PRISSMA Project
Plateforme de Recherche et d'Investissement pour la Sûreté
et la Sécurité de la Mobilité Autonome
04/2021 - 04/2024

[L1.5] TESTS AND AUDIT REQUIREMENTS - FINAL REPORT

EXIGENCES D'ESSAIS ET D'AUDITS

Main authors: Rémi Régnier (LNE), Julien Girard-Satabin (CEA), Alessandro Renzaglia (INRIA), Elodie Cheateauroux (Transpolis), Bertrand Leroy (Vedecom), Fouad Hadj Selem (Vedecom), Anne Kalouguine (LNE), Dominique Gruyer (UGE), Cédric Gava (Sphéréa), Christophe Bohn (IRT SystemX), Karla Quintero (IRT SystemX), Jean-Baptiste Horel (INRIA), Radu Mateescu (INRIA), Leo Maisonobe (STRMTG), Sio-Song Ieng (UGE), Mathis Lejosne (LNE), Alexis Savva (LNE), Wei Xu(UGE)

Keywords: Standards, Evaluation, AI systems, Validation requirements, Evaluation protocol, Coverage, Performance, Robustness, Resilience, Traceability, Interpretability, Explainability, Scenario

Abstract. This deliverable aims to provide a set of recommendations and requirements for testing and auditing an autonomous mobility system in relation to the specificities of AI. Based on the ecosystem and ODDs elements from the WP8 information, evaluation and audit protocols will have to be developed, covering the creation of metrics as well as the implementation of coverage plans and the definition of AI-based system test facilities (constraints and costs). Particular attention will be paid to the definition of test scenarios. The recommendations should cover several critical aspects for the evaluation of AI such as performance, robustness, resilience, traceability, interpretability and explainability of responses, or testability of AI-based functions. The outputs of this task will feed into work packages 2, 3, 4, 5 and 6 of the project.

Résumé. Ce livrable a pour vocation de fournir un ensemble de recommandations et d'exigences d'essais et d'audits pour un système de mobilité autonome en relation avec les spécificités de l'IA. En se basant sur des éléments de l'écosystème et des ODDs issus des informations du WP8, des protocoles d'évaluation et d'audits devront être développés, recouvrant la création de métriques ainsi qu'une mise en œuvre des plans de couvertures et de définition des moyens de test des systèmes à base d'IA (contraintes et coûts). Une attention particulière sera accordée à la définition des scénarios de test. Les recommandations devront couvrir plusieurs aspects critiques pour l'évaluation de l'IA comme la performance, la robustesse, la résilience, la traçabilité, l'interprétabilité et l'explicabilité des réponses, ou la testabilité des fonctions à base d'IA. Les sorties de cette tâche viendront alimenter les WP 2, 3, 4, 5 et 6 du projet.

Contents

List of Figures

List of Tables

1 Introduction : purpose of the document

Artificial intelligence (AI) has seen major developments in recent years in many professional sectors and especially for the autonomous mobility systems. The levels of performance, robustness, ethics and explainability achieved by the various AI solutions have yet to be reliably demonstrated. End-users will thus have the guarantees that condition the acceptability of these technologies. They will be able to choose among different existing solutions thanks to objective and unambiguous common references, metrics and evaluation methodologies. The audit certification and evaluation protocol for a homologation process are intended to accompany this profound transformation of society by providing confidence in AI systems, in order to secure their use and promote their deployment.

This document is an update of deliverable 1.4 following feedback from the first POC phase of the PRISSMA project. The purpose of this document is to provide a set of recommendations and requirements for tests and audits for an autonomous mobility system in relation to the specificities of AI, taking into account the possible cooperative nature of its operating environment (augmented infrastructure with perception and classification capabilities, supervision, communication with the road infrastructure, vehicle-to-vehicle, etc.). In particular, a major phase of updating requirements has been carried out. These requirements may be defined directly from the regulations, the specific needs of the AI, in accordance with the automotive ecosystem or linked to the needs of other work packages and they may be described in an unambiguous and complete manner so that future users of the homologation are fully aware of the benefits and limitations of the functionality developed.

As in the previous deliverable, the first task will be to choose the different AI jobs that could lead to an approval process (understanding of the environment through classification algorithms for example, etc.). Then, for each of these jobs and based on the elements of the ecosystem and the ODDs resulting from the information provided in WP8, evaluation and auditing protocols will have to be developed, covering the creation of metrics as well as the implementation of coverage plans and the definition of means of testing AI-based systems (constraints and costs). Particular attention will be paid to the definition of test scenarios, in particular through the generation of scenarios integrating the criteria linked to risk analysis (occurrence, etc.). These recommendations should cover several critical aspects for the evaluation of AI such as performance, robustness, resilience, traceability, interpretability and explicability of responses, or testability of AI-based functions for both the design and certification phases. The outputs of this task will feed into work packages 2, 3, 4 and 6 of the project.

2 Missions and tasks assessed

2.1 Context

To achieve the goal of this document, the project shall determine the boundaries of its homologation related to the use of AI for autonomous mobility, considering the different levels of integration of smart technologies (from the smart sensor embedded on the mobile platform to the system of systems incorporating the infrastructure) in order to define and document its scope and the applicability of the requirements of audit and evaluation protocols.

The homologation perimeter must specify the evaluation activities covered, the mission and the AI subsystems of the autonomous mobility system on test, but also the distribution between the different types of verification (audit, open road test, controlled environment test and simulation) and provide a justification for any exclusion of applicability of the requirements of the future homologation.

Although the global task 1.2 is supposed to give generic recommendations for all the tasks that can be incorporated in an autonomous mobility system that will be covered by this second iteration of this deliverable and so this document will not stop at the more restricted perimeter of the autonomous shuttle and will try to broaden the subject to include all potential Level 4 vehicles.

2.2 Chosen mission

In the deliverable 1.4, the first use case of the autonomous shuttle, the system tested covers in principle the following area of operation:

- Urban
- Narrowing / narrow roads
- Ego speed range up to 50 (70) kph, in practice this could be less in POCS especially on open road (20 kph for example)
- Fluid and congested traffic conditions
- Roadway edges & markings : all possible in urban
- Signage : all traffic signs/road markings/traffic lights in urban
- Objects : all mobile objects in urban (non-classified/classified)
- Large/small static objects
- All weather (eg light/intense rain) & light (day/night) conditions

For this case study, we had to be able to cover the various levels of system decomposition considered:

- Autonomous vehicle system of systems whose aim is to enable the safety of road users (signalling, safety barriers), reduce traffic jams and allow the passage of emergency vehicles;
- Autonomous mobility system;
- Supervision system;
- Autonomous mobility system components especially captors.

The missions and the ODD of such a shuttle can be based on the ADS Tactical and Operational Maneuvers listed by the NHTSA. For example, an ODD and an OEDR have been defined (by the WP2, WP4 and WP8) in the framework of the Paris2Connect POC and can be used as an example for the outlines of the missions given to the AI but the deliverable must be more generic than its various instantiations of the POCS.

All the tasks listed in the Tactical and Operational Maneuver column of the table 1 were addressed by the first deliverable 1.4, it is up to each POC to limit itself with regard to what the technical solution can do, the route and the means of testing as illustrated on this page for the Paris2Connect case.

Each PoC of the first phase have listed the functional requirements related to the system under study (maximum speed, maximum deceleration, etc.) to finish the listing of the missions of the system under study.

For the update of deliverable 1.5, we will have to at least cover the case of the autonomous shuttle previously covered, but we must be able to cover the new applications of the second

Event	Response
Lead vehicle decelerating	Follow vehicle, decelerate, stop, change lane, pass
Lead vehicle stopped	Decelerate, stop, change lane, pass
Lead vehicle accelerating	Accelerate, follow vehicle
Vehicle changing lanes	Yield, decelerate, follow vehicle
Vehicle entering roadway	Yield, decelerate, follow vehicle
Vehicle cutting out	Accelerate, decelerate, stop
Opposing vehicle encroaching	Decelerate, stop, change lane
Adjacent vehicle encroaching	Yield, decelerate, follow vehicle
Lead vehicle cutting out	Accelerate, decelerate, stop
Lead vehicle parking	Decelerate, stop, change lane, pass
Pedestrian crossing road	Yield, decelerate, stop
Pedalcyclist riding in lane	Yield, follow, change lane, pass
Pedalcyclist riding in adj. lane	Yield, decelerate
Pedalcyclist crossing road	Yield, decelerate, stop
Speed limit sign	Accelerate, decelerate
Access restriction	Stop, transition to MRC
Operating outside of ODD	Transition to MRC (fallback-ready user or ADS)

Figure 1: OEDR for the Paris2Connect POC

Technical and Operational Maneuver	Covered	remarks
Parking	Out of ODD	
Maintain Speed	Yes	
Car following	Yes	
Lane centering	Yes	
Lane switching/overtaking	Limited	limited lane switching on track. Overtaking not in PoC scope
Enhancing Conspicuity	Yes	
Obstacle Avoidance	Yes	
Low-Speed Merge	Out of ODD	No merge ramps in test track
High-Speed Merge	Out of ODD	No merge ramps in test track
Navigate On/Off Ramps	Out of ODD	No merge ramps in test track
Right-of-Way decisions	Yes	Limited number of right-of-way on track
Navigate Roundabouts	Out of ODD	No roundabout in test track
Navigate Intersection	Yes	
Navigate Crosswalk	Yes	
Navigate Working Zone	Out of ODD	
N-Point Turn	Out of ODD	
U-Turn	Out of ODD	No U-Turn in test track
Route planning	Limited	limited to Test track Route Planning

Table 1: ODD for the Paris2Connect POC

phase of the POC, such as the autonomous droid (we would move from a predefined route to a predefined area of use, for example, introducing greater variability), and in absolute terms try to be generalizable to all possible level 4 autonomy applications at least for our requirements.

3 AI requirements/criteria and missions

Appendix B contains the full document setting out the requirements specifically adopted by the PRISSMA project.

3.1 Regulatory

As explained earlier in this document, PRISSMA focuses on the evaluation and validation process of road automated transport systems including functions based on AI. These transport systems are to be treated as systems of systems and the vehicle is one of the systems to be evaluated and approved. In the field of L4 automated vehicle type-approval, 2 regulations can be analysed in this report:

- The European Regulation for Automated Driving systems type-approval, also called UE ADS (2022-1426) [1]
- The French project of regulation for the Automated urban shuttle type-approval (NAVURBAUT) [2]

Moreover, the French legal framework for the safety validation and legal authorisations of the operation of road automated transport system is presented in the section.

The description of each of those regulations is not exhaustive. To get all the information and requirements, it is necessary to browse the documents.

3.1.1 EU ADS 2022-1426 [1]

The European commission has published in august 2022 a regulation defining the type-approval procedures and technical specifications for Automated driving system (ADS) of fully automated vehicles. The scope of this regulation is defined in Article 1. This Regulation applies to the type-approval of fully automated vehicles of category M[1] and N[2], with regard to their automated driving system, for the following use cases:

(a) Fully automated vehicles, including dual mode vehicles, designed and constructed for the carriage of passengers or carriage of goods on a predefined area in an urban or suburban environment.

(b) 'Hub-to-hub': fully automated vehicles, including dual mode vehicles, designed and constructed for the carriage of passengers or carriage of goods on a predefined route with fixed start and end points of a journey/trip and which may include urban or suburban or motorway environment.

(c) 'Automated valet parking': dual mode vehicles with a fully automated driving mode for parking applications within predefined parking facilities. The system may use or not external infrastructure (e.g. localisation markers, perception sensors, etc.) of the parking facility to perform the dynamic driving task.

It means that this regulation will be used for type approval of automated shuttles operating with a level 4 of automation. All the technical information, that are relevant for PRISSMA project are presented in the Annexes of this regulation. In Annex II of the EU ADS 2022-1426 regulation, the regulation specifies the ADS performance requirements. These requirements are listed in 12 paragraphs:

1. Dynamic Driving Task (DDT) under nominal traffic scenarios
2. DDT under critical traffic scenarios (emergency operation).
3. DDT at ODD boundaries
4. DDT under failure scenarios
5. Minimal risk manoeuvre (MRM) and Minimal risk Condition (MRC)
6. Human machine interaction for vehicles transporting vehicle occupants
7. Functional and operational safety
8. Cyber security and software updates
9. ADS data requirements and specific data elements for event data recorder for fully automated vehicles
10. Manual driving mode
11. Operating manual
12. Provisions for periodic roadworthiness tests

[1] Vehicles having at least four wheels and used for the carriage of passengers
[2] Power-driven vehicles having at least four wheels and used for the carriage of goods

All the requirements specified shall be taken into account in PRISSMA work as they can be linked to an AI functionality. It is important to note that none of these requirements refers to a possible AI component in the ADS.

Annex III of the EU ADS 2022-1426 regulation presents the compliance assessment. The overall compliance assessment of the ADS is based on:

- Part 1: The consideration of the most relevant scenarios for the ODD
- Part 2: The assessment of the ADS design concept and the audit of the manufacturer safety management system
- Part 3: The tests of the most relevant traffic scenarios
- Part 4: The credibility assessment for using virtual toolchain to validate ADS
- Part 5: The in-service reporting to demonstrate the safety performance in the field

Any requirement in Annex II (of the EU ADS 2022-1426 regulation) may be checked by means of tests performed by the type-approval authority (or its technical service).

3.1.1.1 Part 1: The consideration of the most relevant scenarios for the ODD

A first set of basic functional scenarios is presented and safety requirements for each scenario are presented. The basic scenarios are:

- Lane change manoeuvre
- Turning and crossing scenario
- Emergency manoeuvre scenarios
- Motorway entry
- Motorway exit
- Passing a toll station
- Operation on other road types than motorways
- Parameters to be used for Automated valet parking

Then, principles to derive scenarios from the ODD analysis are presented in the appendix 1 of part 1 of Annex III (of the EU ADS 2022-1426 regulation). These principles are further investigated in the section 4.2.

3.1.1.2 Part 2: The assessment of the ADS design concept and the audit of the manufacturer safety management system

This part of the regulation presents the documentation and the information that the manufacturer have to present to the type-approval authority or its delegated technical service. The type-approval authority or the technical service shall verify through audit targeted spot checks and tests all the safety management and concepts presented by the manufacturer.

The type-approval authority shall assess the documentation package which shall show that the ADS:

(a) is designed and was developed to operate in such a way that it is free from unreasonable risks for a vehicle occupants and other road users within the declared ODD and boundaries;

(b) fulfils the performance requirements of Annex II of this Regulation;

(c) was developed according to the development process/method declared by the manufacturer.

If the ADS contains any embedded AI algorithm, its functioning, implementation principles and all its safety management and verification shall be included in the documentation that will be audited. For example, paragraph 3.5.2 of the EU ADS 2022-1426, the regulation states : "In respect of software employed in the ADS, the outline architecture shall be explained and the design methods and tools used shall be identified. The manufacturer shall show evidence of the means by which they determined the realisation of the ADS logic, during the design and development process."

The requirements and information included in this part of the regulation shall be taken into account by PRISSMA WP6.

Moreover, taking into account the results of the analysis of the manufacturer's documentation package, the type-approval authority shall request the tests to be performed or witnessed by the Technical Service to check specific points arising from the assessment.

3.1.1.3 Part 3: The tests of the most relevant traffic scenarios

The test program is defined by the type approval authority to cover parameter variations inside the system boundaries and its ODD as declared by the manufacturer.

Type-approval testing may be carried out on the basis of simulations (WP2), manoeuvres on the test track (WP3) and driving tests on real road traffic (WP4). However, it may not be based solely on computer simulations and at the time of type-approval, the type-approval authority shall conduct or shall witness at least the following tests to assess the behaviour of the ADS.

A minimum list of basic scenarios or manoeuvres that shall be tested according to the ODD is presented:

1. Lane keeping
2. Lane changing manoeuvre
3. Response to different road geometries
4. Response to national traffic rules and road infrastructure
5. Collision avoidance
6. Avoid emergency braking before a passable object in the lane
7. Following a lead vehicle
8. Lane change of another vehicle into lane (cut-in)
9. Stationary obstacle after lane change of the lead vehicle (cut-out)
10. Parking
11. Navigating in a parking facility
12. Specific scenarios for motorway

13. For dual mode vehicles, transition between the manual driving mode and the fully automated mode

Tests scenarios to assess the performance of the ADS on a test track are points 1, 2, 5, 6, 7, 8, 9 (WP3) and on-road are 3, 4, 10 (WP4).

This list of scenarios shall be tested with at least the variations of these parameters if relevant with the ODD of the ADS :

1. different speed limit signs, so that the ADS has to change its speed according to the indicated values;
2. signal lights and/or stop instructed by a road safety officer / enforcement agents with situations of going straight, turning left and right;
3. pedestrian and cyclist crossings with and without pedestrians/cyclist approaching / on the road.
4. temporary modifications: e.g., road maintenance operations indicated by traffic signs, cones and other signalisation, access restrictions.
5. motorway entry, exit and toll stations.
6. without a lead vehicle;
7. with a passenger car target as well as a PTW target as the lead vehicle / other vehicle.

At the request of the type-approval authority, additional scenarios that are part of the ODD can be executed. If a scenario described in the previous points does not belong to the ODD of the vehicle, it shall not be taken into consideration.

3.1.1.4 Part 4: The credibility assessment for using virtual toolchain to validate ADS

The credibility can be achieved by investigating and assessing five properties of Modelling and Simulation (M&S):

(a) Capability – what can the M&S do, and what the risks are associated with it;

(b) Accuracy – how well does M&S reproduce the target data;

(c) Correctness – how sound & robust are M&S data and algorithms;

(d) Usability – what training and experience is needed;

(e) Fit for purpose – how suitable is the M&S for the ODD and ADS assessment.

The manufacturer shall produce the credibility assessment framework (CAF); This document will be investigated by the type-approval authority. The CAF provides a general description of the main aspects considered for assessing the credibility of an M&S solution together with principles on the role of third parties assessors in the validation process with respect to credibility.

The main CAF components and their relationship are presented on Figure 2.

The CAF and all the requirements stated in this part of Annex III shall be taken into account in the work of PRISSMA WP2.5.

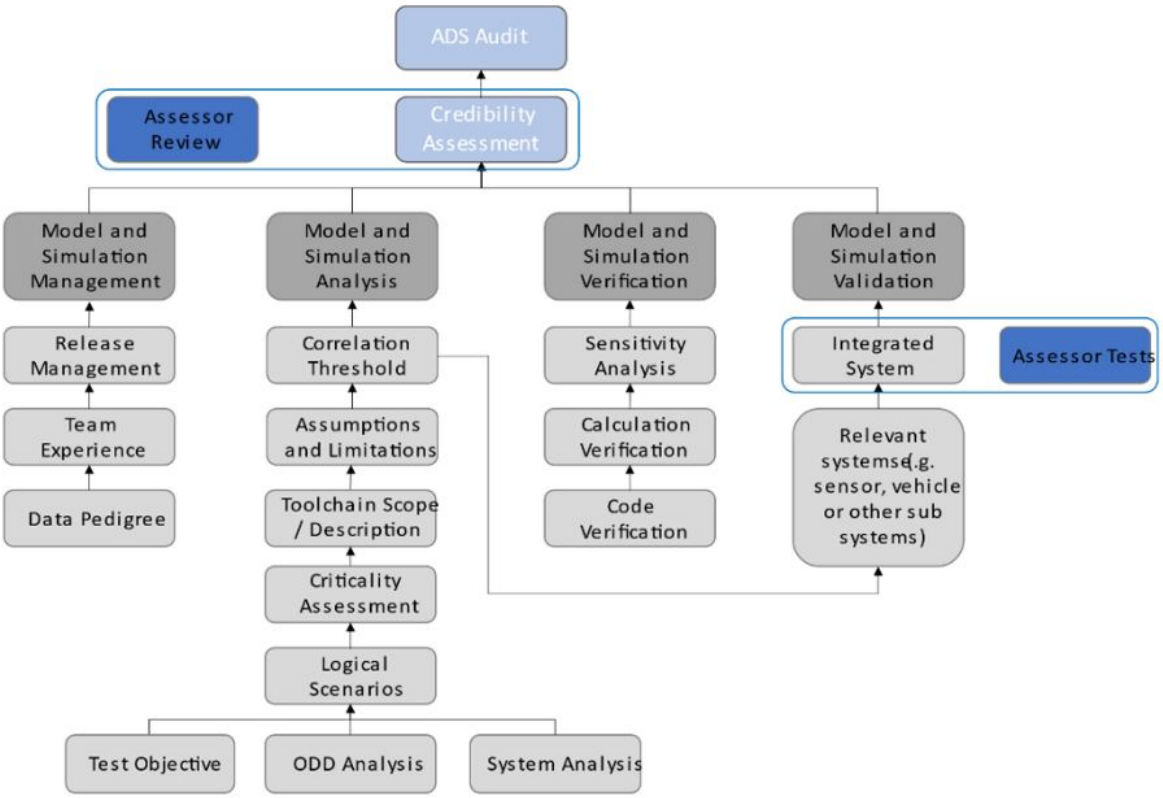


Figure 2: Graphical representation of the relationship between the components of the credibility assessment framework to assess the M&S [1].

3.1.1.5 Part 5: The in-service reporting to demonstrate the safety performance in the field

The manufacturer immediately notifies safety critical occurrences to the type-approval authorities, market surveillance authorities and the Commission.

The manufacturer shall report within one month any short-term occurrences, as described in Table, which needs to be remedied by the manufacturer to the type-approval authorities, market surveillance authorities and the Commission.

The manufacturer shall report every year to the type-approval authority that granted the approval on the occurrences listed in Table 2. The report shall provide evidence of the ADS performance on safety relevant occurrences in the field. In particular, it shall demonstrate that:

(a) no inconsistencies are detected compared to the ADS safety performance assessed prior to market introduction;

(b the ADS respects the performance requirements set by this Regulation;

(c) any newly discovered significant ADS safety performance issues have been adequately addressed and how.

All the requirements defined in part of Annex III shall be taken into account in the work of PRISSMA WP7.

3.1.2 The French automated urban shuttle regulation [2]

An urban shuttle is defined in the article R311-1 of the french highway code such as a motor vehicle designed and built for the transport of people in built-up areas, not meeting the definitions of international categories M1, M2 or M3 and having the capacity to transport, in addition to the driver, at least nine passengers and at most sixteen passengers , four or five of which may be seated. Accordingly, this type of shuttle is framed by French national regulations only. The automated urban shuttle is an highly or a fully automated vehicle as defined in the decree n° 2021-873 - 29 juin 2021 – related to penal responsibilities for automated vehicles and the condition of use (see the subsection below) which can be integrated in an automated road transport system.

In annex I of the regulation, technical specifications are given. Most of the technical specifications refers to the UN regulations setting requirements for type-approval of international categories of vehicles. For instance, on functional safety, cybersecurity and software evolutions, the text refers to the UN regulations respectively UN R 157 (Annex 4), UN R 155 and UN R 156. In addition to the annex, 9 appendices document the requirements on specific aspects such as:

- REQUIREMENTS CONCERNING THE APPROVAL OF VEHICLES AS REGARDS TO STEERING EQUIPMENT ;
- REQUIREMENTS RELATING TO THE APPROVAL OF VEHICLES AS REGARDS TO HORN AND SOUND SIGNALS;
- REQUIREMENTS RELATING TO THE APPROVAL OF VEHICLES AS REGARDS TO BRAKING VEHICLES AND THEIR TRAILERS;
- REQUIREMENTS RELATING TO THE APPROVAL OF VEHICLES AS REGARDS TO INTERIOR LAYOUT AND ACCESSIBILITY OF SHUTTLES;

OCCURRENCE	SHORT-TERM REPORTING (1 month	PERIODIC REPORTING (1 year)
1. Occurrences related to the ADS performance of the DDT, such as		
1.a. Safety critical occurrences known to the ADS manufacturer or OEM	X	X
1.b. Occurrences related to ADS operation outside its ODD	X	X
1.c. Occurrences related to ADS failure to achieve a minimal risk condition when necessary	X	X
1.d. Communication-related occurrences (where connectivity is relevant to the ADS safety concept)		X
1.e. Cybersecurity-related occurrences		X
1.f. Interaction with remote operator (if applicable) related to major ADS or vehicle failures		X
2. Occurrences related to ADS interaction with fully automated vehicle users, such as:		
2.a. User-related occurrences (e.g. user errors, misuse, misuse prevention)		X
3. Occurrences related to ADS technical conditions, including maintenance and repair:		
3.a. Occurrences related ADS failure resulting in a request to intervene to the operator or the remote intervention operator		X
3.b. Maintenance and repair problems		X
3.c. Occurrences related to unauthorised modifications (i.e. tampering)		X
4. Occurrences related to the identification of new safety-relevant scenarios	X (if modifications made by manufacturer to address a newly identified and significant ADS safety issue involving an unreasonable risk, including description of any previously unanticipated scenarios.)	X

Table 2: List of occurrences for in-service reporting [1].

- REQUIREMENTS RELATING TO THE APPROVAL OF VEHICLES AS REGARDS TO SMOOTH DRIVING
- REQUIREMENTS RELATING TO THE APPROVAL OF VEHICLES AS REGARDS TO LIGHTING
- REQUIREMENTS RELATING TO THE APPROVAL OF VEHICLES AS REGARDS TO THE SYSTEM SAFETY AND SAFETY IN DEGRADED MODE;
- REQUIREMENTS RELATING TO THE APPROVAL OF VEHICLES AS REGARDS TO THE DETECTION OF PRIORITY VEHICLES AND RESPONSE TO LAW FORCES;
- REQUIREMENTS RELATING TO THE APPROVAL OF VEHICLES AS REGARDS TO CIRCULATION RULES.

To make the link with AI, these requirements are decorrelated from any artificial intelligence brick but can have indirect impacts on the configuration of the latter. These requirements shall all be taken into consideration by PRISSMA for AI design and development if relevant to an AI brick in the system.

In annex II of this reglementation, specifications related to the operator and the circulation of urban shuttles are given.

Those specifications are not directly linked to AI functions.

3.1.3 French legal and technical framework for road automated transport systems

About the safety validation of complete road automated transport systems, the French government has published a legal framework with the Decret n° 2021-873 - 29 juin 2021 – related to penal responsibilities for automated vehicles and the condition of use [18]. The decree defines terms for various automation levels and focuses on a legal framework to authorise the deployment of road automated transport system on a predefined route or zone with level 4 vehicle and without safety driver on board. It is important to notice that this framework focuses on the whole transport system including the vehicle fleet, infrastructure equipments and configuration, the route or zone layouts, off-board devices, the supervision system, the operation systems, the rules for operating or maintaining the system, the communication and location equipments, etc.

The decree covers all the phases of the life cycle from design to operation. To operate an automated road transport system (ARTS), 4 safety validation stages have to be gone through:

1. The vehicle is to be type-approved according to the European [2] or the French [3] regulation. This validation is a prerequisite for the following safety validations.
2. The designer of the technical system shall demonstrate its safety through a technical system design case (dossier de conception du système technique DCST in French). The technical system is designed for a type of routes / zones and not a specific path.
3. The transport service organiser builds a preliminary safety case (DPS). At this stage, the technical system will be deployed on a specific path or zone. The case presents the test program and the operator safety management system project that will validate and rule the system.
4. The transport service organiser builds a safety case (DS). It is a complete presentation of all the tests and evaluations done to build the safety of the system and their results. It includes also complete presentation of the safety management system.

The commissioning of an Automated Road Transport System requires a decision taken by the service organiser on the basis of the three cases (DCST, DPS, DS) quoted above with the positive reviews by the qualified bodies (OQA, see below).

Type-approval of the vehicle is validated by a technical service of a state member and a Type Approval Authority. The proof of type-approval including the type-approval certificate shall be included in the DCST at stage 2.

The decree states that the cases from stage 2 to 4 will be reviewed and validated by qualified bodies (organismes qualifiés agréés, OQA, in French). The qualified bodies shall provide a review on each case (DCST, DPS and DS). These qualified bodies are qualified by STRMTG (national technical service in charge of safety for ropeways and guided transport) for 7 technical domains:

1. Functional safety of embedded systems
2. Functional safety of connectivity and positioning devices
3. Cybersecurity
4. Infrastructures and road equipment safety
5. Safety of road behaviour of the vehicles
6. Safety management system for operational stage
7. Global evaluation of the system safety.

OQA shall be present during tests preceding commissioning. The STRMTG or the service organiser may prescribe tests before commissioning, in addition to the safety demonstration. Those tests may concern AI.

For an AI algorithm embedded in the vehicle, two cases can happen:

1. The validation of this AI brick embedded in the vehicle is reviewed during the type-approval i.e. stage 1.
2. The validation of this AI brick embedded in the vehicle is not reviewed during the type-approval i.e. stage 1.

In the first case, the OQA of the first domain (Functional safety of embedded systems) shall take into account the evaluations done during the type-approval process.Then, the OQA of the technical domain 1 has for mission to assess:

- the safety level compatibility of the embedded system by analysis of the safety requirements;
- the identification of the safety requirements exported to the operation, the path / zone infrastructure arrangement, and other equipment;
- at stage 3 and 4, the solutions exported toward the maintenance.

In the second case, the domain 1 OQA shall assess:

- the relevance and the comprehensiveness of the safety demonstration;
- the compliance of the embedded system with the safety requirements;
- the allocation of requirements to sub-systems and functions of the embedded systems;

- the identification of the safety requirements exported to the operation, the path / zone infrastructure arrangement, and other equipment;
- at stage 3 and 4, the solutions exported toward the maintenance.

The validation of an AI algorithm out of the vehicle (infrastructure equipments, supervision system for instance) shall be reviewed during stage 2, 3 and 4 by an OQA. The designated OQA depends on the system under consideration.

The mission of the different types of OQAs are described in a guide made by the STRMTG named in French ”Guide d’application relatif à la mission de l’organisme qualifié agréé ” (Application guide related to the mission of the qualified bodies). This guide is available on the STRMTG website (https://www.strmtg.developpement-durable.gouv.fr/guide-d-application-relatif-a-la-mission-de-l-a800.html).

The part of the safety demonstration involving AI may involve all the different technical domains.

For instance, if one or several requirements of an AI brick are exported towards other subsystems, equipments, components, then an OQA of the technical domain 7 shall check these requirements. This OQA has also the responsibility to consider and to process all the interface requirements between the various components of the system including the AI bricks, in particular through the evaluation of the relevance and exhaustiveness of the successive security analyses at the system level as well as the consideration of external risks to the system. In addition to that, the OQA technical domain 7 has the responsibility of looking after the coordination of the other OQAs at every stage.

An other example, if one or several requirements of an AI brick are exported towards organizational rules (operation or maintenance), then the OQA of the technical domain 6 who is in charge of the SMS for the operational stage shall assess :

- at stage 2, the identification, relevance and acceptability of the operation, care and maintenance principles exported to the operation and maintenance of the ARTS.
- at stages 3 and 4:
  - the effective consideration of the safety requirements identified during the various system development phases and exported to operation and maintenance in the operation and maintenance documentation;
  - the completeness of the provisions of the SMS ;
  - the principles and conditions for operating and maintaining the system;
  - the acceptability of the safety requirements identified during the development of the system and exported to operation and maintenance;
  - the clarity of the documentation formalising the SMS.

After commissioning of the vehicle, an annual audit is foreseen to control the SMS of the operator of the system. An OQA technical domain 6 shall perform this audit.

The annual external audit must in particular cover the following topics:

- Missions of the operator;

- Organisation of the operator, including the identification and management of operational safety documents;
- Operating rules and procedures, including those related to staff;
- Maintenance policy, rules and procedures;
- Organisation of feedback;
- Skills management, including those related to safety tasks;
- Permanent system of internal control and evaluation of the level of safety;
- Elements relating to quality, including document management;
- Relations with State services on the occasion of accidents and incidents in operation;
- Evolution of the SMS in the past year;
- Follow-up of the action plan (if any);
- Adequacy of the safety management system to the evolution of operational safety issues.

The OQA audit input data includes in particular the analysis of accidents and incidents occurring during the operation of the ARTS as well as the results of the operation of the ARTS since the previous audit.

Another mission of OQA is to analyse accidents and severe incidents on the basis of the analysis of the operator (article R. 3152-22 of the highway code) if the prefect represented by its technical service the STRMTG requires it.

The OQA mission is to analyse the accident report and issue a review on the relevance of the measures taken to prevent the recurrence of the accident.

This legal framework shall be taken into account in PRISSMA work and PRISSMA shall provide the public authorities and the OQAs with knowledge, tools or methods.

3.2 Dedicated to AI

The use of a AI functionality implies several new requirements to ensure safety or performance and new ethical, transparency, maintenance or explainability issues to name but a few. Therefore, new requirements that may go beyond standards and regulations can be designed to provide good practice guidance for design, evaluation and maintenance.

As part of the project, we drafted a long document on these requirements, which you will find in Appendix B of this deliverable, the appendix will be the reference document for WP2, WP3, WP4 and WP6.

Initial regulatory work is nevertheless underway at the European level with the AI ACT, which notably puts forward measures on the transparency of the algorithms used:

- High-risk AI systems shall be designed and developed in such a way to ensure that their operation is sufficiently transparent to enable users to interpret the system's output and use it appropriately. An appropriate type and degree of transparency shall be ensured, with a view to achieving compliance with the relevant obligations of the user;
- The logging capabilities shall ensure a level of traceability of the AI system's functioning throughout its lifecycle that is appropriate to the intended purpose of the system;

- High-risk AI systems shall be accompanied by instructions for use in an appropriate digital format or otherwise that include concise, complete, correct and clear information that is relevant, accessible and comprehensible to users;
- The measures shall enable the individuals to whom human oversight is assigned to do the following, as appropriate to the circumstances: (a) fully understand the capacities and limitations of the high-risk AI system and be able to duly monitor its operation, so that signs of anomalies, dysfunctions and unexpected performance can be detected and addressed as soon as possible; (b) remain aware of the possible tendency of automatically relying or over-relying on the output produced by a high-risk AI system ('automation bias'), in particular for high-risk AI systems used to provide information or recommendations for decisions to be taken by natural persons; (c) be able to correctly interpret the high-risk AI system's output, taking into account in particular the characteristics of the system and the interpretation tools and methods available.

These AI ACT transparency measures also aim to improve interpretability but are not sufficient to ensure full explicability of outputs. We can therefore add new requirements on the subject when it makes sense:

- During testing, the system must provide explanatory elements. These elements provided by the AI functionality following an automatic decision shall be justified with regard to the regulations, the contractual specifications and the criticality of the system. The explanations during the test shall be saved for a period of time depending on the criticality of the AI functionality, in particular for the purpose of a posteriori analysis, in case of an accident;
- In the design phase, the developer must be able to keep a trace (log or other) that can give explanations of the elements provided by the AI functionality following an automatic decision. The possible explanations will be saved for a period of time depending on the criticality of the AI functionality, in particular for a posteriori analysis purposes, in case of an accident.

To this, we can also add confidentiality or ethical requirements (compliance with the GDPR for example).

Now that these points have been addressed, the specificity of AI leads to further complications in ensuring minimum performance, safety, robustness or resilience. These parameters are also very much related to the type of AI under study, so a learning algorithm will not be evaluated in the same way as an expert system for example because it has its own particularities such as its learning database.

To meet all its specific requirements, the PRISSMA project has produced a comprehensive document listing all the requirements adopted for the project dedicated to AI. This part will be found in the appendix B.

3.3 Selected influencing factors

One possible way to identify influencing factors is to follow the SOTIF standard "ISO/DIS 21448 - Road vehicles — Safety Of The Intended Functionality" [REF]. This standard is not dedicated to AI systems exclusively. It addresses new functionalities on road vehicles for Advanced Driving Assistance Systems (ADAS) and Advanced Driving Systems (ADS) which could lead to accidents or safety critical issues without failure. However, such systems rely

on AI-based components to operate in an open driving area with computer vision sensors to interpret the vehicle environment. Therefore the SOTIF and AI are extremely linked in the autonomous vehicle evaluation.

The SOTIF process is focused on safety related issues which are the main subjects of certification. AI issues related to performance, quality of service and other not safety-related aspects are not taken into account. Classical system level evaluation (without AI) should be sufficient to evaluate these aspects.

The SOTIF process starts with a robust system regarding design, quality assurance and functional safety demonstration. Its goal is to identify if the system is robust enough in the SOTIF point of view. Three main phases allow expert judgement to accept the system's delivery for operational use.

- Step 1: Evaluation by analysis
- Step 2: Evaluation of known potential hazardous scenarios
- Step 3: Exploration and evaluation of unknown scenarios

For AI requirements identification only the first step is pertinent and will be described.
This first step "Evaluation by Analysis" is composed by 2 sub-steps:

- SOTIF related Hazard Identification and Risk Evaluation
- Identification and evaluation of functional insufficiencies and triggering conditions

3.3.1 SOTIF related Hazard Identification and Risk Evaluation

The objectives of this first sub-step is to identify the hazards and risks related to SOTIF at vehicle level and to define acceptance criteria to evaluate residual risks.

For Hazard Identification we can proceed from functional Safety hazardous scenarios and subsequently identify hazards in common with SOTIF.

This first hazardous scenario set will be completed with complimentary approaches such as analytical, experience-based, expert knowledge.

After identification of these scenarios, a hazard qualitative evaluation shall be performed. This evaluation criteria is the same as the one proposed at ISO-26262 without reaching the ASIL quotation.

In the aim of evaluation, acceptance criteria must be defined. These criteria allow defining when residual risks can be considered acceptable once functional modifications are applied. In addition, it supports the definition of a V&V strategy.

The risk is considered mitigated if one of the following situations is reached: scenario severity does not lead to harm or every driver can control the hazardous situation.

3.3.2 Identification and evaluation of functional insufficiencies and triggering conditions

The aim of this sub-step is to identify the functional insufficiencies and triggering conditions and to evaluate the associated system response.

The following definitions should be considered:

Triggering conditions are defined as specific conditions of a scenario that serve as initiators for a subsequent system reaction leading to hazardous behaviour. This means that in the absence of the triggering condition, the scenario is considered acceptable but in the presence of such it fails.

Functional insufficiencies are defined as performance limitations (of sensors) or insufficiency of specification.

A performance limitation is defined as a limitation of the technical capability leading to hazardous behaviour in combination with one or more triggering conditions.

An insufficiency of specification is defined as a specification, possibly incomplete, leading to hazardous behaviour in combination with one or more triggering conditions.

A mixed approach is used for identification of both triggering conditions and functional insufficiencies:

- From known triggering conditions, reveal new functional insufficiencies
- From known functional insufficiencies, identify new triggering conditions

As a result of this process a new consolidated list of triggering conditions is generated. Subsequently, the same process described in subsection 3.3.2 is applied to evaluate the new scenarios containing the new identified triggering conditions in order to identify if the new risk is acceptable or not.

If the risk is not acceptable in itself, system response to the new scenarios has to be evaluated from SOTIF point of view:

- If the system response is unacceptable, design needs to be improved
- If the system response is theoretically correct, next evaluation steps of the SOTIF process can be applied.

3.3.3 Synthesis on Influencing Factors Identified by SOTIF

The subsequent steps of the SOTIF process imply defining test campaigns on the basis of the identified risky scenarios. The triggering conditions and the functional insufficiencies are the main influencing factors regarding evaluation of AI-related scenarios.

The SOTIF process aims to provide acceptance to a specific system before exploitation. In this way, it validates a correct design. If not, the design team needs to improve the system performance regarding failed scenarios.

The newly modified system shall again be submitted to the entire SOTIF process.

[18]

4 Evaluation protocol

4.1 Choice of the test method

The GRVA, an UNECE working group, has recently published guidelines about new assessment and test methods for Automated Driving (NATM) [3]. This methodology focuses on the vehicle validation and it is based on 5 pillars and a scenario catalogue (See Figure 3).

1. Simulation/virtual testing,
2. Track testing
3. Real world testing
4. Audit/assessment
5. In-service monitoring and reporting

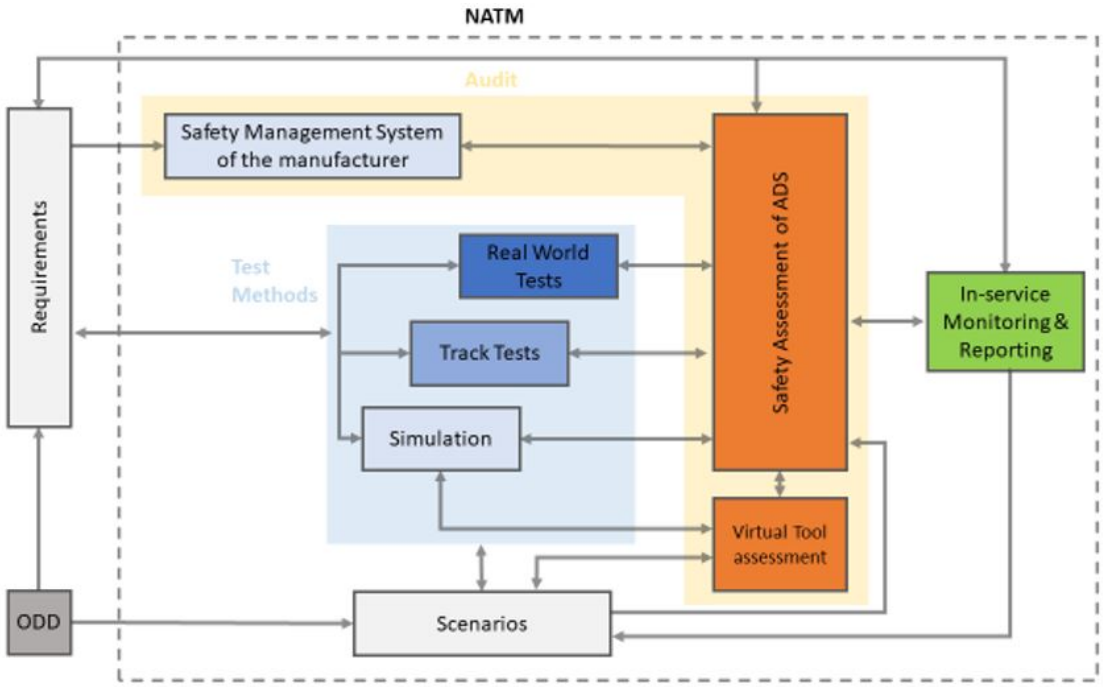


Figure 3: Diagram showing the relationship between the VMAD Pillars, Scenarios (NATM) and the Safety Requirements from FRAV [3].

The NATM master document presented to the 184^{th} World Forum for Harmonization of Vehicle Regulations [17] gives the strengths and the weaknesses of all the NATM pillars. Alongside the GRVA work, a methodological report by the DGITM is planned concerning an adaptation of the principles set in the NATM document for the Automated Road Transport Systems (ARTS). The vehicle-centric vision of the NATM document is extended to the ARTS. The document should soon be available on the DGITM website. This work is part of the deliverables to define how to use scenarios in the safety demonstration of ARTS. The following Tables 3, 4, 5, 6, 7 give an overview of all these weaknesses and strengths. Moreover, PRISSMA experts shall enrich this work. PRISSMA contributions are presented in red in the Tables. The quantities of tests of each pillar for a full validation shall be defined by the vehicle / system manufacturer. These tables provide valuable information for PRISSMA to recommend appropriate test programs for each pillar according to the IA type, or the sub-system or the whole system to be evaluated or validated.

For the PRISSMA project, we will therefore use a similar test protocol. In fact each part of the NATM method is covered by a PRISSMA WP :

Table 3: Strengths and Weaknesses of the Virtual Testing Pillar [17]

Strength	Weakness
Controllability – Virtual testing affords an unmatched ability to control many aspects of a test. **Agility** – Virtual tests allows for system changes to be reevaluated immediately. **Efficiency** – In MIL and SIL, virtual tests can be accelerated faster than real-time so that many tests can be run concurrently in a relatively short amount of time. **Cost effectiveness at test execution** – In spite of the investments required to develop, validate and maintain a virtual testing toolchain, the running costs connected to its use are considerably lower than those required by physical testing. **Wide scenario coverage** – Compared to other testing methods, virtual testing allows a wider exploration of safety-critical scenarios. By properly combining the experiments parameters it can for example reduce the space of the known unknowns and to the extent possible that of the unknown unknowns (including the effect of system failures).This scenario coverage enables also better analysis of the vehicle ODD and its boundaries. **Data gathering and analysis** - Virtual testing offers a convenient and error-free platform for data gathering and analysis of the ADS performance. Once Qualified, that data can serve as a significant contribution for assessing the risk from the ADS. Note: this advantage benefits to the complete automated transport system validation **Repeatability and replicability** – Simulation affords the re-execution of the same virtual test without deviations due to stochastic phenomena. Faults in the functioning of the ADS can thus be identically replicated at any moment.Note: this advantage benefits to the complete automated transport system validation **A deep OD analysis** - Before a L4 ADS deployment (on a specific route), modelling this route with the infrastructure, road equipment, lightning orientation, etc. can provide evidences of the efficiency of new road equipment or help urban planners to choose safer infrastructure for the future ADS services	**Lower environmental fidelity/reliability** – It is difficult, and likely impossible for models to completely reproduce the environment, responses, as well as the behaviour of the vehicle, other road users etc. in the real world. Also the validation process cannot prove the validity of the simulation across all possible scenarios. **Risk of over-reliance**. Without proper consideration of models' intrinsic limitations, a risk exists to put too much emphasis on virtual testing results without sufficient proof of their validity by physical testing. **Expensive software life-cycle**. The availability of a simulation model to execute virtual testing requires covering certain aspects of the software life-cycle which can be costly and time-consuming

Table 4: Strengths and Weaknesses of the Track Test Pillar [17] - PRISSMA additions in red

Strength	Weakness
Controllability – Track testing allows for control over many of the test elements, including certain aspects of the ODD. **Fidelity** – Track testing involves functional, physical ADS-equipped vehicles and lifelike obstacles and environmental conditions. **Reproducibility**– Track testing scenarios can be replicated in different locations by different testing entities. **Repeatability** – Track testing allows for multiple iterations of tests to be run in the same fashion, with the same inputs and initial conditions. **Efficiency** – Compared to real-world testing, closed-course testing can accelerate exposure to known rare events or safety critical scenarios by setting them up as explicitly designed test scenarios. Road testing by contrast could be an inefficient way to test less co manifesting by chance. Track testing can be used to **validate the quality of the simulation** toolchain by comparing an ADS' performance within a simulation test with its performance on a test track when executing the same scenario. **Testing system limits** - using appropriate test equipment with a efficient safety management, dangerous scenarios and system limits can be evaluated on tracks while it is not possible to gather those data with real world tests **VIL benefits** - VIL mixes virtual tests and real tests on tracks. It enables to assess dangerous scenarios more safely (high speed collision scenarios) or to test more complex scenarios with multiple targets for example.	**Significant time** –Track testing can take a significant amount of time to set up and execute. **Costly** – Track testing may require a substantial number of personnel and specialized test equipment (e.g., obstacle objects, measurement devices, safety driver). **Limited variability** – Track testing facility infrastructure and conditions may be difficult to modify to account for a wide variety of test elements (e.g., ODD conditions). They are restricted to their geometries, dimensions, size and ODD limitations such as weather conditions, time of day, number and type of other traffic agents. **Safety risks** – Track testing with physical vehicles and real obstacles presents a potentially uncertain and hazardous environment for the test participants (e.g., safety driver and experiment observers). But track safety management and appropriate test equipment minimise these risks **Representativeness** even with its increased fidelity. Whilst things like pedestrians can be included, these won't typically be real people due to safety reasons and the clutter or real-world environments cannot be replicated.

Table 5: Strengths and Weaknesses of the Real World Test Pillar [17]

Strength	Weakness
High environmental validity – allows for validation of the vehicle in its intended ODD(s) and the diverse conditions these may present. Can be used to **test scenarios elements**, such as weather and infrastructure (e.g., bridges, tunnels), that are **unavailable through track testing.** Real-world testing may be used to **validate the simulation and track-testing** by comparing an ADS' performance within a simulation and track test with its performance on in a real-world environment when executing the same scenario. Can be used to assess aspects of the ADS performance related to its **interaction with other road users**, e.g. maintaining flow of traffic, being considerate and courteous to other vehicles. **Model, single software, and toolchain validation**	**Restricted controllability** – Public-road scenarios afford a limited amount of control over ODD conditions. **Restricted reproducibility** – Public-road scenarios are difficult to replicate exactly in different locations. **Restricted repeatability** – Public-road scenarios are difficult to repeat exactly over multiple iterations. **Limited scalability** – Public-road scenarios may not scale up sufficiently. **Costly** but not as costly as track testing – Requires a number of resources and is **time-consuming**. Potential impact on **traffic** and safety authorities New competencies may need to be developed by **authorities** **Safety risks:** on-road testing could subject test personnel and the public to significant risks of unsafe behavior.

Table 6: Strengths and Weaknesses of the Audit Pillar. Table derived from [17]

Strength	Weakness
Partial maturity – Risk analysis, safety-by-design concepts as well verification/validation test methods are standard development methods used in the automotive industry for years to ensure functional safety of electronic system (fail safe). It is expected that similar methods will be followed by manufacturers to minimize unsafe and unknown scenarios for ADSs in a systematic manner (operational safety beyond failures). **Robustness** - Regarding the safety assessment, the tools under this pillar will provide a more robust demonstration on the ADS safety (coverage) than a few test runs. The manufacturer's safety case will be reinforced if it is assessed by an independent auditor and confirmed by targeted physical or virtual tests.	**Audit scope definition** - Test runs will in particular be needed to demonstrate that the vehicle exhibits minimum performances for standard manoeuvres (e.g. normal lane keeping, lane change), key critical scenarios (e.g. emergency braking) and in traffic conditions (e.g. smooth integration in the traffic). It remains to be decided at this stage whether these tests shall be standardized across manufacturers for some defined situations or shall be tailored to the results of the risk assessment/design of the ADS or both.

Table 7: Strengths and Weaknesses of the In service monitoring and reporting Pillar. Table derived from [17] See 89. To 94.

Strength	Weakness
Most Realistic – Data from the field will be the most realistic way to assess the safety performance of an ADS over a wide range of real driving traffic and environmental conditions. **Scenario Database update** - Data from the field are also instrumental to ensure that the scenario database is updated with the latest scenarios, in particular those deriving from the increasing use of ADS. **Learning from experience** - Regarding safety recommendations, learning from in-service data is a central component to the safety potential of ADSs. Lessons learned from a crash involving ADSs could lead to safety developments and subsequent prevention of that crash scenario in other ADS. Feedback from the operational experience is recognized as best practice for safety management in the automotive sector as well as in other transport sectors (e.g. already in place in aviation, railway and maritime sectors). Field operation data can also provide evidence of the positive impact of ADs on road safety.	**Limitations** might derive from the quantity of data to be handled (too much data is as problematic as too little data), availability of tools for automatic scenario generation, and identification of responsibility handlers. Therefore, the outcome shall be a proportionate, efficient and uniform system. Methods to verify the **reliability** of collected data should be developed. The data collected should be comparable amongst manufacturers. It will create challenges on which data and how these data are collected and reported (definition of suitable reporting criteria). Timewise, another challenge is the development of the in-service safety monitoring framework in a timely manner in order to serve AVs market deployment. Data privacy should also be taken into account. A standardized format for communication of information will be needed to allow processing by authorities in a standard manner and that any outcomes are easily shareable or open for analysis by other authorities. Different type of data may be needed depending on the purpose of the data collection. Processes for reporting the operational feedback from AVs should be developed for the automotive sector taking into account the higher number of monitored vehicles and events to be recorded.

- ”ODD” is covered by WP8
- ”Requirements” is covered by WP1 (see annex of this document)
- Test methods by simulation is covered by WP2
- Test methods by track tests is covered by WP3. WP3 extended to the test bench also.
- Test methods by real world tests is covered by WP4.
- Safety Management System of the manufacturer is covered by the audit in WP6
- service Monitoring and reporting is covered by WP7
- Virtual Tool assessment is covered by the tasks 2.5 and 1.3
- Scenarios is covered by the tasks 2.4 and 1.2 (see section 4.2 of this document)

This is why our protocol will be broadly based on that of NATM.

Having already focused on the requirements part applicable to the various tests, monitoring and auditing, this document will look at the problems linked to the scenarios part and the coverage problems before giving advice on the right test methods to apply. The control of tools (virtual or otherwise) will be covered in task 1.3.

4.2 Generation of scenarios and test cases

There is a consensus in the automotive industry that all AV test protocols should be based on a scenario approach. The first task of any protocol is therefore to describe how these scenarios are selected and constructed.

4.2.1 Introduction to the scenario approach

The objective of this section is to present the PRISSMA approach to generate scenarios and test cases to evaluate, validate and certify an AI. First, after hours of semantic discussion between the project participants, a paragraph giving clear definitions and a framework was found out to be necessary. Then, the context of the global scenario approach pushed by the legal frameworks is exposed and applications to different bricks of AI are studied. This scenario management is analysed at the functional and logical levels. Finally, the last paragraph is focused on the generation of concrete scenarios and test cases.

4.2.2 Definitions

PRISSMA as a French project will take as references the following sources:

- The NATM (ONU work) which addresses the vehicle equipped with ADS type approval [19],
- The European regulations [1] which addresses the vehicle equipped with ADS type approval,
- The French ministry deliverables ([20], [21],or [16], [22]) about ARTS safety demonstration.

The NATM Annex 1 of [19] gives this definition of **“Traffic scenario”** (or scenario for short) : a sequence or combination of situations used to assess the safety requirements for an ADS. Scenarios include a driving maneuver or sequence of driving maneuvers. Scenarios

can also involve a wide range of elements, such as some or all portions of the DDT; different roadway layouts; different types of road users and objects exhibiting static or diverse dynamic behaviours; and, diverse environmental conditions (among many other actors). **"Complex Scenarios"** means a traffic scenario containing one or more situations that involve a large number of other road users, unlikely road infrastructure, or abnormal geographic/environmental conditions.

These definitions do not give a clear and exact list of what must be included in a traffic scenario.

The EU ADS act gives to a scenario the exact same definition as the NATM paper.

The DGITM [3] gives this definition: **"scenario"**: Sequence of scenes and events and/or actions. A scenario is the temporal development of scenes. A scenario consists of at least an initial scene, an event or an action, and a final scene [21]. Moreover, in a later deliverable, DGITM describes the content of a scenario with 5 layers [16]:

- **Static environment:** road description,signs, etc.
- **EGO manoeuvre:** such as turn left, follows an other vehicle, etc.
- **Hazards:** collision precursory event or system failure, etc.
- **System response:** such as braking or avoiding an obstacle, etc.
- **Hazards affecting system response :** such as environmental conditions, masking of an obstacle, slippery road, etc.

According to the NATM [19], it exists 2 categories of scenarios:

- **"Nominal Scenarios"** means a traffic scenario containing situations that reflect regular and non-critical driving manoeuvres.
- **"Critical Scenarios"** means a traffic scenario containing a situation in which the ADS needs to perform an emergency maneuver in order to avoid/mitigate a potential collision, or react to a system failure.

The EU ADS act defines 3 categories of scenarios:

- **"Nominal traffic scenarios"** mean reasonably foreseeable situations encountered by the ADS when operating within its ODD. These scenarios represent the non-critical interactions of the ADS with other traffic participants and generate normal operation of the ADS.
- **"Critical scenarios"** mean scenarios related to edge-cases (e.g. unexpected conditions with an exceptionally low probability of occurrence) and operational insufficiencies, not limited to traffic conditions but also including environmental conditions (e.g. heavy rain or low sunlight glaring cameras), human factors, connectivity and miscommunication leading to emergency operation of the ADS.
- **"Failure scenarios"** mean the scenarios related to ADS and/or vehicle components failure which may lead to normal or emergency operation of the ADS depending on whether or not the minimum safety level is preserved.

[3] Direction Générale des Infrastructures, des Transports et des Mobilités - a department of the French ministry for Transport

It can be noticed that in the definition of critical scenarios in the NATM document, it includes the system failures. Furthermore, the definitions of EU ADS act are more precised compared to the NATM document.

Finally, the DGITM does not classify scenarios in categories but rather describes types of scenarios depending on the sources from where scenarios can be built. 4 sources of scenarios are listed in the [16]:

- **"Scenarios derived from design of the system"** representing scenarios compiled from ODD and OEDR description. It only consists of nominal scenarios as in EU ADS definition.
- **"Accident scenarios"** representing scenarios derived from in-depth analysis of physical and material accidents extracted from referenced databases.
- **"Scenarios derived from risks analysis"** representing scenarios of hazardous situations covering the reasonably foreseeable risks affecting the system. It represents all the scenarios of functional safety, i.e. failure scenarios as defined in the EU ADS act but also the scenarios derived from the SOTIF approach tackling critical scenarios.
- **"Scenarios derived from driving"** representing scenarios of hazardous situations still unknown. They can be identified by numerical and physical driving before commissioning and after commissioning with in-service monitoring.

These scenarios can be completed by scenarios from experts. This approach differs from the EU ADS act or the NATM document. Nevertheless, it remains consistent as the categories of scenarios defined in these texts are covered by the different sources.

Documents of the DGITM, the EU ADS act and the NATM document agree on the definition of 3 levels of detail of scenarios (see also 4):

- **"Functional Scenario"**: Scenarios with the highest level of abstraction, outlining the core concept of the scenario, such as a basic description of the ego vehicle's actions; the interactions of the ego vehicle with other road users and objects; and other elements that compose the scenario (e.g. environmental conditions etc.). This approach uses accessible language to describe the situation and its corresponding elements. For the scenario catalogue, such an accessible (i.e., natural and non-technical) language needs to be standardised to ensure common understanding between different ADS stakeholders about the scenarios.
- **"Logical Scenario"**: Building off the elements identified within the functional scenario, developers generate a logical scenario by selecting value ranges or probability distributions for each element within a scenario (e.g., the possible width of a lane in meters).
- **"Concrete Scenarios"**: Concrete scenarios are established by selecting specific values for each element. This step ensures that a specific test scenario is reproducible. In addition, for each logical scenario with continuous ranges, any number of concrete scenarios can be developed, helping to ensure a vehicle is exposed to a wide variety of situations.

These definitions are extracted from the NATM document. The EU ADS act does not define these levels of details and the levels of details in the DGITM document are equally defined as the NATM document.

The level of details of the scenarios is used depending on the step of the safety demonstration. For instance, functional scenarios are used at the beginning of the implementation of the

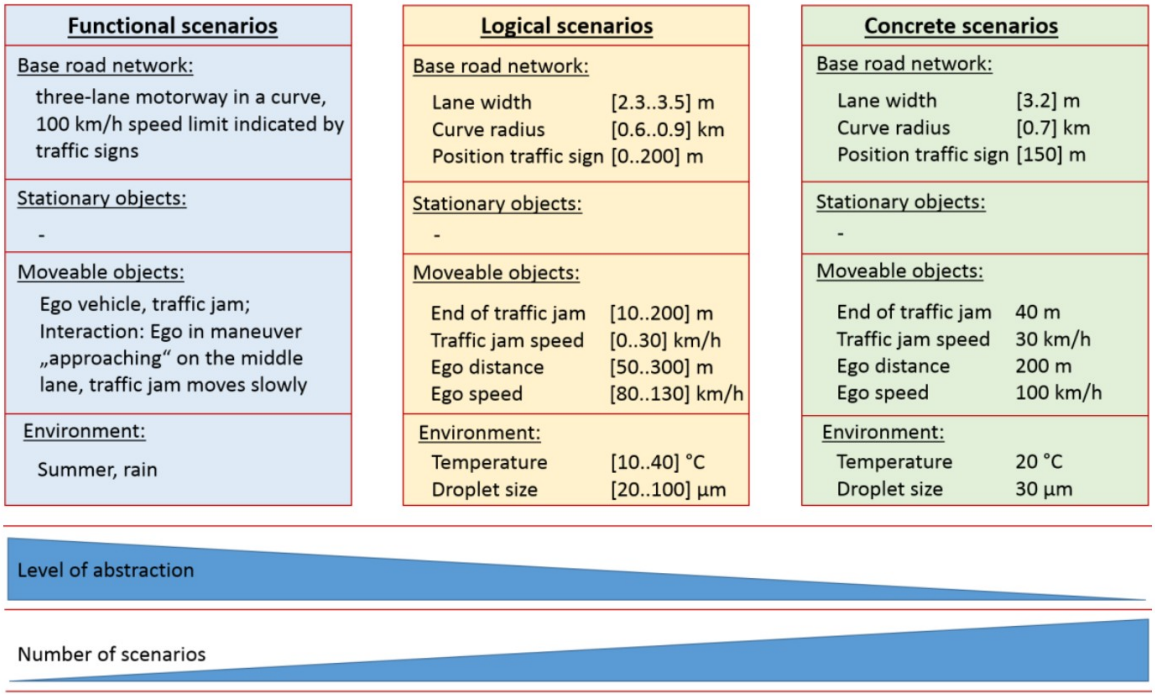


Figure 4: 3 levels of scenarios defined by NATM [3].

scenario approach. The closer we get to implementing the tests, the more precise the scenarios are, to the point of selecting certain concrete scenarios to cover the whole range of reasonably foreseeable scenarios for a specific system.

4.2.3 Scenarios for ADS and ARTS Validation: Regulation Scope

The general technical and legal frameworks of the safety validation of automated and autonomous vehicles and automated road transport systems (ARTS) are still under definition concerning the methodology the parties involved have to follow.

The EU ADS Act [1], published in 2022, is the regulation defining the methods for an ADS type approval. The automated driving system is embedded in the vehicle. The EU ADS act is focused on the automated vehicle approval.

In France, the DGITM [4] and the STRMTG [5] have launched several working groups (WG) to clarify the scenarios approach in the safety demonstration. The scope of these WG is focused on the ARTS, meaning the fleet of automated vehicles and all the systems around.

Note : Both the EU ADS act and the DGITM WGs take the NATM as the theoretical basis to be applied.

A first methodological report was released by the DGITM in February 2022 to present a theoretical approach of the use of scenarios to validate automated road transport systems [21]. It attempts to detail how scenarios feed off ODD, OEDR and pathway description. On the other hand, the STRMTG works on the concept of the GAME demonstration [23], “GAME” like Globally at least equivalent. This principle aims at defining safety levels and objectives that the

[4] Direction générale des infrastructures, des transports et des mobilités,(Branch of ministry of Transportation)
[5] French Technical service in charge of safety for ropeways and guided transports

ARTS shall reach. However, the links between the scenario approach and the GAME principle stay unclear. The DGITM deliverable [21] also characterises scenarios as a description in 5 layers (see the section Definitions above). Each description layer is illustrated with examples. At the end of the document, lists of scenarios without critical hazards are suggested.

The second document published by the DGITM : Scenario generation, supply, and enrichment [16] complements the first one with a reminder of certain definitions, a method of supplying scenarios and a process for scenarios enrichment. At the end of the methodological report, examples of the combinatory of scenarios are given. First, this report presents terms, basic concepts and definitions. All the definitions are consistent with those presented in PRISSMA L8.1 (Terminologie commune au projet PRISSMA). Then, the document details the method of scenario enrichment to adopt:

- A scenario definition by layers,
- The comparison of scenario descriptors with attributes, from different sources,
- The enrichment with new descriptors and attributes of these descriptors,
- The combination of these descriptors and attributes to enrich the initial list of scenarios.

Furthermore, the descriptors and the values of descriptors may be updated following feedback or the addition of a new use case for instance.

As explained in the 4.2.2, DGITM adopts another approach to classify scenarios compared to the EU ADS act and the NATM document. Indeed, scenarios are categorized by sources and not by types.

For comparison, the NATM [17] also presents a list of methods to gather scenarios:

(a) Analyzing human driver behaviour, including evaluating naturalistic driving data;

(b) Analyzing collision data, such as law enforcement and insurance companies' crash databases;

(c) Analyzing traffic patterns in specific ODD (e.g., by recording and analyzing road user behaviour at intersections);

(d) Analyzing data collected from ADS' sensors (e.g., accelerometer, camera, radar, and global positioning systems);

(e) Using specially configured measurement vehicle, onsite monitoring equipment, drone measurements, etc. for collecting various traffic data (including other road users);

(f) Knowledge/experience acquired during ADS development;

(g) Synthetically generated scenarios from key parameter variations;

(h) Engineered scenarios based on functional safety requirements and safety of intended functionality.

Most of the sources of scenarios overlap between the DGITM document and the NATM document. For instance, knowledge and experience scenarios (NATM) correspond to scenarios derived from design of the system as described in the deliverable of the DGITM (Scenario generation). On the contrary, it seems that synthetically generated scenarios from key parameter variations (NATM) are not covered by any of the DGITM sources of scenarios. At the end, PRISSMA shall take into account all of the sources of scenarios.

As quoted above, a third methodological document of the DGITM "Utilisation des scénarios pour la démonstration de la sécurité des STRA" (Use of the scenarios for safety demonstration of ARTS) [22] addresses the adaptation of the NATM principles to the ARTS.

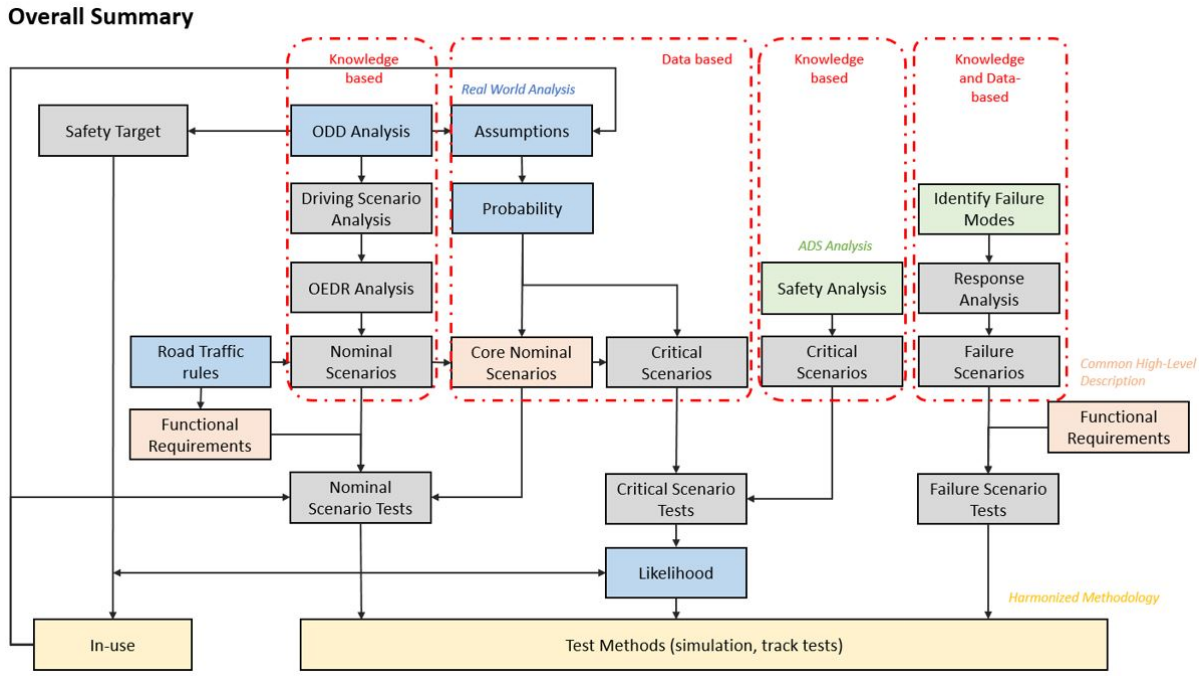


Figure 5: Scenario Framework specified by NATM [3] and [1]

All DGITM and STRMTG documents are focused on the safety demonstration of ARTS. However, the first step to build an ARTS is to plan the deployment of a Type-approved automated vehicle. As the EU ADS act [1] is the regulation that defines the approval methods, it is important for PRISSMA to take it into account. The EU ADS act in appendix 1 part 1 of annex III takes the global framework given in the NATM document to manage the different types of scenarios as presented in Figure 5. This approach is developed to validate the ADS with scenarios. It is important to notice that functional requirements play a major role in the scenario management shown in Figure 5. PRISSMA requirements are further developed in Annex B.

The following subsections 4.2.4 and 4.2.5 present the different scenarios shown on Figure 5.

4.2.4 Nominal and Critical Scenarios

A clear and clean decomposition of scenarios into 'Nominal' and 'Critical' has proved to be rather simplistic in the autonomous and automated road transport systems scope. The following statements illustrate this reasoning and are however crucial to grasp the importance of the nature and potential evolution of each scenario type:

- a scenario that is initially considered as nominal (i.e. no potential and immediate risk) can become a critical scenario given certain conditions.
- today, an OEM may consider a scenario as no longer 'nominal' but rather 'nominal on degraded conditions' in which the system is design to adopt a resilience functioning mode, distinct from the nominal one, in the light of risk prevention.
- a scenario can be considered as a 'near miss', and 'edge case' but not necessarily 'critical', this however does not make it 'nominal'.

- the level of criticality of a scenario is inevitably determined by the function being tested. It is pertinent, given the capabilities of the function or of the vehicle, to assess the possible parameter combinations that would result into a scenario that the vehicle can and should manage safely and to evaluate the potential repercussions of such actions in the actors of the environment.
- an OEM should have the liberty and the obligation to provide safety concepts and their documentation that is auditable so that according to the functions being tested and their attached ODD, evidence can be provided on the strategy used for testing and the assessment of scenarios that are considered as nominal and those which are not.
- some situations can be judged subjectively as critical or as nominal when the criticality is rather (subjectively) low and therefore represent a challenge when annotating datasets for AI developement and testing.

In this sense, to this day, it seems that trying to define in a first stage generic classes of scenarios as 'Nominal' or 'Critical' is not only a challenge but a non suitable solution for a complex problem. The frontier between these classes is rather blurry and scenarios may be rather be considered as 'nominal' or 'non-nominal' where the 'non-nominal' class encompasses a subset of critical and edge-cases scenarios.

In the light of the previous statements, it is not obvious to establish the frontier between types of scenarios: nominal scenarios (normal functioning mode), nominal on degraded conditions (subject to disturbances), and critical scenarios (where risk is present and the SUT's behavior is uncertain).

The frontier of this typology will depend on the design of the SUT and therefore on the audit performed on this design.

- **Nominal Scenarios:** can be covered by audit and light sampling tests in order to confirm what has been audited.
- **Nominal Scenarios on Degraded Conditions:** are tested through a more detailed sampling in order to verify that the transition are properly performed (DDT fallback) and that the frontiers between the nominal and nominal degraded mode are properly implemented.
- **Critical Scenarios:** are a key aspect to address due to the uncertainty of the system's response in the occurrence of a potential accident. In this context, among scenarios leading to potential accidents, it would be useful to identify those where the system can avoid the accident, those where the system cannot avoid it but can limit the consequences, and those where the system cannot avoid the accident. In this case, the testing approach should be the following:

  - In the case of scenarios of accidents that should be avoided by the system and that where its behavior is uncertain, a systematic test sampling shall be done.

  - In the case of scenarios where the accident cannot be avoided but the consequences are expected to be limited by the system with uncertainty in implementation, an expert test sampling shall be done.

  - In the case of scenarios where the accident cannot be avoided, no testing is proven to be useful.

A PRISSMA method to generate nominal and critical scenarios from ODD and OEDR analyses is presented in Annex A.

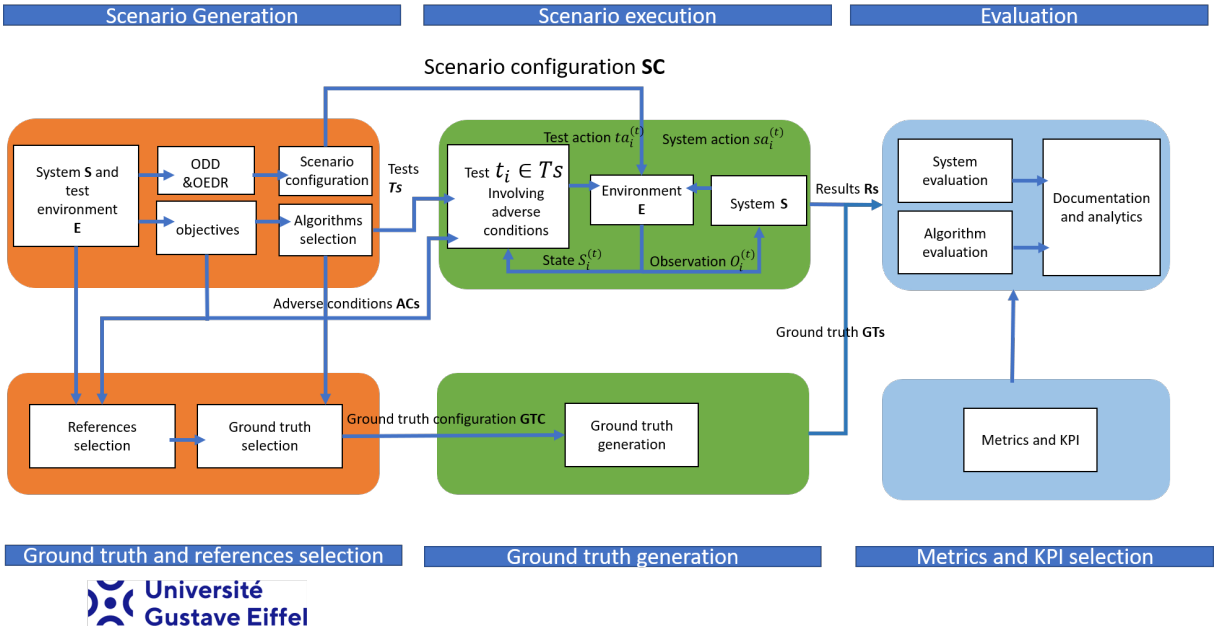


Figure 6: Generic Framework for evaluation of AI-powered systems in ADS

4.2.5 Failure Scenarios

These scenarios are related to ADS and/or vehicle components failure. They may lead to emergency maneuver or minimal risk maneuver. Knowledge about these scenarios may come from the same sources than other scenarios as explained in paragraph 4.2.3.

However, PRISSMA considers that the principles of the ISO26262 [24] shall be applied to the ADS to define a correct set of failure scenarios.

For instance, it can happen that the ADS finds itself operating outside of its ODD. For example, a automated shuttle that is not supposed to run under heavy rain can find itself under a rain shower. These kind of scenario can be considered as failure and it is important to include them in the scenario database to assess the adequate responses of the system.

4.2.6 Focus on the IA components

An AI-powered system in autonomous driving is a component that employs AI algorithms and techniques to carry out various functions critical for an automated vehicle in its operating environment. These functions include perception, localization, decision-making, path planning, control, and more. A (simulated) sensor suite is needed to enable the vehicle to sense its surroundings, including cameras, LiDAR, radar, etc.. This comprehensive sensor setup aims to contribute to a more realistic and complex data source, allowing a more accurate and reliable perception of the surrounding environment. By leveraging advanced AI techniques, autonomous vehicles can perceive, understand, and navigate their environment safely and efficiently. In order to evaluate the AI powered system, it is important to have a generic framework for testing the system with realistic scenarios in real and simulated conditions. The evaluation Framework proposed in [25] can be split into three modules: the scenario generation, the scenario execution and the evaluation as it is by the figure 6.

4.2.6.1 The scenario generation

The scenario generation module is crucial in building the framework, generating scenarios, necessary configurations, and selecting algorithms for evaluation. As it is illustrated in the Algorithm 1, it is responsible for generating configurations of evaluation scenarios based on Operational Design Domain (ODD) and Object and Event Detection and Response (OEDR).

Algorithm 1 SCENARIO GENERATION

procedure GENERATE ▷ S: system, E: environment
  ODD , $OEDR$, $Objs \leftarrow$ DEFINE(S , E) ▷ Define ODD, OEDR, and also objectives $Objs$
  $SC \leftarrow$ CONFIGURE(ODD , $OEDR$) ▷ Generate configurations of scenarios SC
  $ACs \leftarrow$ GENERATE($Objs$) ▷ Generate adverse conditions ACs
  $As \leftarrow$ SELECT($Objs$, $Dataset$) ▷ Select the algorithms As based on objectives $Objs$, also the dataset
  $GTC \leftarrow$ CONFIGURE($Objs$, S , E , A_s) ▷ Generate configurations of ground truth GTC
  return SC, GTC, $Objs$, As ▷ Return configurations, objectives, adverse conditions, and selected algorithms
end procedure

It also selects candidates of AI algorithms for the framework according to specific objectives, then evaluates and validates them based on a representative real-world scenario and dataset. Moreover, the generator component generates the configuration of the ground truth for the executor based on the selected algorithms, ensuring the accuracy and reliability of the evaluation process. The scenario generation should be a carried out using the PRISSMA method detailed in the Annex A. However, when configuring scenarios, it is imperative to define the types of objects the AI system under testing should detect. Furthermore, the scenario should include events that the system should recognize and respond to, such as sudden lane changes, emergency braking, or any other relevant mapping. By incorporating these elements, the scenario enables the evaluation and improvement of the system's perception and response capabilities.

To select algorithm candidates for an AI-powered system, it is essential to establish the domain of AI first. In the case of a visual perception system, deep learning methods like Convolutional Neural Networks (CNNs) have demonstrated promising results and are commonly used for image detection and segmentation tasks. Once the domain is determined, specific tasks should be extracted based on the system's objectives. After an extensive investigation of algorithms suitable for these tasks within the chosen AI domain, potential candidates can be identified. These candidates will undergo training using relevant datasets, and if possible, the models will be fine-tuned. Subsequently, the performance of the trained models will be validated to ensure they meet the necessary criteria for further consideration.

4.2.6.2 The scenario execution

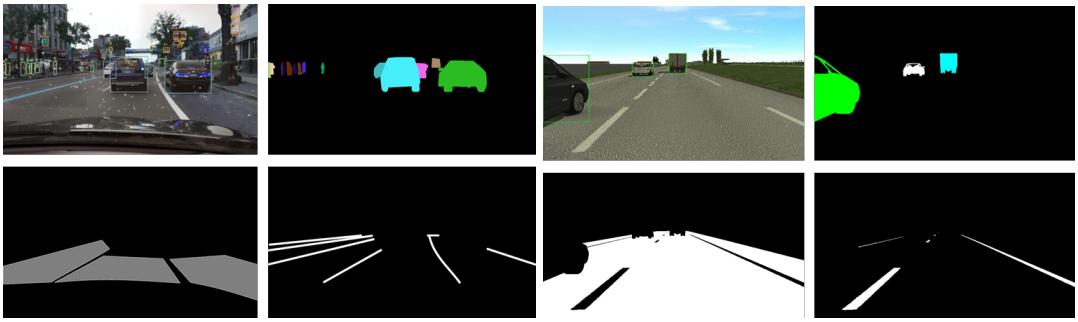
The scenario execution module is responsible for executing the different test cases on the integrated platform and tools, which are built by the output from the generator component of the framework, and also generating different types of results. The module must ensure that the system is executing properly and that the intermediate results are being generated correctly, and then passed back to the generator component as feedback. This process aims to refine the parameters inside the configuration of scenarios and adjust ODD or OEDR if needed. If there are any issues or errors in the execution, it needs to be resolved before passing on the final results to the evaluator component. Once the execution is complete, the final results are passed

to the evaluator component for assessment against the different types of evaluation metrics. The process of the execution is expressed by the Algorithm 2.

In real-world environments, ground truth can be generated through manual annotation or by using calibrated and accurate sensors or devices to capture the actual values of the variables being measured. In the proposed framework, this type of ground truth is used for training and preliminary validation of the selected algorithm. The existing datasets such as the BDD100k dataset [26] has been widely used in visual perception research and provides ground truth labels for various tasks as shown in figure 7a. In the evaluation framework, the selection and configuration of the ground truth are based on the chosen algorithms and the characteristics of the environment. Ground truth data can be generated by using a physics engine to model the behavior of the vehicle and its interaction with the environment, figure 7b shows the different ground truth for visual perception tasks in Pro-SiVIC$^{\text{TM}}$. In the simulation evaluations, the ground truth generated by the simulator will be collected by the executor and used for the final evaluation process while in real world evaluation, references and groud truth must most often be pre-processed by an operator.

Algorithm 2 SCENARIO EXECUTION

procedure EXECUTE(SC, As, GTC, ACs) ▷ Outputs from generator as inputs of executor
  $Ts \leftarrow$ BUILD(As , ACs) ▷ Build the test cases with different algorithms As and adverse conditions ACs
  for $Ti \in Ts$ **do**
    SI , $EI \leftarrow$ INSTANTIATE(S with Ai in Ti , E with SC , P) ▷ Instantiate the system SI with algorithm Ai in test case Ti and environment EI with scenario configuration SC on the integrated platform P
    $Ta_i^T \leftarrow T$.GENERATE(S_i^T , ACi in Ti) ▷ Generate the test action Ta_i^T based on the environment state S_i^T and the adverse condition ACi in the test case Ti
    $Sa_i^T \leftarrow S$.GENERATE(Obs_i^T) ▷ Generate system action Sa_i^T based on the observation Obs_i^T
    S_i^T , $Obs_i^T \leftarrow E$.UPDATE(Ta_i^T , Sa_i^T) ▷ Environment updates based on system actions Sa_i^T and test actions Ta_i^T
  end for
  GTs , $Rs \leftarrow P$.GENRATE(Obs , GTC) ▷ P records final results Rs and ground truth GTs (based on ground truth configuration GTC) from the observer Obs
  return Rs, GTs ▷ Return final results and the ground truth
end procedure



(a) BDD (b) Pro-SiVIC$^{\text{TM}}$

Figure 7: Ground truth of visual perception from the real world and simulation

4.2.6.3 The evaluation

The evaluation module is responsible for evaluating the performance of the AI-powered systems. It applied the selected evaluation metrics and KPI to the output from the executor, and then hereby evaluate the results combining corresponding ground truth. The overall process can be abstracted as shown in the Algorithm 3. The metrics used in the framework are chosen based on the different levels inside the evaluation objectives of the system, such as component level, system level, and scenario level.

Algorithm 3 EVALUATION

procedure EVALUATE ($Rs, GTs, Objs$) ▷ Evaluates the final results with ground truth
  $LEs \leftarrow$ LEVEL$(Objs)$ ▷ Define different levels of Evaluation LEs
  $Metrics \leftarrow Select(LEs)$ ▷ Select metrics for different evaluations
  $R_{metrics} \leftarrow Process(Rs\ ,\ GTs\ ,\ Metrics)$ ▷ Calculate the result of metrics $R_{metrics}$
  return $Visualize(R_{metrics}), Analyze(R_{metrics})$ ▷ Visualize and analyze the result of metrics $R_{metrics}$
end procedure

system evaluation: In order to evaluate the high-level quality of AI-powered system in ADS, such as a visual perception system, it is necessary to implement a full mobility service and propose relevant and representative scenarios involving an exhaustive set of conditions/configurations/situations allowing for quantification of the performances and the quality of the service. The metrics (a case of visual perception system) can refer to a set of specific Key Performance Indicators (KPIs):

- Risk specific: Longitudinal and lateral distance, Time to collision (TTC), Time Exposed Time-to-Collision (TET), Deceleration Rate to Avoid a Crash (DRAC), etc.,
- Task (detection/tracking) specific: Success rate, Loss, Distance, etc.,
- Time specific: Frequency, Time to detect/track, False alarm frequency.

Component evaluation: This level of evaluation focuses on the performance of individual algorithms or functions within the AI-powered system. The metrics are typically related to the functionalities of the perception function such as detection, segmentation and tracking [25].

scenario evaluation: This level of evaluation involves testing the performance of an AI-powered system under various challenging conditions, such as adverse weather, low lighting, and unexpected obstacles. These complex scenarios can be difficult to replicate in real-world testing, which makes simulation tools and virtual environments more essential. The use of simulation allows for the creation of complex scenarios that can be repeatedly tested, analyzed, and modified to evaluate, analyze, and improve the performance of the system. The related metrics can vary depending on the specific application and system requirements. However, some common metrics for this level of evaluation include:

- **Robustness:** This metric evaluates the ability of the system to perform consistently and accurately in various challenging and unforeseen situations, and is usually reflected in various performance metrics, such as accuracy, precision, recall, and F1-score, etc. Robustness can be measured by analyzing these performances in different scenarios and under different conditions, and also by assessing their ability to maintain performance levels over time.
- **Reliability:**This metric evaluates the system's ability to make reliable decisions in emergency situations or other adverse situations.

Figure 8: MOSAR- Scenario Manager. Figure from [4]

4.2.7 From functional scenarios to test cases

4.2.7.1 MOSAR Methodology and Tool

To illustrate our point, we have decided to use the MOSAR tool provided by IRT SystemX, but any tool that does similar work can be used.

MOSAR is a methodology and associated platform developed by IRT SytemX in collaborative projects including OEMS. MOSAR Scenario Manager is part of a suite with a precise focus on scenario management. PRISSMA's deliverable D2.6 [4] further presents this scenario management methodology and tool. This paragraph is an abstract of the information about MOSAR, for more details D2.6 can be consulted [4].

MOSAR Scenario Management platform allows users to have a database to register, store, classify, search, trace, analyze, import and export scenarios from and into the database. This tool can be used to manage descriptive scenarios which can also be transformed into test cases for simulation in different formats (OpenScenario, SCANeR studio, etc.). Scenarios are organized in a tree-like structure based on the three description levels: functional, logical and concrete. Scenarios are stored in specific data structures called containers which provide the means for storage, access restriction in order to ensure confidentiality (among other requirements), and collections access in order to describe scenarios; this is access to specific collections, i.e. elements related to infrastructure, actors, equipment, and behaviors among others.

The statistical analysis feature existing in MOSAR allows users to visualize the existing data in the platform based on the selection of parameters or conditions predefined by the operators. This feature is particularly useful to address analytically real world driving and visualize parameter distributions of encountered situations.

The structure and management of scenario containers can allow to assess on the coverage of requirements of system design.

4.2.7.2 Using Formal Conformance Testing to Generate Concrete Scenarios for Autonomous Vehicles

Starting from a test purpose, generating suitable concrete scenarios to test the behavior of AVs (Autonomous Vehicles) in relevant and potentially critical situations is a fundamental brick in the validation chain. However, due to the complexity of the involved systems and the dimension of the configuration space, obtaining interesting scenarios and a rigorous coverage guarantee is a challenging problem in autonomous driving. For the concrete scenarios generation, two techniques are commonly used: constrained random generation and manual specification [27]. Randomly generated scenarios can be easily produced, but their relevance might be difficult to assess, since they can present a high level of redundancy, which is hard to detect and strongly limits their coverage [27]. On the other hand, manually specifying a large number of concrete scenarios is extremely time consuming and a satisfactory coverage of all possible situations is hardly achievable.

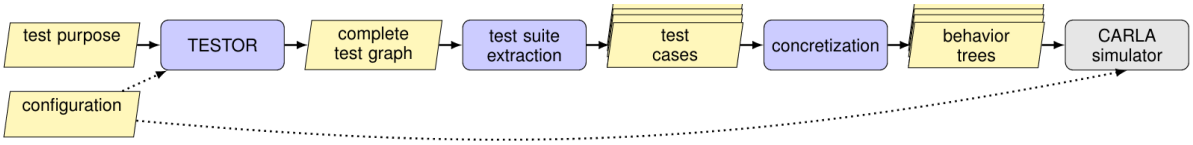


Figure 9: Overview of the approach based on formal conformance testing to generate behavior trees from a configuration and a test purpose. Image from [5].

To automatically generate concrete scenarios that are guaranteed to be relevant for testing AV's behavior in a particular situation (e.g., collision, near miss, etc), we propose to apply formal methods [5]. More precisely, we here present a conformance testing tool to generate concrete scenarios from a formal model and a test purpose characterizing the situation. This tool fits into the framework of scenario abstraction levels defined by the Pegasus project and PRISSMA delivrable L2.1. The generated concrete scenarios are concrete scenarios as defined, while representation level of the formal model and the test purpose is between abstract and logical scenario. It is more abstract than the logical scenario as the behavior of the scenario is unknown before its concretization, and many different concrete scenarios are generated from one formal model and one test purpose. It can be less abstract than the abstract scenario as the test purpose can precisely specify localized events (e.g. collision at a defined location between two specific actors), however the spatial or temporal localization of such events can be left undefined.

Formal Model of an Autonomous Vehicle and its Environment The first necessary step is to devise a formal model of the vehicle and its surrounding environment. We propose to devise such a model in LNT [28, 29], the main modeling language for concurrent systems supported by the CADP toolbox [30]. This perception-focused model has been used to generate a large number of concrete scenarios later executed on CARLA simulator (see Section 4.2.7.2).

The architecture of the LNT model is illustrated in Figure 10. The various elements are represented as concurrent processes interacting by multiway rendezvous. Hereafter we describe briefly the main elements, the full model being available in [6].

Obstacles. The obstacles are actors on the map that represent various objects and elements of the environment. Obstacles can have various sizes (minimum one cell) as they can be pedes-

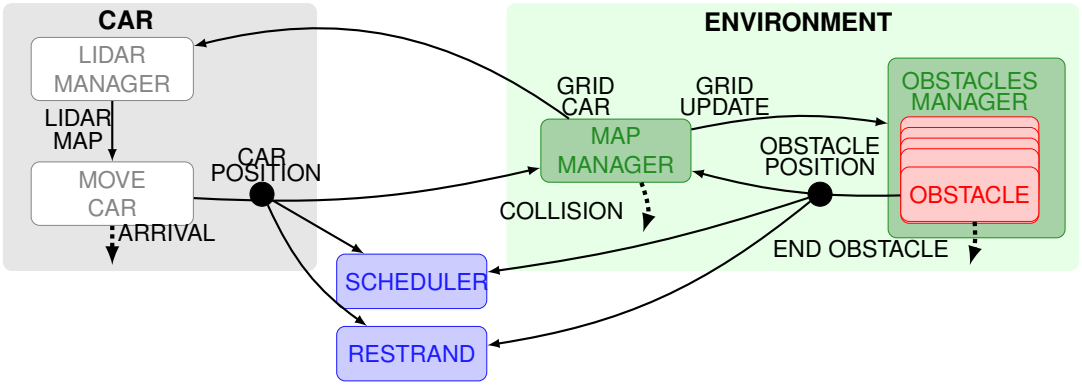


Figure 10: Architecture of the AV model in LNT. Image from [6].

trians, cars, or buildings. Obstacles can be static or mobile, the latter ones being able to move in any direction. Each move consists in traversing a number of cells determined by the obstacle speed. Some obstacles can hide the view (e.g., a car or a building) and some cannot (e.g., a pedestrian or a pole).

In a configuration, each obstacle has a list of moves defining its behavior. An obstacle may also choose not to move or may randomly choose between several possible directions, which adds nondeterminism to the model and enables the exploration of further scenarios. Obstacle moves must not lead to collisions (i.e., end on occupied cells of the map) to yield relevant scenarios. To keep the model size tractable, we enable full random moves only for obstacles close enough to the car, the random moves of the farther obstacles being restricted to directions bringing them closer to the car.

Each obstacle is modeled by an instance of the OBSTACLE process. Before attempting the next obstacle move (at the head of the list), process OBSTACLE obtains, via gate GRID_UPDATE, the current map from the MAP_MANAGER process. Based on the map, on the current obstacle information (position and speed), and on the direction of the next move, the obstacle determines whether the move is valid, i.e., does not lead to a collision. Then, process OBSTACLE sends on gate OBSTACLE_POSITION the previous position, next position, and new direction of the obstacle (including the case when the obstacle does not move) to process MAP_MANAGER, in charge of updating the map. When the list of moves is finished, process OBSTACLE performs an END_OBSTACLE action and stops moving, except when it has a cyclic behaviour, in which case it starts again using its list of moves given initially. The process OBSTACLES_MANAGER is the parallel composition of all OBSTACLE processes in the considered configuration.

Map. The map is essentially a grid-based representation of the environment in which the different actors move. The map is represented as a 2–dimensional array composed of cells with different values: free when there is no obstacle nor car on the cell, occupied (obstacle) when the cell is occupied by an obstacle (including all the obstacle information), and car_pos when the cell is occupied by the car (we consider only one car on the map).

The MAP_MANAGER process is a central part of the model, in charge of maintaining the map and of communicating with the other processes to update the position of the actors. The map is initialized with the positions of static obstacles and the initial positions of mobile obstacles. It is sent on gate GRID_UPDATE to the OBSTACLE processes to determine their next moves, and is also sent on gate GRID_CAR (accompanied by the car position) to the process LIDAR_MANAGER to generate the perception grid. If at some moment the position of the car

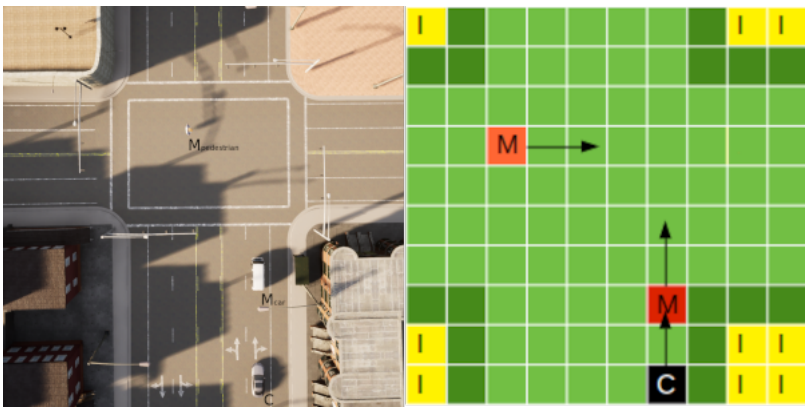


Figure 11: Representation of the map with the car and two mobile obstacles. Image from [6].

becomes the same as one of the obstacles, MAP_MANAGER performs a COLLISION action and stops, entailing the termination of the whole scenario. The ENVIRONMENT process is the parallel composition of MAP_MANAGER and OBSTACLES_MANAGER.

Car. The CAR process is the parallel composition of the LIDAR_MANAGER and CAR_MOVE processes, the latter being in charge of managing the car moves (similarly to process OBSTACLE). The car moves essentially in the same way as the obstacles, except that it can also move to an occupied cell and thus trigger a COLLISION action. Upon each move of the car, its previous and current positions are transferred to the MAP_MANAGER process to be updated on the map. If the car has finished its list of moves, it performs an action ARRIVAL, which terminates the scenario.

LiDAR. The perception grid represents the perception of the car (as computed by the LiDAR) up to a certain distance. It is modeled as a 2–dimensional array centered on the car position. The cells of the perception grid have different values from those of the map: F for free cells, C for the car position, O for occupied cells, M for cells that were free on the last grid but became occupied, T for cells occupied by a transparent obstacle, N similar to M but for cells occupied by transparent obstacles, and U for unknown cells, i.e., those out of the map (if the grid exceeds the map boundaries) or those hidden from view (behind an opaque obstacle).

The perception grid is maintained by the LIDAR_MANAGER process, which sends on gate LIDAR_MAP the new value of the grid and map to process MOVE_CAR to compute the next car moves.

Scheduler and Restrand. Two auxiliary processes optimize the model regarding both its scalability and its realism when connected to an AD simulator. The SCHEDULER process introduces additional synchrony in the model to bring it closer to its physical counterpart, by allowing all actor moves between two TICK actions to be performed in parallel, yielding realistic movements, as opposed to jerky ones induced by equivalent, but less realistic interleavings in the absence of TICK actions.

The RESTRAND process limits the random moves of the obstacles to keep them in a meaningful neighbourhood of the car. This is useful both for specifying scenarios with relevant obstacle trajectories (obstacles close enough to be perceived by the LiDAR) and for reducing the size of the state space.

Scenario module. To easily build various configurations of the LNT model, a scenario module enables to choose the map, the initial positions of (static and dynamic) obstacles, and the behaviour of the car.

Figure 12: Progressing of a test case whose configuration is shown Figure 11.

The LNT model has 1059 lines (excluding the scenario module, the size of which depends on the configuration) dispatched in eight modules, containing 13 types, 38 functions, seven channels, and eleven processes. Using CADP, for the map of size 10×10 represented in Figure 11 and two obstacles, we generated (in less than a minute on a standard laptop) the corresponding LTS with 27,168 states and 50,719 transitions (14,595 states and 28,287 transitions after strong bisimulation minimization).

This LNT model focuses on a particular component (i.e., the perception), with a grid-based representation of the geographical map. The advantage of this focus is the possibility to refine the precision of the moves of the obstacles and the car (e.g., by increasing the resolution of the map and perception grid) and to fine-tune the model to cover a large number of relevant AD perception scenarios. For instance, this model enables random trajectories for the obstacles with different speeds around the car, within an area of parameterized size managed by the RESTRAND process.

Conformance Testing for Scenarios Concretization The formal model previously presented is specialized to a given configuration, which includes a scene map and several actors with their initial positions and constraints on their trajectories. The sequence of actions of all the actors will be automatically induced by the generated scenarios. In general, each test purpose will yield several scenarios, with guarantees to cover all relevant variations of the behavior related to the test purpose. These scenarios are then automatically transformed to be used as input for a driving simulator. To ensure the generation of relevant and critical scenarios, test purposes (e.g., reaching a collision) and test configurations can be defined based on critical situations emerging from road accident data [31]. As stated before, these scenarios are equivalent to concrete scenarios, they describe the environment of the scenario, the actors and their behaviors (i.e. their trajectories).

In connection with the PRISSMA WP2, we illustrate our approach with CARLA simulator [32] by providing a method to translate the scenarios into behavior trees. Our approach is initially evaluated on ten configurations, involving three scene maps (T-crossing, highway, and X-crossing) and various actors, for which we generated several scenarios featuring collisions of the AV with other actors, near-misses of such collisions, and arrivals at the destination. For more details on these results, see [5].

Figure 9 gives an overview of the proposed flow. Its first input is a configuration defining the scene with its objects and their behavior, from which a formal model and a corresponding

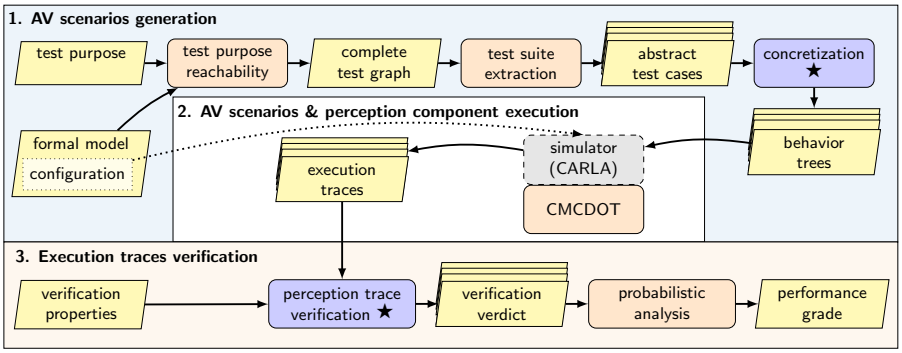


Figure 13: Overview of the approach proposed in [7] to verify AV perception components. The approach generates for a formal model all possible AV scenarios (behavior trees) addressing a specific situation to simulate (test purpose). The generated AV scenarios are then executed on an AV simulator (e.g., CARLA) connected to a perception component (e.g., CMCDOT) to obtain execution traces, on which to perform formal verification and probabilistic reliability analysis.

simulator configuration are derived. The second input is a test purpose, describing the intent all test cases should focus on. The, we extract a CTG (complete test graph) from the model and a TP (test purpose) using the TESTOR tool [33]. A TP is an automaton with special "ACCEPT" labels characterizing the states to be reached by the scenario, and a CTG is an automaton that contains all transition sequences leading to these states. When computing a CTG, only the transitions corresponding to a controllable input or observable output of the SUT (system under test, in our case the CARLA simulator) are necessary. Thus, we can hide—and reduce the model—all other transitions (e.g., the broadcast of the ground truth map) that are useful for validation, but irrelevant for test generation. In general, a CTG contains states for which several inputs can lead to a successful run. Thus, we apply the techniques presented in [10] to extract a test suite, i.e., a set of TCs (test cases) covering all transitions of the CTG. Each TC is an automaton interacting with the SUT to drive it towards the accepting states specified by the TP. Thus, for a given model, several different TCs (and hence, scenarios) can be generated. The TCs extracted from the model and a TP are represented in an abstract form as automata, and must be finally transformed into a more concrete form to be used as simulation scenarios. This last step is dependent on the considered simulator. In the framework of the PRISSMA WP2 (task 2.4), we present the translation of the generated test cases into behavior trees to drive the CARLA simulator. The details of this final concretization step and the results obtained in simulation can be found in the Deliverable 2.6.

Verifying Collision Risk Estimation using Autonomous Driving Scenarios Derived from a Formal Model This scenario generation method has been used in [7] in combination with other tools in a more complete AV validation process. An overview of the entire proposed approach is shown in 13. This work aims to formally validate the perception component of an AV by verifying formal properties on the collision risk estimated by the perception component. First, a set of scenarios is generated from several configurations and test purposes covering the ODD. These scenarios and the AV perception component are simulated and the collision risk estimation is recorded. Then, formal model checking techniques and statistical analysis are used on the recorded data to formally validate the collision risk estimation.

Figure 14: Image from [8]. From a base scenario, the generative model can generate realistic-looking altered images.

Using generative models to provide calibrated input samples Generating realistic test-case for a system can be difficult, depending on a given operational domain. Let us take as an example a perception program that should detect pedestrian under several weather conditions, and takes images as inputs. Assuming the system (including the data) is developed in France, we can assume that there may be less test samples under unusual French weather (for instance, strong snow in plains) than under usual French weather. This lack of data may result in an insufficient coverage.

Using generative models, it is possible to sample realistic inputs from a learned probability distributions. Such inputs could then be altered to better suit the operational domain and fill the missing datas. Following our example, one could generate images from a basic weather condition and add realistic-looking snow or fog on those. See [34, 8] for an example of such technique. Also see 14 for an example of such image generation.

4.3 How to ensure minimum coverage?

4.3.1 Exploration of the scenario space

The biggest problem with a scenario approach is ensuring that the scenarios are covered. In absolute terms, we need to find an optimum (find the necessary and sufficient scenarios) to ensure that we cover enough without causing the number to explode. In this section, we provide advice and methods for dealing with coverage problems. At the time of writing, this remains an open research problem, so it will be more of a guide to good practice and good methods at any given time.

4.3.2 Metamorphic testing

Testing usually requires an oracle to compare the expected result against the computed result. Such an oracle may not exist in some cases (lack of specification), or being prohibitively costly to use. For certain properties however, it is possible to use a technique called Metamorphic Testing that alleviate the lack of oracle problem.

Let P be a program. Let us denote by x an input for P, and $P(x)$ the result of the computation of x by P. A metamorphic relation is a necessary condition that links a set of inputs $[x_1, ..., x_n]$ and the corresponding outputs $[f(x_1), ..., f(x_n)]$. As an example, given a program that computes the absolute value of a number. Here, one possible metamorphic relation would be that "The result of the computation is the same for a number and its negative." Given the set of input $[1, 2]$, it is possible to generate derived inputs: $[-1, -2]$. The output of P on this set of input is

expected to be to be $[|1|, |2|, |-1|, |-2|] = [1, 2, 1, 2]$. Note that the metamorphic relation does not limit to equality of inputs: For the ACAS benchmark, presented in the previous deliverable of WP1, properties could be of the form "given a family of inputs that are below a certain angle, the output of the program will never be to move".

Metamorphic testing consists on the following steps:

1. compute a set of source inputs $[x_1, ..., x_k]$ by P
2. generate a set of derived inputs $[x_{k+1}, ..., x_n]$ using the metamorphic relation
3. compute $P(x_i), i \in (k+1, ..., n)$
4. checks if $[x_1, ..., x_k, x_{k+1}, ..., x_n, P(x_1), ..., P(x_k), P(x_{k+1}), ..., P(x_n)]$ respects the metamorphic relation. If the metamorphic relation does not hold, it means P is not working as intended

Metamorphic testing relies on well-defined metamorphic relations. Defining those relations is a manual process; a program P may have an huge set of metamorphic relations, some of them not relevant for the test campaign goal. Given well-chosen metamorphic relations, the test campaign can be augmented by a set of derived inputs, leading to an overall better coverage.

Metamorphic testing was successfully used to detect faults in GCC and LLVM, and to industrial grade software. See [35] for a comprehensive survey on this approach.

There are some work that apply metamorphic testing to autonomous vehicles. For instance, the authors of [8] use generative models to design new inputs that are supposed to yield the same outputs; failing this test displayed that the neural network was not behaving properly. Although not explicitly using metamorphic testing, the authors of [36] defined equivalence classes between neurons of same activation sign in a linear region; such classes are expected to yield the same output, hence a metamorphic relation can be constructed from there. Authors of [37] generate inputs that maximize neuron coverage (given a sample, it is the ratio between activated neuron on the total number of neuron) from inputs that are not faulty. The insight here is that a car should keep some behaviours similar under a transformation of the image. For instance, the car should keep steering at a road angle, even with different weather conditions. AIMOS, a program developed at CEA and at the time of writing, use on industrial use cases for Grand Défi IA de Confiance, is a tool specialised on metamorphic transformation applied to neural networks.

4.3.3 Coverage testing

Metrics for coverage testing applied to neural networks

Testing classical software usually relies on a family of metric, called "coverage metrics". To briefly summarise, coverage is a measurement of how much a given test scenario explores the behaviour of the program. It can for instance study how many branches are taken (branch coverage).

An analogy of this metric can be found in neuron coverage. Proposed several times in the literature with slight variations [38], this metric can be seen as an analog of branch coverage. Let f be a neural network with N neurons. Here, a neuron n_i is to be understood as a function $n_i : \mathbb{R} \mapsto \mathbb{R}^{+*}$. When the output of n_i is strictly above 0, n_i is active. When the output of n_i is equal to 0, n_i is inactive. Let T an test set comprised of samples x for f. Neuron coverage

consist on calculating the amount of active neurons over the total number of neurons for this input:

$$\frac{|\{n_i > 0\}, f(x), \forall x \in T|}{N}$$

Techniques such as DeepXPlore [38], DeepTest [37] or DeepConcolic [39] leverage this definition to produce test cases that aim to maximise neuron coverage.

There are several issues with this metric however. In classical coverage testing setting, one would like to maximise the coverage of both the functional and the error management parts. However, contrary to classical programs, neural networks do not have clearly defined failure mode, such as dedicated return values (exceptions or error codes). As such, there is a potentially very high overlap between a correct execution and a faulty one. Second, test case generation tools relying on neuron coverage usually provide a high amount of faulty inputs [34], leading to an increased cost. Since neuron coverage does not provide any oracle whether the input is correct or not, it is up to the human operating the test to triage the input, leading to increased cost with little benefits. Authors of [34] propose to add a generative model to generate only valid test cases, reducing the overall human cost.

1. fraction of activated neural networks [34]
2. this paper [40]

4.3.4 Border analysis

The exploration and caracterization of failing scenarios is an important task to ensure model safety in operating conditions. However, a trade-off must be performed between the number of explored scenarios (which ensures precision) and the computational complexity. Indeed, a coarse discretization of the parameters space may cause some critical cases to go undetected if their parameters fall between the values of the discretization grid. On the other hand, if the discretization step is too low, the computation power required to browse the scenario space becomes unsustainable.

4.3.4.1 Building a ”map” of the scenario space

For each use case, thousands of scenarios are generated by variation of input parameters. Each parameter has a finite domain of evolution, and the number of parameters defines the dimension of the configuration space. For safe operation, it is necessary that the domain of functioning and the domain of failure are determined, and the behavior of the model must be predictable when conditions are getting close to failure (border cases) In the work [9], the author presents a simulation-based method for characterizing the failure domain. The algorithm is the following:

- A criterion of failure, or NOGOOD, is defined (for example, in the use-case where an autonomous vehicle must follow a vehicle ahead of it, the failure criterion is the safety time gap between the vehicles)
- An initial “Find One Failure” algorithm tests random scenarios with the goal to find one failure far from the other scenarios. It is an optimization algorithm on failing scenarios
- All outputs from this algorithm are stored and with enough iterations, a “map” of the use-case configuration space is produced.

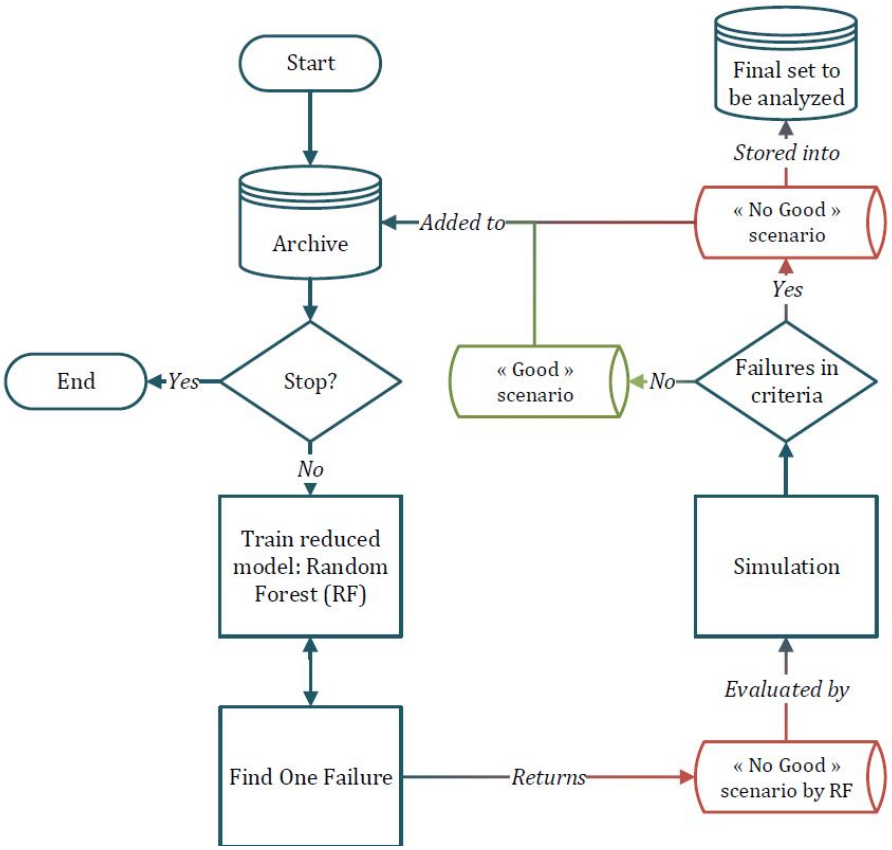


Figure 15: Algorithm used to research all failures in the configuration space. Diagram taken from [9]

4.3.4.2 Caracterising border cases

After this algorithm outputs a map of the use-case space of chosen density, a subset of the global space of scenarios is defined as "border cases". The border case scenarios are those that have both GOOD and NOGOOD neighbour scenarios. A neighbour scenario is one where all parameters differ by at most one step, for example for a point on a 2D grid it would be the 9 direct and diagonal neighbours. The number of border cases is large (almost 1/3 of all scenarios in the provided example) because of the large dimension of the configuration space.

The author then proposes to use specific "border models" which predict whether a given scenario is on the border while limiting the number of calculations. First, a Neural Network model is proposed, with learning on a balanced subset of the scenarii (same number of border and non-border cases). However, although different dataset sizes and optimization approaches were tried, it was found that the robustness of the learning process was insufficient. Furthermore, as the NN approach lacks in explainability, the author proposes a Mixed Integer Linear Programming (MILP) approach.

The MILP approach consists in calculating an exact mathematical description of the border, in the form of a set of equations. The computation of the model is costly, but its application is near immediate. The computation of the model can be accelerated by defining a margin of errors (number of GOOD scenarii that can be classified as NOGOOD). The equations used are either affine (defining a hyperplane in the configuration space) or quadratic (defining a quadratic manifold). The MILP classification is applied as follows:

- each equation $f_1(X) \leq 0$ divides the configuration space in 2, such that one half of the space contains only GOOD scenarios
- For any scenario X, X is NOGOOD if $f_i(X) > 0 \; \forall i$

The function f used can be affine or quadratic and takes as parameters the coordinates of the scenario in the configuration space.

When the NOGOOD scenarios form a single cluster, the MILP classification algorithm is simply recursive:

- Search for an equation that eliminates at least one GOOD scenario
- Remove from set all GOOD scenarios classified by this equation
- Restart with the new reduced set, until no equation can be found.

.

When the shape of NOGOOD subset is more complex, such as several separated clusters, the classification is adapted by splitting the set into clusters and applying the recursive algorithm to each cluster, as illustrated by figure 16.

This work proves that it is possible to build scenario classifiers based on a sampling of the classification space. Although the initial computation cost is high, these models are later relatively easy to implement, even by industry standards.

4.3.5 Model-Based Testing and Coverage

MBT (Model-Based Testing) [41, 42] encompasses the range of methods that exploit a model of the SUT (System Under Test) to automate testing. MBT enables to keep tests in close correspondence with the SUT's requirements and reduces the cost of the test activity, at the price

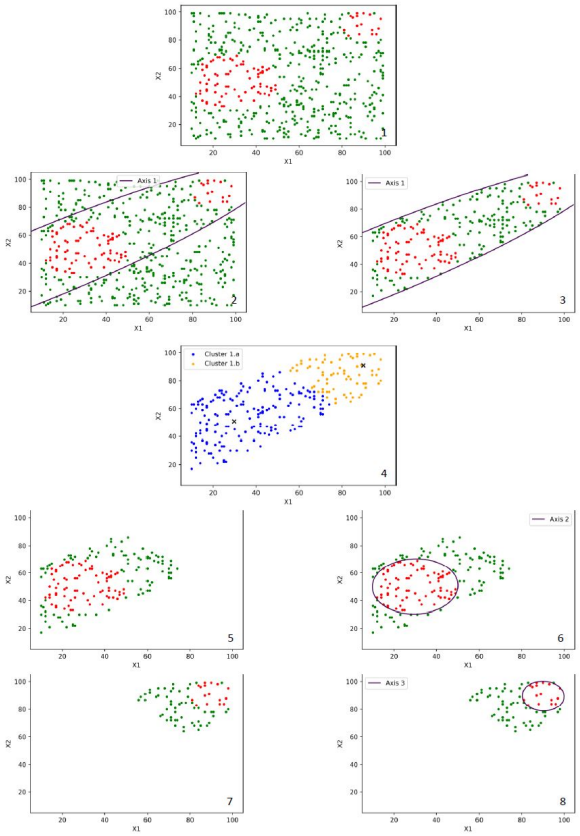


Figure 16: Step-by-step progress of MILP with cluster separation. Diagram taken from [9]

of developing a model of the SUT. Conformance testing is a form of black-box MBT seeking to establish that an SUT behaves according to a model, which serves as an oracle. This approach relies on the hypothesis that the behaviour of the model and the SUT can be represented as an IOLTS (Input-Output Labelled Transition System) [43], which is a convenient semantic representation for high-level formal languages.

A popular conformance relation for IOLTSs is *ioco* [44], which serves as basis for on-the-fly test case generation guided by test purposes, as implemented in the TGV [43] and TESTOR [33] tools. This approach allows the tester to build a test plan, i.e., set of test purposes at a similar abstraction level as the system requirements. The test plan must then be transformed into a test suite, i.e., a set of concrete, deterministic test cases to be executed on an SUT. Each test purpose directs the test case extraction and enables to handle large models by ignoring those parts of the model irrelevant to the considered test purpose. The tester is confronted with the questions of when to stop the testing process (by devising no more test purposes) and how thoroughly the SUT has been tested. These well-known questions in the testing domain are classically addressed using coverage criteria [45] measuring the degree to which the internal structure of an SUT was exercised during the testing process. For the *ioco*-based conformance testing, a suitable coverage criterion is transition coverage, which consists of covering each transition in the IOLTS.

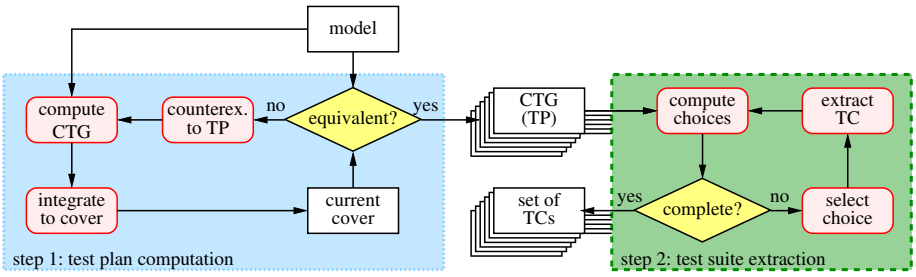


Figure 17: Overview of the transition coverage approach (image from [10])

An approach was proposed recently [10] to automatically generate a set of test purposes with their corresponding CTGs (Complete Test Graphs), each of which contains all necessary information to drive a (conformant) SUT towards the corresponding test purpose (if possible). The approach, illustrated on Figure 17, is iterative: in each iteration, a new test purpose is derived from a counterexample illustrating a not yet covered transition of the model. It is also possible to start from an existing, non-trivial (i.e., not empty) test plan, completing it to cover all transitions, as well as detecting redundant test purposes that do not increase the coverage. Because a CTG is not necessarily controllable (e.g., there might be a non-deterministic choice between inputs to be sent to the SUT), a deterministic test suite covering all its transitions is further automatically extracted from each CTG. The union of all such generated test suites thus ensures transition coverage of the IOLTS model. This approach was implemented on top of TESTOR[6] and the CADP toolbox[7] [30], and experimented on several distributed systems.

4.3.6 A proposal for a validation protocol

1. definition of the scenario space is necessary

[6] http://convecs.inria.fr/software/testor
[7] http://cadp.inria.fr

2. scenario generation test techniques should be employed

   (a) model-based scenario generation

   (b) metamorphic testing

3. coverage testing metrics have limited usefulness

4.4 Choice of metrics, KPIs and criteria

4.4.1 Safety metrics

Several metrics are used to assume the safety of the autonomous vehicle. The standard one is the one proposed by MobilEye : the "Responsibility-Sensitive Safety (RSS)[8] proposal [46] and its NHSTA implementation [47] complement the classic use of "Time-to-X" metrics (Time to Brake, Time to Collision [48], [49], etc.) and the avoidance metrics ([50]) .
The most natural approach is to measure the time remaining before the occurrence of the dangerous event in order to avoid it. This approach has produced a family of metrics generally referred to as Time To Event or as it is named as Time-to-X" (TTX) or temporal proximal indicators. The TTX is a natural measure that makes it possible to decide the most suitable action to avoid the danger given the remaining time. It is also usually to illustrate the different situations before the dangerous event appears on the time axis as illustrated in the figure 18, taken from the article [11]. In this figure we see that the situations evolve from normal driving to an unavoidable accident just before the collision as well as the different assistance systems adapted at the considered moment. The TTX metrics allow us or to assess the effectiveness of an ADAS system in avoiding accidents or driving difficulties. The more the TTX metrics are large the more efficient and the more comfortable is the driving and in the figure 18, this is illustrated with the colors green to red.

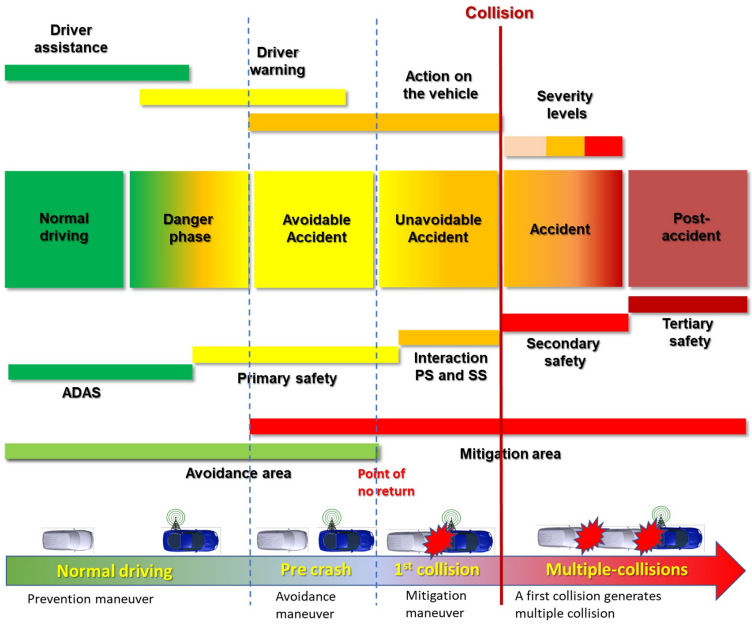


Figure 18: The different risk levels and situation intervals. Updated modeling from ([11])

[8] https://static.mobileye.com/website/corporate/rss/rss_on_nhtsa.pdf

At a minimum, any evaluation should take into account such metrics. We will first give a quick overview of these metrics before going into more detail on how to use them and in which cases.

4.4.1.1 Inter-Vehicle Time (IVT)

This metric is simply the calculation of the inter-vehicular time between the EGO vehicle and the primary target, i.e. the time required for the EGO vehicle to travel the distance to the target at the current constant speed. Let $v(t)$ be the speed of the EGO vehicle and $d(t)$ the distance to the target then the mathematical formula is :

$$IVT(t) = \frac{d(t)}{v(t)} \tag{1}$$

4.4.1.2 Time to Brake (TTB)

Time after which a braking maneuver has to be started to prevent the collision. If the TTB is smaller than 0, a collision can not be avoided by braking [51].

4.4.1.3 Time to Steer (TTS)

Time after which an evasive maneuver has to be started to prevent the collision. If the TTS is smaller than 0, a collision can not be avoided by steering [51].

4.4.1.4 Time to collision advanced (TTC_a)

The Time to Collision (TTC) has been studied intensively, and is based on a microscopic 1D vehicle trajectory model for longitudinal collisions of two vehicles. This limitation is important and the improvements made allowing to calculate both longitudinal and lateral collisions on a 2D plane is proposed in this TTC_a metric by as it is proposed in [49],[11] or [52]. This metric is the calculation of the time to collision between the EGO vehicle and a target. This extension of the TTC collision time is used in Mobileye systems [48].

Let $(x(t), y(t))$ the EGO vehicle position vector, $(v_x(t), v_y(t))$ the EGO vehicle speed vector, $(\gamma_x(t), \gamma_y(t))$ the EGO vehicle acceleration vector, $\left(x^T(t), y^T(t)\right)$ the target position vector, $\left(v_x^T(t), v_y^T(t)\right)$ the velocity vector of the target and $\left(\gamma_x^T(t), \gamma_y^T(t)\right)$ the acceleration vector of the target, we try to estimate X which corresponds to the collision time, noted , in the following second order equation [49] :

$$d_{ij} + \dot{d}_{ij}X + \frac{1}{2}\ddot{d}_{ij}X^2 = 0 \tag{2}$$

we can deduce a Distance to Time to Collision advanced ($dTTC_a$) as the calculation of the pre-collision distance between the EGO vehicle and a target and thus the distance to the TTC_a. The formulation is therefore

$$dTTC_a(t) = TTC_a(t) \times v_x(t)$$

4.4.1.5 Lateral Avoidance metrics

Here we model indicators that highlight the instantaneous lateral acceleration ACC_{lat} or the instantaneous deceleration required to avoid a vehicle in its lane DCC_{long}.

The lateral avoidance acceleration can be written as follow ([50]):

$$\begin{aligned} ACC_{lat}(t) &= \gamma_y^T(t) - \frac{2}{TTC_a(t)^2}(-\left(y^T(t) - y(t)\right) \\ &\pm \frac{\left(W\sin\left(\delta(t)\right) + W^T\sin\left(\delta^T(t)\right)\right)}{2} \\ &- \left(v_y^T(t) - v_y(t)\right)TTC_a(t)) \end{aligned} \tag{3}$$

With W the width of the EGO vehicle and W^T the width of prior target

4.4.1.6 Longitudinal Avoidance metrics

The instantaneous deceleration required to avoid a vehicle in its lane can be written:

$$DCC_{long}(t) = \min\left(\gamma_x^T(t) - \frac{\left(v_x^T(t) - v_x(t)\right)^2}{2 \times d(t)}, 0\right) \tag{4}$$

4.4.1.7 Dealing with the uncertainty

The indicators presented below are mainly based on deterministic vehicle and driving models. However, on the road, the the detected targets IA algorithms are subject to uncertainties on the classification and the positioning of dynamic elements over time. This uncertainty may results from the physical possibilities due to the degrees of freedom and from the behavior of the road users or from the perception perturbations due to environment factors or to the limits of the sensors. We can present here two interesting approaches that assess the uncertainty using probability modelling or using mechanical modelling such as friction circle, as known as Kamm's circle that models the area where the dynamic object should be [53]. The probabilistic approaches can be illustrated by the method proposed in [54] Lambert et al. The goal is to assess collision using the distance of a vehicle to an object thanks to a provided Gaussian normalised function on multiple-dimensional space (x, y, V). Besides, a definition of the collision risk is presented as the product of the collision probability function and the cost of collision, approximated as the Energy Equivalent Speed(EES) [55]: $costcoll(V) = EES(V)$. The collision probability $Pcoll$ used is a multivariate distribution on 2D real space:

$$Pcoll = \int_D P_v(x_v, y_v, \theta_v)P_o(x_o, y_o, \theta_o)dx_v dy_v d\theta_v dx_o dy_o d\theta_o$$

Hence, the risk of collision Riskcoll(v) is computed with: $Riskcoll = Pcoll \cdot costcoll(v)$ and the TTC is computed for the predicted position where the the risk of probability of collision is the highest. In the figure 19, we show an illustration of a Ego-vehicle detecting an obstacle and the probability of collision $Pcoll$ is computed.

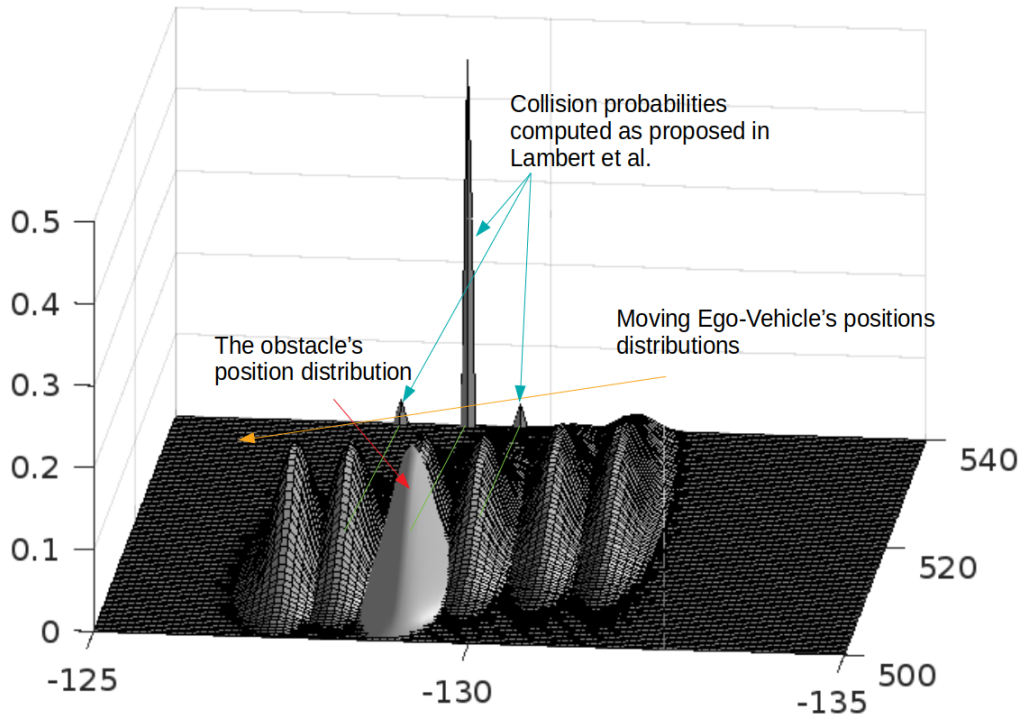


Figure 19: The probabilities of the Ego-vehicle moving in straight line towards the obstacle and the probability of collision.

In more recent works, [56] proposed an advanced collision risk modelling for autonomous vehicles, extended from an interaction-aware motion modelling based on Dynamic Bayesian Networks (DBN). They took in consideration a more global view of the current situation (network and traffic level) in order to estimate a "network level collision prediction". This prediction capacity allows to have a risk anticipation which clearly improves the risk assessment in complex and large traffic scenarios. Later, modelling relying on octagonal representation of the surrounding space is proposed by [57] as an analytic approach to assess the collision risk with obstacles. They used two probabilistic methods to calculate the risk: the Collision State Probability (CSP) in real-time and the Collision Event Probability (CEP) density. They developed the octagon concept which models the trace of the obstacle centroids when it moves around the ego-vehicle (represented by a rectangular box). This concept provides a spatial multidimensional safety indicator.
The mechanical approach is using the so-called Kamm's circle [53] that predict the area where the vehicle or the obstacle should be when the size, the speed and the acceleration of the vehicle and the obstacle are provided by sensors. The position of the dynamic obstacles $(x_o(t), y_o(t)$ and of the vehicles $(x_v(t), y_v(t)$ are modelled by a very simple linear model when the speeds v_o and v_v are given:

$$\begin{aligned} x_{\cdot}(t) &= v_{\cdot,x}t + x_{\cdot}(0) \\ y_{\cdot}(t) &= v_{\cdot,y}t + y_{\cdot}(0) \end{aligned} \tag{5}$$

where $\cdot$ is o for obstacle or v for vehicle. The dynamic objects' position are known with uncertainty and it is modelled with circle centred in $(x_{\cdot}(t), y_{\cdot}(t)$ with radius $r_{\cdot}(t) = \frac{1}{2}a_{\cdot}t^2$. The radius is increasing with t as fast as t^2 modelling the increasing uncertainty over time. The vehicle and the obstacle size are also represented by circles that contain these objects with the radii ρ_v and ρ_o. The collision occurs when $r_v + \rho_v + r_o + \rho_o \geq \sqrt{(x_v - x_o)^2 + (y_v - y_o)^2}$ and the shortest time for the vehicle to hit the obstacle is the metric so-called Worst-Time-To-Collision (WTTC).

The article of Wachenfeld et al. [53] shows that the WTTC is a better metric to identify situations that are actually dangerous and reduce the number of situations that are not critical in order to focus on the dangerous ones. In our opinion, the probabilistic approaches that lead to the estimation of risk can also be used for the same purpose. However, the comparison between the two approaches need to be studied.

4.4.2 Robustness: uncertainty and out-of-distribution, active learning, calibration

Over the past decade, significant progress has been made in artificial intelligence (AI), particularly in fields such as autonomy and robotics. AI has also shown promise in other high-risk areas like medicine and healthcare. However, despite these advancements, there remains a noticeable gap between the innovation of these technologies and their practical application in everyday life. Many of these AI technologies were developed several years ago, yet they are not widely accessible in our daily lives. The reason why we can't easily buy self-driving cars or the absence of robots assisting in surgical procedures is primarily due to the challenges and failures AI has encountered. These failures have been observed in various safety-critical domains, ranging from accidents involving autonomous vehicles to healthcare errors. Addressing these issues is crucial before deploying AI in critical domains. To do so, we must focus on innovating in the fields of safe and robust artificial intelligence.

Two fundamental issues inhibit AI's deployment into high-risk domains. The first involves hidden biases present in training data. Bias arises when machine learning models perform better for specific groups over others. Algorithms trained on skewed datasets may generate solutions that are not universally effective in real-world situations. The second problem revolves around unmitigated and uncommunicated uncertainty. This happens when AI models don't know when they can or can't be trusted. Indeed, machine learning models can behave erratically when presented with data dissimilar to the training data, or what is can referred to as non-nominal data. This uncertainty can, for instance, lead to self-driving cars continuing to operate even when their confidence in their environment is less than 100%, rather than surrendering control to human operators. This element of uncertainty needs to be addressed to ensure safe and reliable AI deployment.

The EASA figure 20 illustrates the impact of different data types on AI's operational system. The green zone indicates nominal operations within the in-distribution data, where the system's responses are as trained and expected. The yellow zone represents in-distribution but non-nominal conditions, where the data is unusual but still within the range of the AI system's experience, and the system continues to respond accurately. The orange zone marks the boundary of out-of-distribution data, indicating increasing severity and potential for error, but not yet critical. In contrast, the red zone highlights where the system encounters severe out-of-distribution data, leading to potential system failure.

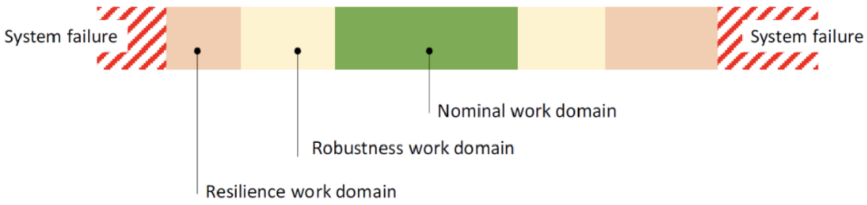


Figure 20: Illustration of the work domains as reported in [12]. From central green bar to side yellow/orange/red bars, the nominal domain shifts and the severity increases in parallel

Our aim is to build AI systems with an introspective understanding of their knowledge boundaries—AI systems that know what they don't know. Achieving this would mean creating AI that is adept at maintaining high predictive performance while also being capable of identifying and reacting to out-of-distribution data. We need a robustness model that clearly demarcates in-distribution data from out-of-distribution scenarios. This demarcation is critical so the AI system can proactively signal when it is unable to make reliable predictions and should therefore transfer control to a human operator. In essence, an AI system's reliability in high-risk environments is contingent on its robustness and its capacity for recognizing its own operational limits.

A proposal for a validation protocol

The evaluation and validation protocol (figure 22) that we propose should allow to have clear and precise answers on the following questions:

1. How trustworthy are the uncertainty estimates of our model under perturbations ?
2. How robust are the prediction of our model under perturbations?
3. How do uncertainty and accuracy of different methods co-vary under perturbations

Concretely, we would like the model predictions to become more uncertain with increased data distribution shift, as far as shift degrades accuracy. This is usually called "covariate shift". Hereafter, we start by selecting a subset of perturbations, following state of the art results, allowing model evaluation and validation with reduced cost. Next we explain decision process.

1. Data perturbations

   (a) Data-set shift: We propose the following shift for autonomous driving system:

   - Time of day / Lighting
   - Geographical location (City vs suburban)
   - Changing conditions (Weather / Construction)

   They may be simulated using domain adaptation technique [58] that has emerged as a new learning technique to address the lack of massive amounts of labeled data by using labeled data in one or more relevant source domains to execute new tasks in a target domain. In our context, we propose the following validation condition.

Figure 21: Examples domain adaptation technique for autonomous driving system

(b) Adversarial perturbations

(c) General corruptions

(d) OOD samples

2. Robustness validation : This refers to the ability of AI algorithms to function effectively in the face of unexpected or adverse situations. In general, there are two different approaches one can take to evaluate the robustness of a neural network: attempt to prove a lower bound, or construct attacks that demonstrate an upper bound. The former approach, while sound, is substantially more difficult to implement in practice, and all attempts have required approximations. On the other hand, attacks used in the latter approach are not sufficiently strong and fail often, the upper bound may not be useful. Moreover, there exist different types of adversarial attacks and defenses for machine learning algorithms which makes assessing the robustness of an algorithm a laborious task. thus, there is an intrinsic bias in these adversarial attacks and defenses to make to further complicate matters. For instance an evaluation process must avoid a model dependence behavior, insufficient evaluation, a perturbation dependent results. This requires a model agnostic adversarial robustness assessment. In [59], authors have recently observed that dual synchronised attacks based on L_0 and L_∞ distance-norms allow a good robustness assessment on several neural network architectures. Moreover, their results suggest that L_1 and L_2 metrics alone are not sufficient to avoid spurious adversarial samples and it is better to combine dual norms (1 and ∞) to construct an upper bound on the robustness of the model.
3. Uncertainty validation : Naturally, we expect the accuracy of a model to degrade as it predicts on increasingly shifted data, and ideally this reduction in accuracy would coincide with increased forecaster entropy. A model that was well-calibrated on the training and validation distributions would ideally remain so on shifted data. On the completely non-nominal data, one would expect the predictive distributions to be of high entropy. Essentially, we would like the predictions to indicate that a model "knows what it does not know" due to the inputs straying away from the training data distribution.

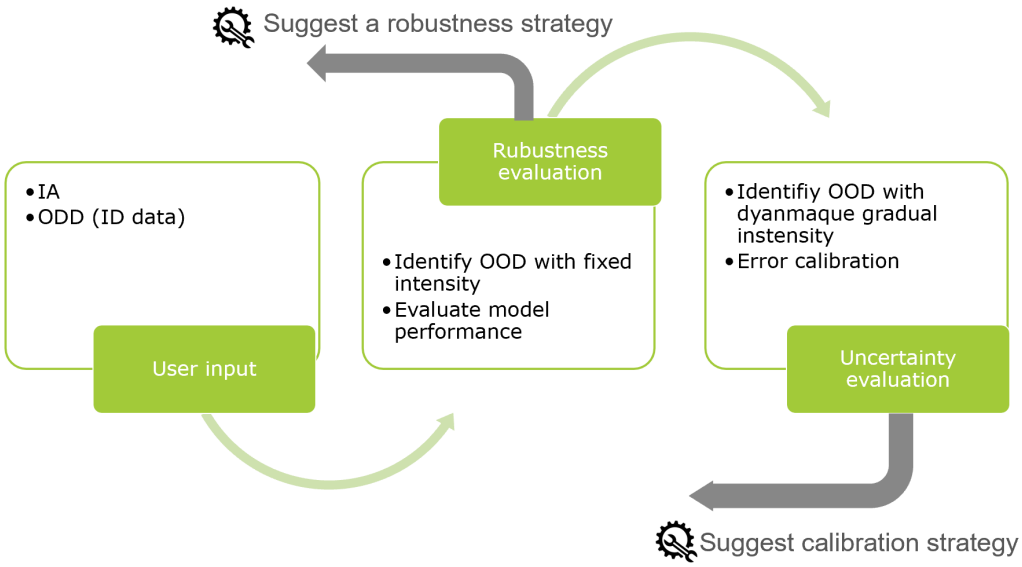


Figure 22: A proposal for a validation protocol

In the following we will delve deeper into various aspects of the proposed protocol. First we will delve into the impediments to AI deployment in high-risk domains : Bias and OOD. We'll then explore the concept of AI robustness. Next, we will examine the quantification of uncertainty, crucial for comprehending AI's decision-making process and assessing its reliability. Finally, we will consider the robustness of uncertainty quantification, crucial to ensure that decisions made by AI are not only reliable but also precise.

4.4.2.1 Bias

Recent research has brought to light the vulnerabilities of AI-based systems to bias, a phenomenon that can be quantified and mathematically defined. Two main types of bias have been identified: sampling bias and selection bias, both of which can occur during different stages of the AI lifecycle.

Sampling bias occurs when certain regions of our input data distribution are over-sampled, while others are under-sampled. This can lead to skewed representations of various groups in the data, affecting the fairness and generalization of the AI model. On the other hand, selection bias refers to the biases introduced during the data collection and preparation process, which may not accurately represent the real-world scenarios and contribute to biased outcomes.

Biases can be further propagated throughout the AI model's training cycles and persist even after the model is deployed in the real world. Distribution shifts can lead to unexpected biases emerging during deployment. It's crucial to continuously monitor and address these biases to ensure fairness and accuracy in the AI system's predictions for all users.

One crucial aspect of bias mitigation is evaluating the model's performance accurately. While a model may demonstrate high accuracy overall, it might not perform as well on specific subgroups of the population. If we solely rely on evaluation metrics that do not include testing on subgroups, we risk facing evaluation bias, which can perpetuate disparities and further marginalize certain user groups.

To create more reliable and fair AI-based systems, it's essential to address bias at each stage of the AI lifecycle and incorporate thorough evaluation processes that consider the performance across diverse subgroups.

4.4.2.2 Out-of-Distribution (OOD)

Machine learning models are typically trained with the assumption of a closed-world scenario, where the test data, denoted as $p_{test}(X, y)$, is drawn independently and identically distributed (i.i.d.) from the same distribution as the training data, denoted as $p_{train}(X, y)$. However, in real-world scenarios, when these models are deployed, they might encounter test samples that come from a different distribution.

$$p_{test}(X, y) \neq p_{train}(X, y)$$

In the presence of such a divergence between the training and operational distributions, it is essential for the system to be able to detect and raise an alarm. The reason behind this is that the performance of the machine learning model may no longer match what was initially measured during the training phase. This could lead to unexpected outcomes and potentially harmful consequences in real-world applications. It has been observed in [60] that neural networks do not generalize under distribution shift on Imagenet data and that accuracy drops with increasing shift.

The distributional shifts between training and operational phases can be caused by several factors, which include:

- Semantic Shift: New classes may manifest during the testing phase that were not present during training. The model needs to be able to handle these new classes gracefully.
- Covariate Shift: In this case, the distribution of input data $p(x)$ changes, while the conditional distribution of labels given inputs $p(y|x)$ remains fixed. Covariate shift can occur due to adversarial attacks, where data samples are modified intentionally to cause the machine learning model to fail with high confidence. It can also happen due to corrupted data, where unwanted changes occur in the data.
- Label Shift:

  This occurs when the distribution of labels,$p(y)$, changes, while the conditional distribution of the input given the label, $p(x|y)$, stays the same. It commonly takes place when the data is inaccurately labeled, leading to an unexpected label distribution.

In the following, we will explore the concept of covariate shift in more detail, focusing on two specific cases: adversarial attacks and corrupted data

Covariate Shift: Adversarial attack. For a long time, the most universal way to measure the quality of a trained learning model has been the empirical error on testing samples which has been the sole focus of researchers. Around 2013 a remarkable paper by Szegedy et al. [61] warns against intriguing behavior of some classification models. In fact, despite their excessive accuracy, they show a worrying instability. This was illustrated by the ability to make them predict false results and with probabilities close to 1. Since then, the greatest challenge for researchers has become the tracking of this kind of disturbance that drives the models crazy and especially how to vaccinate AI models depths of these attacks. In this section we recall the great episodes in this research and which led to more robust models, but also to new challenges.

The main results from the Szegedy work [61] are the following counterintuitive properties of neural networks :

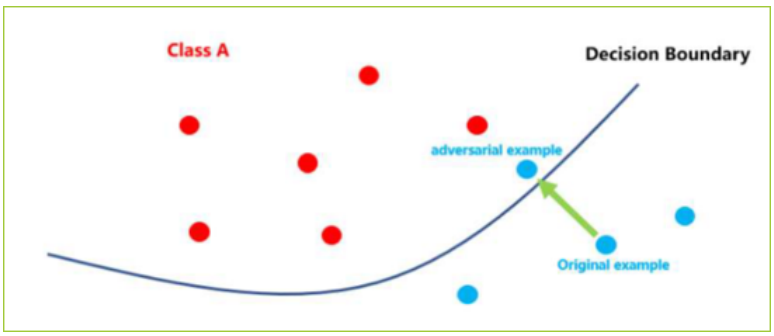


Figure 23: Principle of adversarial perturbation: Find adversarial examples near the decision boundary

1. . The existence of adversarial examples suggests that being able to explain the training data or even being able to correctly label the test data does not imply that our models truly understand the tasks we have asked them to perform. Instead, their linear responses are overly confident at points that do not occur in the data distribution, and these confident predictions are often highly incorrect.
2. The semantic information in the high layers of neural networks are contained in the space (multidimensional), rather than the individual units. This implies the absence of local generalization at unit level and so local imperceptible deviations from a data point in the input space can cause the neural network to change its prediction.
3. Deep neural networks learn input-output mappings are fairly discontinuous to a significant extend. Thus, it is possible to cause the network to misclassify an image by applying a certain imperceptible perturbation, which is found by maximizing the network's prediction error.
4. The specific nature of these perturbations is not a random artifact of learning: the same perturbation can cause a different network, that was trained on a different subset of the dataset, to misclassify the same input.

Since these interesting observations, much work has turned to researching increasingly shocking examples to illustrate this failure. These techniques are called adversarial attack: one type of attack is to attempt to perturb a data point x_0 to another point x_1 in the same space such that x_1 belongs to certain target adversarial class. For example if x_0 is a feature vector of a car image, by adversarial attack we meant to create another feature vector x_1 which will be classifier as a person (or another class specified by the attacker). In some scenarios, the goal may not be to push initial input to a specific target class, but just push it away from its original class or understand how the model works. Since 2013, the main adversarial attack families may be resumed as follows.

1. On basis of model : On basis of Threat model
   (a) White Box Attacks: attacker has access to the model's parameters
   (b) Black Box Attacks: attacker has no access to the model's parameters
2. On basis of Objective
   (a) aim is to enforce the model to misclassify adversarial inputs

(b) aim is to get the image classified as a specific target class different from the true class

(c) aim is to reverse Engineering the model in order to either reconstruct the model or extract the data that it was trained on

3. On basis of Distance metrics

(a) L_0 attacks: minimize the total number of different input features between clean and adversarial inputs

(b) L_2 attacks: minimize the square difference between input features of clean and adversarial inputs

(c) L_∞ attacks: minimize the maximum input feature difference between clean and adversarial inputs

Since 2013, a large number of adversarial attacks have been introduced. These attacks have become a significant research topic, as evidenced by the increasing number of articles published each year in this field. This research has led to the development of more sophisticated attack methods, such as APGD [62], which are faster and more effective than previous approaches.

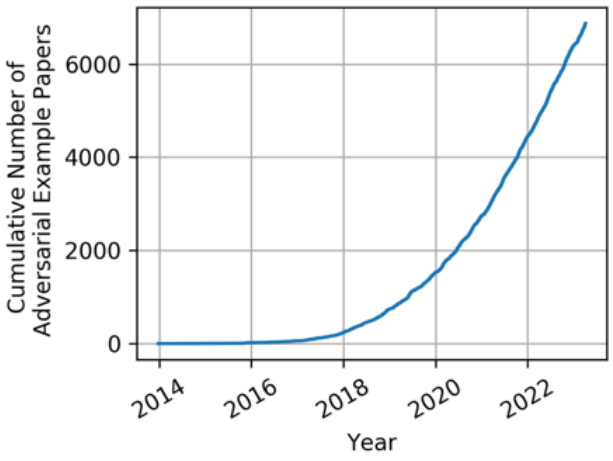


Figure 24: Cumulative number of adversarial example papers

We present in the following, in chronological order, the main approaches that have improved the state of the art by further reducing accuracies while simplifying and accelerating the generation of examples.

1. White-box attacks

- Szegedy formulation 2013 [61]
- Fast gradient sign method: FGSM: Goodfellow et al. 2014 [63]
- DeepFool: Moosavi-Dezfooli et al. 2015 [64])

- Jacobian-based saliency map attack: JSMA, Papernot et al. 2016 [65]
- Universal adversarial perturbations, Moosavi-Dezfooli et al. 2017, [66]
- One Pixel Attack for Fooling Deep Neural Networks, Su et al. 2017 [67]

2. Black-box attacks

- Practical Black-Box Attacks, Papernot et al.2017 [68]
- ZOO: Zeroth Order Optimization Based Black-box Attacks, Chen et al.2017, [69]

Adversarial attacks extend beyond digital manipulations and can be executed through physical perturbations. This method involves the use of tangible, external objects - commonly patches or stickers - to cause disruptions. These tangible attacks are particularly alarming as they can be executed in the real world, posing a direct threat to the reliability and robustness of AI systems operating in physical environments.

Physical perturbation attacks leverage the vulnerability of machine learning models to imperceptible alterations in their input data. By strategically placing patches or stickers on objects or surfaces, adversaries can exploit these vulnerabilities and deceive AI systems into misclassifying or misinterpreting the environment. These physical perturbations are carefully designed to appear benign to the human eye but have a significant impact on the AI's decision-making process. This exploitation of machine perception vulnerabilities can impact a broad range of applications, from facial recognition and surveillance systems to self-driving vehicles.

To execute such attacks, attackers usually employ sophisticated optimization algorithms to find the optimal location and size of the patches or stickers. The goal is to maximize the impact on the model's output while ensuring that the changes are minimal enough to avoid detection by human observers. One critical implication of physical adversarial attacks is the potential for real-world consequences. For example, an adversarially perturbed stop sign could be misclassified by an autonomous vehicle as a speed limit sign, leading to hazardous traffic incidents

Figure 25: Stickers on stop signs [13].

In regression tasks, unlike classification tasks, there are no natural margins for decision boundaries, which makes adversarial learning more challenging. Defining adversarial attacks, evaluating their success, and establishing appropriate evaluation metrics pose difficulties in the regression setting. Despite the growing number of works on adversarial attack generation, research specifically focusing on regression tasks remains relatively limited.

One notable contribution by Tong et al. (2018) explored adversarial attacks in the context of an ensemble of multiple learners. They investigated the interactions between these linear learners and an attacker in the regression setting, modeling it as a Multi-Learner Stackelberg

Game (MLSG). However, the use of linear models in this work limits its ability to address the larger class of non-linear models commonly encountered in real-world scenarios. On the other hand, Ghafouri et al. (2018) addressed an important problem concerning the selection of optimal thresholds for each sensor against adversaries in regression tasks within cyber-physical systems. This work sheds light on practical aspects of adversarial defenses in the context of regression tasks. In a different context, Deng et al. (2020) introduced the concept of adversarial threshold, which relates to the allowable deviation between the original prediction and the prediction of an adversarial example. This concept is particularly relevant in the context of driving models, where an acceptable error range must be defined to ensure safety and stability.

Covariate Shift: Corrupted data. Corrupted data can arise due to various non-malicious reasons such as errors in data collection, sensor noise, data transmission issues, or natural variations in the data over time. These alterations can introduce noise or outliers, disrupting the normal distribution of the data. As a result, the AI system may encounter unseen or unexpected patterns during testing, leading to decreased performance and inaccurate predictions.

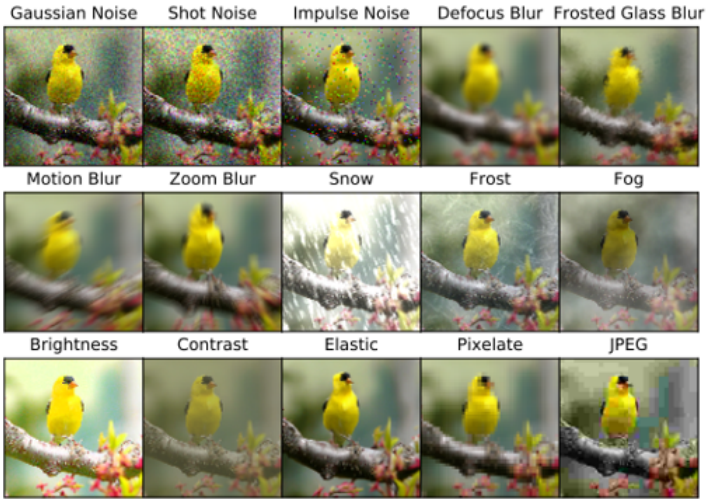


Figure 26: 15 types of algorithmically generated corruptions from noise, blur, weather, and digital categories

The first common corruption is Gaussian noise. This corruption can appear in low-lighting conditions. Shot noise, also called Poisson noise, is electronic noise caused by the discrete nature of light itself. Impulse noise is a color analogue of salt-and-pepper noise and can be caused by bit errors. Defocus blur occurs when an image is out of focus. Frosted Glass Blur appears with "frosted glass" windows or panels. Motion blur appears when a camera is moving quickly. Zoom blur occurs when a camera moves toward an object rapidly. Snow is a visually obstructive form of precipitation. Frost forms when lenses or windows are coated with ice crystals. Fog shrouds objects and is rendered with the diamond-square algorithm. Brightness varies with daylight intensity. Contrast can be high or low depending on lighting conditions and the photographed object's color. Elastic transformations stretch or contract small image regions. Pixelation occurs when up-sampling a low-resolution image. JPEG is a lossy image compression format that increases image pixelation and introduces artifacts. Each corruption type may be tested with depth due to its five severity levels, and this broad range of corruptions allows to test model corruption robustness with breadth. Moreover, one adversarial attack is to

modify in purpose several bits or adding an imperceptibly small vector in order to misclassify adversarial inputs.

To ensure the robustness and reliability of machine learning models in an open-world setting, it is crucial to develop techniques that can identify and handle out-of-distribution samples effectively, as well as detect and adapt to distributional shifts that may occur during deployment. This will enable the models to perform consistently and safely in real-world applications

4.4.2.3 Debiasing

There are various debiasing approaches to address biases that may be present in the data. One effective approach to mitigate class imbalance is sample re-weighting. Instead of uniformly sampling from the dataset, we sample at a rate inversely proportional to the incidence of each class. By doing so, we give higher importance to underrepresented classes, allowing the model to focus more on learning patterns from these scarce samples, which can improve its ability to recognize and predict minority classes accurately.

Another technique to address class imbalance is loss re-weighting. Rather than treating all mistakes made by the model equally, we assign different weights to samples based on their class representation. Samples from underrepresented classes are assigned higher weights, making their misclassifications contribute more significantly to the total loss function. This enables the model to prioritize learning from these samples and better adjust its decision boundaries to account for the minority classes' characteristics.

Lastly, batch selection can be employed to tackle class imbalance. In this approach, we randomly choose samples from each class to form a batch, ensuring that every batch contains an equal number of data points from each class. This balanced representation in each batch helps the model receive a fair and representative distribution of data during training, reducing the risk of the majority class dominating the learning process.

Even in cases where the classes are completely balanced, there can still be other forms of bias present. While we've successfully addressed the issue of underrepresented classes, we must now focus on the problem of variability within the same class, particularly when there is feature imbalance. One approach to mitigate this bias is to employ a de-biasing algorithm that utilizes the latent features learned by a variational auto-encoder (VAE) to perform under-sampling and over-sampling within our dataset.

To begin, we need to train a VAE using the provided dataset to learn the underlying latent features. Once we have successfully captured the latent structure, we can use it to calculate a distribution of the inputs across each latent variable. This distribution helps us identify areas within our data where the density is high and others where it is sparse.

By having this distribution information, we can now make informed adjustments to our dataset. Specifically, we can under-sample samples belonging to the denser areas of the distribution and over-sample data points from the sparser regions. This process ensures a more balanced representation of the data while preserving the overall integrity of the information present in the latent features

4.4.2.4 Robustness

Before discussing robustness in AI models, let's first distinguish between accuracy and robustness. Accuracy in classification refers to the fraction of inputs that the model correctly classifies. Given a dataset, accuracy measures the percentage of observations for which the

model correctly identifies the class. In contrast, accuracy in regression refers to the closeness of the model's predicted values to the actual target values for the given dataset.

Robustness, on the other hand, quantifies the model's ability to maintain its accuracy or predictive accuracy when the input data is slightly modified. More precisely, robustness measures the fraction of input points x for which the predicted class (in classification) or predicted value (in regression) is close to the actual class or value, respectively, for all input points y belonging to a ball of radius ε around x. This notion of robustness depends on the norm L_p and the intensity of perturbation ε.

It's important to note that in both classification and regression, adversarial examples have revealed that models with high accuracy can have a zero robustness score. Adversarial examples are carefully crafted inputs with imperceptible modifications that lead the model to misclassify (in classification) or make inaccurate predictions (in regression).

To mitigate adversarial attacks, various defense methods have recently been proposed. These can be broadly classified into two categories: (a) Reactive defenses that modify the inputs during testing time, using image transformations to counter the effect of adversarial perturbation, and (b) Proactive defenses that alter the underlying architecture or learning procedure e.g. by adding more layers, ensemble/adversarial training or changing the loss/activation functions. Proactive defenses are generally more valued, as they provide relatively better robustness against white-box attacks. Nevertheless, both proactive and reactive defenses are easily circumvented by the iterative white-box adversaries. In this paragraph, we recall the main defense technique that reached the state of the art robustness performances.

1. Minmax Mardy et al. formulation, 2017 [70] : The key point is that the loss function we usually minimize is jut a proxy to improve accuracy of the model. Recall that the accuracy is the fraction of inputs which are correctly classified however, the robust accuracy is the faction of inputs such that the predicted class remains unchanged on a ball around it. The size and form of this stable neighborhood depends on the used norm and the intensity of the supported perturbation. It is worth noticing that adversarial examples have shown that highly accurate models may have zero robust accuracy scores. Thus, in [70], the author proposed to minimize the maximum of the loss on a given ball instead of minimizing the loss. So, the adversial training solution may be estimated as

$$\min_{w} \max_{x \in B(x,\epsilon)} Loss(x, y, W)$$

instead of

$$\min_{w} Loss(x, y, W)$$

.

However, the estimation of $\max_{x \in B(x,\epsilon)}$ is not an easy task. In [70], it has been proposed to estimate the max on an L_∞ ball using an ascent projected gradient PGD. Once a solution x_{max} is estimated, the network weights are optimised as usually but with fixed input : $\min_w Loss(x_{max}, y, W)$. The obtained results suggest an significant improve in the robustness sore while keeping high accuracy.

2. Robustness and accuracy tradeoff formulation : TRADES, Zhang et al. 2019 [**?**] :TRADES is based on the following observation. Accuracy by itself is not good since adversarial examples easily fool the model and, on the other hand, robustness by itself is not good since constant models are robust but not relevant. Thus, the best solution have to be a

tradeoff between both measures. The obtained algorithm, called TRADES, try to solve the following equation :

$$\min_{w}\{Loss(x, y, W) + \lambda \max_{\tilde{x}\in B(x,\epsilon)} KL(f(x), f(\tilde{x}))\}$$

3. Universal Adversarial Training, Shafahi et al. 2019: [71] : This is a defense strategy against universal adversarial perturbations (UAP). It showed better results against UAP than previous defenses including PGD adversarial training. It also worth noticing that contrary to other defense approaches not easily scalable for large data sets, the UA training scales better to ImageNet than adversarial training because in this case one adversary is constructed for many images.
4. Randomized Smoothing (Cohen et al. 2019) [72]: This method focuses on constructing robust models for both regression and classification tasks. It consists in training a foundational model using Gaussian data augmentation. Following this, a smoothing function is applied to this base model to develop a new predictive model,

   In the context of classification tasks, randomized smoothing guarantees a specified level of accuracy within a certain radius, where perturbations are constrained. This implies that the model's predictions are ensured to maintain their accuracy for inputs falling within this defined radius. Similarly, in the case of regression tasks, randomized smoothing generates an interval where the prediction is assured to fall, thereby providing a measure of certainty for the output.
5. Lipschitz model Szegedy et al., 2014, Goodfellow et al., 2015: Lipschitz constraint is a property that characterizes a function's behavior by ensuring that a small change in its input results in a small change in the output. In the context of neural networks, this means that a slight perturbation to the input data should lead to only minor fluctuations in the model's predictions.Mathematically a function f is said to be Lipschitz when the norm of its first derivatives is bounded by some constant L which it's minimum value is called the Lipschitz constant of the function.

   $$\|f(x_1) - f(x_2)\| \leq L\|x_1 - x_2\| \quad \forall x_1, x_2,$$

   This means that if x_1 and x_2 are close to each other, then their predictions $f(x_1)$ and $f(x_2)$ will be close to each other too.

   Formally, for a neural network model, maintaining a Lipschitz constraint is crucial for enhancing robustness. It limits the sensitivity of the model to small changes in the input and can be particularly beneficial for classification tasks. When a classification model adheres to a Lipschitz constraint, a slight modification in the input data will correspond to a proportionate adjustment in the classification logits. As a result, the model becomes less susceptible to noise and minor variations in the input.

   However, the significance of Lipschitz constraint is not limited to classification alone. It can also be extended to regression tasks. In regression, the model's objective is to predict continuous values, and the Lipschitz constraint plays a similar role in ensuring that small changes in the input data lead to only marginal changes in the output predictions.
6. Convex Outer Adversarial Polytope, Wong et al. 2017, [73]
7. Data dependent Randomised smoothing, Alfaraa et al. 2020, [74]

8. Protecting classifiers against adversarial attacks using generative models, Samangouei et al. 2018, [75]

   To conclude on the subject of adversarial attacks and defenses, we can say that there exists a real ping-pong game, or an endless cycle between these two aspects 27. Indeed, each time a defense is established, within a few months a new adversarial attack is created to bypass it. This situation has been repeating itself for many years, with many different cycles.

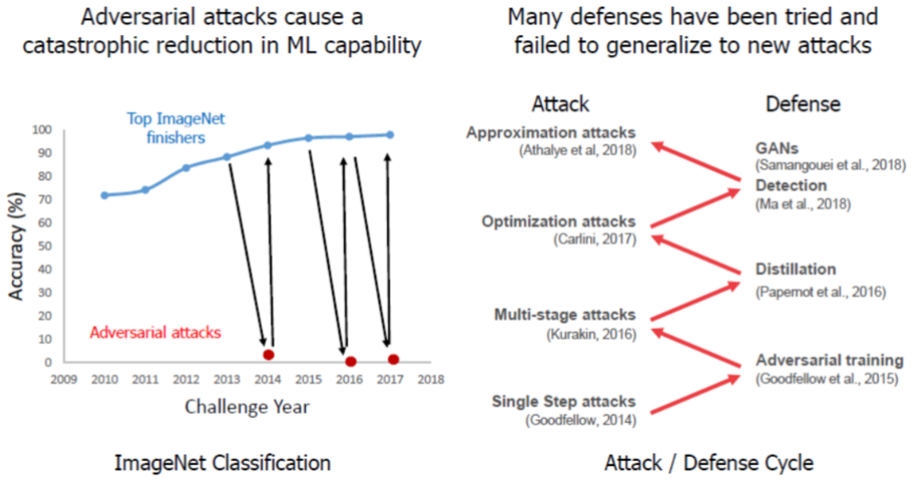


Figure 27: Attack vs Defense

In the 2020 study "Benchmarking Neural Network Robustness to Common Corruptions and Perturbations [76]" conducted by Hendrycks et al., it was observed that the robustness of neural networks against adversarial attacks does not ensure their robustness against everyday corruptions.

Nevertheless, the researchers proposed an effective solution in this paper to bolster the robustness of neural networks against such prevalent corruptions. They suggested a methodology to enhance the resilience of neural networks against a wide array of image corruptions, emphasizing the adaptability of the AI system.

4.4.2.5 Uncertainity quantification

Now, let's consider we've applied every available method to construct a robust AI model. Our next step would be to examine whether our model can identify out-of-distribution (OOD) inputs. It's crucial to remember that a reliable AI in high-risk domain must not only be robust but also capable of detecting OOD. To achieve this, we'll delve into uncertainty quantification

The main questions we will try to answer in this section are :

- What is the uncertainty in the machine learning context?
- How to o quantify it? How to evaluate it? What method and scores to do it?
- What the "uncertainty evaluation" is useful for ?

Uncertainty means working with imperfect or incomplete information. It is fundamental to the field of autonomous car, yet it is one of the major obstacles that could delay the secure commercial deployment of autonomous driving system on a large scale. We can identify two sources of uncertainty in AI models: epistemic and aleatoric uncertainty.

Epistemic uncertainty is the uncertainty represented in the model parameters and captures the ignorance about the models most suitable to explain our data. This type of uncertainty can be reduced with additional training data and therefore carries the alternative name “reducible uncertainty”. A model will broadcast high epistemic uncertainty for inputs far away from the training data and low epistemic uncertainty for data points near the training data.

Aleatoric uncertainty captures noise inherent to the environment i.e., the observation. Compared to epistemic uncertainty, this type cannot be reduced with more data but with more precise sensor output. This two-levels uncertainty can theoretically help users understand if a model is incorrect because it lacks data or because the example is intrinsically ambiguous. The third type is called predictive uncertainty which is the conveyed uncertainty in the model’s output. Predictive uncertainty can combine epistemic and aleatoric uncertainty.

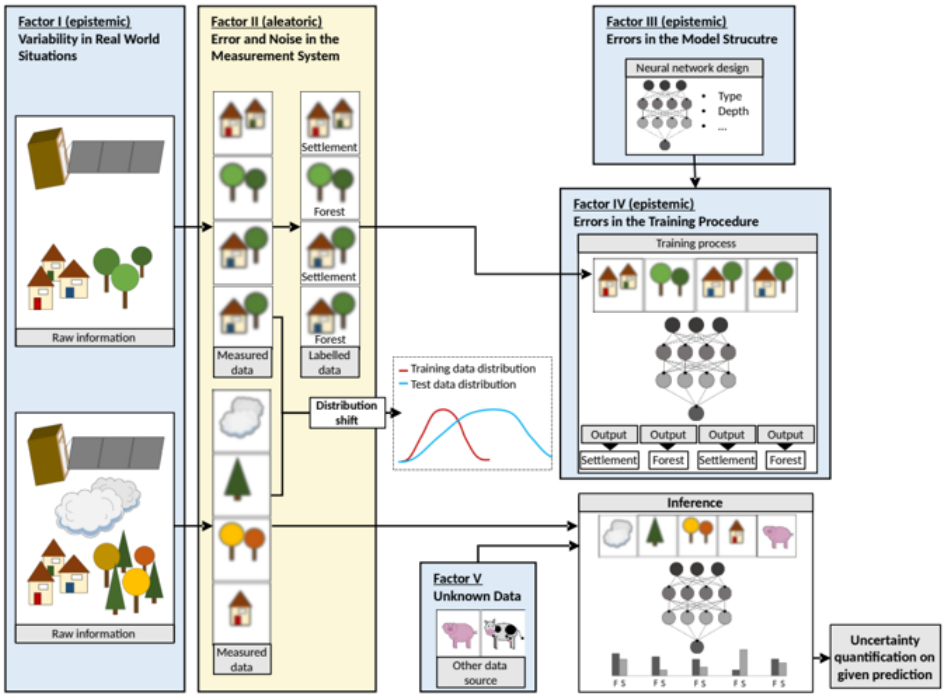


Figure 28: Epistemic and aleatoric uncertainty [14]

In classification tasks, quantifying uncertainty entails providing an output class label along with the confidence level associated with the prediction. This confidence is often represented as a probability, which indicates the likelihood of the data point belonging to the assigned class.

In regression tasks, quantifying uncertainty involves not only providing the predicted mean output value but also offering a measure of the uncertainty associated with this value. This measure of uncertainty can be expressed as either the variance or the standard deviation of the prediction. These metrics provide an indication of how the predicted values are dispersed around the mean.

The Uncertainty estimation may be useful for:

- Knowing when to trust model's predictions, especially under dataset shift
- Better decision making: Calculating the risk vs reward associated with prediction (worst case vs average case)
- Active learning: Getting more data in regions where the model is uncertain
- Open set recognition
- Lifelong learning
- Exploration in reinforcement learning

In this document we will focus on one main application : validate robustness of models on tasks in connection with uncertainty estimation. Valid models according to this approach are "excepted to know what they don't know". They are supposed to notice when they are unsure about a prediction. This could be partially achieved if it is able, while maintaining high prediction score, to detect the existence and intensity of the following disruptions and corruptions for which work has already been done :

- Detecting adversarial examples.
- Distinguishing between in-distribution (ID) and non-nominal data.
- Detecting common corruptions and perturbations

It is imperative to distinguish between two closely related properties: model robustness and model uncertainty. Model robustness refers to its ability to make accurate predictions even in the presence of perturbations and corruptions. On the other hand, model uncertainty pertains to how well the model's uncertainty reflects the presence and intensities of those disturbances.

The ideal validation for a model involves verifying its robustness in making correct predictions while detecting the presence and intensity of difficulties. This validation should continuously hold true until the point where the model becomes unable to predict accurately. In such cases, the model should detect and inform about its limitations promptly. This concept is encapsulated by the famous phrase, 'models have to know what they don't know.'

In practical terms, we introduce 'calibration error' as a metric to evaluate a model. Calibration error represents the difference between a model's expressed confidence in its predictions and the actual accuracy of those predictions.

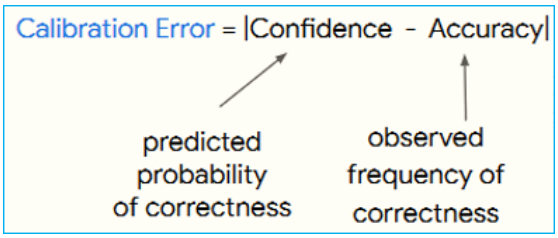
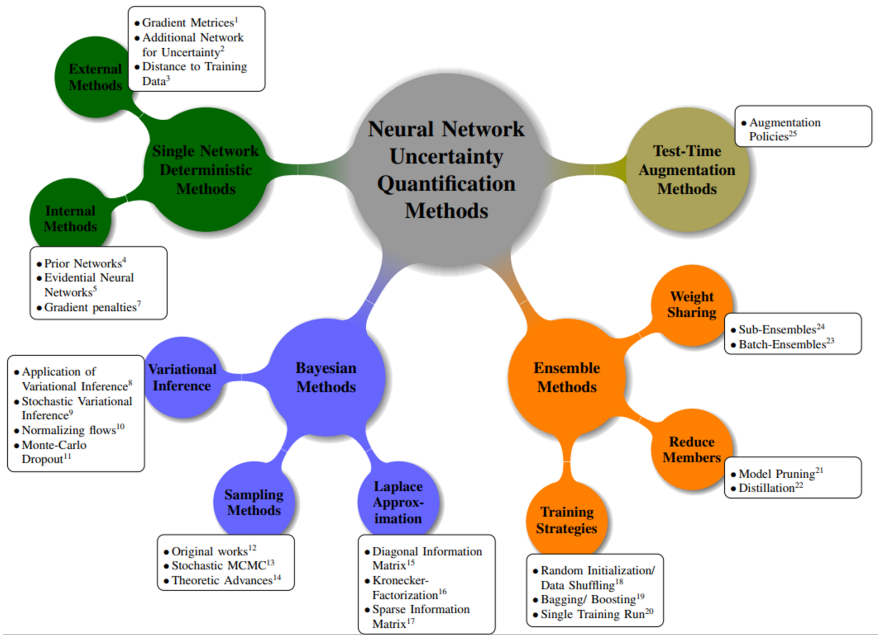


Figure 29: Calibration error

Essentially, it measures how well the model's confidence aligns with reality. The relationship between accuracy and confidence is proportional: as accuracy increases, confidence should also increase, and vice versa. Therefore, to validate our AI model effectively, we expect the calibration error to remain close to zero.

If the calibration error deviates significantly from zero, our AI model might be either overconfident or under-confident, indicating a lack of proper calibration. In such cases, it becomes crucial to improve the quality of uncertainty by calibrating the AI model



Extract from Gawlikowsk et al. 2022

Figure 30: Visualization of the four different types of uncertainty quantification methods [14]

4.4.2.6 Calibration

Calibration is a vital method used to assess the confidence of a prediction model in its predictions relative to their actual accuracy. In simpler terms, it evaluates how well a model's predicted probabilities align with the real-world frequency of the events it predicts. It has been proved that there exists a strong correlations between adversarial robustness and calibration. In fact, it has been found across many datasets that adversarially unrobust data points, where small adversarial perturbations to the input are able to fool the model into wrong predictions, are more likely to have poorly calibrated and unstable predictions.This has lead to new use for adversarial robustness as a means to more generally improve model trustworthiness, not just by limiting adversarial attacks but also improving calibration and stability on unexpected data. More, the adversarial example defense can yield substantial robustness gains on diverse and common perturbations corruption.

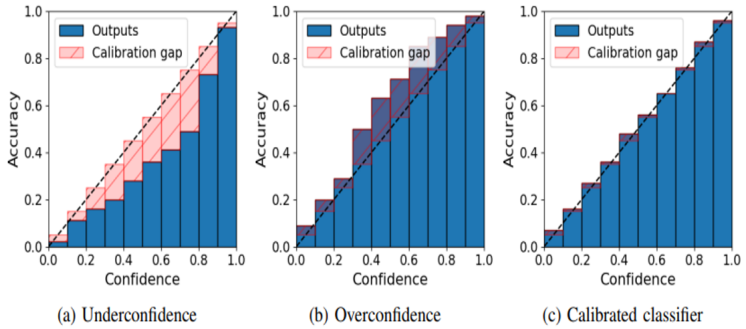
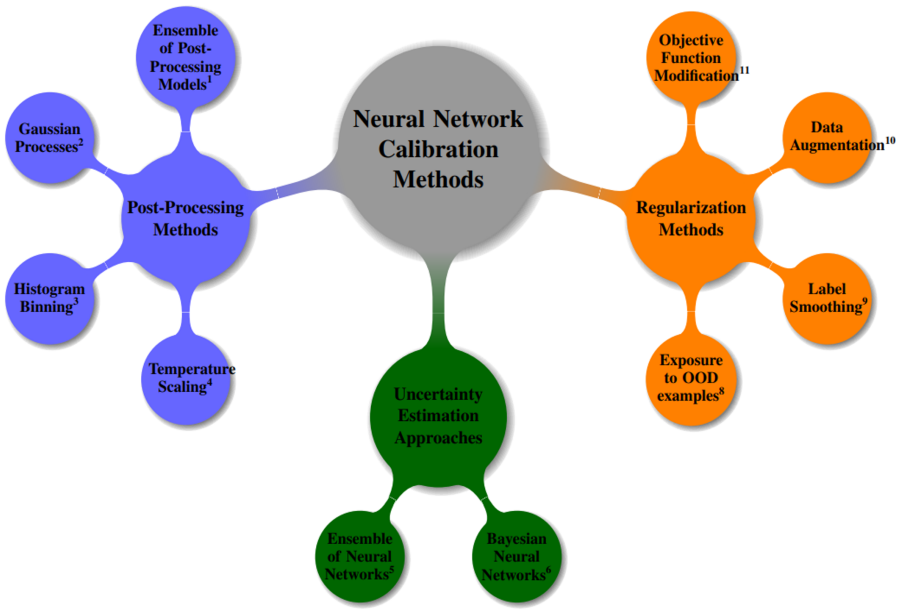


Figure 31: Temperature Scaling : Instead of computing the Softmax, all the logits (values just before the final activation, here Softmax) are divided by the same value called temperature. [14]

Calibrated predictive uncertainty is important because it enables accurate assessment of risk, allows practitioners to know how accuracy may degrade, and allows a system to abstain from decisions due to low confidence. It involves re-calibration of probabilities on a held-out validation set through temperature scaling which was shown by Guo et al. [77] to lead to well-calibrated predictions on the i.i.d. test set. Temperature Scaling (TS) is in fact a state-of-the-art among measure-based calibration methods which has low time and memory complexity as well as effectiveness.



Extract from Gawlikowsk et al. 2022

Figure 32: Visualization of the different types of uncertainty calibration methods [14]

4.4.2.7 Active learning : an application of uncertainty estimation

Beside AI safety, there exist many applications which rely on model uncertainty. These applications include choosing what data to learn from, or exploring an agent's environment efficiently. Common to both these tasks is the use of model uncertainty to learn from small amounts of data. This is often a necessity in settings in which data collection is expensive (such as the annotation of individual examples by an expert), or time consuming (such as the repetition of an experiment multiple times). In this section, we will focus on the active learning technique in the context of computer vision with deep learning. This approach approach stems from the observation that there's no need to annotate all the data because most instances are not informative to give better performance [78]. Therefore, one can construct a strategy to select and annotate a smaller, but informative subset of the abundant unlabeled data to train the model. This process is usually employed iteratively, i.e. selecting and annotating a number of subsets in so-called cycles and re-training the model on the so-far collected data. Moreover, an active learning strategy usually assumes that the output from a model trained on data from previous cycles can be used to select a subset for annotation in the current cycle. Such informed query-based procedure motivates the name active learning.

4.4.2.8 Active learning depends on the learning task

In the active learning context, the significance of instances depends on their importance in the learning process, i.e. their improvement of the predictive capacity of the model. Thus, the measure of this weight, necessarily depends on the nature of the task. In the context of classification, it is a question of measuring the difficulty that the model encounters when assigning a class to a given image. This uncertainty could be measured using the entropy for example or simply using the maximum probability over all the classes : the lower this value, the more informative the instance. In the context of regression, the proposed ideas are focused on the dispersion of both the dependent (Y) and independent (X) variables. This allows to detect most of the variability in the regression model and thus reduce its variance. Finally, in the detection case, to the best of our knowledge, there is no consensus on a measure to estimate the totality of the information provided by an instance. Indeed, optimizing the model in this case, consists in increasing its capacity to detect objects, find their corresponding classes and also their bounding boxes. Moreover, this should be measured over all detected objects of each image. Reasoning in this way, several problems arise. First, should one or two criteria be privileged or should all three be treated together? Second, How would it be possible to aggregate heterogeneous scores? Next, should we aggregate the three criteria on each object and then find an aggregation on the whole image or aggregate each score over all objects first? Finally, which aggregation criterion would be suitable for each measure?

Most of these questions remain open today. All the proposed works are limited to an aggregation of a single measure (often of classification) on all the instances while ignoring all the others. In this document, we start by describing the main active learning approaches in the context of classification, then regression and finally we consider the case of detection.

4.4.2.9 Active Learning for classification

The key idea is always conceived on the notion of uncertainty, i.e. how uncertain the model is about the class to predict for a given image. In the following , we list a not exhaustive list but we cover the great majority of the confirmed approaches having given significant results on classic data sets

1. Uncertainty Sampling : One way to reduce labeling cost is to identify the data points that the underlying model finds most difficult to classify and provide labels only for those. We score a data point as simple or complex based on the soft-max output for that point. Suppose the model has N output nodes and each output node is denoted by z_j. Thus, for an output node z_i from the model, the corresponding soft-max would be

$$p_i := \frac{\exp z_i}{\sum_{j \in 1:N} z_j}, \text{ and for the predicted class c : } p_c = \arg\max_i \{p_i\}$$

.

Using those probabilities, many approaches may be defined :

- Least Confidence : the probabilities are used to pick elements for which the model has the lowest confidence, i.e. the smallest p_c
- Margin Sampling then margin sampling would pick elements using the distance between the two largest probabilities in each image.

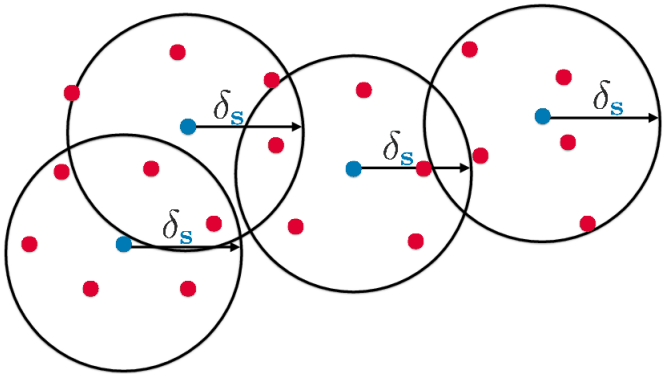


Figure 33: Corest selected points

- Entropy : then Entropy sampling would pick elements using with larges values of : $-\sum_i p_i \log(p_i)$

It is well known that both least confidence sampling and margin sampling pick some data points that have pairwise confusion however entropy focuses on the data points which have confusion among most of the labels.

2. BADGE [79]: Batch Active learning by Diverse Gradient Embedding (BADGE) samples groups of points that are disparate and high magnitude when represented in a hallucinated gradient space, a strategy designed to incorporate both predictive uncertainty and sample diversity into every selected batch. This allows it to trades off between uncertainty and diversity without requiring any hand-tuned hyper-parameters. Here at each round of selection, loss gradients are computed using the hypothesized labels. While other approaches sometimes succeed for particular batch sizes or architectures, BADGE consistently performs as well or better, making it a useful option for real world active learning problems.
3. Adversarial Techniques: they are motivated by the fact that often the distance computation from decision boundary is difficult and intractable for margin-based methods. Adversarial techniques such as Deep-Fool, BIM(Basic Iterative Method) [80] etc. have been tried out in active learning setting to estimate how much adversarial perturbation is required to cross the boundary. The smaller the required perturbation, the closer the point is to the boundary.
4. CORESET [81]: This technique tries to find data points that can represent the entire data set. For this, it tries to solve a k-Center Problem on the set of points represented by the embedding obtained from the penultimate layer of the model. Embedding from the penultimate layer can be thought of as the extracted features, therefore, solving the k-Center Problem in this new feature space can help us get representative points. The idea in Coreset strategy is that if those representative points are labeled, then the model will have enough information. For example, as illustrated in figure 33, Coreset strategy would select the blue points if the union of red and blue points were given as input and the budget was 4.
5. FASS [82]: Filtered Active sub-modular Selection (FASS) combines uncertainty sampling idea with Coreset idea to most representative points. To select the most representative points it uses a sub-modular data subset selection framework to select a subset based

on uncertainty sampling using the sub-modular functions as 'facility location', 'graph cut', 'saturated coverage', 'sum redundancy' and 'feature based' to we select a subset of images. Here sub-modular functions are often used to get the most representative or diverse subsets.

6. GLISTER-ACTIVE [83]: performs data selection jointly with parameter learning by trying to solve a bi-level optimization problem. First, an inner level optimization very similar to the problem encountered while training a model except that here the data points used are from a subset. Therefore this tries to maximize the log-likelihood with the given subset. Next, an outer level Optimization which is also a log-likelihood maximization problem. The objective here is to select a subset S that maximizes the log-likelihood of the validation set with given model parameters. This bi-level optimization is often expensive or impractical to solve for general loss functions, especially when the inner optimization problem cannot be solved in closed form. Therefore, instead of solving the inner optimization problem completely, a one-step approximation is made while solving the outer optimization.

By comparing the performance of these active learning algorithms against the strategy of randomly selecting points to label, the labeling efficiency of these active learning algorithms becomes clear. Here are some of the results obtained on common datasets using some of the active learning algorithms:

1. CIFAR 10 : the best strategies show 2x labeling efficiency compared to random sampling. BADGE does better than entropy sampling with a larger budget, and all strategies do better than random sampling.
2. CIFAR 100 : all strategies exhibit a gain over random sampling, but the per-batch version of BADGE performs similarly to random sampling. (Regular BADGE does not scale to CIFAR-100!)
3. MNIST : all strategies exhibit a gain over random sampling, and both entropy sampling and BADGE achieve a 4x labeling efficiency compared to random sampling.
4. FASION MNIST : all strategies exhibit a gain over random sampling, and both entropy sampling and BADGE achieve a 4x labeling efficiency compared to random sampling.
5. SVHN : all strategies exhibit a gain over random sampling, and both entropy sampling and BADGE achieve a 3x labeling efficiency compared to random sampling.
6. Robustness against redundancy : compared to random sampling, all algorithm even entropy sampling handles redundant data poorly while BADGE handles redundant data proficiently.

4.4.2.10 Active Learning for Regression

There is an abundance of literature examining the applicability of active learning to problems of classification. However, the use of active learning for regression has received considerably less attention [84]. Nevertheless, the theoretical capability of active learning to significantly improve the estimation of a function in the presence of noise has been shown by Castro et al. [85]. This study shows that the learning rate may be increased when learning functions "whose complexity is highly concentrated in small regions of space" i.e., functions generally better suited

to kernel-based models. This is due to the ability of active learning to quickly isolate interesting regions of the version space using techniques such as model uncertainty and local variance. Although the study does not find that active learning could provably significantly outperform passive sampling in learning a general function without localized complexity, the goal of data set labeling is not necessarily to approximate a general function, applicable to unseen data, but rather to discover that function which best approximates the finite sample of data in the unlabeled pool. Later, Niyogi [86] showed promising practical results in applying active learning to estimating polynomial functions which do not have this property of localized complexity. On the other hand, a number of active learning selection strategies initially developed for use in classification have been shown to perform well when used in regression problems:

1. Expected Model Change Maximization [87] which aims at choosing the unlabeled data instances that result in the maximum change of the current model once labeled. The model change is quantified as the difference between the current model parameters and the updated parameters after the inclusion of the newly selected examples. In light of the stochastic gradient descent learning rule, the change as the gradient of the loss function is approximated with respect to each single candidate instance. Experimental results on both UCI and StatLib benchmark data sets have demonstrated a clear acceleration in the selection process.
2. Transductive Experimental Design [88] employs statistical techniques from "Optimal Experiment Design" to assess the utility of an instance based on its non-label features. This approach avoids the need to train additional models and reduces the overhead incurred in employing an active learning selection strategy.
3. Query By Committee [89] is an example of an ensemble-based approach to active learning. QBC trains a committee of models using different views of the available data; and selects for labelling the unlabelled instance on which each of the models in this committee most disagree. Burbidge et al. [89] have explored the application of the QBC algorithm to linear regression models, finding it to perform favourably against a random baseline.
4. Expected Gradient Length [90] is a selection strategie that assign an expected utility of each unlabeled instance based on the output of models generated using the currently labeled data. Like QBC, it builds a committee of models using samples of the labelled dataset. However, unlike QBC, unlabelled data is scored on the basis of the disagreement between the aggregated output of the committee on the one hand, and the predicted outcome of a model built on the entire labelled dataset — the output model — on the other. The idea behind EGL is that those instances which maximise the change in the output model are most likely to improve the model's performance.
5. Kernel Farthest-First [91] : is based on farthest-first traversal sequences in kernel space. The KFF algorithm seeks to label the unlabeled instance which is least similar to (i.e. farthest from) the currently labeled data set, with the distance between a point and a set defined as the minimum distance between that point and any instance belonging to the set. The KFF algorithm has been shown to outperform a random baseline on a simple XOr classification problem.
6. Density Based Selection Strategy [92]: density, or closeness to the labelled data, is considered as a selection strategy and implemented as the inverse of the Kernel Farthest-First Diversity algorithm described above to label those instances closest to the currently la-

beled set. Using cosine distance as a similarity measure between instances, density-based selection strategies has not shown as effective as the other approaches described above.

7. Exploration Guided Active Learning [93]: is a classifier-independent approach which offers computational advantages over committee-based alternatives. The EGAL algorithm is a model-free approach to active learning based on a combination of density and diversity measures. Unlabelled data is compared using a similarity measure, cosine for instance, but the approach is independent of the particular measure used. Only those instances which are sufficiently distant from the currently labeled data set (candidate set) are considered for labeling. Within this candidate set, instances are ranked according to their density within the data set as a whole, and those in stances with the greater density are preferred. EGAL works on the assumption that the densest instances are most representative of the current data, allowing EGAL to balance a bias for selecting dissimilar instances for labeling with a resilience to labeling outliers which are not representative of the data as a whole.

4.5 Formal methods

Finally, this document will present some approaches and good practices on the use of formal evidence in the evaluation of VAs.

4.5.1 Foreword

Formal methods are a scientific and technical field aiming to design techniques bringing strong mathematical guarantees on the behaviour of software systems. Applying formal methods to critical industrial software was met with numerous successes. For instance the Paris subway lines 1 and 14 are fully automated; the correct behaviour of their software was proven using the Method B and Atelier B[94]; the Frama-C for C code analysis platform proved the absence of runtime error in critical code for EDF [95]. More globally, saying that formal methods as a scientific discipline and an industrial practice contributed to make software safer is not an understatement [96].

Usually, formal methods can be characterized by their soundness: if a method answers that a property is true, then the property is actually true. One key point is thus to express the specification in a sufficiently unambiguous language. ISO 26262 expresses those languages as formal (or semi-formal) languages. There exist a plethora of formal languages, for there are multiple possible abstraction levels. Examples of formal language are SMTLIB [97], used for SMT solvers; or the ACSL specification language used to specify properties for C languages [98].

Once the specification is available in a formal language, it is then possible to apply a formal technique to obtain a verified answer. See figure 34 for a rough description of formal methods for verification of programs.

Multiple norms describe how to include formal methods in the development process of transportation systems, for instance ISO 26262 and CENELEC 50128. Regarding cybersecurity, the Common Criteria for Information Technology Security Assessment define an Evaluation Assessment Level (EAL), ranging from 1 to 7. EAL5, EAL6 and EAL7 define (Semi)formally Verified Design and Tested, and certify the use of formal verification during the design, development and evaluation phases for a given security target. Among EAL5-7 certified systems are network monitoring software and hardware and embedded execution environments on individual vehicles. Cybersecurity guidelines for software development and assessment (Bureau Veritas - SW200) includes ”the use of formal verification tools to check the absence of code

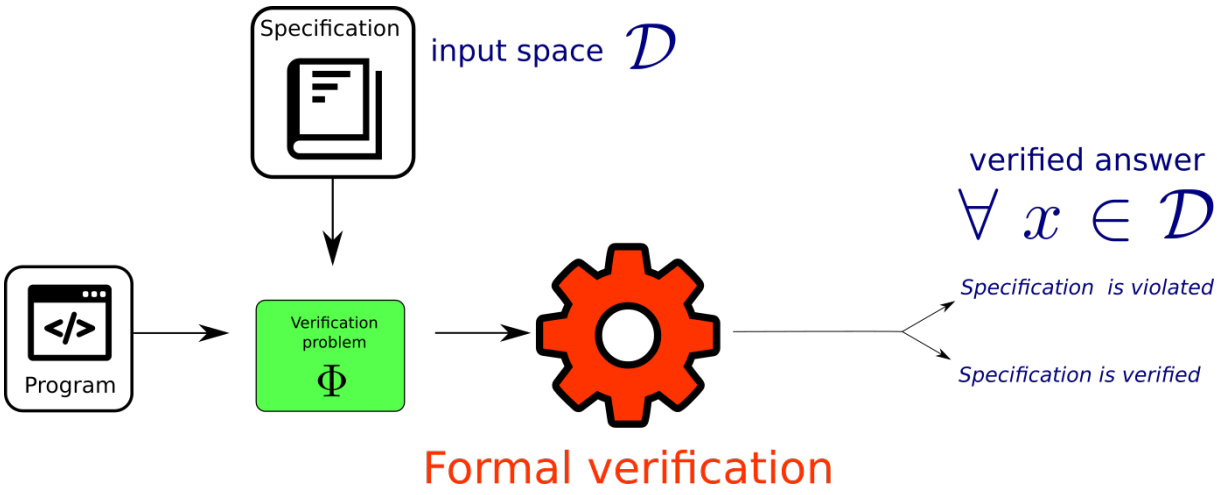


Figure 34: Formal methods prove the behaviour of a component against a formal specification. Adapted from [15].

weaknesses is considered highly efficient, in particular for critical logic components and security functions."

4.5.2 Formal verification of artificial intelligence software: challenges ahead

In the last decade, the boom of data-based machine learning programming and its subsequent diffusion in the software industry and society as a whole led to consider their integration in critical systems, among which are transportation software systems. The very fact that the PRISSMA project exists is a strong evidence of this endeavour.

Data-based machine learning have several key specificities that prevent the direct application of formal methods [15].

4.5.2.1 The specification problem

Expressing a specification for data-based programs is difficult. For instance "a picture with a pedestrian" has no mathematical, unambiguous definition. This comes from the fact that modern machine learning programs only get a specification of their behaviour through examples; they also manipulate complex concepts, that embed culture, education and past political choices that cannot be accurately translated to computers. More precisely, this means that the definition of the Operational Design Domain is ambiguous. This ambiguity is difficult to mix with formal languages. At the moment, formal techniques can check properties that are either local (like adversarial robustness [99]) or functional properties, provided by expert knowledge [100]. There are some preliminary work on formally assessing the fairness of a machine learning program [101].

The CAISAR platform [102], or more generally neuro-symbolic verification[103] aim to embed machine learning programs into formal languages in order to express more complex properties.

4.5.2.2 The misalignment problem

Coverage testing, symbolic execution and trace analysis are common analysis techniques that rely on exploring the execution tree of a program. They rely on the underlying assumption

that this structure yields a semantic. Modern data-based programs - especially neural networks - do not necessarily respect this assumption. The result of a learning procedure is usually a succession of operations on vast arrays of numbers. This sheer size induces a complexity that makes the structure of a machine learning program difficult to make sense of. There is a divorce between the intend of the developer and the resulting program structure.

The whole field of explainable artificial intelligence (xAI) aims to produce interpretable models, either by design or post-hoc.

4.5.2.3 The scalability problem

Verification decision procedures have difficulties to scale to modern neural networks; their depth and the use of non-linear operations are still a challenge even for the most recent tools, like α-β-CROWN [104].

There is an ongoing work in the academic community to standardize the evaluation of formal methods applied to artificial intelligence systems. For instance, the Competition for Verification of Neural Networks VNN-COMP https://sites.google.com/view/vnn2023 aims to evaluate tools on commonly agreed benchmarks. The full report of 2021 venue is available at [105].

ISO/IEC TR 24029-1 defined a taxonomy of methods used to assess the robustness of machine learning programs, including formal methods. The ISO/SAE 21434 (cybersecurity of road vehicle systems) also include a section on formal methods.

4.5.3 Suggestion for requirements

4.5.3.1 Functional properties

guarantee that a program respect a functional property on a whole domain: there is a requirement for a mathematical proof that the program will respect the functional property. Like each proof, there are hypothesis, and those hypothesis can describe for instance the operational domain, a certain class of functional property, and so on.

4.5.3.2 Absence of bugs

guarantee that a program is devoid of a certain class of bugs on a whole domain can be necessary. For instance, ”this program will classify all inputs and their neighborhood similarly on this operational domain”

4.5.3.3 Quality of the dataset

requirements must include the dataset, to ensure a sufficient protection against tempering (such as poisoning attacks [106]), a measure of unwanted biases, its intended purpose [107]. A machine learning program behaviour will largely depends of its datasets; Confiance.ai’s project 5 is solely focusing on data engineering best practices

4.5.3.4 Security and privacy

Neural network present notorious issues for privacy, as it is possible to identify samples used for training [108] or reconstruct said inputs. As of today, to the best of our knowledge there exist no norms that qualify or enforce the level of privacy integrity that a deep learning program must

comply with. The state-of-the art currently rely on Differentialy private learning [109], altough there are some doubts on the actual usefulness of this approach in realistic use cases [110].

4.5.4 Suggestion for a protocol

Overall, there are use cases where needs arise for proving the correctness of systems against a specification; often in critical systems (such as automotive transport). Components that are responsible for perception, decision and planning are potential targets for those requirements.

4.5.4.1 Defining scope

target the scope of where the guarantees are expected: cybersecurity, functional properties, safety of operation

4.5.4.2 Inclusion during design, conception and test

include formal or semiformal methodologies during the design phase, or design the program in a way that eases the formal application of said methods

4.5.4.3 Defining a formal language appropriate to specification

definition of a specification is key. To be checked against, a specification should be phrased into a semiformal or formal notation. This is especially difficult for machine learning systems, as they are expected to work on high-level abstractions that are embedded with social significance that cannot be phrased to a computer. As an illustration, it is impossible to obtain a formal definition of a picture of ”pedestrian”; and defining whether a person in a wheelchair is to be considered a ”pedestrian” or not is subject to human debate that the machine cannot capture

the choice of formal method depends on the properties to check (which is embedded in the formal specification)

4.5.5 Examples of techniques

4.5.5.1 Abstract interpretation and bound propagation

Modern neural networks are computing inputs from high dimensional datasets (images, sounds, text corpuses) to output complex answers - as the probability to belong to a certain class for a classification model, or a sampling over a distribution on a generative model. The size of those input spaces makes methods solely based on sampling very brittle: high dimensional spaces we consider here have counter intuitive properties, like a non-uniform distribution in the input space[111].

Abstract interpretation[112] consists on building an over-approximation of a program's behaviour that is easier to analyze. In the context of neural network verification, the overappoximation is a neural network that handles numerical sets computations - the simple intervals or more accurate but costly zonotopes. Formal verification of a neural network's robustness boils down to the following:

1. express a numerical set describing possible perturbations: for instance a ball centered on a given input x with radius ϵ can be described as an interval $[x - \epsilon, x + \epsilon]$
2. compute this numerical set on the abstract neural network

3. collect the output of the abstract and check whether it satisfies the property at hand (typically, a correct classification result)

Here are some of the most prominent tools that leverage abstract interpretation or bound propagation: the ERAN framework [99, 113, 114], constraint propagation frameworks like α, β CROWN [115, 104]or the nnenum tool [116]. In the context of Confiance.IA first pillar, the tool PyRAT (Python Reachability Assessment Tool) developed by CEA was used successfully to check properties on several industrial use cases.

There exist multiple refinements to bounds propagation. For instance, the tool VeriNet [117] makes use of symbolic variable propagation. Symbolic variable propagation consist on keeping track of the relationships between variables using their symbols. This can be used to refine results of otherwise imprecise numerical domains like intervals.

4.5.5.2 SMT calculus

Satisfaction Modulo Theory (SMT) calculus is a technique that aims to combine the power of Boolean calculus with more expressive theories. Such theories include for instance real numbers arithmetic, arrays or uninterpreted functions. Example of state-of-the-art SMT solvers include Z3 [118]. Such solvers are the result of decades of research, and are able to solve difficult problem instances.

One of their main drawback, however, is their inability to efficiently deal with neural network activation functions. For instance, the sigmoid function $f : x \in \mathbb{R} \mapsto \frac{1}{1+\exp^{-x}}$ make use of the exponential function, which is difficult to model. Rectified linear unit (ReLU) $f : x \in \mathbb{R} \mapsto \max(x, 0)$ a piecewise-linear function. When encountered, such function must be splitted in two linear variant: either $x < 0$, thus $\max(x, 0) = 0$, or $x > 0$, thus $\max(x, 0) = x$. Such case-splitting occurs for each occurrence of ReLU, which leads to a prohibitively vast search space (2^n possible cases where n is the number of neurons).

Reluplex and its successor Marabou [100] constitute a line of work that aims to adapt classical SMT routines to piecewise-linear functions. The authors proposed a modified simplex algorithm to handle ReLUs, drastically reducing the number of necessary case-splitting. It allowed previously intractable problems to be solved, for instance the ACAS [119] benchmark.

4.5.5.3 Mixed Integer Linear Programming methods

It is possible to model a neural network using mixed integer linear programming (MILP). Such modelling is used for instance in MIPVerify [120]. MILP tend to be slightly less expressive than SMT approaches to express properties: for instance, conjunctions of disjunctions are difficult to express, and non-linear properties are impossible to prove directly. In the case of non-linear properties, it is necessary to produce linear relaxations, which give less precise results.

4.5.5.4 Deductive verification and Weakest Precondition Calculus

Although solvers like Alt-Ergo [121] or tools like Coq, Agda and Isabelle are used for deductive program verification, there exist to the best of our knowledge little application of weakest precondition calculus to formal neural network verification. The only work we found is that of Vehicle-lang [122], a tool to embed neural network programs into proofs in the Agda theorem

prover. In some sense, the CAISAR platform [102] can interface with the Coq theorem prover, but we did not find any application yet.

4.5.6 Limitations

Formal methods for machine learning verification and validation is still a nascent field. Although the tools are increasingly efficient, scalability is still an issue that prevent the verification on certain use cases that deal with complex programs, or perceptual data like hi-res images or sound. For instance, modern architectures like transformers, deep detection models and diffusion models are currently not handled by most of the presented tools.

Tools are evolving really fast: a tool developed two years ago could be replaced by a new, better performing tool but with a different interface, requiring work for adapting the verification problem to the new tool. To mitigate this issue, platforms such as CAISAR [102] or DNNV [123] aim to provide a unified modelling interface. GOOSE [124] is an upcoming meta-solver that automatically select the proper solver for a given problem instance.

Literature currently focus on a subset of properties, such as local robustness against a perturbation, or well-known, academic benchmarks on low-dimensional inputs. During the course of verification of programs embedded in autonomous mobility systems, properties that do not fit in those two definitions will be encountered; which may limit the use of existing tools for checking those properties.

Ultimately, formal methods require a specification to check. Specifically, there need to be some kind of mathematical characterisation of the neural network behaviour. Producing such mathematical formulation can be difficult, due to the dimension of the spaces we consider or the conceptual complexity of the inputs [15]. For instance, it is impossible to formally define what is an image of a pedestrian considering all camera angles, weather conditions or brightness conditions. Verifying that a program does not take a certain subset of decisions when presented an image of pedestrian would be impossible, unless reducing the definition of what a pedestrian is to the point of harming the performance of the system.

Formal methods for machine learning usually require full access to the model, and sometimes to the data it was trained on. Due to legal or technical reasons, this may not be achievable. A partial access to the program and its data (for instance, synthetic data) could still be workable; it would however require a description of the process used to obtain the synthetic dataset.

REFERENCES

[1] European Commission, “Regulation (eu) 2022/1426 - laying down rules for the application of regulation (eu) 2019/2144 of the european parliament and the council as regards uniform procedures and technical specifications for the type-approval of the automated driving system (ads) of fully automated vehicles - 5 august 2022,” Brussels, 2022.

[2] MINISTÈRE DE LA TRANSITION ÉCOLOGIQUE - TRANSPORTS. Arrêté du xxx définissant les conditions d’homologation, d’exploitation et de circulation des navettes urbaines équipées d’un système de conduite automatisé.

[3] GRVA, “New assessment/test method for automated driving (natm) - guidelines for validating automated driving system (ads),” UNECE, Tech. Rep., VMAD -24th session, February 2022. [Online]. Available: https://wiki.unece.org/display/trans/VMAD+-+24th+session

[4] "Definition of procedures of scenario management and results analysis," PRISSMA project, PRISSMA Deliverable D2.6, Mar. 2022.

[5] J.-B. Horel, C. Laugier, L. Marsso, R. Mateescu, L. Muller, A. Paigwar, A. Renzaglia, and W. Serwe, "Using formal conformance testing to generate scenarios for autonomous vehicles," in Design, Automation and Test in Europe - Autonomous Systems Design (DATE/ASD'2022). IEEE, 2022.

[6] L. Marsso, R. Mateescu, L. Muller, and W. Serwe, "Formally Modeling Autonomous Vehicles in LNT for Simulation and Testing," in Proceedings of the 5th Workshop on Models for Formal Analysis of Real Systems (MARS'2022), Munich, Germany, ser. EPTCS, vol. 355, Apr. 2022, pp. 60–117.

[7] J.-B. Horel, P. Ledent, L. Marsso, L. Muller, C. Laugier, R. Mateescu, A. K. Paigwar, A. Renzaglia, and W. Serwe, "Verifying collision risk estimation using autonomous driving scenarios derived from a formal model," Journal of Intelligent & Robotic Systems, vol. 107, pp. 1–28, 2023. [Online]. Available: https://api.semanticscholar.org/CorpusID:258242307

[8] M. Zhang, Y. Zhang, L. Zhang, C. Liu, and S. Khurshid, "DeepRoad: GAN-based metamorphic testing and input validation framework for autonomous driving systems," in Proceedings of the 33rd ACM/IEEE International Conference on Automated Software Engineering. New York, NY, USA: Association for Computing Machinery, Sep. 2018, pp. 132–142. [Online]. Available: https://doi.org/10.1145/3238147.3238187

[9] M. Nabhan, "Models and algorithms for the exploration of the space of scenarios : toward the validation of the autonomous vehicle," Theses, Université Paris-Saclay, Dec. 2020. [Online]. Available: https://tel.archives-ouvertes.fr/tel-03144001

[10] L. Marsso, R. Mateescu, and W. Serwe, "Automated transition coverage in behavioural conformance testing," in Testing Software and Systems - 32nd IFIP WG 6.1 International Conference, ICTSS 2020, Naples, Italy, December 9-11, 2020, Proceedings, ser. Lecture Notes in Computer Science, V. Casola, A. D. Benedictis, and M. Rak, Eds., vol. 12543. Springer, 2020, pp. 219–235.

[11] F. Jiménez, J. Naranjo, and F. García, "An improved method to calculate the time-to-collision of two vehicles," International Journal of Intelligent Transportation Systems Research, vol. 11, no. 1, pp. 34–42, 2013.

[12] H. Delseny, C. Gabreau, A. Gauffriau, B. Beaudouin, L. Ponsolle, L. Alecu, H. Bonnin, B. Beltran, D. Duchel, J. Ginestet, A. Hervieu, G. Martinez, S. Pasquet, K. Delmas, C. Pagetti, J. Gabriel, C. Chapdelaine, S. Picard, M. Damour, C. Cappi, L. Gardès, F. D. Grancey, E. Jenn, B. Lefèvre, G. Flandin, S. Gerchinovitz, F. Mamalet, and A. Albore, "White paper machine learning in certified systems," CoRR, vol. abs/2103.10529, 2021. [Online]. Available: https://arxiv.org/abs/2103.10529

[13] I. Evtimov, K. Eykholt, E. Fernandes, T. Kohno, B. Li, A. Prakash, A. Rahmati, and D. Song, "Robust physical-world attacks on machine learning models," CoRR, vol. abs/1707.08945, 2017. [Online]. Available: http://arxiv.org/abs/1707.08945

[14] J. Gawlikowski, C. R. N. Tassi, and M. Ali, "A survey of uncertainty in deep neural networks," Artificial Intelligence Review, July 2023. [Online]. Available: https://link.springer.com/article/10.1007/s10462-023-10562-9

[15] J. Girard-Satabin, "Verification and validation of machine learning techniques," phdthesis, Université Paris-Saclay, Nov. 2021. [Online]. Available: https://tel.archives-ouvertes.fr/tel-03547545

[16] DGITM & IRT SystemX , "Démonstration de sécurité des systèmes de transports routiers automatisés : apport des scénarios de conduit - livrable 1. génération, alimentation et enrichissement des scenarios," Tech. Rep., 2022.

[17] GRVA, "New assessment/test method for automated driving (natm) - master document," presented at World Forum for Harmonization of Vehicle Regulations – 184th session, Geneva, 22-24 June 2021.

[18] MINISTÈRE DE LA TRANSITION ÉCOLOGIQUE - TRANSPORTS. Décret n° 2021-873 du 29 juin 2021 portant application de l'ordonnance n° 2021-443 du 14 avril 2021 relative au régime de responsabilité pénale applicable en cas de circulation d'un véhicule à délégation de conduite et à ses conditions d'utilisation. [Online]. Available: https://www.legifrance.gouv.fr/jorf/id/JORFTEXT000043729532

[19] NATM, "New assessment/test method for automated driving (natm) guidelines for validating automated driving system (ads) – amendments to ece/trans/wp.29/2022/58," 2022. [Online]. Available: https://unece.org/sites/default/files/2022-09/GRVA-14-16e_0.pdf

[20] DGITM , "Véhicules et systèmes de transports automatisés : Premiers principes et questions pour la définition des odd" - document de cadrage méthodologique – version d'initialisation," Tech. Rep., May 2022. [Online]. Available: https://www.ecologie.gouv.fr/sites/default/files/DGITM_ODD-descriptors-2022.pdf

[21] DGITM, "Démonstration de sécurité des systèmes de transport routier automatisés : Apports attendus des scénarios de conduit," Tech. Rep., February 2022. [Online]. Available: https://www.ecologie.gouv.fr/sites/default/files/DGITM_Approche-par-scenarios-fevrier-2022_0.pdf

[22] ——, "Utilisation des scénarios pour la démonstration de la sécurité des systèmes de transports routiers automatisés," 2023. [Online]. Available: https://www.ecologie.gouv.fr/sites/default/files/DGITM-Utilisation-scenarios-NATM-fevrier_20231.pdf

[23] STRMTG, "Technical guide related to game demonstration for arts," 2022. [Online]. Available: https://balise.documentation.developpement-durable.gouv.fr/docs/Balise/0065/Balise-0065872/GuideDemonstration%20GAME_STRA_v1_EN.pdf

[24] "Road vehicles — functional safety," International Organization for Standardization, Geneva, CH, Standard, Mar. 2000.

[25] W. Xu, D. Gruyer, and S.-S. Ieng, "Generic simulation framework for evaluation process: Applied to ai-powered visual perception system in autonomous driving," in Proceedings of the IEEE International Conference on Intelligent Transportation Systems (ITSC), 2023. IEEE, 2023.

[26] F. Yu, H. Chen, X. Wang, W. Xian, Y. Chen, F. Liu, V. Madhavan, and T. Darrell, "Bdd100k: A diverse driving dataset for heterogeneous multitask learning," in The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2020.

[27] S. Riedmaier, T. Ponn, D. Ludwig, B. Schick, and F. Diermeyer, "Survey on scenario-based safety assessment of automated vehicles," IEEE Access, vol. 8, pp. 87 456–87 477, 2020. [Online]. Available: https://doi.org/10.1109/ACCESS.2020.2993730

[28] D. Champelovier, X. Clerc, H. Garavel, Y. Guerte, C. McKinty, V. Powazny, F. Lang, W. Serwe, and G. Smeding, "Reference Manual of the LNT to LOTOS Translator (Version 7.0)," Mar. 2021, INRIA, Grenoble, France.

[29] H. Garavel, F. Lang, and W. Serwe, "From LOTOS to LNT," in ModelEd, TestEd, TrustEd – Essays Dedicated to Ed Brinksma on the Occasion of His 60th Birthday, ser. lncs, vol. 10500. sv, oct 2017, pp. 3–26.

[30] H. Garavel, F. Lang, R. Mateescu, and W. Serwe, "CADP 2011: A Toolbox for the Construction and Analysis of Distributed Processes," Springer International Journal on Software Tools for Technology Transfer (STTT), vol. 15, no. 2, pp. 89–107, 2013.

[31] W. Ding, B. Chen, M. Xu, and D. Zhao, "Learning to collide: An adaptive safety-critical scenarios generating method," in International Conference on Intelligent Robots and Systems (IROS). IEEE, 2020, pp. 2243–2250.

[32] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "CARLA: An open urban driving simulator," in Proceedings of the 1st Annual Conference on Robot Learning, 2017, pp. 1–16.

[33] L. Marsso, R. Mateescu, and W. Serwe, "TESTOR: A Modular Tool for On-the-Fly Conformance Test Case Generation," in Proceedings of the 24th International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS'18), Thessaloniki, Greece, D. Beyer and M. Huisman, Eds., vol. 10806, Apr. 2018, pp. 211–228.

[34] S. Dola, M. B. Dwyer, and M. L. Soffa, "Distribution-aware testing of neural networks using generative models," in 2021 IEEE/ACM 43rd International Conference on Software Engineering (ICSE), May 2021, pp. 226–237.

[35] T. Y. Chen, F.-C. Kuo, H. Liu, P.-L. Poon, D. Towey, T. Tse, and Z. Q. Zhou, "Metamorphic testing: A review of challenges and opportunities," ACM Computing Surveys (CSUR), vol. 51, no. 1, pp. 1–27, 2018.

[36] D. Mukhopadhyay, K. Madhukar, and M. Srivas, "Permutation Invariance of Deep Neural Networks with ReLUs," arXiv:2110.09578 [cs], Oct. 2021.

[37] Y. Tian, K. Pei, S. Jana, and B. Ray, "Deeptest: Automated testing of deep-neural-network-driven autonomous cars," CoRR, vol. abs/1708.08559, 2017. [Online]. Available: http://arxiv.org/abs/1708.08559

[38] K. Pei, Y. Cao, J. Yang, and S. Jana, "Deepxplore: Automated whitebox testing of deep learning systems," in Proceedings of the 26th Symposium on Operating Systems Principles, ser. SOSP '17. New York, NY, USA: Association for Computing Machinery, 2017, p. 1–18. [Online]. Available: https://doi.org/10.1145/3132747.3132785

[39] Y. Sun, M. Wu, W. Ruan, X. Huang, M. Kwiatkowska, and D. Kroening, Concolic Testing for Deep Neural Networks. New York, NY, USA: Association for Computing Machinery, 2018, p. 109–119. [Online]. Available: https://doi.org/10.1145/3238147.3238172

[40] F. Adjed, M. Mziou-Sallami, F. Pelliccia, M. Rezzoug, L. Schott, C. Bohn, and Y. Jaafra, "Coupling algebraic topology theory, formal methods and safety requirements toward a new coverage metric for artificial intelligence models," Neural Computing and Applications, may 2022.

[41] J. Zander, I. Schieferdecker, and P. J. Mosterman, Eds., Model-Based Testing for Embedded Systems, ser. Computational Analysis, Synthesis, & Design Dynamic Systems. CRC Press, 2011.

[42] M. Utting, A. Pretschner, and B. Legeard, "A Taxonomy of Model-based Testing Approaches," Software Testing, Verification and Reliability, vol. 22, no. 5, pp. 297–312, Aug. 2012.

[43] C. Jard and T. Jéron, "Tgv: Theory, principles and algorithms – a tool for the automatic synthesis of conformance test cases for non-deterministic reactive systems," Springer International Journal on Software Tools for Technology Transfer (STTT), vol. 7, no. 4, pp. 297–315, Aug. 2005.

[44] J. Tretmans, "Conformance Testing with Labelled Transition Systems: Implementation Relations and Test Generation," Computer networks and ISDN systems, vol. 29, no. 1, pp. 49–79, Dec. 1996.

[45] H. Zhu, P. A. V. Hall, and J. H. R. May, "Software Unit Test Coverage and Adequacy," ACM Computing Surveys, vol. 29, no. 4, pp. 366–427, 1997.

[46] S. Shalev-Shwartz, S. Shammah, and A. Shashua, "On a formal model of safe and scalable self-driving cars," arXiv preprint arXiv:1708.06374, 2017.

[47] A. Shashua, S. Shalev-Shwartz, and S. Shammah, "Implementing the rss model on nhtsa pre-crash scenarios," tech. rep, 2018.

[48] M. M. Minderhoud and P. H. Bovy, "Extended time-to-collision measures for road traffic safety assessment," Accident; Analysis and Prevention, pp. 89–97, 2001.

[49] J. Ward, G. Agamennoni, S. Worrall, and E. Nebot, "Vehicle collision probability calculation for general traffic scenarios under uncertainty," 2014 IEEE Intelligent Vehicles Symposium Proceedings, pp. 986–992, 2014.

[50] J. Janson, "Collision avoidance theory with application to automotive collision mitigation," Ph.D. dissertation, Linköping Universitet, 2005.

[51] C. Ackermann, R. Isermann, S. Min, and C. Kim, "Collision avoidance with automatic braking and swerving," IFAC Proceedings Volumes, vol. 47, no. 3, pp. 10 694–10 699, 2014, 19th IFAC World Congress. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1474667016433136

[52] J. Eggert and T. Puphal, "Continuous risk measures for adas and ad," in 2017 4th International Symposium on Future Active Safety Technology towards Zero-Traffic-Accidents (FAST-zero), Sep. 18 - 21, 2017, Nara Kasugano International Forum, Nara, Japan, 2017.

[53] W. Wachenfeld, P. Junietz, R. Wenzel, and H. Winner, "The worst-time-to-collision metric for situation identification," in 2016 IEEE Intelligent Vehicles Symposium (IV), 2016, pp. 729–734.

[54] A. Lambert, D. Gruyer, and G. S. Pierre, "A fast monte carlo algorithm for collision probability estimation," in International Conference on Control, Automation, Robotics and Vision 2008 (ICARCV 2008), Hanoi, Vietnam, December 2008.

[55] P. Moravcová, K. Bucsuházy, M. Bilík, M. J. Belak, and A. Bradác, "Let it crash! energy equivalent speed determination," in VEHITS, 2021.

[56] C. Katrakazas, M. Quddus, and W.-H. Chen, "A new methodology for collision risk assessment of autonomous vehicles," in In Proceedings of Transportation Research Board 96th Annual Meeting (TRB 2017), volume 8, Washington D.C., USA, January 2017, 2017.

[57] A. Philipp and D. Goehring, "Analytic collision risk calculation for autonomous vehicle navigation," in In Proceedings of IEEE 2019 International Conference on Robotics and Automation (ICRA'19), page 1744–1750, Montreal, QC, Canada, May 2019., 2019.

[58] A. Farahani, S. Voghoei, K. Rasheed, and H. R. Arabnia, "A brief review of domain adaptation," CoRR, vol. abs/2010.03978, 2020. [Online]. Available: https://arxiv.org/abs/2010.03978

[59] D. V. Vargas and S. Kotyan, "Model agnostic dual quality assessment for adversarial machine learning and an analysis of current neural networks and defenses," CoRR, vol. abs/1906.06026, 2019. [Online]. Available: http://arxiv.org/abs/1906.06026

[60] Y. Ovadia, E. Fertig, J. Ren, Z. Nado, D. Sculley, S. Nowozin, J. V. Dillon, B. Lakshminarayanan, and J. Snoek, "Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift," 2019. [Online]. Available: https://arxiv.org/abs/1906.02530

[61] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," 2013. [Online]. Available: https://arxiv.org/abs/1312.6199

[62] F. Croce and M. Hein, "Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks," CoRR, vol. abs/2003.01690, 2020. [Online]. Available: https://arxiv.org/abs/2003.01690

[63] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," 2014. [Online]. Available: https://arxiv.org/abs/1412.6572

[64] S. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, "Deepfool: a simple and accurate method to fool deep neural networks," CoRR, vol. abs/1511.04599, 2015. [Online]. Available: http://arxiv.org/abs/1511.04599

[65] R. Wiyatno and A. Xu, "Maximal jacobian-based saliency map attack," CoRR, vol. abs/1808.07945, 2018. [Online]. Available: http://arxiv.org/abs/1808.07945

[66] S. Moosavi-Dezfooli, A. Fawzi, O. Fawzi, and P. Frossard, "Universal adversarial perturbations," CoRR, vol. abs/1610.08401, 2016. [Online]. Available: http://arxiv.org/abs/1610.08401

[67] J. Su, D. V. Vargas, and K. Sakurai, "One pixel attack for fooling deep neural networks," IEEE Transactions on Evolutionary Computation, vol. 23, no. 5, pp. 828–841, 2019.

[68] N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z. B. Celik, and A. Swami, "Practical black-box attacks against machine learning," 2016. [Online]. Available: https://arxiv.org/abs/1602.02697

[69] P.-Y. Chen, H. Zhang, Y. Sharma, J. Yi, and C.-J. Hsieh, "Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models," 10th ACM Workshop on Artificial Intelligence and Security (AISEC) with the 24th ACM Conference on Computer and Communications Security (CCS), 2017.

[70] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," 2017. [Online]. Available: https://arxiv.org/abs/1706.06083

[71] A. Shafahi, M. Najibi, Z. Xu, J. Dickerson, L. S. Davis, and T. Goldstein, "Universal adversarial training," 2018. [Online]. Available: https://arxiv.org/abs/1811.11304

[72] J. M. Cohen, E. Rosenfeld, and J. Z. Kolter, "Certified adversarial robustness via randomized smoothing," 2019. [Online]. Available: https://arxiv.org/abs/1902.02918

[73] J. Z. Kolter and E. Wong, "Provable defenses against adversarial examples via the convex outer adversarial polytope," CoRR, vol. abs/1711.00851, 2017. [Online]. Available: http://arxiv.org/abs/1711.00851

[74] M. Alfarra, A. Bibi, P. Torr, and B. Ghanem, "Data-dependent randomized smoothing," 2022. [Online]. Available: https://openreview.net/forum?id=ZFIT_sGjPJ

[75] P. Samangouei, M. Kabkab, and R. Chellappa, "Defense-gan: Protecting classifiers against adversarial attacks using generative models," 2018. [Online]. Available: https://arxiv.org/abs/1805.06605

[76] D. Hendrycks and T. G. Dietterich, "Benchmarking neural network robustness to common corruptions and perturbations," CoRR, vol. abs/1903.12261, 2019. [Online]. Available: http://arxiv.org/abs/1903.12261

[77] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," CoRR, vol. abs/1706.04599, 2017. [Online]. Available: http://arxiv.org/abs/1706.04599

[78] B. Settles, "Active learning literature survey," University of Wisconsin–Madison, Computer Sciences Technical Report 1648, 2009.

[79] J. T. Ash, C. Zhang, A. Krishnamurthy, J. Langford, and A. Agarwal, "Deep batch active learning by diverse, uncertain gradient lower bounds," 2020.

[80] A. Kurakin, I. Goodfellow, and S. Bengio, "Adversarial examples in the physical world," 2016. [Online]. Available: https://arxiv.org/abs/1607.02533

[81] X. Cao and I. W. Tsang, "Bayesian active learning by disagreements: A geometric perspective," 2021. [Online]. Available: https://arxiv.org/abs/2105.02543

[82] K. Wei, R. Iyer, and J. Bilmes, "Submodularity in data subset selection and active learning," in Proceedings of the 32nd International Conference on Machine Learning, ser. Proceedings of Machine Learning Research, F. Bach and D. Blei, Eds., vol. 37. Lille, France: PMLR, 07–09 Jul 2015, pp. 1954–1963. [Online]. Available: https://proceedings.mlr.press/v37/wei15.html

[83] K. Killamsetty, D. Sivasubramanian, G. Ramakrishnan, and R. K. Iyer, "GLISTER: generalization based data subset selection for efficient and robust learning," CoRR, vol. abs/2012.10630, 2020. [Online]. Available: https://arxiv.org/abs/2012.10630

[84] W. Cai, Y. Zhang, and J. Zhou, "Maximizing expected model change for active learning in regression," in 2013 IEEE 13th International Conference on Data Mining, Dallas, TX, USA, December 7-10, 2013, H. Xiong, G. Karypis, B. M. Thuraisingham, D. J. Cook, and X. Wu, Eds. IEEE Computer Society, 2013, pp. 51–60. [Online]. Available: https://doi.org/10.1109/ICDM.2013.104

[85] R. Willett, R. Nowak, and R. Castro, "Faster rates in regression via active learning," in Advances in Neural Information Processing Systems, Y. Weiss, B. Schölkopf, and J. Platt, Eds., vol. 18. MIT Press, 2006. [Online]. Available: https://proceedings.neurips.cc/paper/2005/file/4ea6a546c19499318091a9df40a13181-Paper.pdf

[86] K. Sung and P. Niyogi, "Active learning for function approximation," in Advances in Neural Information Processing Systems, G. Tesauro, D. Touretzky, and T. Leen, Eds., vol. 7. MIT Press, 1995. [Online]. Available: https://proceedings.neurips.cc/paper/1994

[87] W. Cai, M. Zhang, and Y. Zhang, "Batch mode active learning for regression with expected model change," IEEE Transactions on Neural Networks and Learning Systems, vol. 28, pp. 1668–1681, 2017.

[88] K. Yu and J. Bi, "Active learning via transductive experimental design," in In Machine Learning, Proceedings of the Twenty-Third International Conference (ICML. ACM Press, 2006, pp. 1081–1088.

[89] IDEAL'07: Proceedings of the 8th International Conference on Intelligent Data Engineering and Automated Learning. Berlin, Heidelberg: Springer-Verlag, 2007.

[90] M. Shukla, "Bayesian uncertainty and expected gradient length – regression: Two sides of the same coin?" 2021. [Online]. Available: https://arxiv.org/abs/2104.09493

[91] Y. Baram, R. El-Yaniv, and K. Luz, "Online choice of active learning algorithms," J. Mach. Learn. Res., vol. 5, p. 255–291, dec 2004.

[92] E. Boiy and M. francine Moens, "A machine learning approach to sentiment analysis in multilingual web texts," Information Retrieval, pp. 526–558, 2009.

[93] R. Hu, S. Jane Delany, and B. Mac Namee, "Egal: Exploration guided active learning for tcbr," in Case-Based Reasoning. Research and Development, I. Bichindaritz and S. Montani, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 156–170.

[94] E. Sekerinski and K. Sere, Program Development by Refinement Case Studies Using the B Method. Springer London, Limited, 2012.

[95] P. Baudin, F. Bobot, D. Bühler, L. Correnson, F. Kirchner, N. Kosmatov, A. Maroneze, V. Perrelle, V. Prevosto, J. Signoles, and N. Williams, "The dogged pursuit of bug-free c programs: The frama-c software analysis platform," Commun. ACM, vol. 64, no. 8, p. 56–68, jul 2021. [Online]. Available: https://doi.org/10.1145/3470569

[96] G. Klein, J. Andronick, M. Fernandez, I. Kuz, T. Murray, and G. Heiser, "Formally verified software in the real world," Commun. ACM, vol. 61, no. 10, pp. 68–77, 2018.

[97] C. Barrett, P. Fontaine, and C. Tinelli, "The SMT-LIB Standard: Version 2.6," Department of Computer Science, The University of Iowa, Tech. Rep., 2017, available at `www.SMT-LIB.org`.

[98] J. Signoles, E-ACSL: Executable ANSI/ISO C Specification Language. [Online]. Available: http://frama-c.com/download/e-acsl/e-acsl.pdf

[99] M. N. Müller, G. Makarchuk, G. Singh, M. Püschel, and M. Vechev, "PRIMA: General and precise neural network certification via scalable convex hull approximations," Proceedings of the ACM on Programming Languages, vol. 6, no. POPL, pp. 1–33, Jan. 2022.

[100] G. Katz, D. A. Huang, D. Ibeling, K. Julian, C. Lazarus, R. Lim, P. Shah, S. Thakoor, H. Wu, A. Zeljić, D. L. Dill, M. J. Kochenderfer, and C. Barrett, "The Marabou Framework for Verification and Analysis of Deep Neural Networks," in Computer Aided Verification, ser. Lecture Notes in Computer Science, I. Dillig and S. Tasiran, Eds. Cham: Springer International Publishing, 2019, pp. 443–452.

[101] C. Urban, M. Christakis, V. Wüstholz, and F. Zhang, "Perfectly Parallel Fairness Certification of Neural Networks," Proceedings of the ACM on Programming Languages, vol. 4, no. OOPSLA, pp. 1–30, Nov. 2020. [Online]. Available: https://hal.inria.fr/hal-03091870

[102] M. Alberti, F. Bobot, Z. Chihani, J. Girard-Satabin, and A. Lemesle, "CAISAR: A platform for Characterizing Artificial Intelligence Safety and Robustness," in AISafety, ser. CEUR-Workshop Proceedings, Vienne, Austria, Jul. 2022. [Online]. Available: https://hal.archives-ouvertes.fr/hal-03687211

[103] X. Xie, K. Kersting, and D. Neider, "Neuro-symbolic verification of deep neural networks," in Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22, L. D. Raedt, Ed. International Joint Conferences on Artificial Intelligence Organization, 7 2022, pp. 3622–3628, main Track. [Online]. Available: https://doi.org/10.24963/ijcai.2022/503

[104] S. Wang, H. Zhang, K. Xu, X. Lin, S. Jana, C.-J. Hsieh, and J. Z. Kolter, "Beta-CROWN: Efficient Bound Propagation with Per-neuron Split Constraints for Complete and Incomplete Neural Network Robustness Verification," arXiv:2103.06624 [cs, stat], Oct. 2021.

[105] S. Bak, C. Liu, and T. Johnson, "The second international verification of neural networks competition (vnn-comp 2021): Summary and results," 2021. [Online]. Available: https://arxiv.org/abs/2109.00498

[106] M. Goldblum, D. Tsipras, C. Xie, X. Chen, A. Schwarzschild, D. Song, A. Madry, B. Li, and T. Goldstein, "Dataset Security for Machine Learning: Data Poisoning, Backdoor Attacks, and Defenses." [Online]. Available: http://arxiv.org/abs/2012.10544

[107] T. Gebru, J. Morgenstern, B. Vecchione, J. W. Vaughan, H. Wallach, H. D. III, and K. Crawford, "Datasheets for datasets," Commun. ACM, vol. 64, no. 12, p. 86–92, nov 2021. [Online]. Available: https://doi.org/10.1145/3458723

[108] N. Carlini, S. Chien, M. Nasr, S. Song, A. Terzis, and F. Tramer, "Membership inference attacks from first principles," 2022.

[109] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang, "Deep learning with differential privacy," in Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. ACM, oct 2016. [Online]. Available: https://doi.org/10.1145%2F2976749.2978318

[110] E.-M. El-Mhamdi, S. Farhadkhani, R. Guerraoui, N. Gupta, L.-N. Hoang, R. Pinot, S. Rouault, and J. Stephan, "On the impossible safety of large ai models," 2023.

[111] J. Gilmer, N. Ford, N. Carlini, and E. Cubuk, "Adversarial examples are a natural consequence of test error in noise," in Proceedings of the 36th International Conference on Machine Learning, ser. Proceedings of Machine Learning Research, K. Chaudhuri and R. Salakhutdinov, Eds., vol. 97. PMLR, 09–15 Jun 2019, pp. 2280–2289. [Online]. Available: https://proceedings.mlr.press/v97/gilmer19a.html

[112] A. Miné, "Tutorial on Static Inference of Numeric Invariants by Abstract Interpretation," Foundations and Trends® in Programming Languages, vol. 4, no. 3-4, pp. 120–372, 2017.

[113] W. Ryou, J. Chen, M. Balunovic, G. Singh, A. Dan, and M. Vechev, "Scalable Polyhedral Verification of Recurrent Neural Networks," in Computer Aided Verification, A. Silva and K. R. M. Leino, Eds. Cham: Springer International Publishing, 2021, vol. 12759, pp. 225–248.

[114] M. Balunović, M. Baader, G. Singh, T. Gehr, and M. Vechev, "Certifying geometric robustness of neural networks," Advances in Neural Information Processing Systems 32, 2019.

[115] H. Zhang, H. Chen, C. Xiao, S. Gowal, R. Stanforth, B. Li, D. Boning, and C.-J. Hsieh, "Towards Stable and Efficient Training of Verifiably Robust Neural Networks," arXiv:1906.06316 [cs, stat], Nov. 2019.

[116] S. Bak, "Nnenum: Verification of ReLU Neural Networks with Optimized Abstraction Refinement," in NASA Formal Methods, ser. Lecture Notes in Computer Science, A. Dutle, M. M. Moscato, L. Titolo, C. A. Muñoz, and I. Perez, Eds. Cham: Springer International Publishing, 2021, pp. 19–36.

[117] P. Henriksen and A. Lomuscio, "Deepsplit: An efficient splitting method for neural network verification via indirect effect analysis," in Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21, Z.-H. Zhou, Ed. International Joint Conferences on Artificial Intelligence Organization, 8 2021, pp. 2549–2555, main Track.

[118] L. de Moura and N. Bjørner, "Z3: An Efficient SMT Solver," in Tools and Algorithms for the Construction and Analysis of Systems, ser. Lecture Notes in Computer Science, C. R. Ramakrishnan and J. Rehof, Eds. Berlin, Heidelberg: Springer, 2008, pp. 337–340.

[119] G. Manfredi and Y. Jestin, "An introduction to ACAS Xu and the challenges ahead," in 2016 IEEE/AIAA 35th Digital Avionics Systems Conference (DASC). Sacramento, CA, USA: IEEE, Sep. 2016, pp. 1–9.

[120] V. Tjeng, K. Xiao, and R. Tedrake, "Evaluating Robustness of Neural Networks with Mixed Integer Programming," in International Conference on Learning Representations (ICLR), 2019.

[121] S. Conchon, A. Coquereau, M. Iguernlala, and A. Mebsout, "Alt-Ergo 2.2," in SMT Workshop: International Workshop on Satisfiability Modulo Theories, Oxford, United Kingdom, Jul. 2018. [Online]. Available: https://hal.inria.fr/hal-01960203

[122] M. L. Daggitt, W. Kokke, R. Atkey, L. Arnaboldi, and E. Komendantskya, "Vehicle: Interfacing neural network verifiers with interactive theorem provers," 2022. [Online]. Available: https://arxiv.org/abs/2202.05207

[123] D. Shriver, S. Elbaum, and M. B. Dwyer, "DNNV: A Framework for Deep Neural Network Verification," in Computer Aided Verification, ser. Lecture Notes in Computer Science, A. Silva and K. R. M. Leino, Eds. Cham: Springer International Publishing, 2021, pp. 137–150.

[124] J. Scott, G. Pan, E. B. Khalil, and V. Ganesh, "Goose: A meta-solver for deep neural network verification," in Proceedings of the 20th Internal Workshop on Satisfiability Modulo Theories co-located with the 11th International Joint Conference on Automated Reasoning (IJCAR 2022) part of the 8th Federated Logic Conference (FLoC 2022), Haifa, Israel, August 11-12, 2022, ser. CEUR Workshop Proceedings, D. Déharbe and A. E. J. Hyvärinen, Eds., vol. 3185. CEUR-WS.org, 2022, pp. 99–113. [Online]. Available: https://ceur-ws.org/Vol-3185/extended678.pdf

A A PRISSMA method to generate scenarios from the ODD and the OEDR

Using the scenario approach of DGITM and the one presented in the ADS act, as well as the ODD from WP8 and requirements, a PRISSMA database scenario generation method is proposed. This method aims at defining nominal and critical scenarios. Failure scenario generation is not included in this work. The ODD taxonomy structure proposed by WP8 is shown on figure 35.

The parameters to generate the database will be define by considering scenario layer by scenario layer and use a principle of increasing complexity. This method should aloow to explore the ODD space and shall be combine with requirement analyses. The requirements are presented in Annex B.

First observation, infrastructures are built to allow manoeuvres, a first classification in 5 categories relating infrastructures and manoeuvres is proposed for a passenger transport system based on automated vehicles. See Figure 36.

First stage: To generate scenarios with a growing complexity, the first stage is to consider only the static infrastructure, the manoeuvres and the responses of the system which respectively corresponds to the layers 1/2 and 4 defined in the DGITM document (see Figure 37. In the ODD taxonomy, the branches physical infrastructures, scenery and digital infrastructures are taken into account. The traffic conditions, weather conditions or any other road user behaviour are not yet examined.

To generate a first set of scenarios, manoeuvres and infrastructure categories should be analysed one by one. All infrastructures elements included in the ODD have to be listed and associated with manoeuvres and the intended responses. This analysis shall be run using functional and technical requirements. The boundaries of the ODD are important to be investigated. A set of a manoeuvre, a response and a large category of infrastructure leads to the definition of a functional scenario.

It is possible to organise the scenario database considering that each level includes scenarios from the lower levels, as done in the MOSAR platform. (See PRISSMA deliverable L2.1). For instance, each functional scenario can include several sets of Logical scenarios. The Logical scenarios contain descriptors of the infrastructures, the manoeuvre and response parameters, with defined ranges of metrics. By setting a value for each logical scenario parameter, a concrete scenario is obtained.

To generate a relevant scenario database in a practical way, it may be more convenient to define functional scenarios by refining the infrastructures / manoeuvres categories presented in figure 36. Then the descriptors and metrics of the ODD static and digital infrastructures,

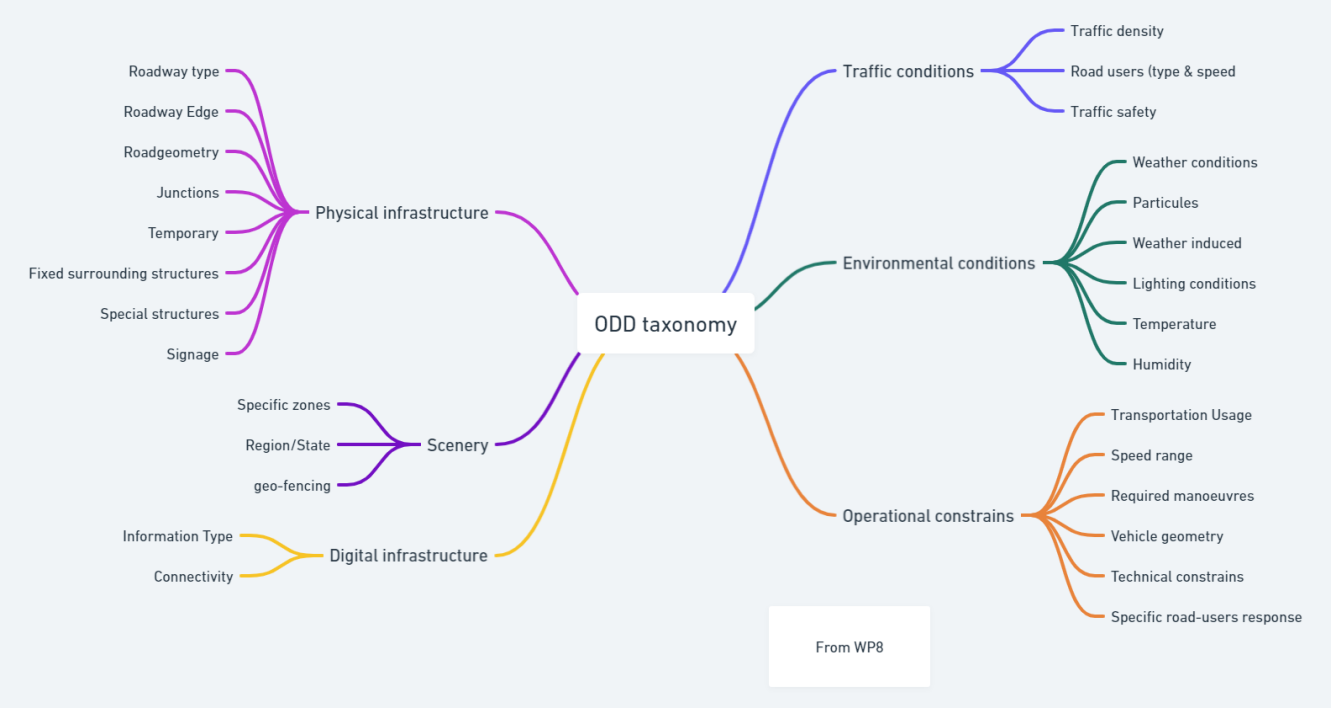


Figure 35: ODD taxonomy defined in WP8 (only the first levels are presented)

manoeuvres and the scenery shall be included in the database by defining appropriate logical scenarios for each functional scenario category. The ranges of parameters contained in the logical scenarios shall include the boundaries of the ODD.

Second stage: For all infrastructures included in the ODD, traffic hazards have to be taken into account. In the ODD taxonomy, the Traffic condition branch is included in the process. Only reasonably likely events have to be considered. By adding traffic conditions and other road user behaviours, new manoeuvres and responses shall be defined. Again, their definitions shall be based on functional and technical requirements. This work corresponds to the OEDR analysis defined in the ADS Act method [1].

It is important to notice that adding events involving other road users may lead to define **nominal scenarios** and also **critical scenarios**, i.e. with a possible accident outcome. Again, this work consists in defining functional scenarios including a manoeuvre, a response, other actors and their behaviour in a category of infrastructure. Within each functional scenario, logical scenarios have to be specified to take into account all the ODD descriptors and boundaries.

Third stage: Two different tasks are to be done.
First, all scenarios defined at stage 1 and stage 2 shall be reanalysed to add the environmental conditions. It consists in adding the environmental condition branch of the ODD to the process. This work shall be done at the logical scenario level by defining the conditions descriptors and metrics of the ODD.

Secondly, masks can affect the response of the systems. They can be static or dynamic. In the DGITM document Scenario Generation, 6 types of masks are defined:

Figure 36: Categories of manoeuvres according to the infrastructures.

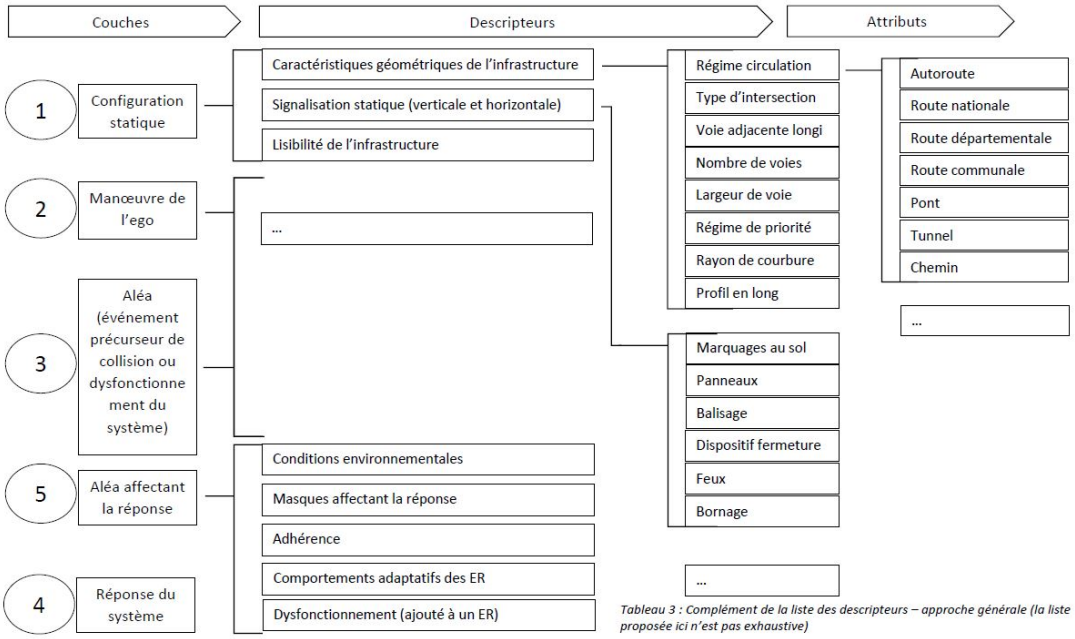


Figure 37: Scenario layers defined by DGITM [16].

- the intrinsic legibility of the infra, independent of the perception sensor (erased markings)
- the static masks (wall, billboard)
- the temporary masks (scaffolding, work zone, vegetation in front of a billboard)
- the fleeting masks (parked vehicle)
- the dynamic masks (vehicle in motion masking other users)
- the environmental masks due to weather conditions (fog)

Again they are to be included in the database at the logical scenario level. However, the presence of a specific mask can lead to the definition of a new functional scenario according to the requirements. As a result new functional scenarios can be created with their associated logical and concrete scenarios.

Figure 38 presents an overview of the whole method.

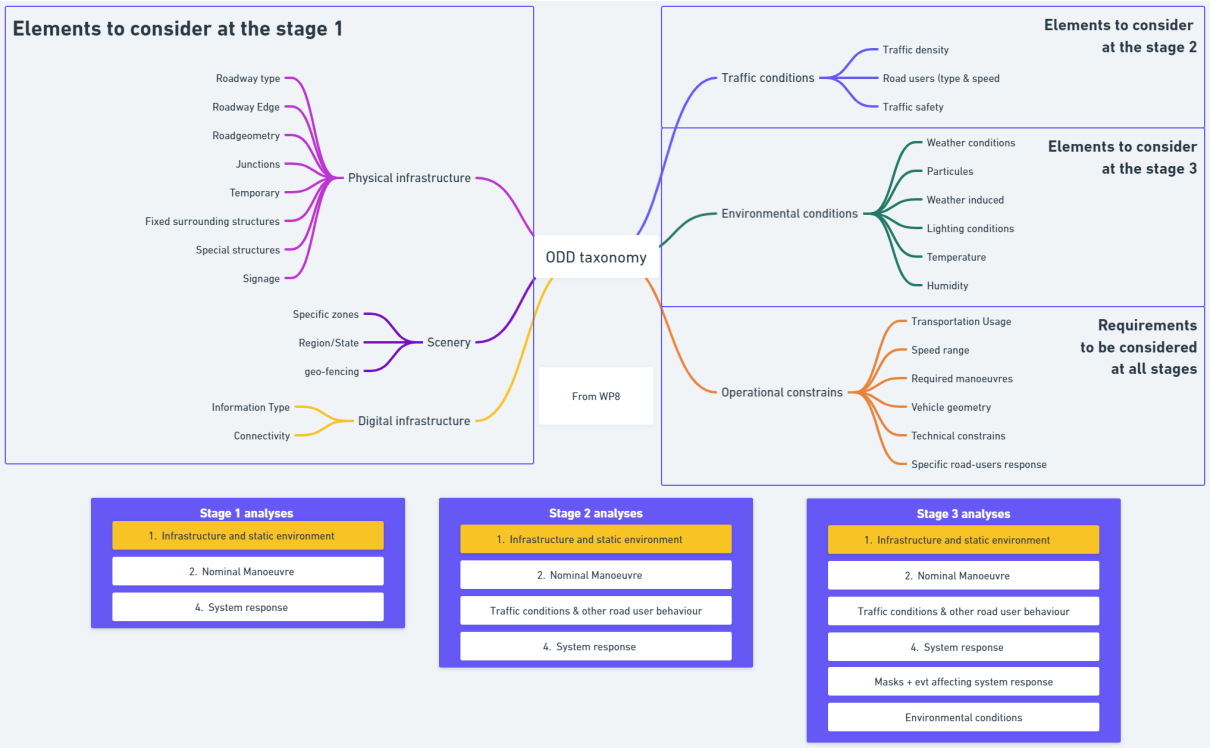


Figure 38: Scenario generation from ODD and requirements

B PRISSMA requirements

1 INTRODUCTION

1.1 Objectives of the PRISSMA Project

To evaluate AI systems for automated & autonomous mobility and ensure their operation, several challenges related to autonomous systems and AI must be addressed:

1. *Accounting for the "non-deterministic" nature of AI techniques.*
2. *Managing the life cycle and evolution capabilities of systems and functions, particularly after the use of AI-based techniques.*
3. *Maintaining auditability, robustness, and safety requirements specific to critical functions and systems.*
4. *Standardizing the methods considered to enable compatibility with international work and to enable their deployment on a large scale.*
5. *Managing the inherent complexity of a system of systems.*

Regarding the PRISSMA project, three objectives have been set:

1. *Identify and list safety and reliability objectives for AI-based autonomous mobility systems and develop complete validation processes for reliability aimed at the commercial operation of SAE Level 4 autonomous mobility services by 2024.*
2. *Ensure the availability of shared concepts to address the complexity of AI-based autonomous mobility systems, which can be used internationally.*
3. *Participate in implementing prerequisites that will enable France to position itself at the European level to host one of the autonomous mobility test centers that will be developed in the coming years.*

To ensure the safety and reliability of systems to be deployed for commercial operation, PRISSMA's first mission will be to identify the characteristics of an AI-based system and its components (the "system under consideration" of the PRISSMA project), as well as the key indicators (KPI) corresponding to the objective to be achieved to demonstrate mastery of the system and the methods and processes to be implemented.

Once the demonstration objects and objectives are identified, PRISSMA will need to develop questions to be asked to the actor wishing to obtain the commissioning of an AI-based autonomous mobility system, determine acceptable evidence, and specify the means of demonstrations and associated tools allowing this actor to demonstrate the safety of their system. The demonstration objectives will be consolidated into a common reference and may result from ongoing work at the French, European, and international levels.

PRISSMA will thus determine the means of qualifying simulation tools and associated databases, as well as requirements for processes using them. Close collaboration with pillar 1 will be necessary, as it will propose AI-based system design tools and processes that can be evaluated by the actors using the PRISSMA method to be declared applicable and sized to provide acceptable proof elements (tool qualification concept to validate test results).

One of the project's challenges is the integration of simulation as a means of demonstration through the provision of acceptable proof. Indeed, the eventual use of this process as a necessary step in demonstrating mastery of functions is commonly recognized as essential to bringing autonomous mobility services to market due to the complexity of combinations of events and situations that may arise. PRISSMA will need to propose elements enabling the demonstration of this mastery, going up to homologation and incorporating improvements throughout the automated & autonomous road transport system life cycle.

PRISSMA will rely on concepts and definitions from international work (UNECE, NHTSA, etc.), supplemented by the work of French industry working groups to ensure compatibility with their results, benefit from their results, and provide technical and scientific support for industry positions in return.

The France Autonomous Vehicle Plan (FVA) has been working for several years to coordinate the efforts of the automotive and shared transportation industry to develop the technical and regulatory framework necessary for the deployment of autonomous vehicles. Thematic working groups, such as the Homologation & Testing Means working group for Autonomous Shared Transportation Systems (STPA), the Technical Regulations working group for Passenger Vehicles, the Validation working group, and the Automotive Technical Standardization Committee are all instances that bring together French experts on these topics to align the industry's positions and initiate joint projects aimed at testing and demonstrating the safety and reliability of autonomous systems.

Finally, the certification framework and methodologies developed within the framework of PRISSMA should enable France to emerge as a leader in autonomous mobility at the European and international level. With this in mind, France aims to apply for the European call for tenders to support a limited number of test sites in Europe in the coming years. The location of one of these sites in France will assert the French industry's position in the race towards autonomy, while offering an advantage to the industry and economic and social repercussions that will benefit the chosen territory and its ecosystem.

[REF: PRISSMA project application form to BPIFrance, section 1.2 - Objectifs]

1.2 Purpose of document

The purpose of this document is to define the requirements that must be satisfied by the PRISSMA method for demonstrating, justifying, or arguing the safety and security of an AI-based Automated Road Transport System (ARTS). When feasible, the aim isn't to prescribe what the method should do, but rather to specify the outcomes this method should ideally achieve. These outcomes are crucial in the context of its certification or the homologation of its vehicle by a third-party, under the delegation of governments, for authorizing the operation of such a system in public areas. An ARTS achieving to have the PRISSMA certification can be operated with a sufficient level of confidence in its safety and security.

ARTS supplier

The term ARTS supplier is a generic term that can be declined by different entities during the implementation of the method. In some cases of applicable local regulation, this term can designate:

- ARTS Operator
- Service Organizer
- technical ARTS system provide

As an example, the "European ADS act" [EU 2019/2144] has defined the following compliance assessment process to its requirements:

- Part 1: The consideration of the most relevant scenarios for the ODD

- Part 2: The assessment of the ADS design concept and the audit of the manufacturer safety management system
- Part 3: The tests of the most relevant traffic scenarios
- Part 4: The credibility assessment for using virtual toolchain to validate ADS
- Part 5: The in-service reporting to demonstrate the safety performance in the field

Rather than defining the process, methods, and tools required to enable such safety demonstration and compliance assessment of ARTS, this document will focus on the requirements that this method should comply with in order to achieve the intended goals. In doing so, it provides a framework for the various working groups involved in the process, while avoiding restricting or biasing the proposals of methods that need to emerge from these working groups.

Some of these requirements may not be met during this project, as the state-of-the-art in the field of AI safety assessment is still under development. However, ideally, the implementation of the PRISSMA method that would satisfy all the requirements of this specification should be able to assess the safety of an ARTS and enhance its safety over time .

This document is structured around the following 3 main parts:

- Section 2: Specification on how the PRISSMA method should qualify the requirements applicable to the design of the ARTS (which corresponds to the descending part of the V cycle). These requirements should then be verified during the qualification of the AI components or ARTS itself.
- Section 3: Specification on how the PRISSMA method should qualify the inputs (both documents, material or immaterial) used for the evaluation (or the training) of the ARTS, considering that these artefacts are used to verify that the produced ARTS is compliant with its requirements, qualified by the requirements detailed in the section 2
- Section 4: Specification on what the PRISSMA method should verify on how the artefacts qualified in the section 3 have been used to demonstrate that the ARTS is compliant with it's requirements expressed in the section 2.

1.2.1 Key issues regarding safety assessment for AI

In traditional system engineering,the safety insurance is based on the quality insurance principles : Plan-Do-Check-Adjust where it is possible to check that the results comply with the expectation, in iterative enhancement process. In safety critical systems, the generic process for safety insurance is comparable with the generic process detailed below [doi: 10.1109/ISSREW.2019.00091].

- **Hazard Analysis:** Identifying potential hazards associated with the system's usage.
- **Safety Requirements:** Establishing specific requirements to mitigate these hazards at system, software, and hardware levels.
- **Risk Mitigation:** Developing and implementing measures to reduce the identified risks.
- **Verification:** Demonstrating that the risk mitigation measures effectively reduce the risk to an acceptable level.

- **Iteration:** Repeating the process until the safety level is deemed acceptable.

The SOTIF (ISO 21448:2022) relies on the hypothesis that the vehicle functional safety has been demonstrated through the application of the ISO 26262:2018, which in turns relies on the generic quality insurance principle.

States-of-the-art (REF: PRISSMA) in AI shows that such hypothesis does not apply in AI and, more specifically for supervised machine learning algorithms

- **specificability:** behaviors easy to train for with datasets are very difficult to specify using requirements (example is pedestrian detection. What a pedestrian is ? Does it means that people in a wheelchair are not included in this category?)
- **hazard assessment impossible without specification:** How to define risk mitigation requirements when functional requirements are not defined?
- **risk mitigation verification is not possible:** We cannot present irrefutable arguments demonstrating that these risk mitigation requirements are met (neither proof nor postulates that would demonstrate this coverage exist, due to issues of causality and non-linearity).
- **achieved quality level is not quantifiable**: it remains unclear when to stop this retraining process. Iterative improvement of this quality level is not possible
- **isolation of defect:** is almost impossible inside a neural network at the state of the art.
- **quality assurance composition:** demonstrating the system's quality assurance through the quality assurance of its AI components, similar to estimating system MTBF (Mean Time Between Failures) through the MTBF of internal components, is currently not possible at the state-of-the-art.

1.2.2 The qualification strategy of PRISSMA method

The PRISSMA certification of an Automated Road Transport System (ARTS) is based on the successive qualification of its constituent AI components and functions, as well as the concepts and processes of its life cycle:

- Homologation of the vehicles
- Qualification of other system components (supervision, connected infrastructure)
- Qualification of the ARTS supplier process, whether this process involves the integration of existing components or includes, directly or indirectly, the complete development of each component
- Qualification of the ARTS operator and maintainer process, including it's safety management system (Système de Gestion de la Sécurité en Français)

The homologation of the vehicle relies on:

- The qualification of this vehicle supplier's process
- The qualification of the AI components used in this vehicle

All qualification processes through the PRISSMA method are based on equally qualified inputs: requirements, performance and safety objectives, Operational Design Domain (ODD), Object and

Event Detection and Response (OEDR), routes, scenarios, and metrics.

1.2.3 Issues and open topics

1.2.3.1 TRL level of engineering methods used in PRISSMA

Regarding the TRL (technological readiness level) of the possible methods expected to comply with the requirements expressed in this specification, the PRISSMA partners have decided to stay close to the state of the art of the engineering methods (TRL 6 minimum), with the following complement details:

- Require justifications for completeness, coverage, that allow for a consensus between industrialists and certification authorities
- For level 4 shuttle types on a given route (typically the Paris2Connect case): TRL 8 minimum is required (therefore, including trying to comply with current standards, even if they are difficult to apply).

Requirements that can only be satisfied by engineering methods at at low TRL are excluded from this specification (for example requiring a high usage of formal languages to describe the ODD / OEDR to enable automatic reasoning and automatic test generations based on these reasoning are out of scope of this document).

1.2.3.2 Expected level of safety demonstration

The PRISSMA partenrs have identified three possible level of safety demonstration:

1. Require formal proofs (<-> no tests required) of the safety of the expected functionality
   a. Hypothesis: We don't know how to formalize the requirements of certain AI functions
   b. Proofs of coverage of a target space in simulation
2. Require objective evidence that the level of risk is verified at the level of confidence aimed for (Beta) (confidence level, not trustworthiness level)
   a. Specified safety level (hence specified risk level Alpha (GAME))
   b. Obligation of results
   c. Proofs provided by the system provider
3. Require a level of safety assurance
   a. Only trust in the efforts made in the proofs produced to demonstrate the safety of the STRA
   b. Obligation of means
   c. Level of security assurance (like cybersecurity)

Decision: The scope of the PRISSMA method is the item 2: seeking for obligation of results for the safety demonstration

1.2.3.3 Scope of activites impacted

1. Applicable for the verification/validation in the ascending part of the V cycle in the internal process of the ARTS supplier
2. Applicable for the evaluation by a third-party authority knowing that the verification/validation has already been carried out by the system provider
   a. We assume that the STRA provider has already iterated enough that the probability of observing a dangerous behavior is almost nil
   b. We evaluate the proofs of the safety demonstration made by the provider with the two pillars:
      i. Audit of the entire process
      ii. The tests carried out by the third-party aim to give confidence in what has been audited
      Confirm that there is correspondence between the documentary analysis and the system
      Explore the "expert opinion" scenarios that seem insufficiently covered

Decision: Item 1 scope is addressed by Pillar 1 (Confiance IA), and the scope of PRISSMA is Item 2

1.2.3.4 Scope of autonomous vehicles

The target for vehicle autonomy in the proof-of-concepts is indeed SAE 4, even though valid studies for SAE 4 and 5 are entirely possible .

1.2.3.5 System life cycle considered

In the scope of this document, the system life cycle considers the following stages:

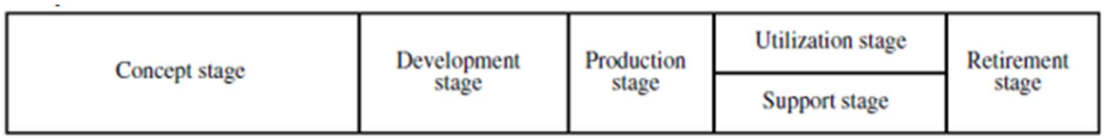


Figure 1 ISO/IEC TR 24748-1 standard generic life cycle stages

Life cycle stages	Purpose
Concept	Define the problem space, characterize the solution space Identify stakeholders' needs, explore idea and technologies, explore feasible concepts, propose viable solutions

Development	Define/Refine system requirements Create solution description – architecture and design Implement initial system Integrate verify and validate the system
Production	Produce the system Inspect and verify
Utilization	Operate the system to satisfy stakeholders needs
Support	Provide sustained system capability
Retirement	Store, Archive or dispose of the system

1.2.3.6 Open comments

The following comments are still open in the V1 version of this document, and will be reviewed in another version:

1.2.4 Glossary

Auditability
The extent to which an independent examination of the development and verification process of the system can be performed [DEEL].

Automated and Autonomous Road transport system (ARTS)
Technical system for automated road transport deployed on *predetermined routes or traffic areas*, and complemented by operational, maintenance, and service rules for the purpose of providing a public collective or private passenger road transport service, excluding transport subject to Decree No. 2017-440 of March 30, 2017, relating to the safety of guided public transport .

Automated and Autonomous Driving System (ADS)
A vehicle belonging to an ARTS. ADS means the hardware and software that are collectively capable of performing the entire Dynamic Driving Tasks on a sustained basis in a specific operational domain design [UE ADS act art 2, def 1]

Confidence
Confidence represents, after a step of processing, combination, or merging/fusion of data, the degree of validity of the result obtained for a specific function (detection, tracking, identification of an obstacle, detection and tracking of a road marking, etc.)

Corner case
A **corner case** is a type of problem or situation that occurs only outside of normal operating parameters—specifically one that manifests itself when **multiple environmental variables or conditions are simultaneously at extreme levels**, even though each parameter is within the specified range for that parameter.

Data Quality
The extent to which data are free of defects and possess desired features. [DEEL]

Edge case
On the other hand, an **edge case** is a problem or situation that occurs when one parameter is at an extreme level . This could involve maximum or minimum inputs, or something unusual like a leap year date. The term "edge" comes from the idea of being on the 'edges' of what is considered normal or typical for the system.

Evaluation
Evaluation refers to the process of assessing some properties of a system of interest.

Explainability
Refers to the ability to explain why the model gave a certain prediction by providing information in a complete semantic format that is accessible to a novice.
The extent to which the behavior of a Machine Learning model can be made understandable to humans [DEEL].

Fidelity
Closeness of agreement between the results of successive measurements of the same measure and carried

out under the same conditions of measurement. [ISO 3534-2:2006 Statistics — Vocabulary and symbols]
NOTE 1 - Fidelity is generally expressed numerically by characteristics such as the standard deviation, variance, or coefficient of variation under specified conditions.
NOTE 2 - The specified conditions may be, for example, repeatability conditions, intermediate precision conditions or reproducibility
conditions (see ISO 5725-1:1994).
NOTE 3 - Fidelity is used to define the repeatability, intermediate precision, and reproducibility of measurement.
NOTE 4 - The term "measurement fidelity" is sometimes improperly used to denote measurement accuracy.

Interpretability

Refers to the ability to understand how the model works by providing sufficient information about the AI model as well as the data used. Interpretability is usually dedicated to machine learning or expert systems .

OD

The Operational Domain (OD) describes what the *environment of the system* world actually is. Whereas the ODD refers to the system capabilities to handle operating conditions, the OD (Operational Domain) refers to the real *environment of the system*
world, describing the real operating condition the system vehicle encounters [derived WP8.11]

Operational Design Domain (ODD)

Means operating conditions under which a given [ARTS] is specifically designed to function, including, but not limited to, environmental, geographical, and time-of-day restrictions, and/or the requisite presence or absence of certain conditions (like traffic or roadway characteristics for [ARTS] [derived from UE 2022/1426 - 19, generalized to any system])
ODD = espace de descripteurs de l'état possible du véhicule vs son environnement (WP8.6)

Qualification

Qualification is "the process of evaluating the capability of a design, procedure, process, item, material, or system to perform its intended function(s) adequately and safely, under specified conditions" [MIL-STD-882E, Section 3.2.34]

Rare event

In the context of machine learning, a "rare event" is an occurrence that happens much less frequently than normal events, often in an imbalanced classification context. Its prediction is challenging due to its rarity. Specific techniques are used to enhance the detection of these rare events despite the inherent data imbalance.

Repeatability

Fidelity of measurement under a set of repeatability conditions. [ISO 3534-2:2006 Statistics — Vocabulary and symbols]

Repeatability conditions

Conditions of measurement in a set of conditions that include the same measurement procedure, the same operators, the same measuring system, the same conditions of use, and the same location, as well as repeated

measurements on the same or similar objects over a short period of time. [ISO 3534-2:2006 Statistics — Vocabulary and symbols]

Reproducibility

Fidelity of measurement under a set of reproducibility conditions.

Reproducibility conditions

Conditions of measurement in a set of conditions that include different locations, operators, and measuring systems, as well as repeated measurements on the same or similar objects. [ISO 3534-2:2006 Statistics — Vocabulary and symbols]

Resilience

The ability of AI functionality to maintain compliance with expected requirements in the presence of inputs outside its use domain (e.g., in the event of failure, intentional or unintentional incident, cyberattacks and/or extreme stress).[REF]

Ability for a system to continue to operate while an error or a fault has occured [DEEL].

Robustness

The ability of AI functionality to maintain compliance with expected requirements in the presence of input data within the intended use domain. [DEEL white paper]

Alternate def: The ability of AI functionality to maintain compliance with expected requirements in the presence of input data within the intended use domain. [REF]

(Global) Ability of the system to perform the intended function in the presence of abnormal or unknown inputs

(Local) The extent to which the system provides equivalent responses for similar inputs.

Validation

Validation is the process of verifying that a system or component meets its intended requirements and operates as intended. Validation includes testing, analysis, inspection, and demonstration to ensure that the system meets its specified requirements for performance, reliability, maintainability, and safety. [MIL-STD-882E]

Specificability

The extent to which the system can be correctly and completely described through a list of requirements. [DEEL]

Verifiability

Ability to evaluate an implementation of requirements to determine that they have been met [DEEL, adapted from ARP4754A].

2 ARTS DESIGN REQUIREMENTS QUALIFICATION

All the inputs used in the PRISSMA method must be previously qualified. Within the scope of the PRISSMA merhod, qualification corresponds to the complete verification of the compliance of these inputs with the requirements applicable to them in this document. In most cases, this verification must be done by the ARTS supplier before initiating the certification of this ARTS by a competent authority for this certification (for both vehicle homologation and complete ARTS certification).

2.1 Identification of AI components, AI functions-of-interest, AI activities-of-interest

One of the first objective of the PRISSMA method is to verify that the functions and components of the ARTS based on AI technologies comply with all the laws & regulations requirements applicable to all the pathways, countries and areas where the ARTS will be operated . The processes involved in any of the lifecycle of the ARTS (design, development, verification, utilization, maintenance, retirement) can also be impacted by the applicable laws and should be taken into account for the identification of the AI dependent elements (whether they would be physical components, requirements, technical data).

AI agent

An AI agent is automated entity that senses and responds to its environment and takes actions to achieve its goals [ISO/IEC 22989:2022]

AI component

An AI component functional element that constructs an AI system artificial intelligence / AI: <discipline> research and development of mechanisms and applications of AI systems
Note 1: Research and development can take place across any number of fields such as computer science, data science, humanities, mathematics and natural sciences. [ISO/IEC 22989:2022]

Artificial Intelligence system/AI system

An AI intelligent system or AI system is an engineered system that generates outputs such as content, forecasts, recommendations or decisions for a given set of human-defined objectives
Note 1 to entry: The engineered system can use various techniques and approaches related to artificial intelligence (3.1.3) to develop a model (3.1.23) to represent data, knowledge (3.1.21), processes, etc. which can be used to conduct tasks (3.1.35).
Note 2 to entry: AI systems are designed to operate with varying levels of automation (3.1.7) [ISO/IEC 22989:2022]

AI tainted activity

Any activity in the lifecycle of an AI technology (including development phase, the production phase, the maintenance phase, or the decomissioning phase) **or** any activity directly using AI technology .

AI activities-of-interest

The set of AI tainted activity for supplying the ARTS by the manufacturer (including for example design, training, validation) and possibly some activities that depends on those AI tainted activities.

AI function-of-interest

In the scope of the PRISSMA Method, the *functions-of-interest* are the functions of the ARTS which depends on an AI component. Therefore, all the functions involved in a functional chain of the ARTS containing an AI component are functions-of-interest.

Therefore, the first steps of the PRISSMA method consist of the following tasks :

- Identification of the AI components and *AI functions-of-interest*
- Identification of the *AI tainted activities* and AI activities-of-interest
- Identification of the *operational domain* and *operational design domain* of the ARTS
- Identification of the *applicable regulations requirements*

PM-937 - Identification of AI components and AI functions-of-interest

The PRISSMA method shall verify that all the *AI components* and *AI functions-of-interest* have been identified by the ARTS supplier .

Note 1: Verification may rely on simple declaration, audit or can rely on more intrusive methods

Note: An update of the ARTS shall trig an update of this verification

Rationale: The known set of AI functions-of-interest is to be known to assess the proper level of evaluation to be undertaken on the ARTS or to check that all the applicable regulations are identified.

PM-1116 - Identification of ODD

The PRISSMA method shall verify that the ODD of the ARTS has been identified by the ARTS supplier.

PM-1139 - Identification of the AI functional domain

The PRISSMA method shall verify that the AI functional domain of the AI functions-of-interest has been identified by the ARTS supplier.

PM-1013 - Identification of OEDR and DDT

The PRISSMA method shall verify that all the *OEDR (Object-Event-Detection-Response)* and the *DDT (Dynamic Driving Tasks)* have all been identified by the ARTS supplier .

Note: OEDR could be classified as a functional requirement of the ARTS, but have been identified separately for clarification regarding autonomous vehicle state-of-the-art

PM-938 - Identification of AI activities-of-interest

The PRISSMA method shall verify that all the *AI activities-of-interest* have been identified by the ARTS Supplier.

Note 1: An obvious activity-of-interest of the ARTS supplier are the Development and Safety insurance process of the ARTS

Note 2: Verification may rely on simple declaration, audit or can rely on more intrusive methods

Note 3: An update of the ARTS shall trig an update of this verification

Rationale: The known set of AI activities-of-interest is to be known to assess the audit to be done on the development process.

2.2 Qualification of applicable regulation requirements

PM-939 - Qualified regulation requirements

Based on the identified *AI functions-of-interest (*PM-937 *), AI activities-of-interest (*PM-938*) and operational domain)*PM-936*)* the PRISSMA method shall verify that the ARTS supplier has:

1. Identified the list of all the applicable regulations to the ARTS and ARTS chain of suppliers (the ARTS supplier and the suppliers of the ARTS subsystems, recursively) and extract the applicable regulations requirements from this list.
2. Conducted safety assessment on those regulations requirements to demonstrate possible inconsistencies and risk on the impact of the whole set of applicable regulation requirements .
3. Traced the selected requirements with the applicable regulations they are extracted from
4. Setup a process to regularly identify any update in the applicable regulations requirements

Note: This activity should be addressed in WP8 "high level requirements"

Example: From the European ADS act, the regulation specifies the performance requirements of level 4 automation vehicle classified in the following 12 categories.

1. Dynamic Driving Task (DDT) under nominal traffic scenarios
2. DDT under critical traffic scenarios (emergency operation).
3. DDT at ODD boundaries
4. DDT under failure scenarios
5. Minimal risk maneuver (MRM) and Minimal risk Condition (MRC)
6. Human machine interaction for vehicles transporting vehicle occupants
7. Functional and operational safety
8. Cyber security and software updates
9. ADS data requirements and specific data elements for event data recorder for fully automated vehicles
10. Manual driving mode
11. Operating manual
12. Provisions for periodic roadworthiness tests

2.3 Qualification of ARTS Supplier's activities-of-interest

2.3.1 Critical system engineering and safety insurance

PM-989 - Critical system engineering activity
The PRISSMA method shall verify that the ARTS supplier is able to demonstrate the compliance of its critical system engineering process with ISO 26262, SOTIF (ISO/DIS 21448) standards and applicable standards for the design and configuration management of critical systems.

In addition to the AI activities-of-interest identified as basements of the method (see PM-938-Identification of AI activities-of-interest) the PRISSMA method has particular interest on the safety assessment process:

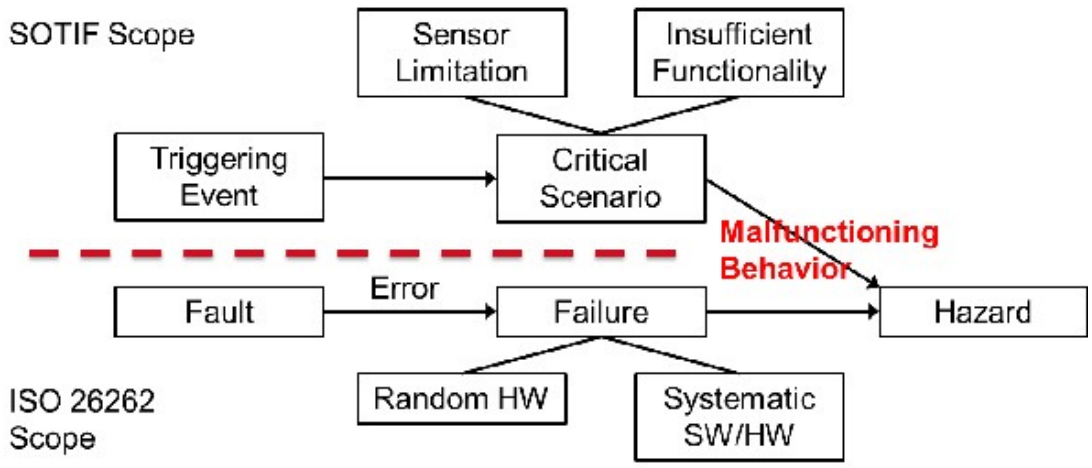


Figure 2 : SOTIF vs ISO 26262

PM-990 - Safety assessment on recorded hazardous situations
The PRISSMA method shall verify that the ARTS supplier is able to demonstrate the safety of the ARTS when the triggering conditions which led to a hazardous behavior of the ARTS (accident or near-accident) are reproduced.

PM-1006 - Compliance with particular PRISSMA requirements
The PRISSMA method shall verify that the ARTS supplier is able to demonstrate the compliance of its process with the relevant qualified requirements from the PRISSMA baseline for this ARTS.

2.3.2 Cyber security and privacy assessment

PM-916 - Cyber-security assessment activity
The PRISSMA method must include security assurance activity to ensure mitigation of the impact of cyber-attacks , and in particular the following aspects:

1. Verification of the resilience evaluation process by the AI suppliers or ARTS suppliers [REQ20221_050]

Note: This security assurance activity shall cover the AI lifecycle data to show mitigation to reach an acceptable level of risks [PM-811 - REQ202211_078]
Note: The definition of this activity is covered by the PRISSMA WP5 project

PM-991 - Data privacy preservation
The PRISSMA method shall verify that the AI activities-of-interest and particularly the data recording activities of the ARTS supplier does not violate local laws on user's data (RGPD).

2.3.3 Maintenance

The STRA provider must have a monitoring activity for hazardous events recorded from the following sources, while maintaining traceability between the source of the recorded event and the event itself :

- from internal STRA sources (vehicle sensors, infrastructure, or supervision)
- from external sources (other STRAs using equivalent infrastructures)
- accidentology

PM-985 - Maintenance and feedback activity
The PRISSMA method shall verify that the ARTS supplier implements maintenance and feedback activities achieving the following outcomes:

1. Update the catalog of scenarios, including misuses, to be used for safety argumentation for the updates of the ARTS.
2. Ensure the recording of pertinent vehicle data (sensor inputs, decisions) in order to provide feedback to the ARTS activities-of-interest in case of system's failure, accident or near-accident in order to fix the ARTS functions.
   Note 1: The access to the recorded video by the local infrastructure to collect the potential hazardous behavior of an ARTS that has not detected near-accident or hazardous behavior should also be considered (to complete the set of data that can be used for post analysis).
   Note 2: Sensors provided only AI-computed output, and not raw input, should be avoided (as this might hide the triggering condition recording)

3. Demonstrate rigorous configuration management practices for the update of the ARTS and AI components in addition to risks assessments and mitigations in the updates of AI components [PM-799 - REQ202211_066, PM-797 - REQ202211_064]

Note: This is implemented in WP7

2.4 Qualification of performance, safety & security objectives

The performance, safety & security objectives come from three different sources:

1. The qualified *regulation requirements*
2. The ARTS supplier
3. The PRISSMA method itself

Only qualified objectives, qualified KPI, qualified metrics applicable to AI function-of-interest or AI activities-of-interest can be used in the PRISSMA method. The ones coming from the applicable regulation requirements are, by essence, qualified to be used in the PRISSMA method. the other shall follow a qualification process defined in the following sections.

2.4.1 Performance, safety & security objectives from regulations

PM-888 - Qualified performances objectives, KPI and metrics from regulations

The PRISSMA method shall verify that the ARTS supplier has identified all the performances objectives, their associated KPI and metrics available from the *applicable regulation requirements and applicapble to* . All these objectives, KPI and metrics are qualified to be used in the PRISSMA method.

Rationale: if performances objectives, KPI and metrics are defined in regulations, then they are applicable.

PM-891 - Safety & Security objectives and risk measurements from regulations

The PRISSMA method shall retrieve all the applicable security objectives & risks measurements from the *applicable regulation requirements*.

Rationale: if security objectives and risk measurements are defined in regulations, then they are applicable

PM-904 - Traceability with regulations

All the requirements, objectives, KPI, measurements retrieved from applicables regulations shall included in the PRISSMA method with the traceability links to their statements in the regulations.

Rationale: To enable periodic review and updates of requirements based on regulations, a precise traceability link is added to the requirements to enable impact analysis.

PM-889 - Identification of missing performance, safety & security objectives from regulations

The PRISSMA method shall verify that the ARTS supplier has conducted activities to identify the missing performance, safety and security objectives from the applicable regulations .

Note: This identification shall target both the AI functions-of-interest and AI activites-of-interest.

2.4.2 Performance, safety & security objectives from ARTS supplier

PM-897 - Verification of performance, safety & security objectives from supplier

The PRISSMA method shall evaluate the performance, safety & security objectives provided by the ARTS supplier.

Note: A list of evaluation activities shall be added based on the following examples:

- Are the objectives compliant with applicable regulations ?
- Are some objectives not addressed by applicable regulations ? If yes, are they accompanied by justification files that justifies how they comply to the objectives, KPI and metrics qualities requirements of the PRISSMA method ?

2.5 Qualified requirements baseline

PM-940 - Qualified requirements baseline

The PRISSMA method shall verify that ARTS supplier has completed the qualified regulation requirements with additional requirements from safety and security assessment of the particular AI function-of-interest of the ARTS.

The set of qualified regulation requirements and the supplier's additional requirements makes together the "qualified requirements baseline".

3 ARTS EVALUATION INPUTS QUALIFICATION

The elements specified in this section are used for the verification of the compliance of the ARTS with the **qualified requirements baseline** defined in the preceding section.

3.1 Qualification of KPI, metrics and data

PM-892 - Definition of additional KPI from regulations

When no KPI is associated to a given performance objective from applicable regulations, the PRISSMA method shall associate a KPI to this performance objective.

Any quantitative KPI used in the PRISSMA method shall have an acceptance threshold (also know as "quality acceptance level" in sampling and measuring) and the PRISSMA method shall verify the relevance of the acceptance threshold.

Note: How to demonstrate the chosen KPI is correct ? Should we complete the requirement to state that the selection process of the KPI shall be justified ?

PM-893 - Definition of missing risks measurements from regulations

When no risk measurement is associated to a given security objective from applicable regulations, the PRISSMA method shall associate a risk measurement to this security objective.

Note: How to demonstrate the chosen risk measurement is correct ? Should we complete the requirement to state that the selection process of the risk measurement shall be justified ?

PM-895 - Definition of additional metrics for KPI

When a KPI has no associated metric for its evaluation, the PRISSMA method shall either:

1. Define an metric associated to the KPI and justify the quality of the associated metric, when the KPI can be evaluated by the use of a metric (see § TBD for the requirements to qualify the metric)
2. Define a method to evaluate the KPI and justify the quality of this evaluation method, when the KPI cannot be evaluated by the use of a metric
3. Any quantitative KPI used in the PRISSMA method shall have an acceptance threshold (also know as "quality acceptance level" in sampling and measuring) and the PRISSMA method shall verify the relevance of the acceptance threshold.

3.1.1 Metrics qualification requirements

The objectives of this section is to define the requirements aimed at ensuring the quality of metrics and data used to evaluate and validate the performance and safety objectives of the ARTS using AI.

To verify the requirements of this chapter: AI experts will propose metrics of poor quality, and low-quality data. These metrics and data should be non-compliant with these requirements and be detected by the PRISSMA method.

In the context of the PRISSMA Method, we define the terms "metric", "reference" and "observation" as follows:

Metric
A metric is an operator qualifying the quality of an observation relative to a reference. [NoRef]

Observation
An observation is a quantity produced at a given time by an agent (a human, a machine, a system composed of human and machine). [NoRef]

Reference
A reference is a "base of a comparison, person or thing from which one defines, estimates, calculates, etc." [Larousse] or "the use of a source of information in order to ascertain something" [Oxford]
By extension, AI lifecycle data are data used a source of in information during the processes for developing, evaluating, operating, maintaining and retiring ARTS. Common examples of references data in the field of AI based autonomous system are "annotated test dataset", "ground truth" or "maps".

Only **qualified metrics** (PM-908- Qualified metrics usage) can be used in the PRISSMA method. Such metric are either:

- Defined in an applicable regulation and are named "regulatory metrics" (see PM-888- Qualified performances objectives, KPI and metrics from regulations) and qualified in the scope of the PRISSMA method (see PM-906- Regulatory metrics qualification activity)
- Defined in addition to the normative metrics and are named "additional metrics" and qualified in the scope of the PRISSMA method (see PM-907- Additional metrics qualification process)

PM-908 - Qualified metrics usage
The PRISSMA method shall demonstrate that it uses only **qualified metrics**.

PM-906 - Regulatory metrics qualification activity

To qualify a normative metric used in the PRISSMA method, the PRISSMA method shall verify that the definition of this metric is compliant with the applicable regulation at the date of the evaluation. This verification shall be executed on a regular basis to maintain up-to-date baseline of applicable normative metrics.
Note: traceability to regulations shall also be maintained with the metric usage

PM-907 - Additional metrics qualification process
To qualify an additional metric (as opposed to normative metric) the PRISSMA method shall execute the follow activity for this metric:

1. Metric specification: The metrics characteristics and properties must be specified. The requirements may involve characteristics such as accuracy, precision, linearity, sensitivity, and specificity of the method using a range of samples or standards.
2. Metric selection: The selection of an appropriate metric may involve reviewing published literature, consulting with experts, and considering factors such as cost, speed, and complexity.
3. Metric validation: The validation activities shall ensure that the chosen characteristics are suitable for the intended purpose. The selection of the persons realizing the validation must be justified (in particularly can they should qualified experts different from the persons who have specified the metric).
4. Metric verification: The metric must be verified to ensure that it performs consistently and reliably in routine use. This may involve analyzing a set of control samples or using proficiency testing programs to assess performance. The selection of the people realizing the verification should also be justified.
5. Metric monitoring and maintenance: Finally, the metric must be regularly monitored and maintained to ensure continued accuracy and reliability.

PM-905 - Metrics relative to AI lifecycle data
The PRISSMA method shall demonstrate that the metrics relative to reference data use **qualified AI lifecycle data** as references.

3.1.2 AI lifecycle data qualification requirements

Data used for a critical system is essential for ensuring the accuracy, reliability, and safety of the system. It plays a critical role in the design, development, testing, and maintenance of critical systems and should be selected and qualified carefully to ensure that it meets the specific requirements of the system.

AI lifecycle data

Refers to the comprehensive data set used throughout the lifecycle of an Artificial Intelligence system (design, maintenance, utilization):

1. Maps: Topographical maps and ground truth data from both simulated and real-world environments.
2. Databases: Involves databases for learning, testing, and validation, supporting supervised or semi-supervised machine learning approaches. This data may serve as a hypothesis or a reference.
3. Accident Statistics: Databases of accidents and statistics serve as a foundation for learning and improvement.
4. Reference Data: Data used as reference for comparison

PM-911 - Qualified AI lifecycle data usage

The PRISSMA method shall verify that the ARTS supplier uses only **qualified AI lifecycle data** and that only qualified AI lifecycle data are used in the PRISSMA method.

PM-909 - Qualification of AI lifecycle data

To qualify a reference data, the PRISSMA method shall verify that the supplier of the data has followed a process to qualify the data, with at least the following steps:

1. AI lifecycle data specification: The reference data constraints and **properties** (PM-912 and PM-913) must be specified, in compliance with any regulation or norm relative to the aspects reference by the data.
   In particular, if the reference needs to evolve in a controlled manner over time, with a specific objective, it is in the specification of this reference that this should first be described. Example: a test base must evolve regularly to prevent the AI developer from knowing the test base, and thus the references associated with this test base will need to follow this evolution.
2. AI lifecycle data selection or definition: The reference data must be extracted from a verified source or created to meet the specified objectives.
3. AI lifecycle data validation: The validation activities shall ensure that the chosen characteristics are suitable for the intended purpose. The selection of the persons realizing the validation must be justified (in particularly they should be qualified experts different from the persons who have specified the metric).

4. AI lifecycle data verification: The reference data must be verified to ensure that it performs consistently and reliably in routine use. This may involve analyzing a set of control samples or using proficiency testing programs to assess performance. The selection of the people realizing the verification should also be justified.
5. AI lifecycle data monitoring and maintenance: Finally, the reference data must be regularly monitored and maintained

PM-912 - Qualified AI lifecycle data properties

The PRISSMA method shall verify that the AI lifecycle data supplier have at least specified, verified and justified the the AI lifecycle data has the following properties in the operational domain of the system of interest regarding the qualified performance, safety & security objectives applicable to this system-of-interest

1. **Suitability**: The AI lifecycle data should be appropriate for the intended use and meet the specific requirements of the critical system.
   Note: Suitability has been preferred over representativity - The relevance of the data depends on the ODD application. This could mean representativity if our goal is to train a calibrated model predicting balanced statistics. It can also mean exhaustively when we aim to over-represent rare phenomena and classes in order to improve the ability to detect them.
   Note: In contrary, for a road map, the suitability is a synonym of representatitivy, because all the features (road signs, road marking, etc..) shall be present in the road map.
2. **Accuracy:** The AI lifecycle data should be accurate and have a known level of uncertainty. The accuracy of the reference data should be justified regarding the performances of the ARTS and the applicable regulations.
   Note: For the position of a micro vehicle, a map with a 10 m precision is insufficient. If road map is used, it icludes accurate road map road features (road marking positioning and type, width of lane, number of lane, curvature, …)
3. **Acquisition Repeatability:** The AI lifecycle data acquisition should produce consistent results when used repeatedly under the same conditions.
4. **Acquisition Reproductibility:** The AI lifecycle data acquisition should produce consistent results when used by different operators/annotators or in different laboratories.
5. **Traceability:** The AI lifecycle data should be traceable to a recognized standard or calibration process, or appropriate observation which ensures that the data is reliable and trustworthy.
6. **Stability:** The AI lifecycle data should remain stable over time if no changes to the data is required, without significant changes in its properties, such as composition or physical characteristics.
7. **Robustness:** The AI lifecycle data shall remain stable over time en in case of occurrence of expected disturbances, critical or hazardous events.
8. **Completeness & identification of missing data:** The AI lifecycle data shall be complete with regard to the objects and space represented by these data.

Note: The resilience is not a property of the reference, but a property of the system using the reference (independantly of the nature of the system, it can be a technical system like an ARTS or a process like the AI training process).

PM-913 - Additional properties for qualified annotated AI lifecycle data

In addition to the properties listed in PM-912, annotated AI lifecycle data shall have the following properties:

1. Include a target population specification which includes
   a. completeness analysis [REQ202211_074]
   b. rare event analysis: [REQ202211_075]
2. **Qualified annotation:** see next requirements ... TBD briefly define annotation qualification process [REQ202211_077]
   Qualification of automatic annotation
   Qualification of human annotation: analysis to to limit cognitive bias [REQ202211_080] // influence factors [REQ202211_045] // intra-annotator qualification, inter-annotator qualification// Identification of influence factors
3. Justification of unbiased selection
4. Qualified balance: - adressed by delivrable 1.4 [REQ202211_075] (PM-920 & PM-921)
5. Qualified sampling: If the reference data are obtained by sampling a given data set, the requirements of the sampling methods are specified and are therefore included in the qualification process of the AI lifecycle data. [REQ202211_081]

Note 1: Like any AI lifecycle data, its properties must be specified so that AI lifecycle data can be qualified (PM-909), so the standard classes are obviously defined.
Note 2: An imbalanced dataset refers to any dataset where the proportions of various classes are not strictly identical. (Note: Class balance generally yields better results in machine learning.) It is crucial to quantify this imbalance. This consideration is only relevant when there is a notion of class or categorization involved.

PM-925 - Qualification of human annotation

The PRISSMA method shall verify that human annotations used for qualified CCR reference data have followed a qualification process to asses the following properties of the annotation:

1. Accuracy of annotations: an expert supervise the annotation
2. Repeatability of annotations: qualification intra annotator
3. Reproductibility of annotations: qualification inter annotator
4. Traceability: record of the identity of the annotator

Note: Example of human annotation qualification process:

1. Define the annotation guide that the annotators will follow to limit human cognitive bias and justify how to control the influence factors and qualify this annotation guide
2. Selection of annotators
3. Training of annotators

4. Qualification of annotators by semantic and syntactic verification on a first sample of each annotator separately in order to eliminate or correct defects
5. Possibly a second qualification phase on a second sample according to the results of the first qualification phase 6) Global annotation of the database
6. Global annotation of the database
7. Inter- and intra-annotator qualification on the complete database
8. Adjustment of annotations following the qualification of the complete database (deletion of uncertain annotations...

Note: Syntax verification involves scrutinizing the 'form' of the data to ensure it is computationally tractable, meaning it is coherent and uniform.
Semantic verification, which can be of various types, includes manual checking of annotations made on a random sample for each annotation source. The objective is to ensure these are in alignment with the guidelines provided in the annotation guide. Intra-annotator verification aims to confirm an annotator's consistency in their annotation work over time. Inter-annotator verification, on the other hand, seeks to ensure that annotations among different annotators are mutually coherent.

PM-946 - Qualified annotation guide
the PRISSMA method shall verify that the rules defined in the annotation covers **all the situations** that the annotators should face, and verify the following aspects of the guide:

- The rules shall be understandable
- The guide has no cognitive bias itself

Note: When the annotation tool is updated, the guide shall be updated.

PM-915 - Continuous improvement of completeness of qualified AI lifecycle data
The PRISSMA method shall have a dedicated process to evaluate and improve the completeness of AI lifecycle data.
Note: this activity is defined in the WP6 and WP7
Note: See reference on the rare event analysis

3.1.2.1 Completeness of annotated AI lifecycle data

PM-952 - Completeness analysis
The PRISSMA method shall verify that the supplier of the AI lifecycle data provides a completeness analysis of the AI lifecycle data with regard to the total expected population of events/objects represented by the annotated AI lifecycle data

- Elicitation of atomic event/object possible
- Extrapolation by computational techniques (combinatorial combinations, etc...)
- Identification of missing data

Note: Cf WP1.2 working group for completeness

PM-953 - Unbiased selection justification

The PRISSMA method shall verify that the supplier of the annotated AI life cycle data provides demonstration that no selection bias has been created during the creation of the annotated AI life cycle data reference (cf PM-913-3) .

3.1.2.2 Rare events and balance of annotated AI lifecycle data

PM-943 - Rare events analysis

The PRISSMA method shall verify that the supplier of the annotated AI lifecycle data provides a rare event analysis of this annotated AI lifecycle data with regard to the total possible population of events/objects represented by the annotated AI lifecycle data.

- Elicitation of rare events and probability and severity assessment
- Justification of the weighting of data for balance or sampling

Note: Update of the Rare event analysis based on utilization and maintenance feedback of the ARTS (WP7]

PM-920 - Maximization of balance of annotated AI lifecycle data qualification

The PRISSMA method shall verify that the AI lifecycle data provider has maximized the balance of the data .

Note 1: The objective is to achieve a balanced distribution of data across all classes, where the number of data points in Class 1 is ideally equal to that in Class n. In instances where perfect balance is unattainable, it is imperative to minimize the imbalance. Any residual disparity must be duly justified, and solutions such as the implementation of focal loss should be employed to address the imbalance.

Note 2: One method has been described in the deliverable 1.4

PM-921 - Unbalanced annotated/supervised AI lifecycle data qualification

If the balance of the CCR reference data is not possible, the PRISSMA methode shall assert that the provider has justify the impossibility to provide balanced CCR reference data, and justify the strategy implemented to use the unbalanced reference data for the IA component.

Note: this activity is described in the deliverable 1.4 (zero shot, one shot, two shots, focal loss...)

PM-942 - Qualified sampled annotated/supervised AI lifecycle data

If the s-data is obtained by sampling an original source of data, the PRISSMA method shall verify:

- Traceability vs original data
- The sampling method has been analyzed to demonstrate that it does not add undesired bias to the original data (example: over-representing rare events in the sampling data is a desired bias to be able to detect those rare events).

Note: It is essential to incorporate focal loss functions to address class imbalance in the context of detection.From a development point-of-view, the test set should be partitioned to enable testing on specific subsets related to this requirement.

3.2 Qualification of Simulator and Test sequencer

Simulator

In engineering, a simulator is a hardware or software tool used to replicate the behavior of a physical system for testing, analysis, and design purposes. It allows engineers to experiment with various scenarios without the need for real-world testing. [IEEE Standard 1076-2017]

Test sequencer

Is a tool responsible for the execution of test cases and the collection of test results. It provides an environment in which test scripts, whether automated or manual, are run to assess the behavior and functionality of a particular system of interest.

Model

In science, a model is an intellectual or material construct designed to represent an aspect or process of reality, highlighting certain essential features while disregarding other details." -[Modeling and Simulation in Science and Mathematics Education (A. A. Berry, 2008]

The Simulation is the result of the utilization of the Simulator.

PM-998 - Qualification of Model & Simulation

The PRISSMA method shall verify that all the models and simulation of the ARTS have followed a qualification process, comparable with the "credibility assessment framework" [UE ADS Act] or any process demonstrating that the models & simulations used in the safety argumentation of the ARTS have been specified, verified, validated, documented, maintained.

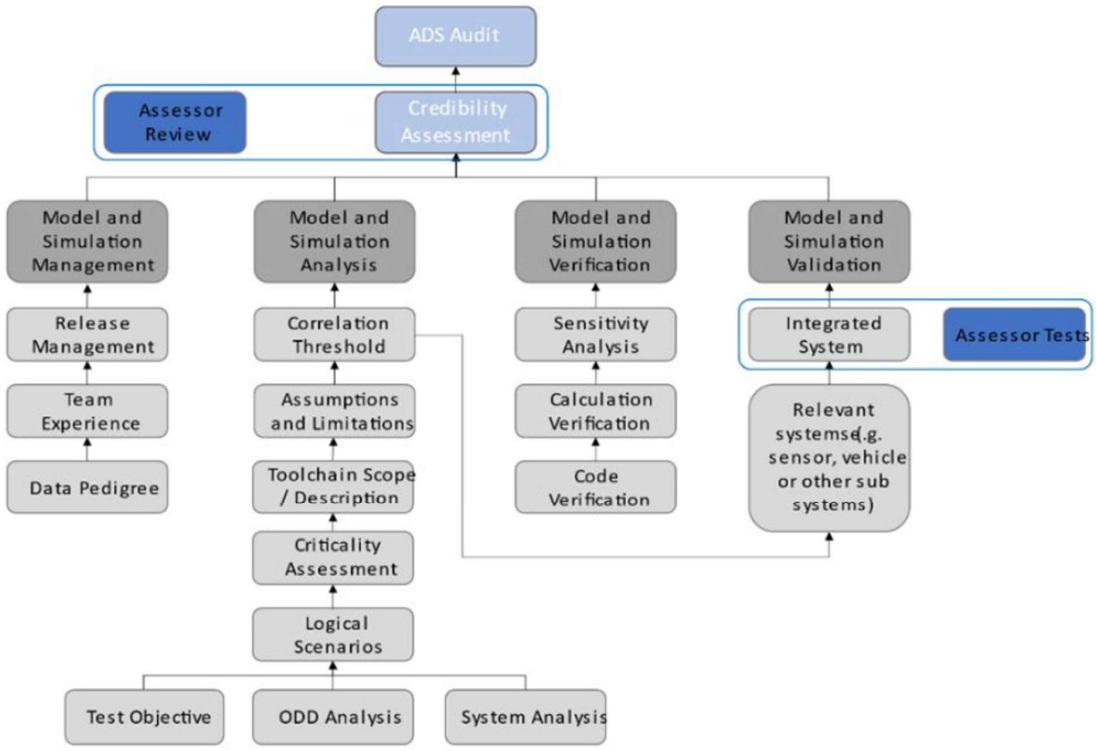


Figure 3 Graphical representation of the relationship between the components of the credibility assessment framework to assess the M&S [1_4]

The next points, listing particular aspects to be verified on particular technologies or components of the simulation, are out of the scope of this document. Rather, they should be listed as particular constraints of the prissma_arts project, or guidelines of the implemented PRISSMA method. Given the focus of the PRISSMA project on AI, these specific requirements are listed below to guide PRISSMA WPs that would use them.

PM-999 - Particular properties for simulation qualification

The PRISSMA method should verify the following properties of the simulation:

1. Assertion of level of fidelity of the sensor (low/middle/high) [PM-834]
2. Assertion of level of fidelity of perception based on sensor with real data or simulated data, particularly in hybrid perception [PM-822, PM-823]
   The bias introduced by the two types of data (real / simulated) must be mastered. System outputs should be similar when using real or simulated data. This implies a level of representativity of the important simulated data.
   Example of verification: the scenario in real and simulation must have the same level of performance
3. Availability of ground truth (reference ?) on the full duration of the scenario in simulated environment
   Element to verify: Ground truth (in simulation) are relative to labelling (objects, road, environment),

very accurate values for the key component state vectors, parameters of sources of disturbance (weather, light, dust, …) [PM-828]

4 ARTS QUALIFICATION

4.1 Introduction

The PRISSMA qualification relative to the safety demonstration of AI based ARTS relies on:

1. The qualification of the AI activities-of-interest of the ARTS supplier
2. The qualification of the applicable requirements and development artifact used for the evaluations
3. The qualification of the AI components (4.2- AI component qualification requirements)
4. The qualification of the ARTS AI functions-of-interest (4.3- ARTS AI functions-of-interest validation requirements)

PM-882 - Scope of evaluation and validation

The PRISSMA method shall evaluate and validate **both AI functions-of-interest of ARTS system and AI components** along with supplier's AI activities-of-interest

4.2 AI component qualification requirements

4.2.1 Qualification of AI Development process

Any ARTS supplier is responsible of the qualification of the subsystems of the ARTS. The suppliers of the subsystems of the ARTS are, in their turn, responsible for the qualification of their component, including the verification of the qualification of the AI component's supplier .

Three scenarios may arise:

- Either the AI component supplier agrees to be audited by the subsystem supplier , in which case the vehicle manufacturer is obligated to follow the recommendations described in the following requirement.
- Alternatively, if the AI component supplier refuses the audit by the subsystem supplier (to, for example, preserve industrial know-how), the qualification of the AI component's development process will then need to be conducted by a third-party organization, accredited to carry out audits of critical system processes. This third-party will provide all the proof elements listed below to the subsystem supplier, who can then provide them to the authority responsible for subsystem certification.
- In the event that an AI component supplier refuses any form of process qualification the component cannot be used in the vehicle.

AI functional domain

The domain of inputs the AI components with it's expected outputs.

PM-956 - Qualification of databases

The PRISSMA method shall verify that the databases used in the whole life cycle of AI components comply with the following requirements:

1. They comply with annotated AI life cycle data qualification requirements (PM-910).
2. Demonstrate a coverage of the qualified scenario library and qualified AI functional domain of the AI function-of-interest the AI component belong to, using a qualified metric .
   Note: metrics coverage is with regard to the performance, safety & security objectives of the ARTS (cf §2.5)
3. The test databases used for the whole life cycle of the AI component are independent of other databases (training and validation).
   Independance: independence between two databases means that data from one database is not found in the other [REQ202211_08]
4. Verify that the provider of the IA component can demonstrates evidences about the use of a strategies to minimize errors in the center or tail of the distribution of the training and validation database

PM-996 - Qualification of AI component development process

The PRISSMA method shall verify that all the AI component of the ARTS have followed a qualification process, comparable with the following one:

1. **Inputs qualification:** All the inputs of the process must be qualified, which particularly targets the artefact//basic elements listed in the sections 2- ARTS design requirements qualification and 3- ARTS evaluation inputs qualification of the present document:
   a. Requirement baselines, OD, ODD, functional domains, Pathway description, expected performance, safety & security objectives, KPIs, metrics and AI lifecycle data must be qualified
   b. All the *AI lifecycle data* used in the process must be qualified, therefore all the inputs of the AI functions must be qualified [PM-909][REQ202211_059]
   c. All the databases used must be qualified [PM-956]
   d. the expected performance must be specified including the acceptable accuracy on this performance. This criteria must be tied to the qualified performance, safety & security objectives of the AI-function-of-interest of the ARTS to be qualified.
2. **AI specification:** Identify the functional and performance requirements, as well as any regulatory and safety requirements that must be satisfied.
   AI functional domain: the requirements must define the functional domain of the AI component and the expected outputs
   Post-treatments: the requirements must define the integrated or suggested post treatments required to use the AI, with description of raw prediction decoding mode [PM-806 - REQ202211_073]
   Interface requirements: the requirements must define the input/output data formats of the IA component, and, in a broader way all the constraints related to the specification of its interfaces [PM-805 - REQ202211_072, PM-803 - REQ202211_070]
   Uncertainty and uncertainty propagation: what is the expected uncertainty of the AI component and how this uncertainty is propagated if many AI stages are used in the AI component
3. **Training, Design and development:** Once the requirements have been defined, the AI must be designed and developed in accordance with these requirements. This may involve developing software, defining the datasets, training of AI and performing test.
   The supplier can demonstrate over-fitting / under-fitting mitigation process, to avoid bias in the training and validation process [REQ202211_060].
   If RGPD applies, the learning process must prevent tracing back to the original data [REQ202211_067]
   Evaluation protocol includes metrics, databases, methods used, test and test systems used. Justify the chosen test protocol.
   A replay (same metrics, same test data sets) of the evaluation by another independent actor must lead to the same results [PM-773]. Test systems should also be qualified test means [PM-770] .
   The PRISSMA method must verify that the provider of the AI component has used convergence measurement techniques during its learning process [REQ202211_063].

Demonstrate **repeatability** of evaluation and identify all the unrepeatable cases [REQ202211_038]
Demonstrate **reproducibility** and justify all the **unproducible cases** (cf metrics fidelity / acceptable jitter in the mectric ?) [REQ202211_037]
In Cyber Security, reproducibility is almost never possible and rely therefore on justification

4. **Verification and validation:** The next step is to verify and validate the AI to ensure that it is compliant with the defined requirements. The evaluation shall be realized by different people involved in the specification, design and production. The verification and validation steps shall be reviewed.
5. **Documentation:** This includes documenting the AI requirements, design specifications, testing results, and other important information.
6. **Certification and accreditation:** This involves demonstrating that the AI meets all regulatory and safety requirements.
   The database used for these evaluations shall be kept secret to the ARTS supplier's, to prevent ARTS supplier for specifically training the AI components to success to the evaluations done during these tests [REQ202211_071]
7. **Maintenance and monitoring:** Finally, once the AI is in use, it must be regularly maintained and monitored to ensure continued safe and reliable utilization.
8. **Traceability:** of all the definition data and artifact used in the AI design process
9. **Independence of the teams:** Demonstration of independence and traceability of development tasks versus the evaluation task, and justification of the skills and knowledge of evaluation team with regard to the AI functional domain .

4.2.2 AI Properties to be evaluated

An important aspect of AI component qualification is to demonstrate the compliance of its functional domain with the one expected during the specification of the AI, with a focus on the outputs at the limits of this functional domain.

The expected **performances** have been qualified prior to the evaluation of the *AI component-of-interest (see section* 2.4- Qualification of performance, safety & security objectives*).*

Considering that AI can be highly non-linear, the figure below summarizes the possible domains to be considered in this characterizations.

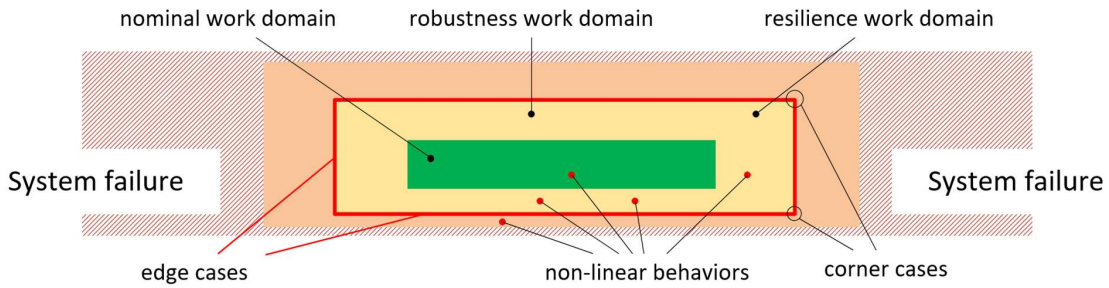


Figure 4 Nominal vs Robustness vs Resilience [Adapted from DEEL]

4.2.2.1 Performance & convergence definition

Reminder : the evaluation of the safety of an AI based system shall rely on the usage of qualified AI lifecycle data .

PM-924 - Qualified performance, safety & security objectives verification

The PRISSMA method shall verify that the provider of the AI components has verified that the AI components meets the qualified performance, safety & security objectives during all the life cycle phases of the AI component (example: learning process, validation).

PM-918 - Center/Tail distribution error minimization strategy

The PRISSMA method shall verify that the provider of the IA component can demonstrates evidences about the use of a strategy to minimize errors in the center or tail of the distribution in the learning process (cf PM-956- Qualification of databases - 4).

Note: this activity is defined in the deliverable 1.4

PM-923 - Convergence measurement

The PRISSMA method must verify that the provider of the AI component has used convergence measurement techniques during its learning.

Rationale: Explains how the supplier has evaluated the performance of the AI component

4.2.2.2 Limits, Robustness & Resilience

PM-1035 - AI robustness qualification

The PRISSMA method shall verify that the supplier of the AI component-of-interest has provided a qualification of the robustness of its AI component providing:

1. Stability analysis (a slight variation of the input parameters should not cause a strong variation of the outputs)
2. Analysis of the performance at the limits of the ODD (critical or precritical scenario analysis)
3. Analysis of the operating range (nominal values, possible disturbed values, influencing factors and parameters)
4. Characterization of AI functional domain outside robustness work domain .
5. Characterization of impact of adversarial attackes (cyber attack)

Note: The SOTIF approach can be used to find pertinent parameters for the characterization of non linear behaviors and robustness

PM-988 - AI resilience qualification

The PRISSMA method shall verify that the supplier of the *AI component-of-interest* has provided a qualification of the resilience of its AI component providing:

1. Characterization of AI functional domain outside robustness work domain.
2. Extreme values
3. Values outside of permitted inputs values (typically out of distribution data)
4. Adversarial attacks (cyber attack)

Note: The SOTIF approach can be used to find pertinent parameters for the characterization of non linear behaviors and resilience

PM-914 - OOD metric for qualified annotated AI lifecycle data

The PRISSMA method shall evaluate the capability of an *AI components-of-interest* to detect **Out of Distribution Data**

Note: this activity is defined in the deliverable 1.4

4.2.2.3 Confidence index

In various types of AI models, a **confidence index (or confidence score)** can be used to indicate the probability that a given prediction or classification is correct. This is common in machine learning algorithms such as logistic regression, support vector machines (SVMs), or neural networks. The confidence index can help in decision-making by providing an estimate of the reliability of a prediction.

Confidence scores can be used in perception algorithms to estimate the likelihood that a detected object is of a certain type (e.g., a pedestrian, a bicycle, or another vehicle). Such scores can be used to inform decision-making in control algorithms.

PM-961 - Confidence index

The PRISSMA method shall verify that *AI components-of-interest* also provide a Confidence Index on their output and a confidence index justification document that details:

1. What are the specifications of this confidence index (how is it computed, what is it's purpose)
2. What are the expected performances of this index
3. How to interpret the different values (range from very good to very poor confidence)

Example: Detection tracking of road markings can provide up to many confidence indexes
1st stage confidence: primitive extractors on the belonging of the pixel to a road marking
2nd stage confidence: occurrence of the tracking - the more the road marking has been positively followed, the better on the overall confidence

4.2.2.4 Interpretability

PM-994 - Logging system

The PRISSMA method shall verify that each AI component is able to log it's output's result based on it's input, and backed up on a duration depending on the criticality of the function.

Rationale: Enable post-analysis in case of incident or accident

Questions: What is the recording frequency? What is the retention period? Do all components need to log? All the time?

Discussion: Confirm with maintenance phase (WP7), that STRA is responsible for recording parameters. Each AI component should log input, output information, explainability, etc. The STRA provider must justify the frequency... (see UE ADS, and UN-R160 Event Data Recorder which are covered by the qualification of regulatory requirements. The following requirement is a reminder.)

PM-1029 - AI component Interpretability functions and justification.

The PRISSMA method shall verify that the AI supplier's has provided the interoperability justification documentation that demonstrate the interpretability of the AI ouputs by domains experts, including the justification of any funtion related to the supply of information required for the interpretation of the outputs (like a logging sytem or additional information about the output).

Example: Detection of dogs or cats, adding pixels to an image to indicate which pixels have allowed to discriminate between dogs and cats.

4.2.2.5 Testability

testability: The "testability" of equipment [...] can be defined as its ability to be tested so that both the equipment manufacturer, [...] user services, and those responsible for providing logistic support can:

- Verify its performance and proper functioning,
- Detect its failures,
- Identify the causes of its failures,
- Remedy its failures.

Within reasonable delay and costs.

PM-1036 - Testability inputs/outputs

The PRISSMA method shall verify that the AI component supplier's has provided the input/outputs point to enable the required qualification level of the AI-Function of interest using this AI component in the rest of the PRISSMA method (including therefore certification, homologation, maintenance and utilization of the ARTS).

Example: If the interpretability require additional outputs for the demonstration, the outputs shall be available

Example: If the homologation of the AV requires additional inputs, for example to enable augmented reality tests in closed road, the inputs must be present (GPS alteration can be also required BUT pay attention to cybersecurity) .

4.3 ARTS AI functions-of-interest validation requirements

4.3.1 ODD, OEDR and Scenario Qualification

4.3.1.1 Qualification of Pathway description and ODD

Considering that the pathway description is a set of parts of the pathway considered to have the same properties, as identified in the ODD grammar, then those properties are considered the same ways as annotation for AI lifecycle data (see SSS-3.1.2-5- Additional properties for qualified annotated AI lifecycle data) .

PM-1022 - Qualification of pathway description

The PRISSMA method shall verify that any parts of the pathway description and ODD, which have been annotated, comply with the **qualification requirements for annotated AI lifecycle data.** All the other parts shall comply with the qualification requirements of AI lifecycle data.

PM-936 - Qualification of the ODD

The PRISSMA method shall verify that all operational design domain has been completely defined by the ARTS Supplier, with particular attention on the aspects of this operational domain linked with a particular regulation.

1. ODD specification: The ODD must be specified, in compliance with any regulation or norm relative to the aspects represented by the ODD [like pas1883].
2. ODD definition: The ODD must be defined, created to meet the specified objectives.
3. ODD validation: The validation activities shall ensure that the chosen characteristics are suitable for the intended purpose. The selection of the persons realizing the validation must be justified (in particularly they should be qualifie experts different from the persons who have specified the ODD).
4. ODD verification: The ODD must be verified to ensure that it performs consistently and reliably in routine use. This may involve analyzing a set of control samples or using proficiency testing programs to assess performance. The selection of the people realizing the verification should also be justified.
5. ODD monitoring and maintenance: Finally, the ODD must be regularly monitored and maintained

Note: CF 8.9(state-of-the art) and 8.11(Taxonomy ODD PRISSMA)
Example: Public/Private zone, country localisation, transport sector (railway, road, open road..)

PM-1046 - ODD taxonomy compliance

The PRISSMA method shall verify that the OD and ODD grammar are compliant to the applicable standards relative to the ODD definition.
Note 1: This aspect should already be covered by the requirements elicitation process defined in section 2- ARTS design requirements qualification, but is reminded here as it is key for the comprehension of the objective of this method.
Note 2: For information, the pas1883 states the following requirements

1. Based on the [used] taxonomy [...], an ODD definition shall be developed and agreed by stakeholders, either individually or in consultation, for the safe operation of the ADS [or ARTS].
2. The ODD definition shall be extensible in a way that allows new attributes or details to be added as a result of stakeholder consultation.
3. The abstraction hierarchy used for the ODD definition shall be at the discretion of the stakeholder. Irrespective of the abstraction level chosen, stakeholders shall specify the ODD attributes used for informing the safety case for the ADS [or ARTS].
4. A stakeholder who defines an ODD choosing a higher abstraction level shall comply with all the sub-attributes, even if they have not been explicitly mentioned in the ODD definition.
5. While performing the DDT, the ADS shall monitor itself and the ODD attributes for the safe operation within the defined ODD, which includes performing the minimal risk maneuver (MRM).
6. As part of the process to show compliance with the defined ODD, ADS developers shall demonstrate test procedures for the defined ODD attributes.

PM-1019 - Qualification of ODD vs OD
The PRISSMA method shall verify that the ARTS supplier can demonstrate the inclusion of the OD inside the ODD or demonstrate that all the spaces of the OD outside the ODD have been identified and addressed.
For each space of the OD outside the ODD the ARTS supplier shall justify how the ARTS remains secure and functional.

4.3.1.2 Qualification of the OEDR

PM-1048 - Qualification of the OEDR
The PRISSMA method shall verify that all OEDR has been completely defined by the ARTS Supplier, with particular attention on the aspects of this operational domain linked with a particular regulation.

1. OEDR specification: The OEDR must be specified, in compliance with any regulation or norm relative to this aspect.
2. OEDR definition: The OEDR must be defined, created to meet the specified objectives.

3. OEDR validation: The validation activities shall ensure that the chosen characteristics are suitable for the intended purpose. The selection of the persons realizing the validation must be justified (in particularly can they should qualified experts different from the persons who have specified the metric).
4. OEDR verification: The OEDR must be verified to ensure that it performs consistently and reliably in routine use. This may involve analyzing a set of control samples or using proficiency testing programs to assess performance. The selection of the people realizing the verification should also be justified.
5. OEDR monitoring and maintenance: Finally, the OEDR must be regularly monitored and maintained

4.3.2 ARTS Qualification (Sim/Closed Road/Open Road)

In the scope of the PRISSMA method, the main objective of the safety demonstration is to verify that the ARTS is at least as safe as human in equivalent situation (GAME principle). This demonstration shall therefore rely on an objective criteria that remains the same in all the situations. The proposal is to demonstrate that the ARTS has no accident resulting in severe or fatal injuries after being operated on a large enough distance or duration (see PM-1051- Qualified safety objective metric) with appropriate justification of the coverage of the evaluation domain, which means the demonstration has been realized on many different situations (see PM-1053- Statistical distribution justification). Ideally, the qualification should also enable to verify that the ARTS has no repeated incidents.

ED: Evaluation Domain

The evaluation domain is the space resulting from the combination of the following spaces:

- The ODD, including:
  - The road infrastructure (Pathway) and the events that can reasonably occur on this pathway (both environmental conditions and actors events)
  - ARTS capabilities limitations with regards to the possible events and environmental conditions (for example, if the ARTS cannot be operated safely at night, then night utilization is out of the ODD)
- ARTS functions and requirements
  - OEDR (ARTS automatic driving requirement, being AI or not)
  - Other system events (risks, failures, functional insufficiency, triggering conditions)

Hypothesis 1:

The vehicles of the ARTS must operate safely even if the other subsystems like the infrastructure or the supervision of the ARTS are dysfunctional.

This hypothesis as an impact only on scenarios where one components of the infrastructure has a failure. For example, if connected traffic lights are used for the utilization of the ARTS, but the ARTS must remains safe when the traffic lights are dysfunctional.

Hypothesis 2:

Any vehicle with SAE 4 level is meant to operate safely inside it's ODD (including its pathway). As the infrastructure is a part of the ARTS, then the pathway is considered inside the system-of-interest.

4.3.2.1 Vehicle AI autonomous driving homologation

PM-1073 - Safety requirements application

The PRISSMA method shall verify that the ARTS supplier has provided safety justification document containing the following information :

1. The evaluation inputs (see 3- ARTS evaluation inputs qualification, 4.2- AI component qualification requirements , 4.3.1- ODD, OEDR and Scenario Qualification)
2. The test campaign specification and coverage analysis versus the evaluation domain
3. The qualification documents for the simulator and the test sequencer used to realize tests in simulation
4. The **safety objectives metric** and the test runs coverage versus this safety objective metric of the autonomous driving functions relying on AI functions-of-interest
5. The identification and coverage % of the test runs on test track versus the test realized in simulation
   Rationale: to give confidence of the correlation between the simulation and the real behavior of the vehicle
6. The identification and coverage % of the test runs on open road versus the test realized in simulation
   Rationale: to constitute a dataset for scenarios encountered on open road and therefore provide proof on the validation of the ARTS
7. The statistical distribution justification of the realized test versus the evaluation domain

PM-1051 - Qualified safety objective metric

The PRISSMA method shall verify that the ARTS provider has defined the safety objective metric in compliance with the state-of-the art and applicable regulations where the ARTS is operated (see 2.2- Qualification of applicable regulation requirements)

Note 1: This metric can be expressed in a distance without fatality (like 275 million fatality free miles) or in hour of travel without fatality (like 10^-7 fatality per hour).

PM-1065 - Qualification of test runs

The PRISSMA method shall verify that the ARTS supplier has performed enough of the necessary, sufficient and representative tests to demonstrate the safety of the ARTS on a sufficient distance (or duration) regarding the safety objective metric qualified announced.

Note 1: This requirement probably implies that huge amount of tests are done with simulation on qualified simulator

Note 2: Due to the statistical distribution justification requirement or in a broader way to increase safety demonstration, the total distance / duration covered by the test has high probability to overflow the initial safety objective metric.

Note 3: The overall safety demonstration relies also on the qualification of the inputs, safety analysis and risks mitigation strategies, all these points are covered in all the previous sections of this document).

Example 1: According to the National Highway Traffic Safety Administration (NHTSA) in the United States, in 2019, there were approximately 1.1 deaths per 100 million miles traveled (about 160 million kilometers). This equates to approximately 0.0000069 deaths per kilometer traveled. To demonstrate that fully autonomous vehicles have a fatality rate of 1.09 deaths per 100 million miles (a reliability of 99.9999989%) with a 95% confidence level, the vehicles would need to be driven 275

million fatality-free miles (440 million fatality-free km) [Sources: National Highway Traffic Safety Administration (NHTSA), "Driving to safety: How many miles of driving would it take to demonstrate autonomous vehicle reliability?" by N. Kalra and S. Paddock.]

Example 2: The example of acceptance criteria indicated in the footnote relies on a safety threshold (10-7 fatalities per hour of operation) based on the analysis of current EU road accidents aggregated data. Such threshold is suitable for the market introduction of ADS based on similar services as the ones which the aggregated data refers to; i.e. buses, coaches, trucks and cars. Therefore, a more suitable reference threshold could be derived specifically for each use-case, also considering the defined operational design domain (ODD) [UE ADS Act].

Example 3: The GAME principle (Globally At Least Equivalent) applies to Automated Road Transport Systems (ARTS). It aims to ensure that the overall safety level of an ARTS is at least equivalent to that of existing or comparable systems. The principle considers users, operating staff, and third parties. It allows for some flexibility by permitting a "system" approach to safety. The guide serves to formalize expectations and provide a framework for industry professionals.[Sources STRMTG GAME guide].

PM-1053 - Statistical distribution justification

The PRISSMA method shall verify that the ARTS supplier has provided the justification of the test run distribution within the evaluation domain.

Note 1: Unless required by an applicable regulation (see PM-939- Qualified regulation requirements) the use of scenario approach is a mean to give confidence in this justification (see 4.3.2.2- Qualification of Scenario)

Note 2: One of the impact of this requirement could be the increase of the safety objective metrics (for example: cumulating 200 Million fatality free miles on highway, and 100 million fatality free miles on crossover , etc, etc...) eventhough the method for the allocation of global objective to different parts of the pathway is not available at the state-of-the-art.

PM-1068 - Qualification of the simulator

The PRISSMA method shall verify that the ARTS simulator, provided by the ARTS supplier, has the following properties:

1. The whole simulator has been provided by a qualified process (specified, designed, evaluated, verified, validated by third party)
2. The validation shall includes a correlation/consistency justification campaign executed with the real vehicles of the ARTS on test tracks and (or) in operational pathway that indicates at least:
   a. usage of qualified correlation/consistency metrics between the simulated ARTS behavior and the real ARTS behavior (digital twin)
   b. % coverage of the tests passed on the real system versus the tests passed through simulatio

PM-1075 - Qualification of the test engine

Any evaluation made in the PRISSMA method and relying on tests ran with the simulator of the ARTS shall use qualified test sequencer that:

1. Enable the PRISSMA evaluation to set it's own test campaign parameters and observe the results of the tests, independently from the one realized by the ARTS supplier
2. Is different from the test engine used by the ARTS supplier or has been certified by an independent organisation
   Rationale: to avoid any bias of the test campaign resulting from a possible bug in the test sequencer

PM-1070 - ARTS Vehicle Evaluation

The PRISSMA method shall realize the following evaluations on the AI autonomous functions of the vehicle:

1. Realize some tests on the simulator using a qualified test engine
2. Realize tests on test tracks and open road if possible
3. Produce the following metrics
   a. The % of the test space covered by track tests (confidence level)
   b. The % of the test space covered by tests in the operational environment (the % will be very low, and related to test permissions on the final route)
   c. The % correlation between the real VEH behavior VS simulated VEH behavior
   d. The % correlation between the results obtained by the ARTS supplier in simulation and the results obtained during this evaluation

4.3.2.2 Qualification of Scenario

This section is under construction

Recommended at the national and international levels for the design and validation of autonomous vehicles.

- NATM (GRAV, WP29 UNECE): scenario database (almost pillar)
- DGITM: GT scenarios.

Allows for representing the complexity of interactions between the system and its environment dynamically and in connection with the AI blocks used. Structuring approach for PRISSMA:

- To link the system level and the AI block
- To link the simulation approach (WP2), controlled environment (WP3), and real environment (WP4)
- To enable a coverage evaluation during the validation of the ARTS through the usage of a "scenario library"

Scenarios describe the contexts that autonomous vehicles (and their AI blocks) will face during utilization=> They are exogenous to the system.
For each logical scenario, the list of **Use Cases** describing each possible behavior of the system must be defined (with the corresponding value ranges for each parameter).
The test strategy is built in the form of a test protocol represented by a set of test cases. Each test case is the instantiation of a situation with specific parameters. To generate useful Test Cases for validation in simulation and the associated stop criteria, it is necessary to associate the expected behavior (DDT - Dynamic Driving Task / OEDR - Object and Event detection and Response implemented in the ART).

Challenges of the scenario approach:

- Numerous inputs to consider: System, components, ODD, OEDR, QoS, Safety, environment, actors, extras, etc.
- Multiple concrete approaches at the national and international levels (see draft L1.5) not yet converged with each other.
- What is the use of scenarios in design?
- What is the use of scenarios in validation?
- What is the use of scenarios in approval/certification?
- Complementarity of analytical and experimental approaches.
- For AI, what use in learning? What use in testing?

Scenario
A scenario is a temporal sequence of action/events (edges) and scenes (nodes). [Ulbricht & al]

Nominal traffic scenarios
Means reasonably foreseeable situations encountered by the ADS when operating within its ODD. These scenarios represent the non-critical interactions of the ADS with other traffic participants and generate normal operation of the ADS [(UE] 2022/1426 - 19].

Critical scenarios
Means scenarios related to edge-cases (e.g. unexpected conditions with an exceptionally low probability of occurrence) and operational insufficiencies, not limited to traffic conditions but also including environmental conditions (e.g. heavy rain or low sunlight glaring cameras), human factors, connectivity and miscommunication leading to emergency operation of the ADS [(UE] 2022/1426 - 20].

Failure scenarios
Means the scenarios related to ADS and/or vehicle components failure which may lead to normal or emergency operation of the ADS depending on whether or not the minimum safety level is preserved [(UE] 2022/1426 - 21].

Note: The definition of "Failure scenarios" are still under discussion at the moment of issuance of this document.

The goal here is to verify the correct implementation of fallback modes specified by the manufacturer, not to verify that we are safe. In the context of an SAE Level 4 autonomous system, all sensors and actuators should be redundant. A failure should therefore lead to a loss of redundancy and a DDT Fallback.
Problem: When we talk about failure, we are talking about the state of the system and not an element of the environment. So, what do we mean by "failure scenario" at the ARTS level?
It might be more relevant to only keep the "critical scenarios" and to consider in these critical scenarios the failures (in the sense of ISO26262) whose consequences could lead to a critical scenario (once risk reduction measures are implemented).

Functional Scenario

Functional Scenario is the predecessor of a group of scenarios describing the same situation in different events that provides a high level of description.

Logical scenario

A logical is one situation of a functional scenario, i.e. each scene, action or event are set. The temporal sequence, the logic of the scene and the action or event is set from the initial scene to the final scene. An interval is defined for each parameter. It is used to describe a behavior or a test.

Concrete scenario

A concrete scenario is an instantiation of a logical scenario giving exact values of each parameter. It is used to describe a test case or to measure/extract a real driving occurrence

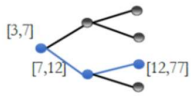
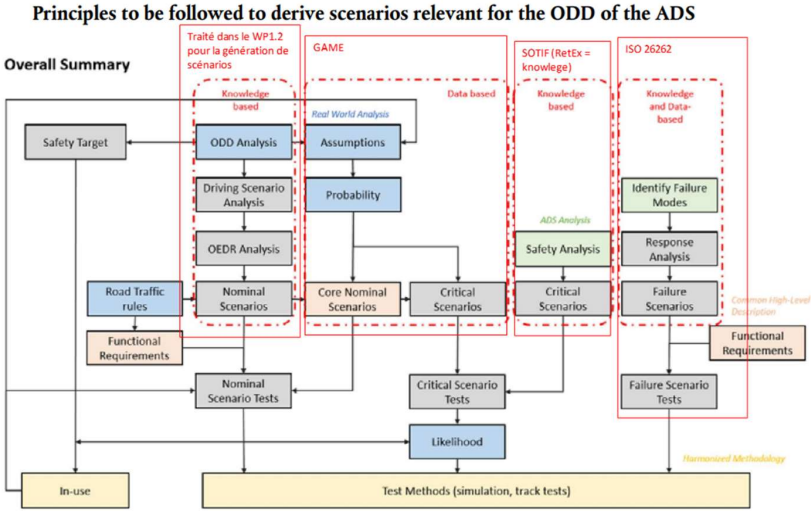
Scénario Scenario	Fonctionnel **Functional**	Logique **Logical**	**Concret** **Concrete**
	Description d'une classe, d'une famille de scénarios sous un nom commun.  Description of a class of scenarios, gathered under a common name	Une possibilité d'évolution du scénario fonctionnel  One line (blue) of evolution of the functional scenario. One temporal sequence is set, i.e. each scene, action or event are set.	Séquence entièrement définie, instanciée.  One fully numerically defined temporal sequence, i.e. a value is given to each parameter of the logical scenario.

Figure 5 Functional, Logical and Concrete scenario



PM-964 - Qualification of functional scenario

If a scenario approach is used for the justification of the statistical distribution (see PM-1053- Statistical distribution justification), the PRISSMA method shall verify that the ARTS supplier has based this elicitation on a qualified process including specification, traceability, verification, ... and in particulary, addressed the following aspects of scenario elicitation:

1. Kinds of scenario in the scope
   a. Nominal scenario - Covering all the functions and functional requirements from the applicable regulation requirements . Compliance of *AI functions-of-interest* (PM-937) to *applicable regulation requirements* (PM-939) falls in nominal scenario scope
   b. Critical Scenarios:
      Notably the edge-cases (PM-781)
      Verify minimization of effect of risk occurrence at acceptable level (Req_FMEA) Failure / Critical Scenario
      Situations where applicable regulations requirements contradictions (PM-842) Critical Scenario
      Example: rouler sur la bande d'arrêt d'urgence OU impact fort dans le véhicule avant (multimodal, multiactor incertitudes & constraints [REF])
   c. Failure scenario
      i. Component Dysfunction (Classic SDF) arising from FMEA at the system level, and which must include failures.
      ii. Out of ODD and out of resilience domain (cf figure in section 4.3.1.1- Qualification of Pathway description and ODD) & PM-782
      iii. Accidentology (or identified near-accidents): derived from past experience feedback (WP7) Critical / Failure Scenarios

Note: As any requirement or piece of information used in critical system engineering, the scenario shall be managed with appropriate lifecycle and configuration management process, relying on scenario management tools.

PM-1020 - Level 4 qualified functional scenario completeness
Based on the analysis of the Qualified Pathway and the Qualified OEDR, the PRISSMA method shall verify that the STRA supplier's has demonstrated the completeness of coverage of functional scenarios versus the pathway and the OEDR.
Note: The elicitation of functional scenario should be realized by the certification authority in addition to the ARTS supplier in order to do this verification

PM-1085 - Logical Scenario Qualification
The PRISSMA method shall verify that the ARTS supplier logical scenario definition has the following properties:

- compliant with standards or state-of-the art method for elicitation (PEGASUS or DGITM)
- include at least the following key-frames: Initial conditions of the actors at T0 (position, speed, state, etc.), Start maneuver parameters (position, speed, type, maneuver related parameters, etc.), End maneuver parameters, Triggering conditions at each keyframe (values or parameters)
- justification of inputs dimensions of the OD discarded as parameters of the logical scenario
- classification of logical scenario:
  All the logical scenario shall be classified in the different a functional scenarios

4.3.2.3 Qualification of the subsystems

PM-1076 - Supporting subsystem qualification
The PRISSMA method shall verify that all the AI component of the ARTS have followed a qualification process, comparable with the following one: Specification, Design, Verification & Validation, Certification and accreditation, Maintenance and Monitoring

4.3.2.4 Qualification of the ARTS in operational environment

PM-1007 - ARTS operation inservice monitoring
The PRISSMA method shall verify that the ARTS supplier is able to demonstrate that the High-Risk AI functions-of-interest of the ARTS can be oversight by human . The expected demonstration is a risk assessment and mitigation with an specification of how the human should interact with the ARTS system to mitigate the risk at acceptable level.
Note: In the french ARTS decree, remote intervention supervision is distinguished.
The possible actions of a remote operator are defined in the ARTS decree:

- To activate, deactivate the system, to give the instruction to perform, modify, interrupt a maneuver, or to acknowledge maneuvers proposed by the system
- To give instruction to the navigation system operating on the system to choose or modify the planning of a route or stopping points for users;

There may be other actions not related to remote intervention but posing a safety issue within the framework of supervision.

PM-1055 - Operational environment evaluation // qualification by monitoring
During the open road evaluation, the STRA must be operated with a human in the loop (monitor) for a sufficient testing period to allow the STRA supplier to build a driving justification file showing:

- The statistical distribution VS the evaluation domain (particularly throughout the year, or even over X years to take into account the seasons, the weather, the particular traffic conditions)
- The absence of dangerous situations attributable to the ARTS, as noted by the human operator (human operator accredited separately without conflicts of interest with the ARTS supplier)

Rationale: The human operator = more guarantee of avoiding an accident during the probationary period.
Note 1: Additional evaluations, provided by a connected infrastructure, will be so much additional credit to provide during this justification.
Note 2: If the human operator is remote, it must also be demonstrated that at no time could their retaking control of the vehicle be prevented (hence the certification of the infrastructure and supervision).

4.3.3 Miscellaneous properties to verify

PM-947 - Verification of AI Resilience
The PRISSMA method shall verify that in case of **Out of Distribution Data** the ARTS still has safe behavior or trigger minimal risk manoeuver.
Note: This situation addresses both normal event out of distribution and rare events out of distribution.
Example: Pedestrian who appears in front of the vehicle outside of a pedestrian crossing.

PM-1031 - Explicability
The PRISSMA method shall verify that the AI component's supplier is providing an explicability justification document along with it's AI that enables a human to assert if an output is correct based on the knowledge of the inputs of the AI component.

PM-995 - AV or ARTS Explicability // Transparency
The PRISSMA method shall verify that the AI component has explicability ouputs and the AI supplier's has provided the documentation enabling third party justification document along with it's AI that enables a human to assert if an output is correct based on the knowledge of the inputs of the AI component.

Example: Detection of dogs or cats, adding pixels to an image to indicate which pixels contributed to distinguishing dogs and cats.

PM-1032 - Logging system

The PRISSMA method shall verify that the ARTShas a logging function in compliance with current regulations and capable of centralizing logs from system components, including AI components, over a sufficient period of time.

Note: in particular, the ability to vary the recorded parameters and their duration based on the system's accident history and maintenance.

PM-1034 - ARTS Interpretability

The PRISSMA method shall verify that the ARTS supplier's has provided the interpretability justification documentation that demonstrate the interpretability of the AI ouputs by domains experts, including the justification of any function related to the supply of information required for the interpretation of the outputs (like logging sytem or additional information about the output) .

5 ANNEX 1: AI FUNCTIONAL REQUIREMENTS

The pure functional requirements of the AI components or specific evaluation techniques are out of scope of this document. Rather they are inputs or outputs to an instantiation of the PRISSMA method applied to a given ARTS.

However, given the focus of the PRISSMA project on AI, these specific requirements are listed below to guide PRISSMA WPs that would use them.

Below is a partial list of the functions to be covered, based on the state-of-the art most important functions regarding ARTS [SAAV]

1. The ARTS manage risks according to the following rules
   a. ARTS does not create accident by its own
   b. ARTS is robust, as far as reasonably possible, to risks caused by others
   c. ARTS complies with applicable driving rules (including those applicable to human drivers) unless it is the only way to avoid an accident
2. Driving policy: The vehicles of the ARTS seeks to maintain safety distance with the preceding vehicle and leaves AD mode after 1st significant shock
3. Understandable: The ARTS operator and the other road users are clearly informed if the vehicle is in AD mode, and it's maneuvers are understandable by the vehicle operator and the other road users (vulnerable or not)
4. Transition to/from AD mode: the ARTS has defined rules to enter or leave AD mode.
5. Takeover procedure: The ARTS operator is able to take over vehicle control at any time
6. Minimum Risk Maneuver: The vehicles of the ARTS has precisely defined Minimum Risk Maneuver that shall be triggered when the ARTS is getting out of its ODD
7. Logging system: The vehicles and ARTS has logging mechanism to record in operation data for post analysis [PM-832]

When qualifying a perception/decision AI function-of-interest, the PRISSMA method should put a particular focus on the following aspects:

1. Validation of static object and dynamic object detection
2. Construction of the dynamic occupancy of the LIDAR sensors
   To validate the construction of a dynamic occupancy grid based on Lidar data, it can be compared with a ground truth dataset. For example, a vehicle equipped with LIDAR sensors can be driven along a route while simultaneously recording the LIDAR data and the ground truth data.
3. Validation of the relative speed estimation of tracked object.
4. Quantification and measurement of uncertainty in speed measurements.
5. Validate and evaluate the time to collision measurement of the vehicles (the time to collision must be true in case of collision and the vehicle shall not hit other vehicles if the time to collision does not detect potential collision).

6. Validation of the clustering and classification for the different perception sensors.
7. Code quality & safety assessment (memory loss, structure é memory loss overflow, etc...) [PM-827]
8. Provide accurate ground truth (have accurate reference as a GPS RTK. Record the proprioceptive data coming from the CAN bus data. This last requirement implies the need of a DBC file in order to parse the frame.) [PM-829]
9. Separate evaluation of clustering and validation if possible, and global validation of the whole classification/clustering[PM-826, PM-825]
10. To propose and dispose of tools, materials, and references (test patterns) for the verification of sensor models in simulation an in real environment.[PM-791 &PM-790]
11. For each exteroceptive sensor: Validate the sensor field of view and sensor performances allowing to detect obstacles and hazardous events potentially dangerous for the ego-vehicule. [PM-789]