{cases} \dfrac{s_{0lon} - l}{v_{lon} - v_{0lon}} : s_{0lon} > l \cap v_{lon} > v_{0lon} \cap s_{0lat} - (v_{lat} - v_{0lat}).\dfrac{s_{0lon} - l}{v_{lon} - v_{0lon}} < w \\ \infty : else \end{cases} \tag{22}$$

$$TTC_{2D} = \min\left(TTC_{lon}, TTC_{lat}\right) \tag{23}$$

where s_0 is the space headway between two vehicles, v_0 and v are the speed of the leading and following vehicles respectively, and l and w are the length and width of the vehicles respectively. The subscripts lat and lon are the corresponding lateral and longitudinal components respectively. Let s_1 be the space headway after TTC_{lon} and TTC_{lat}. Figure 6.5 and Figure 6.6 show the possible scenarios. It can be deduced that only case 2 in each of the scenarios will lead to collision. The time to reach the configuration of case 2 from a configuration with s_0 headway is TTC_{2D}.

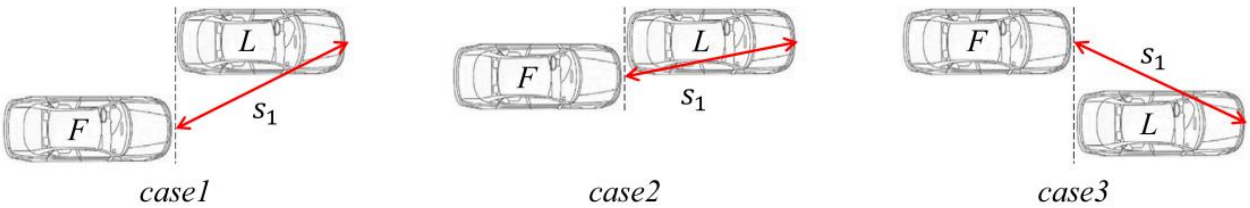


Figure 6.5 Rear-end collision scenarios [25]

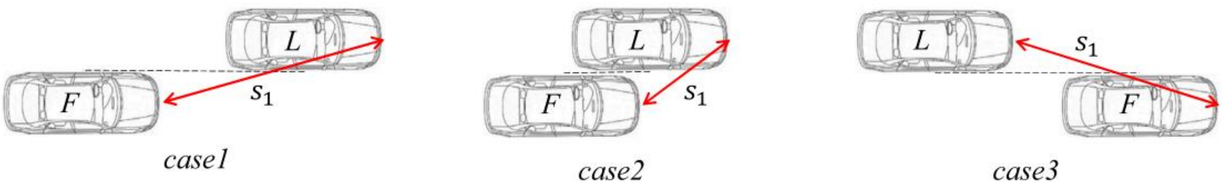


Figure 6.6 Sideswipe collision scenarios [25]

A drawback of the methodology for 2D-TTC in [25] is that the orientation of the vehicle is not considered. This would mean that the actual headways between the vehicles can be less than the computed headways. This will lead to erroneous estimation of 2D-TTC. Hence, the metric is improved to consider the orientation of the vehicles and then extended to articulated vehicles within the framework of the ROADVIEW project. The computation of the 2D-TTC for articulated vehicles (tractor-semitrailer) involves three steps. In the first step, the orientations, and speeds of the articulated vehicle and the car are transformed into a single frame of reference. This is performed by applying necessary transformations. In the second step, the time to collision is calculated for both longitudinal and lateral directions for both units using the transformed speeds. The minimum of these two TTC values for each unit determines the type of potential collision. If the lateral TTC is the minimum, it indicates a sideswipe collision; otherwise, it indicates a rear-end collision. The third step involves considering the minimum of TTCs obtained from both units. The unit with the minimum TTC is the one susceptible to collision. Further details on the expressions of the 2D-TTC metric for articulated vehicles will be provided in a future publication.

2D-TTC can be applied in test cases considering lane-changing situations within the context of UC4 and UC5. Examples of application of 2D-TTC in this context are given in Chapter 8.3.

6.10 Metric application and thresholds

This chapter provides guidelines for the application of metrics. Section 6.10.1 specifies the context in which they may be calculated. In 6.10.2, validation criteria are proposed.

6.10.1 Metrics application according to test case configuration

This section summarises the motion planning metrics with respect to their intended usage in ROADVIEW use cases. The summary can be found in Table 7. As an example, the metrics are then calculated in CARLA simulation assuming different weather conditions. Preliminary results on how to determine a test case threshold (i.e., a threshold value to validate whether the systems-under-test pass or fail during a test case) for each metric are then presented at the end of this section.

Table 7 Summary of motion planning metrics with respect to use cases

Metrics	Use Case				
		UC3			
	UC1 & UC2	Conflict	Passing	Stopping (including UC2)	UC4 & UC5
CLP	X				
$Distance_{min}$	X			X	
TTC_{min}	X		(X)	X	
TET	X			X	
TIT	X			X	
PET			X		
$DRAC_{maximum}$	X			X	
Time Gap	X			X	
2D-TTC					X

6.10.2 Test-case metrics criteria

The metrics chosen in this section are mostly time-based metrics. While time-based metrics such as TTC are widely used and easy to apply in all testing environments, one of its major drawbacks is that it requires a threshold parameter to distinguish between safe and unsafe situations. There is no consensus on how to set this threshold parameter in the literature. For instance, researchers have considered values between 1-3 seconds [9] as a threshold for TTC value.

Furthermore, all values above the threshold are considered “safe” regardless of their difference from the threshold and vice versa. In other words, if the TTC threshold is set to 2 seconds, TTC 2.5 seconds and 10 seconds are both treated equally and considered safe; this similar challenge also applies to TET. However, TIT can be used as a complement to indicate the amount of risk occurred under a situation.

As mentioned above, defining the threshold is crucial for distinguishing between safe and unsafe situations. In the context of this deliverable and T7.2, this threshold will also be used to assess whether the system-under-test pass or fail during the test case. Nevertheless, it should be noted that the discussed thresholds in this section are based on theoretical assumptions of the test cases, and thus these thresholds should be revised once the test data from the PG are available later in the project.

Different weather conditions will have an effect on the friction of a road surface. For instance, when a road surface is wet, a lower friction value can be expected. However, there are several other factors that will affect friction such as speed, temperature, road surface type, etc. Therefore, it is difficult to define the exact value for road surface friction value. Many attempts to measure and estimate the road surface have been made in the past, as summarised in [26]. In this report, we will consider friction values according to [27], which assumes the friction coefficient as 0.7, 0.4, and 0.1, for dry asphalt, wet asphalt, and icy road, respectively.

To provide an example in this deliverable, the friction coefficients were used in CARLA simulation (depicted in Figure 6.7) to obtain example metric values summarised in Table 8. The test case in CARLA is a simplified test case for UC3, where the ego vehicle will keep accelerating until there is an object detected within 50 metres detection range, then it applies a full brake. The simplified test case is used as an example here since actual test cases that will be conducted are defined yet at the time of writing this deliverable.

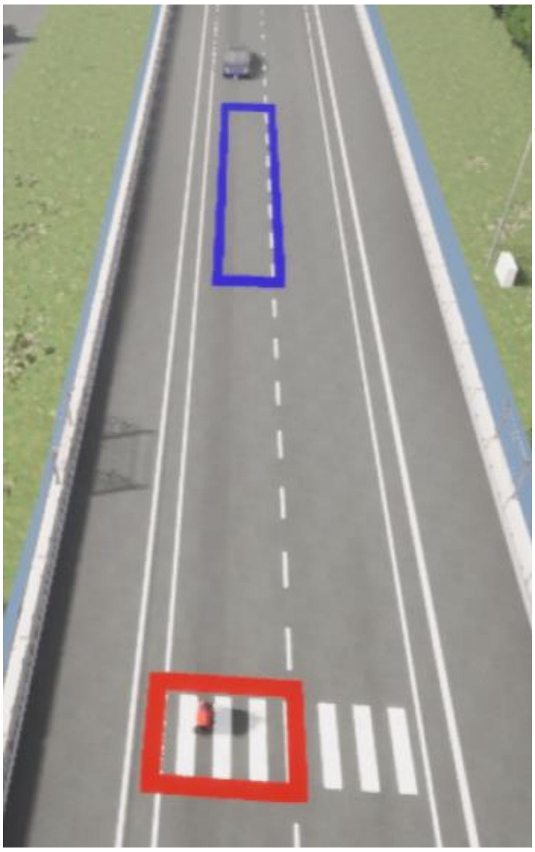


Figure 6.7 CARLA simulation setup (blue box indicates slippery patch on the road; red box a conflict zone)

Table 8 An example metric values for UC3 tests with respect to different weather conditions

Use Case 3	Metric	Metric value		
		Dry	Wet	Ice
Ego vehicle stopping	Distance (minimum)	25.14 m	19.53 m	17.16 m
	TG (minimum)	3.06 s	3.07 s	1.97 s

In order to distinguish between safe and unsafe situations, a threshold for each metric can be defined in several ways depending on the system design and overall safety criteria of the systems-under-test. This is beyond the scope of this deliverable since not all implementations and specifications of the systems under test are ready at the time or writing this deliverable. Nevertheless, as Table 8 suggests, weather conditions have effects on the metric value in different ways even in a simple manoeuvre. Therefore, the weather conditions should be taken into account when setting a threshold for evaluating the performance of the systems under test. Lastly, it is suggested that a combination of metrics shall be used instead of relying on one single metric since different effects can be seen already in a simple example shown in Table 8.

7 Comparison of PG and Simulation results

7.1 Background and context

The aim of this section is to describe the method used to compare the results obtained on the XiL and PG case tests. For each test case in a given configuration (simulated or real), one or more repetitions of the scenario can be carried out. A minimum number of 3 repetitions is preferable if the results are to be reliable.

The first steps, which are not described here, consist of collecting the data from the same repetition of a given test case together. This is because, especially in the real world, the data generally comes from several sensors (several players) and it is necessary for the spatial and temporal reference frames to be identical, or at least for the information needed to synchronise them to be available.

The basic metrics have been chosen so that the data needed to calculate them is always available, whatever the configuration chosen (XiL, HiL, SiL). The test-case aggregation methods can then be applied to each repetition of each configuration.

7.2 Global synchronisation

The first step in comparing simulation data with PG is to transpose the data so that they align with a single spatial and temporal reference frame.

7.2.1 Spatial synchronisation

By choosing a consistent spatial origin, we can accurately compare and analyse vehicle performance under simulated conditions and in the test field. This approach allows all the data to be precisely aligned and synchronised to this common origin. This facilitates a more effective assessment of vehicle behaviour and performance in different test environments.

To achieve spatial synchronisation, it is necessary to establish a common spatial origin for all iterations of the same test case. This origin should be based on a contextual element relative to the ego vehicle but must not be directly linked to the vehicle itself. For example, this reference point could be the initial position of the lead vehicle or a specific point on the road, such as an intersection.

7.2.2 Time synchronisation

Similarly, temporal synchronisation involves establishing a common time origin, simplifying the comparison of data across repetitions and configurations (PG or simulation). Once again, the choice of the temporal origin should be based on an event in the test case. For instance, this could be the moment another vehicle starts moving or the moment another actor crosses a reference line within the global spatial reference frame.

7.2.3 Define a common begin and end

Finally, the last step is to define the start and end points for each scenario. These points should be set in alignment with the test case and the specific aspects being analysed. For instance, if the focus is on observing a vehicle's reaction to a pedestrian crossing the road, the start and end points should clearly encompass this event, ensuring that it is not partially omitted by starting too late or ending too early.

Depending on the situations and metrics (test case metrics), it may be appropriate to select two different sequences within the same scenario. For example, when defining the minimum Time-To-Collision, it might be useful to focus on a local minimum rather than always considering the global minimum throughout the entire scenario. This approach allows for a more precise and relevant analysis of critical moments within the scenario.

7.3 Environment perception – PG and simulation results comparison

For environmental perception (EP), the first step is to calculate the test-case metrics for each test case and each repetition. Depending on the scenario, it may not be possible to calculate all the metrics.

7.3.1 EP - Level 0 – Object detection test case validation criteria

Level 0 of the comparison between the PG and the simulation consists of comparing the result obtained with the scenario tested on the validation criteria. In other words, was the scenario tested a success or a failure for both the PG and the simulation? As mentioned earlier, a failure in a scenario for both the PG and simulated test can however be a good result for the simulation as it has reproduced the same behaviour as the test in the physical world.

Table 9 lists all the possible combinations of results and shows that the simulation can be validated if the validation criteria for the test case are not met.

Table 9 EP - Level 0 simulation validation - Test-case validation criteria satisfied or not

Level 0 validation test	PG	Simulation	Simulation validation
Were the test-case validation criteria satisfied?	Yes	Yes	Yes
	No	No	Yes
	Yes	No	No
	No	Yes	No

7.3.2 EP - Level 1 – Comparison between test-case metrics

Level 1 of the comparison between the PG tests and the simulation consists of an absolute comparison of the results obtained for the various test-case metrics. At this stage, the choice of metrics to be applied depends on their relevance to the scenario. The difference between the results of the simulation and the PG tests is used to assess whether or not the simulation can be validated. Conditions are proposed in Table 10 but these will need to be adjusted if necessary depending on the scenarios and test conditions (weather conditions, traffic conditions).

Table 10 EP - Level 1 simulation validation – Minimum absolute difference to be achieved for each metric

Level 1 validation test	Condition on absolute difference between simulation and PG results to validate the simulation
First detection	< 0.5 sec
Sensitivity	< 15%
Confidence	< 15%
F1-score	< 15%
FPPI	< 15%
MOTA	< 15%
MOTP	< 15%

The calculation of these different metrics may require the introduction of parameters (thresholds). Values are proposed in the chapters describing these metrics, but they should be adjusted if necessary according to the context of the test cases (particularly the weather) and simple sensitivity tests should be carried out (whether or not the validation of the simulation is strongly linked to the choice of parameter).

7.3.3 EP - Level 2 – Global score

Level 2 validation between PG and simulated tests consists of calculating an overall score based on the results obtained for the test cases metrics.

As the reference is considered to be the PG results, we look at the average of the relative differences between the simulation and PG results.[2]

$$Level\ 2\ EP\ Simulation\ score = \sum_{i}^{N_{metric}} \left(\frac{metric_{i,simul} - metric_{i,PG}}{metric_{i,PG}} \right) \times \frac{1}{N_{metric}} \qquad (24)$$

Where N_{metric} is the number of different metric used for the given scenario.

As shown in, it is proposed that the level 2 validation criterion for the simulation should be a minimum threshold to be reached for the 80% score.

Table 11 EP - Level 2 simulation validation – Minimum score to reach

Level 2 validation test	Criteria for validating the simulation
Level 2 EP Simulation score	> 80%

7.3.4 EP - Level 3 – Curve comparison

Level 3 validation between the PG and simulated tests consists of comparing the estimated trajectory curves of the detected objects. Please note that we are not comparing the curves of the detected objects themselves, but rather the curve of their detected trajectory.

As presented in Table 12, the simulated curve must lie within a range defined by the curve obtained by PG plus or minus three standard deviations.

Table 12 EP – Level 3 simulation validation – Curve comparison

Level 3 validation test	Is the simulation curve within the PG curve +/- 3 standard deviation	Simulation validation
Detected object trajectory curves	Yes	Yes
	No	No

7.4 Motion planning – PG and simulation results comparison

As for the environmental perception, the first step for motion planning (MP) is to calculate the test-case metrics for each test case and each repetition. Depending on the scenario, it may not be possible to calculate all the metrics.

7.4.1 MP - Level 0 – Motion planning test case validation criteria

As with the perception of the environment, level 0 of the comparison between the PG and the simulation consists of comparing the result obtained with the scenario tested for a given validation condition. The result obtained must be the same for the test bed and the simulation.

Table 13Table 9 lists all the possible combinations of results and shows that the simulation can be validated if the validation criteria for the test case are not met.

Table 13 MP - Level 0 simulation validation - Test-case validation criteria satisfied or not

Level 0 validation test	PG	Simulation	Simulation validation
Were the test-case validation criteria satisfied?	Yes	Yes	Yes
	No	No	Yes
	Yes	No	No
	No	Yes	No

Level 0 of the comparison between the PG and the simulation consists of comparing the result obtained with the scenario tested on the validation criteria. In other words, was the scenario tested a success or a failure for both the PG and the simulation? As mentioned earlier, a failure in a scenario for both the PG and simulated test can however be a good result for the simulation as it has reproduced the same behaviour as the test in the physical world.

7.4.2 MP - Level 1 – Comparison between metrics

As with the perception of the environment, level 1 of the comparison between field tests and simulation consists of an absolute comparison of the results obtained for the different metrics of the test cases. Here again, the choice of metrics to be applied depends on their relevance to the scenario.

As shown in Table 14, an absolute difference criterion is defined for each metric.

Table 14 MP - Level 1 simulation validation – Minimum absolute difference to be achieved for each metric

Level 1 validation test	Condition on absolute difference between simulation and PG results to validate the simulation
CLP	< X
TTC minimum	< 0.5 sec
TET	< 0.5 sec
TIT	< 0.5 sec²
DRAC	< X
Time Gap	< X

As with perception of the environment, parameters are needed to calculate these metrics, and care must be taken to ensure that the values proposed in this report are consistent with the different situations observed.

7.4.3 MP - Level 2 Simulation validation – Global score

As with perception of the environment, level 2 validation of motion planning between PG and simulated tests consists of calculating an overall score based on the results obtained for the test cases metrics. The equation is then similar.

$$Level\ 2\ MP\ Simulation\ score = \sum_{i}^{N_{metric}} \left(\frac{metric_{i,simul} - metric_{i,PG}}{metric_{i,PG}} \right) \times \frac{1}{N_{metric}} \qquad (25)$$

As Table 15 shows, the score must exceed 80%, which means that on average, the scores obtained on the metrics in the simulation must not be too far from the metrics in the PG.

Table 15 MP - Level 2 simulation validation – Minimum score to reach

Level 2 validation test	Condition on score to validate the simulation
Level 2 MP Simulation score	> 80%

7.4.4 MP - Level 3 – Curve comparison

Level 3 validation between the PG and the simulated tests consists of comparing the trajectory and speed curves as a function of the time of the ego vehicle.

Comparing vehicle trajectories between the PG and the simulation alone is not sufficient, as they could pass through the same points but with different dynamics. This notion of dynamics has a strong impact on the results of the metrics calculated previously for levels 1 and 2.

As Table 16 shows, the simulated curve must lie within a range defined by the curve obtained by PG plus or minus three standard deviations.

Table 16 Level 3 simulation validation – Ego vehicle curves comparison

Level 3 validation test	Is the simulation curve within the PG curve +/- 3 standard deviation	Simulation validation
Ego vehicle position and velocity curves	Yes	Yes
	No	No

8 Example applications in simulation

In this chapter, the methodology presented in this deliverable is applied to two separate examples. In particular, in sub-sections 8.1 and 8.2, all the steps described in chapter 7 are applied to a dataset (Proving ground data: [28], Simulated data: [29]) corresponding to a concrete example very similar to the UC1 described in chapter 4. It should nevertheless be kept in mind that this example application is not part of the test cases that will be carried out in task 7.4 of the ROADVIEW project when the XiL system is ready. This example is being used to verify and validate some proposed metrics, and it has highlighted the limitations that need to be anticipated.

Sub-sections 8.1 and 8.2 therefore correspond to the same example and describe the object detection validation and motion planning validation steps respectively. The scenario is described at the beginning of sub-section 8.1. Section 8.3 is a simulation example to illustrate 2D-TTC.

8.1 Car following bicycle – Object detection evaluation

In this example, a vehicle (ego_vehicle) follows a bicycle (target). Initially, both are stationary, then the vehicle starts up, the bike also starts up, and both go in a straight line one behind the other. The vehicle soon catches up with the bike and has to slow down and then stop to avoid a collision. The bike also stops a little further on.

The part studied in this scenario is the moment when the two actors approach each other.

The scenario was carried out in PG (PG) and simulated (simul) conditions, in daylight (clear weather), at night and in rainy weather. Each scenario was repeated 3 times (DG1, DG2 and DG3)

8.1.1 Global synchronisation

In the initial state, as we can see in Figure 8.1, the data obtained in real and simulated conditions are neither temporally nor spatially synchronised. The first step is therefore to define a common spatial and temporal origin for all the data (real and simulated).

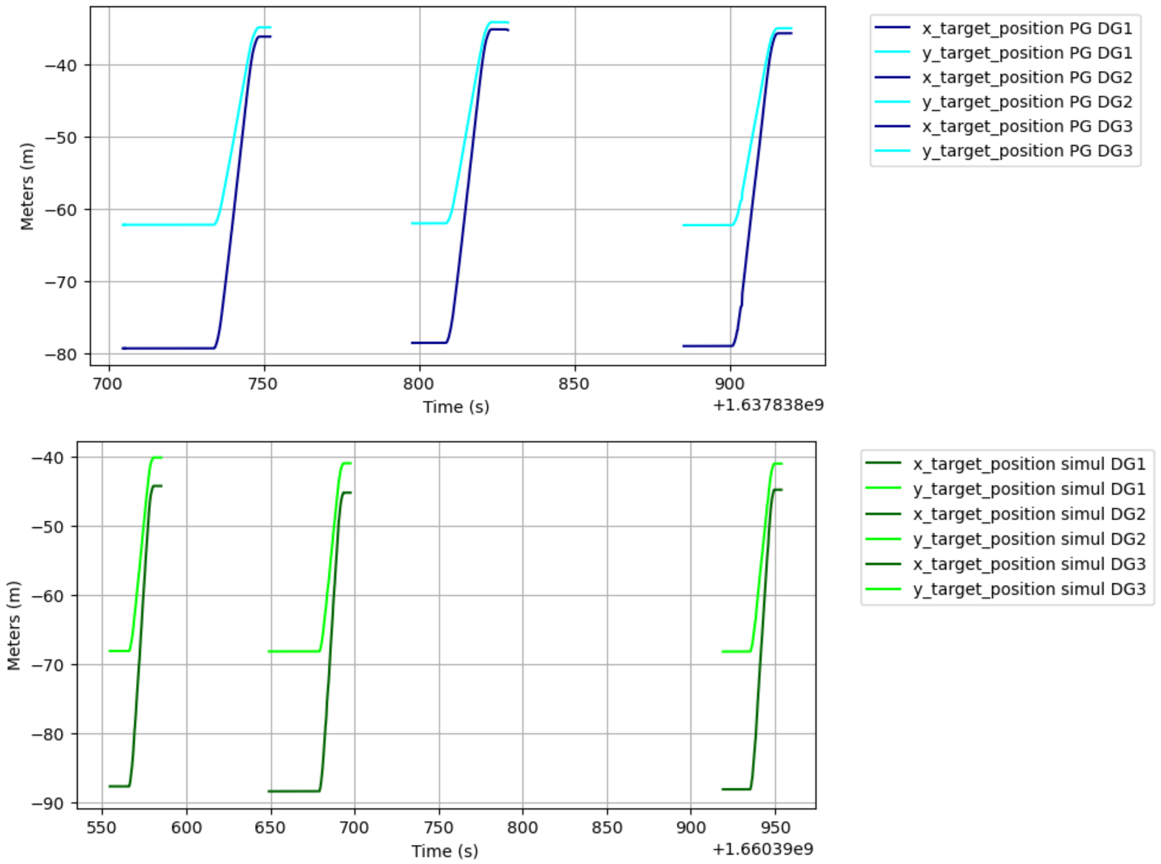


Figure 8.1 Raw data before any synchronisation

For this example, the origin of the spatial reference frame is the initial position of the bike (x_target and y_target), while the origin of the times is when the bike starts moving.

The scenario starts 10 seconds before the bike is set in motion and ends 10 seconds afterwards. The choice of the end of the scenario is consistent with the key moment we wish to analyse, i.e. the moment when the two actors are closest.

Figure 8.2 shows the x and y position of the vehicle (ego) and the bicycle (target) as a function of time, for the real case and the simulated case. The 3 repetitions are plotted one on top of the other using the same colours.

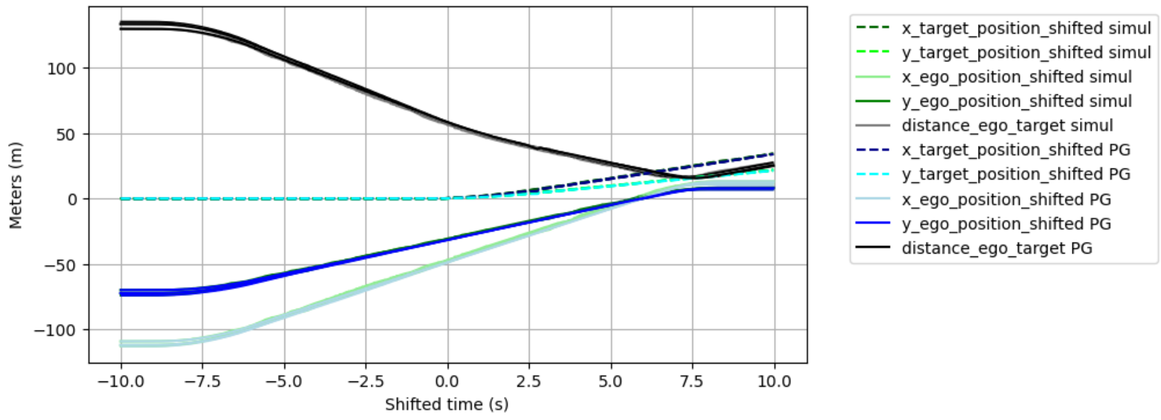


Figure 8.2 Synchronised data.

8.1.2 Basic metrics calculation

To calculate the basic metrics related to object detection, the previously mentioned synchronisations are not the only preliminary steps. Typically, there is also a need for preprocessing and aligning the raw sensor data with the rest of the system's data.

Specifically, the information related to the detected object's location—here the bicycle—is initially provided relative to the position of the sensor itself, and by extension, relative to the ego vehicle's position. However, the ground truth information about the object's location is typically provided in the same global reference frame as the ego vehicle. Therefore, it is necessary to transform the detected data into this global reference frame. This transformation ensures that both the detected positions and the ground truth data are comparable within a consistent spatial framework.

In practice, it is the position of the 8 points forming the bounding box of the detected object that needs to be transformed in order to obtain their coordinates in the global reference frame. In this example, when the bike is detected by the sensor, the coordinates of the 8 points are obtained from the coordinates of the centre of the bounding box, its dimensions (length, width, height) and its rotation around the vertical axis. It is then possible to obtain the coordinates of the 8 points in the sensor frame and transfer them to the global frame.

After the transformation applied to the bounding box points corresponding to the detected object and the real object, a graph with a top-view projection was produced to ensure the validity of the processing. In the example shown in Figure 8.3, it's a top view (2D projection) and then only 4 points among the 8 of the box are plotted. We can see that the two rectangles (green and red) would not overlap (low IoU value) and that the threshold for considering the object detected would certainly not be reached. The graph therefore corresponds to a frame where detection was not good.

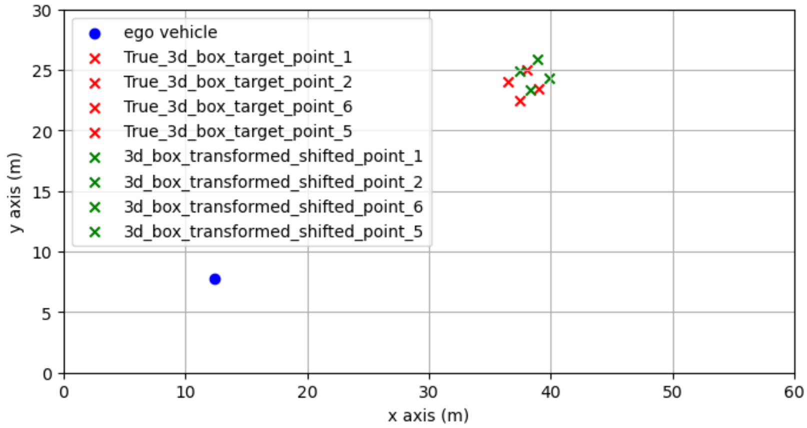


Figure 8.3 Top-view of detected and true bounding boxes in the global spatial reference (PG – Day config).

To get a better idea of the quality of object detection as a function of time, Figure 8.4 gives the coordinates in the (x and y) plane of the same point of the two bounding boxes corresponding to the detected object (target detected box) and the real object (target true box).

In the example studied, the simulation shows very different results from the test cases carried out in real life. This is due to the poor quality of the sensor and the associated algorithm used to generate the data for the example analysed. There is no link to the ROADVIEW project.

In the example, the object first appears to be detected at a position offset from its actual position (the red and blue dots are under the lines of the same colour). From about $t = -5$ s, the detected and actual positions become virtually superimposed. We can see that at the end of the test, for $t > 2.5\ s$, an offset returns.

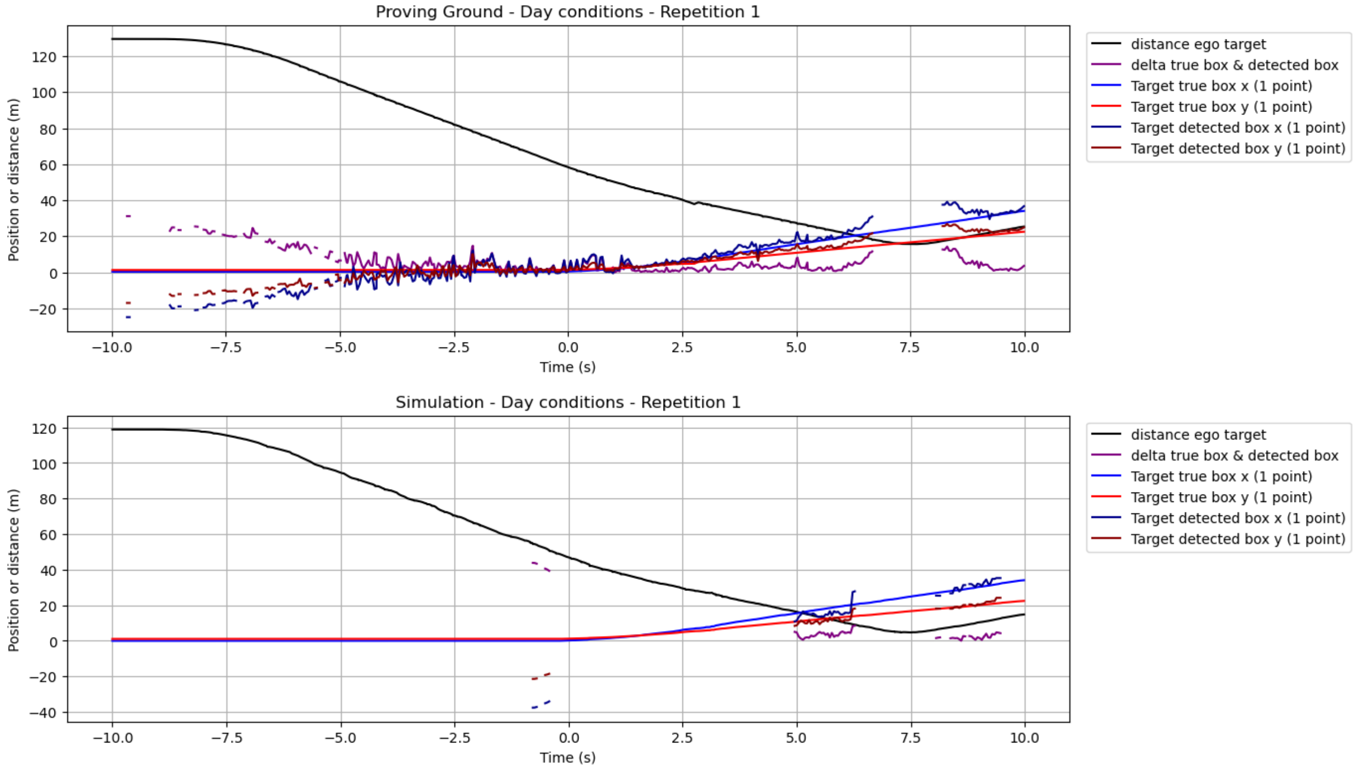


Figure 8.4 Plot of distance between a detected point and its ground truth position (1 of 8 bounding boxes)

Next, it is necessary to calculate the IoU for each frame and each bounding box detected with the ground truth bounding box object. From here, it is possible to determine for each frame and for each bounding box whether it is True Positive, False Positive, True Negative or False Negative. As a reminder, it is also necessary to define when (or where) an object should be detected (which will change a False Positive into a True Positive or a True Negative into a False Negative). In the graph below, the thicker blue curve indicates when the object should be detected.

For the example, the IoUs were calculated as a function of time (for each frame), but as mentioned above, the sensor and associated algorithm produced very poor results that were of little use for subsequent analysis. Thus, the object was considered to be well detected (TP) when the detected position was within 10 metres of the object's true position. This was done with the sole aim of obtaining realistic TP and TN shares for this type of scenario.

Figure 8.5 shows, both for PG and simulated tests (day conditions), the detection state curves (TN, FN, TP, FP) as a function of time. The categorical variable has been transformed into a quantitative variable and coloured to simplify reading.

We can already see from the curves that the simulation gives very different results from those of the PG tests. In this example, we can see that the object should not be detected from the same instant (thick blue curve), because the distance between the vehicle and the target is not identical as a function of time between the real case and the simulated case.

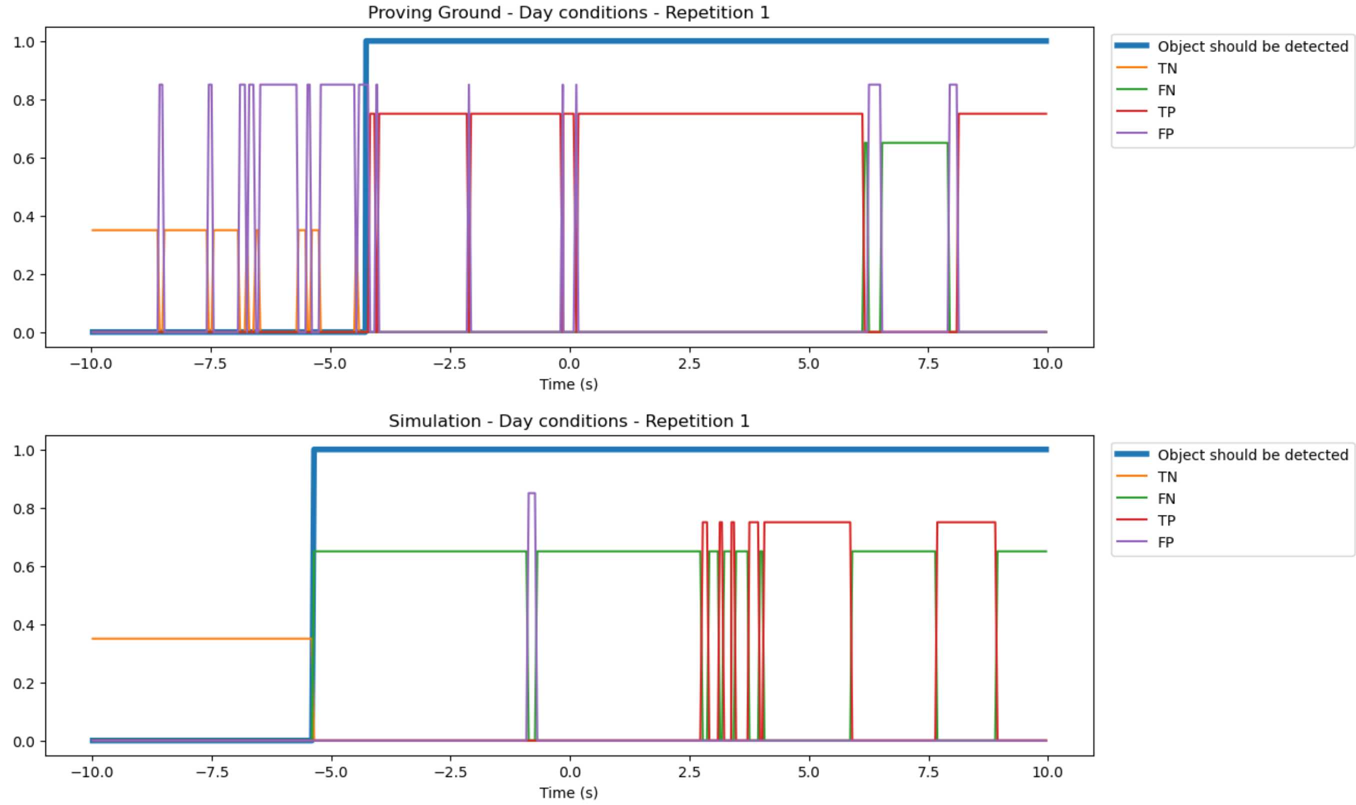


Figure 8.5 Graph showing the distribution of TP, TN, FP, FN as a function of time

By summing the duration of each frame corresponding to each status (TP, TN, FP, FN) we obtain the cumulative duration of each of them. It is possible to verify that the sum is indeed 20 seconds, corresponding to the duration of the scenario retained in this example.

Table 17 Dataframe of total duration calculated for each status (TP, TN, FP, FN)

config	meteo	TN	FN	TP	FP
day	PG	2.694758	1.490926	11.993011	3.781306
day	simul	4.535919	12.327238	2.183710	0.928010
night	PG	5.877273	7.808265	4.423555	1.864303
night	simul	4.766722	13.166633	1.962903	0.060629
rain	PG	5.653833	4.137541	9.355970	0.815028
rain	simul	4.626597	11.786829	3.412164	0.137188

8.1.3 Test-case metrics calculation

Once the distributions of TP, TN, FP and FN have been obtained, it is easy to calculate sensitivity, confidence and the F1-score by applying the corresponding formula. The duration of the frames must be taken into account if the time step is not constant. For the first detection, here it is the instant of the first frame in which a TP is found that has been considered.

Table 18 Dataframe of EP metrics results calculated for all the configuration and weather

config	weather	First Detection	Sensitivity	Confidence	F1-score
day	PG	-4.22	0.89	0.76	0.82
day	simul	5.54	0.16	0.67	0.25
night	PG	-0.93	0.36	0.70	0.48
night	simul	3.62	0.14	0.97	0.24
rain	PG	-3.13	0.69	0.92	0.79
rain	simul	3.50	0.25	0.96	0.40

In this example, there was only one object, so the two metrics MOTA and MOTP are not calculated.

8.1.4 Comparison between Simulation and PG Gap - Level 0 – Validation of the test case

Level 0 corresponds to the comparison of whether or not the test case has been validated according to the defined criteria. For the test case in the example, the validation conditions were as follows:

- First detection before –3 second
- Sensitivity, confidence and f1-score above 0.65

The Table 19 shows that the real test validates all the test-case criteria and has therefore passed the test case. However, the simulation only passed one of the four criteria and did not pass the test case.

Table 19 Verification of the pass/fail criteria of the test cases for the real test and the simulation

Criteria	PG	Simulation
First detection < -3 s	Validated	Not validated
Sensitivity > 0.8	Validated	Not validated
Confidence > 0.8	Validated	Not validated
F1-score > 0.8	Not validated	Not validated

A fortiori, given that the real case passed the test case but not the simulation, level 0 of the comparison between the simulation and the real case is not reached.

Indeed, we noticed that the simulation gave very different results from the outset. The results of the test-case metrics give an idea of the aspects where the simulation is not good enough.

8.1.5 Comparison between Simulation and PG - Levels 1, 2 and 3

The idea is that a given level of comparison between simulation and PG would be observed more in the case where each previous level has been validated.

For this example, the following levels have not been calculated because they are of little interest: the absolute differences between the test-case metrics are very large (level 1), and the overall score obtained on the basis of the average of the relative differences would be low (level 2) and the curves would also be very far apart (level 3).

8.2 Car following bicycle – Motion planning evaluation

The same example for environment perception is used to analyse motion planning. As the first stages of synchronisation are identical, they are not presented here.

The test case was distorted here because the vehicle did not have to detect the target, making it easier to plan its movement. This does not prevent the proposed methodology from being applied.

In the case of ROADVIEW, it is to be expected that poor object detection will make good motion planning performance even more difficult. This is why, the analysis, it is planned to look first at the perception of the environment.

There is, however, a pre-processing step which generates the data concerning the speed of the ego vehicle and the target.

8.2.1 Velocity curves obtained by calculation from position data

This step will not take place as part of the testing and data analysis of the ROADVIEW project, as the data will be available from the outset. However, the presentation of this stage here highlights the importance of the measurement step (or duration of a frame), which is too small here and tends to noisy both curves.

The curves in Figure 8.6 corresponding to the simulated case (in green) are particularly irregular, which has a direct impact on the smoothness of the curves for the basic metrics calculated. This behaviour is the result of the way that the simulation was conducted in this exemplary test. The recorded position of the real targets and ego vehicles are fed in the simulation and the velocity is calculated based on the time step generating the fluctuation on the velocity.

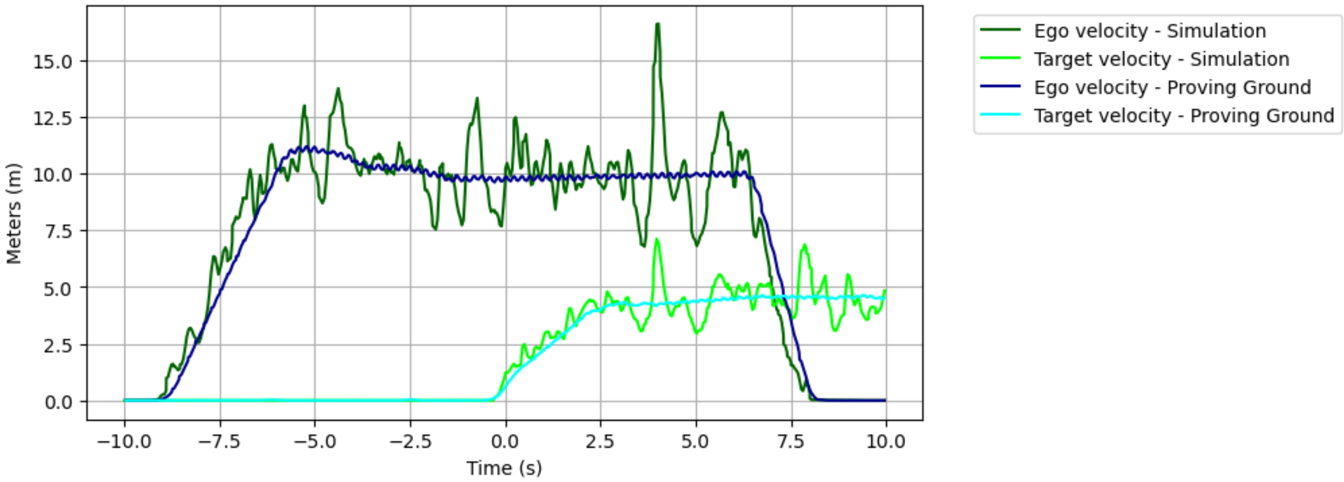


Figure 8.6 Velocity curves for ego vehicle and target, both for PG and simulated test in day conditions

8.2.2 Basic metrics calculation

In the case of motion planning, the only basic metric calculated for this example is time to collision. The different curves for the different repetitions (DG1, DG2 and DG3) for the real and simulated cases are shown in Figure 8.7 and

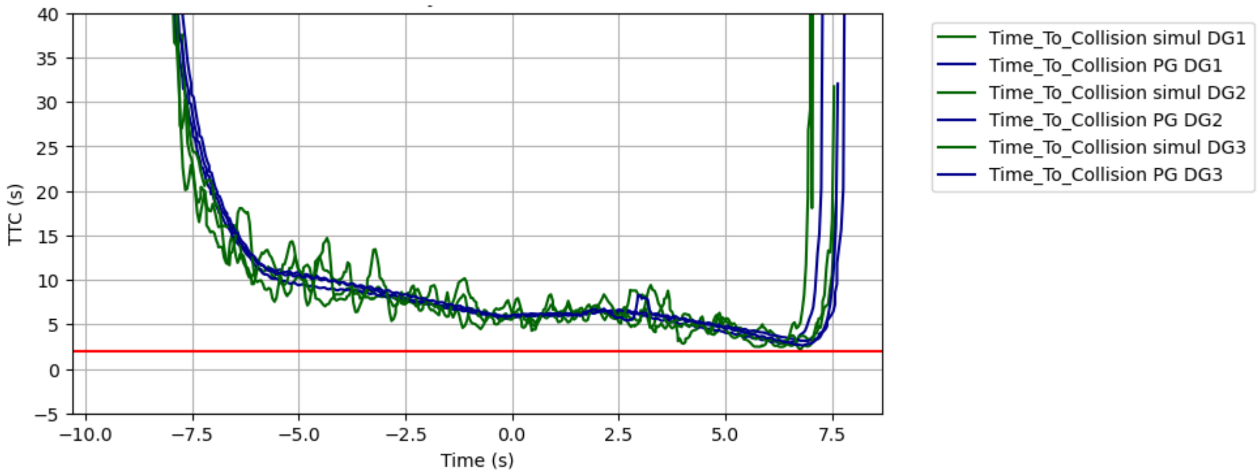


Figure 8.7 The red line shows the test-case validation criterion: TTC must never be less than 2 seconds.

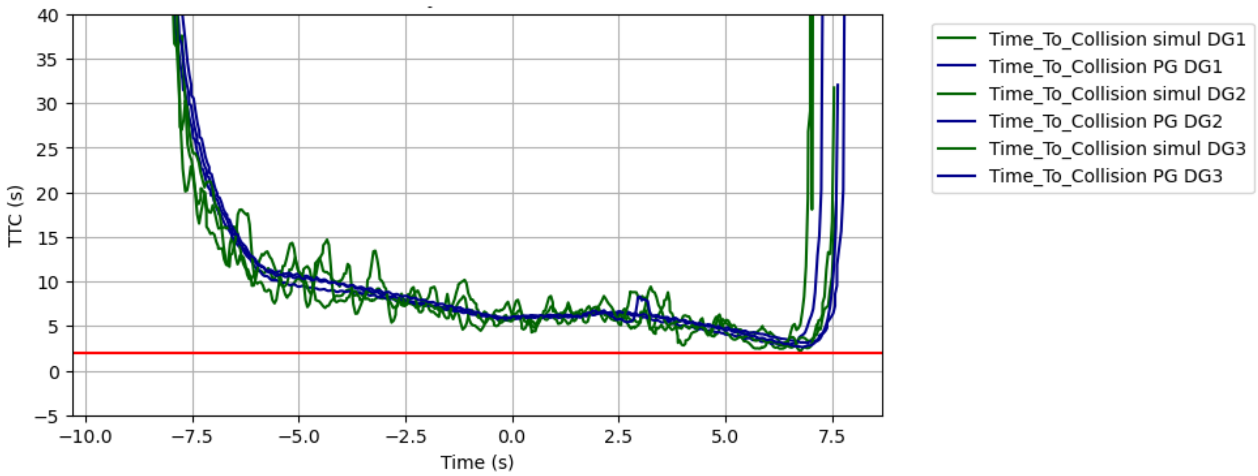


Figure 8.7 TTC curves for day conditions, both for PG and simulated test in a clean weather test

8.2.3 Comparison between Simulation and PG - Level 0 – Validation of the test-case

As mentioned above, for this example it is assumed that the only test-case validation criterion is that the TTC must never be less than 2 seconds.

Table 20 shows that both the real test and the simulated test meet this criterion, and we therefore consider that they both validate the test case.

Table 20 Verification of the pass/fail criteria of the test cases for the real test and the simulation

Test-case criterion	PG	Simulation
TTC > 2 s	Validated	Validated
TET < 3 s	Validated	Validated
TIT< 5 s²	Validated	Validated

What's more, since they both obtained the same result with test case validation criteria, the simulation is also validated against the real case.

8.2.4 Comparison between Simulation and PG - Levels 1 – Test case metrics comparison

The next step is level 1 comparison between simulation and PG, which consists of calculating and comparing test-case metrics.

TTC Minimum

In Figure 8.8, TTC curves are shown again and $TTC_{minimum}$ values are identified with horizontal lines. TIT and TET are then calculated from TTC minimum value.

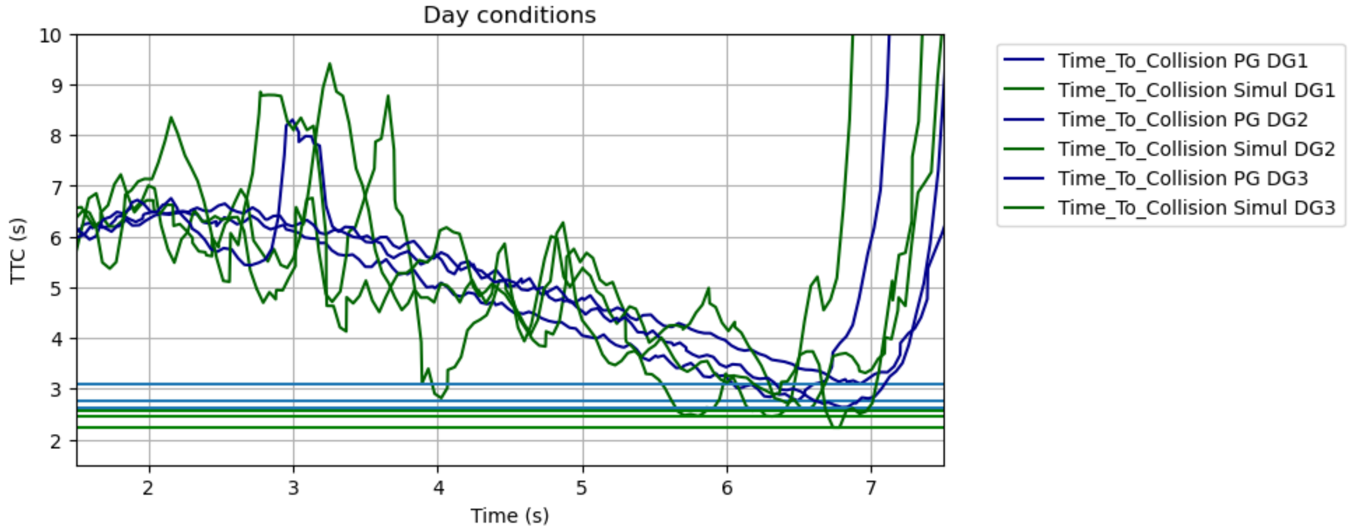


Figure 8.8 Graph showing the TTC curves calculated for PG and simulated tests and TTC minimum focus

In Table 21, results obtained for TTC minimum are given for both PG and simulated tests. Since the difference between the two averages is less than 0.5, the validation criterion for level 1 simulation, the simulation can be validated here.

Table 21 Verification of the pass/fail criteria of the test cases for the real test and the simulation

Configuration	TTC minimum (s)	Average (s)
PG	2.76 3.10 2.64	2.83 s
Simulation	2.46 2.57 2.24	2.42 s
	Level 1 Condition to validate simulation Difference < 0.5 sec	Simulation validated

Time exposed TTC

In Figure 8.9, TTC curves are shown again and the TET parameter is identified, corresponding to $TTC_{minimum}$ value +1 second, so 3.83 seconds here.

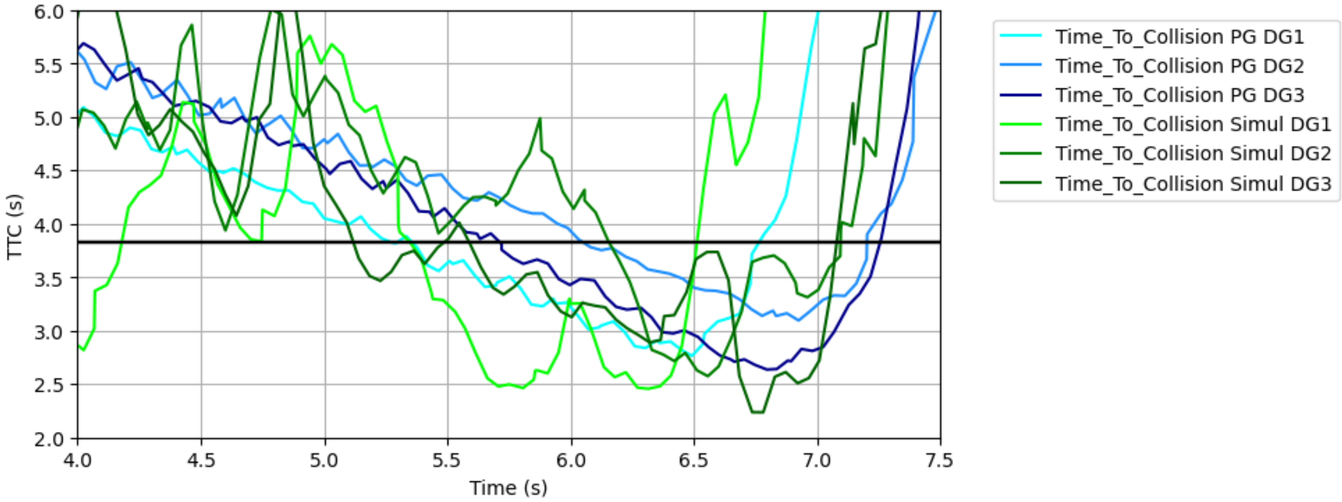


Figure 8.9 Graph showing the TTC curves and the TET parameter used to calculated TET metric

In Table 22, the results obtained for TET for each repetition and the average value are given, both for PG and simulation.

Table 22 Results obtained for TET for PG and simulation tests (day conditions)

Configuration	TET (s)	Average (s)
PG	1.46 1.16 1.54	1.39 s
Simulation	1.42 0.94 1.87	1.41 s
	Level 1 Condition to validate simulation Difference < 0.5 sec	Simulation validated

Time integrated TTC

For TIT, the parameter used is also the minimum TTC value +1 second, so 3.83 seconds here.

In Table 23, the results obtained for TIT for each repetition and the average value are given, both for PG and simulation.

Table 23 Results obtained for TIT for PG and simulation tests (day conditions)

Configuration	TIT (s²)	Average (s²)
PG	4.74 4.00 4.83	4.52
Simulation	4.13 3.02 6.04	4.40
	Level 1 Condition to validate simulation Difference < 0.5 sec	Simulation validated

8.2.5 Comparison between Simulation and PG – Level 2 – Global score

The calculation of the overall score is based on the average relative deviation on the different test case metrics calculated in level 1 with the equation in section 7.4.3, Table 24 summarises the intermediate values and the global score.

Table 24 Average difference between PG and simulated results and global score obtained

TTC minimum	TET	TIT	Global score
0.86	0.98	0.97	0.94

8.2.6 Comparison between Simulation and PG – Level 3 – Curve comparison

Level 3 corresponds to the comparison of trajectory curves for the ego vehicle. The validation condition for the simulation is that the x and y trajectory curves of the simulation must be within 3 standard deviations of the trajectory curves of the real test.

Figure 8.10 shows the mean x and y curves as a function of time for the ego vehicle in the simulated case, which are well within the coloured area corresponding to the mean x and y curves for the PG case +/- 3 variances.

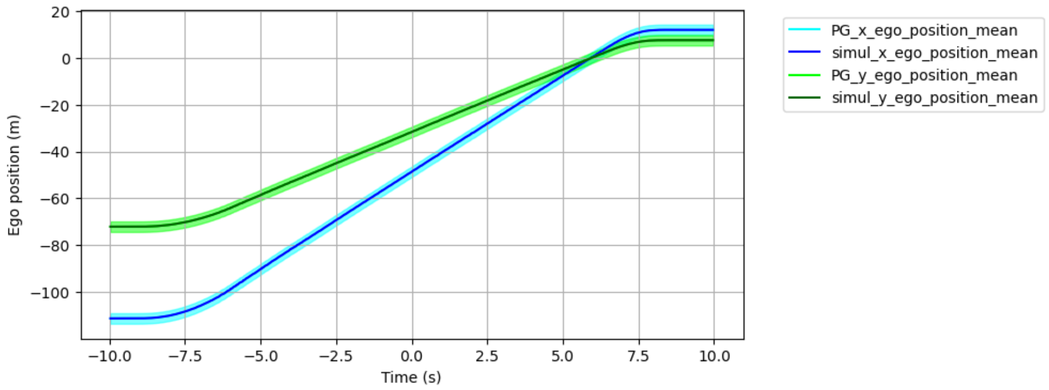


Figure 8.10 Comparison of the trajectory curves of PG and simulated tests

To be complete with regard to the methodology explained in section 7, level 3 validation of the simulation would also require the speed curves of the ego vehicle to be compared. However, this has not been done for this example.

8.2.7 Comparison between Simulation and PG – Level 0, 1 and 2 for rainy and night conditions

The results obtained for the same example in rainy conditions and at night are not detailed, but the tables below summarise the results obtained by applying exactly the same methodology as above. Table 25 presents the results of the test case metrics, while the second table directly compares the PG with the simulation, in order to easily identify cases where the simulation would be validated.

Table 25 Dataframe of MP metrics results calculated for all the configurations and weather conditions

Weather	Configuration	TTC_min	TET	TIT
day	PG	2.83	1.39	4.52
day	simul	2.42	1.41	4.40
rain	PG	2.85	1.27	4.17
rain	simul	3.15	0.79	2.74
night	PG	2.99	2.03	7.07
night	simul	2.77	1.49	4.90

Table 26 shows the absolute differences for each metric and weather calculated from Table 25 (line PG – line simul). As a reminder, the validation condition has been set at 0.5, i.e. the difference between PG and simulation must be less than this condition. It can be seen (red values) that while the simulation was always validated for the 'day' case, in the 'rain' case the results on the TIT did not allow it to be validated, and for the 'night' case it was the results on the TET and TIT that invalidated it (absolute difference > 0.5).

Table 26 Absolute differences between PG and simulation metrics results

Weather	TTC minimum	TET	TIT	Condition	Simulation validation
Day	0.41	0.02	0.12	<0.5	Yes
Rain	0.30	0.48	1.43	<0.5	No
Night	0.22	0.54	2.17	<0.5	No

8.3 2D-TTC metric application

The 2D-TTC metric developed in Section 6.9 for articulated vehicles is evaluated using the use case 'Entering a Lane' listed in Table 2. As shown in Figure 8.11, the ego vehicle, a tractor-semitrailer combination, is initially in the left lane while a car is in the right lane. The ego vehicle then changes to the right. Two types of collisions, rear-end and sideswipe, are possible in this situation depending upon the vehicles' speed and heading. An example scenario is created in Carla for the considered use case. Both vehicles start from a standstill and attain constant speed before the ego vehicle initiates the lane change. Both vehicles are in adjacent lanes at the start point, see Figure 8.11 (left). The scenario is designed such that both vehicles, the car and semi-trailer of the ego vehicle, are on a trajectory that will result in a sideswipe collision. The collision is shown in Figure 8.11 (right). This setup allows for verification of whether the metric can predict a sideswipe collision. The ego vehicle begins moving first, followed by the car (after 10 s of the ego vehicle). The ego vehicle reaches 45 km/h speed, while the car accelerates to 70 km/h before the ego vehicle initiates the lane change as shown in Figure 8.12.

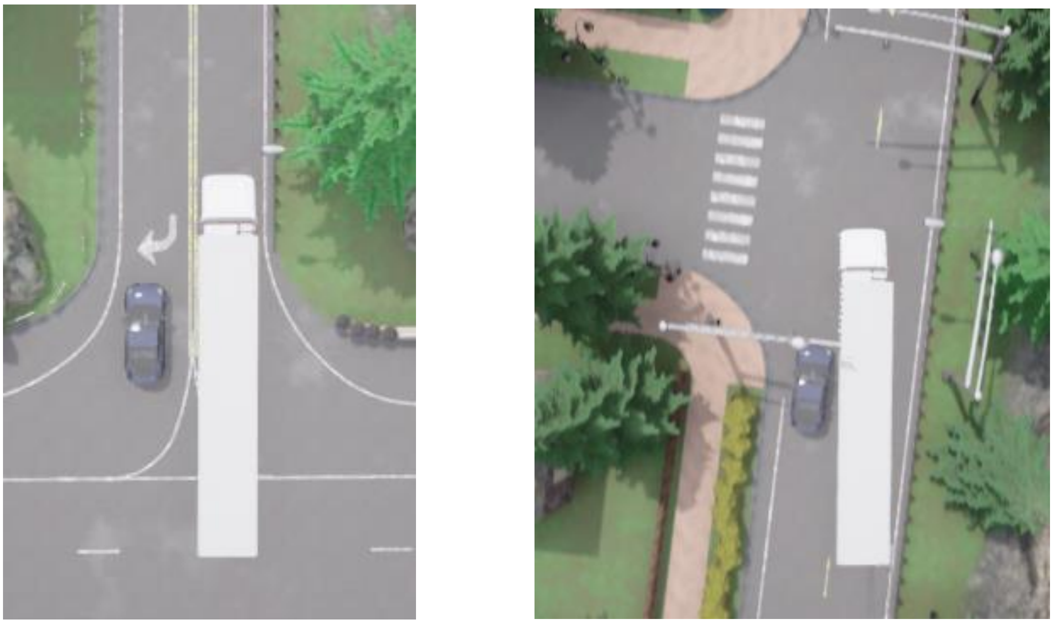


Figure 8.11 Scenario in Carla. Left: Standstill setup, Right: Sideswipe collision

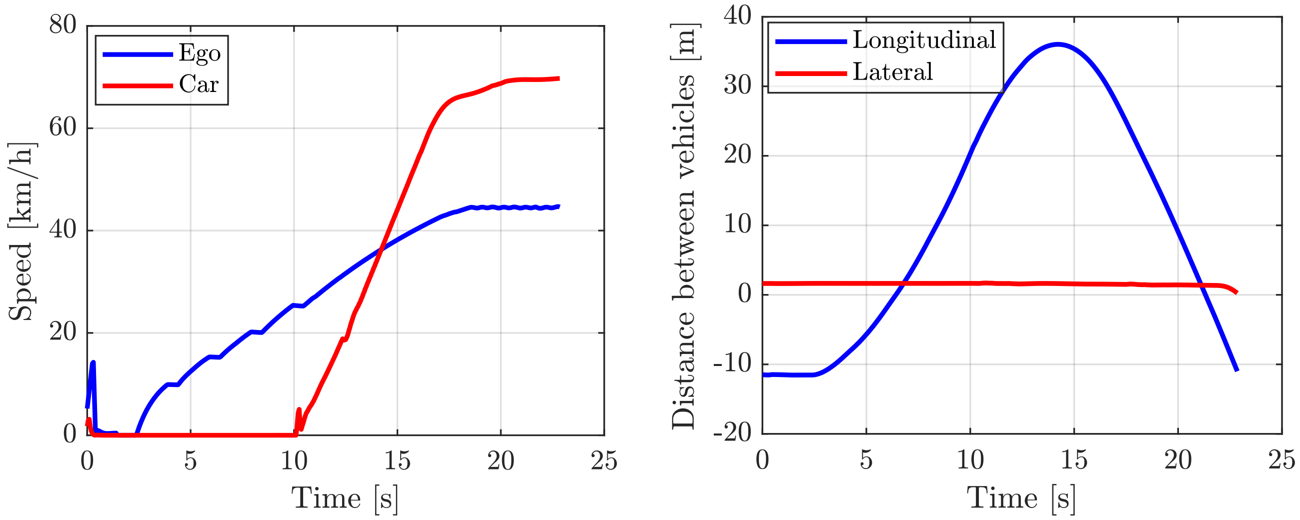


Figure 8.12 Speed and distance (longitudinal and lateral) between the vehicles

Figure 8.12 also shows the longitudinal and lateral spacing between the semitrailer and the car until the crash happens at around 23 seconds. The longitudinal spacing is defined as the difference between the position of the rear edge of the semi-trailer and the front edge of the car. Similarly, the lateral spacing is defined as the difference between the side edges of the considered vehicles. A negative longitudinal distance indicates that the front edge of the car is ahead of the rear edge of the semitrailer, while a positive distance means the front edge of the car is behind the rear edge of the semitrailer. When the car and the semitrailer are in adjacent lanes and next to each other, the longitudinal distance is in the range of (-18.4,0) m. The first entry is the sum of the wheelbases of the semitrailer and the car and refers to the situation when the car’s rear edge aligns with the semitrailer’s front edge. The second entry refers to the situation when the car’s front edge aligns with the semitrailer’s rear edge. There is a risk of sideswipe collision when the vehicles are next to each other and the lateral distance between them is reduced in every time step. For example, in this case, when the car is adjacent to the ego vehicle at around 21 seconds, the latter executes a lane change. The lateral distance decreases and eventually is 0 when there is a collision.

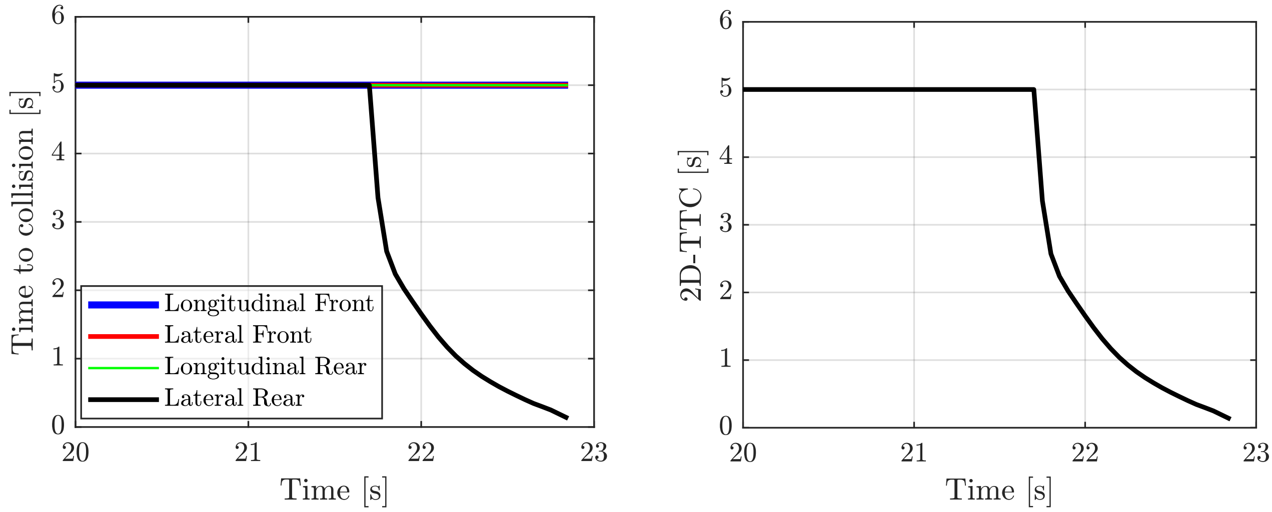
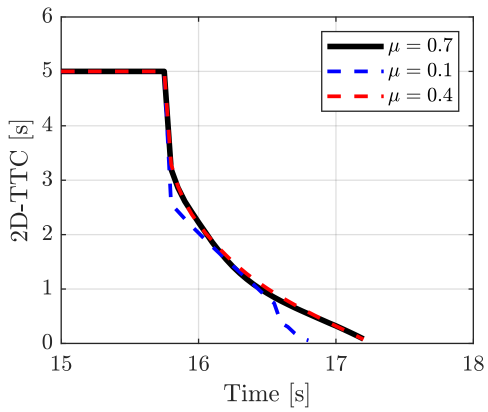


Figure 8.13 Time to collision computed for both tractor (front unit) and semi-trailer (rear unit) and 2D-TTC

Figure 8.13 (Left) shows the time to collisions obtained from both units of the ego vehicle in both longitudinal and lateral directions. The value '5' is used to indicate an infinite time to collision. It is evident from the figure that a sideswipe collision occurs between the rear unit (semi-trailer) and the car, as the lateral TTC corresponding to the semi-trailer (Lateral Rear in the legend) drops close to 0. This result is consistent with the expected outcome, as the scenario was designed to induce a sideswipe collision. Considering the minimum of all TTCs gives the magnitude of 2D-TTC as shown in Figure 8.13 (right). This minimum value corresponds to the TTC of the semitrailer in the lateral direction.



Error! Reference source not found.Error! Reference source not found. shows the 2D-TTC computed in a similar lane-changing scenario as described above in different friction conditions. The friction coefficients (μ) are representative of ice (μ = 0.1), wet (μ = 0.4) and dry (μ = 0.7) roads. The lane change leads to a crash in all the friction conditions. There is not a significant difference in the 2D-TTC computed in all the conditions. The reason behind this is that the vehicles in each of the cases are simulated in an open-loop fashion to understand if 2D-TTC can predict the collision. Hence, the trajectories of the vehicles in each of the frictions are very similar.

9 Conclusions

To conclude, we first revisit the overall objectives: conducting test cases that align with specific use cases is essential for evaluating a system's performance. Simulation serves as a crucial complement to this process by enabling a larger number of tests, which is necessary because validating the behaviour of an autonomous vehicle requires extensive mileage. Thus, it is equally important to validate the performance of these simulations against real-world trials.

In this report, we reviewed the key definitions of the metrics used in the context of CAVs. We distinguished between basic metrics, which are calculated at a given point in time, and test case metrics, which are aggregated, as their name suggests, at the test case level.

The proposed metrics are consistent with the test cases defined in T7.1 and will be applied in T7.4. These metrics are then calculable and applicable to both real-world and simulated tests. We have also defined examples of how success criteria should be formulated to evaluate whether the requirement is met for a test case. These criteria may need to be adjusted to account for specific conditions, such as weather and traffic. To support these adjustments, we have provided decision-making aids. Additionally, we proposed a methodology for assessing the discrepancies between results obtained from real test cases and those from simulations.

The proposed methodology for validating simulations is structured into several levels, where PG tests are compared to simulations, ranging from Level 0 to Level 3. Level 0 involves comparing results based on success criteria for the test cases. Level 1 focuses on the absolute comparison of results obtained from test case metrics. Level 2 involves comparing an overall score derived from several test case metrics, while Level 3 entails comparing the trajectory curves obtained for the test vehicle and objects within the scenario. We also provided an example of how this complete methodology can be applied to clarify the entire process.

The various metrics used, both basic and test case-specific, rely on different parameters and thresholds, the selection of which can significantly impact the results. Sensitivity studies have been proposed but could be further developed for certain specific configurations.

Since PG tests are not performed with UC4 and UC5, simulations in Carla are used to study metrics associated with these use cases. To deal with both longitudinal and lateral dimensional aspects of UC4 and UC5, a new metric ‘2D-TTC’ is defined in the context of an articulated vehicle. The metric is further evaluated in a lane change scenario similar to UC4 and UC5 in different road friction conditions. The scenario consists of an articulated vehicle making a lane change in front of a car. The investigation shows that the metric can detect collisions along both the dimensions of the vehicle. Moreover, the simulations indicate a very small difference in the magnitude of the metric in different friction conditions. This is primarily due to similar vehicle trajectories leading to crashes in all conditions.

As a future work, it may be interesting to create a scenario that does not lead to a crash and evaluate the metric in different friction conditions. This will mean the vehicles adapt to their trajectories based on the metric and therefore, the efficiency of the metric can be explicitly realised.

The next steps of the project will focus on conducting tests corresponding to Use Cases 1 through 3, both in real-world and simulated environments, as outlined in T7.4. During these tests, the metrics described in this deliverable will be applied to evaluate the system's performance. Following this, T7.5 will aim to validate the simulations using the methodology proposed in this deliverable. The ultimate goal is to achieve a comprehensive validation of the entire ROADVIEW system, ensuring its reliability and effectiveness under various conditions. This process will be crucial for confirming that the system meets the project's objectives and is ready for broader implementation.

10 References

[1] N. Kalra and S. Paddock, “Driving to Safety: How Many Miles of Driving Would It Take to Demonstrate Autonomous Vehicle Reliability?,” 2016.

[2] L. B. Markus M. and W. Hermann, “AutonomousDriving: Technical, Legal and Social Aspects,” 2016.

[3] L. Westhofen, “Criticality Metrics for Automated Driving: A Review and Suitability Analysis of the State of the Art,” 2021.

[4] M. N. Sharath and B. Mehran, “A Literature Review of PerformanceMetrics of Automated Driving Systemsfor On-Road Vehicles,” 2021.

[5] C. Katrakazas, M. Quddus, W.-H. Chen and L. Deka, “Real-time motion planning methods for autonomous on-road driving: State-of-the-art and future research directions,” Transportation Research Part C: Emerging Technologies, vol. 60, p. 416–442, November 2015.

[6] A. Dhar, “Object Tracking for Autonomous Driving Systems,” EECS Department, University of California, Berkeley, 2020.

[7] J. Luiten, A. Osˇep, P. Dendorfer, P. Torr, A. Geiger, L. Leal-Taixé and B. Leibe, “HOTA: A Higher Order Metric for Evaluating Multi-object Tracking,” International Journal of Computer Vision, vol. 129, p. 548–578, October 2020.

[8] M. Takahashi, A. Moro, Y. Ji and K. Umeda, “Expandable YOLO: 3D Object Detection from RGB-D Images,” 2020.

[9] S. M. S. Mahmud, L. Ferreira, M. S. Hoque and A. Tavassoli, “Application of proximal surrogate indicators for safety evaluation: A review of recent developments and research needs,” IATSS Research, vol. 41, pp. 153-163, 2017.

[10] C. Wang, Y. Xie, H. Huang and P. Liu, “A review of surrogate safety measures and their applications in connected and automated vehicles safety modeling,” Accident Analysis & Prevention, vol. 157, p. 106157, 2021.

[11] L. Westhofen, C. Neurohr, T. Koopmann, M. Butz, B. Schütt, F. Utesch, B. Neurohr, C. Gutenkunst and E. Böde, “Criticality Metrics for Automated Driving: A Review and Suitability Analysis of the State of the Art,” Archives of Computational Methods in Engineering, vol. 30, p. 1–35, 2023.

[12] S. Shalev-Shwartz, S. Shammah and A. Shashua, On a Formal Model of Safe and Scalable Self-driving Cars, 2018.

[13] J. C. Hayward, “Near miss determination through use of a scale of danger (traffic records 384),” Highway Research Board, Washington, DC, 1972.

[14] K. Ozbay, H. Yang, B. Bartin and S. Mudigonda, “Derivation and Validation of New Simulation-Based Surrogate Safety Measure,” Transportation Research Record, vol. 2083, pp. 105-113, 2008.

[15] J. Eggert, “Predictive risk estimation for intelligent ADAS functions,” in 17th International IEEE Conference on Intelligent Transportation Systems (ITSC), 2014.

[16] F. Reway, A. Hoffmann, D. Wachtel, W. Huber, A. Knoll and E. Ribeiro, “Test Method for Measuring the Simulation-to-Reality Gap of Camera-based Object Detection Algorithms for Autonomous Driving,” October 2020.

[17] A. v. B. Georg Volk and O. Bringmann, “A Comprehensive Safety Metric toEvaluate Perception in Autonomous Systems,” 2006.

[18] K. &. B. R. &. G. J. &. M. D. &. S. P. Stiefelhagen, “The CLEAR 2006 evaluation,” 2006.

[19] W.-L. Zhang, K. Yang, Y.-T. Xin and T.-S. Zhao, “Multi-Object Tracking Algorithm for RGB-D Images Based on Asymmetric Dual Siamese Networks,” Sensors, vol. 20, p. 6745, November 2020.

[20] K. Bernardin and R. Stiefelhagen, “Evaluating Multiple Object Tracking Performance: The CLEAR MOT Metrics,” EURASIP Journal on Image and Video Processing, vol. 2008, p. 1–10, 2008.

[21] N. Bongiorno, O. Pellegrino, A. Stuiver and D. de Waard, “Cumulative lateral position: a new measure fordriver performance in curves,” 2022.

[22] M. M. Minderhoud and P. H. L. Bovy, “Extended time-to-collision measures for road traffic safety assessment,” Accident Analysis & Prevention, vol. 33, pp. 89-97, 2001.

[23] B. L. Allen, B. T. Shin and P. J. Cooper, “ANALYSIS OF TRAFFIC CONFLICTS AND COLLISIONS,” Transportation Research Record, 1978.

[24] L. Zheng and T. Sayed, “Comparison of Traffic Conflict Indicators for Crash Estimation using Peak Over Threshold Approach,” Transportation Research Record: Journal of the Transportation Research Board, vol. 2673, p. 493–502, April 2019.

[25] H. Guo, X. Kun and M. Keyvan-Ekbatani, “Modeling driver's evasive behavior during safety–critical lane changes: Two-dimensional time-to-collision and deep reinforcement learning,” Accident Analysis & Prevention, p. 107063, 2023.

[26] C.-G. Wallman and H. Åström, “Friction measurement methods and the correlation between road friction and traffic safety : A literature review,” 2001.

[27] A. Novikov, I. Novikov and A. Shevtsova, “Study of the impact of type and condition of the road surface on parameters of signalized intersection,” Transportation Research Procedia, vol. 36, pp. 548-555, 2018.

[28] Y. Poledna, “THI dataset - Proving Ground Data,” 2024. [Online]. Available: https://hhse.sharepoint.com/:u:/r/sites/ite-aware_intelligent_systems_roadview-project/Shared%20Documents/WP7/Task%207.2%20Relevant%20metrics%20for%20performance%20assessment/TWICE_CBLA_SMOKE_Correct_position.zip?csf=1&web=1&e=9zaK98.

[29] Y. Poledna, “THI dataset - Simulated Data,” 2024. [Online]. Available: https://hhse.sharepoint.com/:u:/r/sites/ite-aware_intelligent_systems_roadview-project/Shared%20Documents/WP7/Task%207.2%20Relevant%20metrics%20for%20performance%20assessment/TWICE_CBLA_SMOKE_SIMULATED.zip?csf=1&web=1&e=KZqq9g.

[30] S. Cafiso, A. G. Garcia, R. Cavarra and M. R. Rojas, “Crosswalk safety evaluation using a pedestrian risk index as traffic conflict measure,” in 3rd International Conference on Road safety and Simulation, 2011.

[31] W. Wachenfeld, P. Junietz, R. Wenzel and H. Winner, “The worst-time-to-collision metric for situation identification,” June 2016.

[32] T. Verster, “Standard Operation Procedures for conducting the on-the-road driving test, and measurement of the standard deviation of lateral position (SDLP),” in International journal of general medicine. 4. 359-71. 10.2147/IJGM.S19639., 2011.

[33] C. traditional image quality metrics and D. N. N.-b. object detection a case study with, “JOURNAL OF LATEX CLASS FILES, VOL. 14, NO. 8, AUGUST,” 2021.

[34] Q. Song, K. Tan, P. Runeson and S. Persson, “Critical Scenario Identification for Realistic Testing of Autonomous Driving Systems,” April 2023.

[35] M. Saffarzadeh, N. Nadimi, S. Naseralavi and A. R. Mamdoohi, “A general formulation for time-to-collision safety indicator,” 2743.

[36] B. Robustness, in Object, D. Autonomous and D. when, “See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication,” 3345.

[37] F. Reway, W. Huber and E. P. Ribeiro, “Test Methodology for Vision-Based ADAS Algorithms with an Automotive Camera-in-the-Loop,” September 2018.

[38] S. Mammar, S. Glaser and M. Netto, “Time to line crossing for lane departure avoidance: a theoretical study and an experimental setting,” 2010.

[39] S. S. F. X. X. Z. Z. L. Jingwei Cao and J. Marcelo H. Ang, “Robust Object Tracking Algorithm for Autonomous Vehicles in Complex Scenes,” 2021.

[40] Y. A. N. G. Hong, K. OZBAY and B. BARTIN, “Application of Simulation-Based Traffic Conflict Analysis for Highway Safety Evaluation,” 2010.

[41] H. R. et al., “Generalized Intersection over Union: A Metric and A Loss for Bounding Box Regression,” 1902.

[42] A. Eidehall, “Multi-Target Threat Assessment for Automotive Applications,” in 2011 14th International IEEE Conference on Intelligent Transportation Systems Washington, DC, USA. October 5-7, 2011, 2011.

[43] M. F. Drechsler, G. Seifert, J. Peintner, F. Reway, A. Riener and W. Huber, “How Simulation based Test Methods will substitute the Proving Ground Testing?,” June 2022.

[44] C. Comparison, “A Framework for Quantitative Dissimilarity Measurement in Point,” 1004.

[45] A. Badithela, T. Wongpiromsarn and R. M. Murray, “Evaluation Metrics for Object Detection for Autonomous Systems,” in Submitted, 2023 International Conference on Robotics and Automation (ICRA) http://cds.caltech.edu/ murray/preprints/bwm23-icra_s.pdf.

[46] L. &. H. M. &. H. A. Mahmud, “Application of proximal surrogate indicators for safety evaluation: A review of recent developments and research needs,” 2017.

[47] J. Eggert, “Predictive Risk Estimation for Intelligent ADAS Functions,” 2691.