



**HAL**  
open science

## Operational Design Domain Monitoring through Uncertain Observations

Thibault Charmet, Véronique Cherfaoui, Javier Ibanez-Guzman, Alexandre  
Armand

► **To cite this version:**

Thibault Charmet, Véronique Cherfaoui, Javier Ibanez-Guzman, Alexandre Armand. Operational Design Domain Monitoring through Uncertain Observations. SIA VISION, Société des Ingénieurs de l'Automobile (SIA), Oct 2024, Paris, France. hal-04829774

**HAL Id: hal-04829774**

**<https://hal.science/hal-04829774v1>**

Submitted on 10 Dec 2024

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Operational Design Domain Monitoring through Uncertain Observations

Thibault Charmet<sup>1,2</sup>, Véronique Cherfaoui<sup>2</sup>, Javier Ibanez-Guzman<sup>1</sup>, Alexandre Armand<sup>1</sup>

1: Ampere, [name.surname@ampere.cars](mailto:name.surname@ampere.cars)

2: Université de Technologie de Compiègne, [name.surname@hds.utc.fr](mailto:name.surname@hds.utc.fr)

**Abstract:** Intelligent vehicles are constantly increasing their automation levels; thus, it is necessary to ensure that such vehicles operate under conditions designed to guarantee safe operation. This paper proposes a method to address this challenge using the Operational Design Domain (ODD) concept and vehicle onboard real-time monitoring. The proposed method comprises a framework that estimates the state of the Operational Domain (OD) while considering measurement uncertainty to determine whether the vehicle is operating within its ODD. This method utilizes fuzzy sets to represent the vagueness of real-world driving environments. We also introduce a taxonomy-based approach to formally define ODDs with respect to vehicle capabilities, independent of the architecture. The approach is demonstrated through simulation use cases, by monitoring the vehicle's OD at runtime and determining when it operates outside its defined ODD. The results pave the way for the introduction of verifiable safety rules based on OD membership degrees.

**Keywords:** Operational Safety, Intelligent Vehicles, Operational Design Domain, Uncertainty

## 1. Introduction

Defining and limiting the operational conditions of automated systems is a way to guarantee their safe operation. The operational design domain (ODD) concept is often used, either during the design phase of the V cycle, to define the environment in which the designed system should be able to operate, or during the operation phase, as specifications describing the operational domain (OD) in which the system can evolve. This is a transparent way to specify minimum safety requirements for systems, that can be understood by both various non-technical end users like regulators and by experts. It is defined by [1] as "Operating conditions under which a given driving automation system or feature thereof 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 traffic or roadway characteristics."

For example, a valet parking system could be designed to operate in an indoor or outdoor parking lot, but not on the road. A SAE level 3 [1] traffic jam

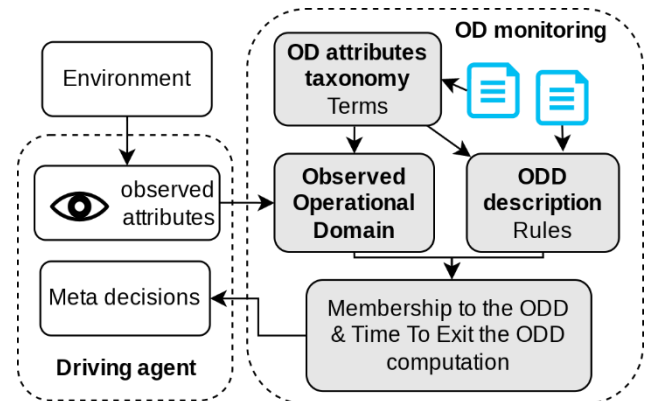


Figure 1: Functional architecture

pilot could be designed for driving in slow traffic, during the day, in favorable weather conditions.

This technical paper, based on recent works published in [2], focus on the role of the ODD concept as a safety guarantee for intelligent vehicles and how it can act as a safeguard for complex mobile systems when combined with real-time monitoring of their OD. ODD Monitoring or OD Monitoring is the task of determining, whether or not a system is operating in the domain for which it was designed. This monitoring can be performed: in advance, during the mission planning phase, using a priori information; and/or during vehicle navigation, using real-time information. This paper is centred on the use of these real-time, often uncertain, measurements to infer the state of the OD relative to the ODD.

ODD descriptions are useful tools to describe the capabilities of a system through the situation it can manage. The ODD is often used to describe the suitable operating conditions for a system. This is done via documents, tables, etc. However, this use remains generally abstract.

This paper aims to contribute to the development of a formal framework for defining and monitoring the ODD through two contributions:

1. A formal description of the ODD.
2. The monitoring of the ODD, at runtime, using uncertain observations.

We will first look at the literature on the topic of ODD monitoring in Section 2, before presenting the ODD description method in Section 3. Then, the proposed method to monitor uncertain OD attributes is presented in Section 4 and the results in Section 5.

## 2. Related works

Related works can be grouped into 4 categories: ODD monitoring, ODD identification, ODD description, and ODD attributes. This work is based on [3] and uses the methodology provided to build an ODD monitoring system. It focuses mainly on ODD monitoring, and secondly on ODD description and attributes. The ODD will be considered as already identified.

Several works have approached ODD monitoring. [4] addresses the need for functional boundaries (i.e. an ODD) and a methodology to define them for driving systems. [5] addresses the restriction of the ODD based on the degraded capabilities of the system. [6] recognizes that the monitoring task should be divided based on the type of ODD attributes and that different strategies may be applied. [7] monitors the ODD of a 2D laser-based localization algorithm by using machine learning (ML) algorithms to look at the extracted feature first. It verifies if the inputs contain distinguishable information that the localization system can use. [8] presents a functional architecture for reasoning with known system capabilities and environment monitoring. Each capability has a contract with preconditions and guarantees. A dependency tree therefore exists between capabilities, so that when one of them becomes unavailable, those downstream also lose their guarantees. An Operational Domain Monitor (ODM) is used to retrieve external information used as input to determine the service quality of a capacity.

[9] employs a statistical approach to identify risky situations by computing their level of compliance with fuzzy requirements. The approach aggregates the values of relevant characteristics of situations from fleet logs. A level of acceptable risk can be defined to accept the most unlikely occurrences. The values are then compared to the requirements to obtain a compliance score between 0 and 1.

The task of ODD monitoring requires the ODD to be described first. It should be noted that there is no single way of representing the ODD in the literature. Works like [10] are based on ODD descriptions. Either using domain specific language or YAML-based descriptions. [11] proposed a two-level language to describe the ODD. The ODD description is represented by a structured natural language that can be converted into a SQL-like, machine-interpretable formal language. Other projects like OpenODD are under conception [12].

Similarly, since these descriptions are based on attributes of the ODD representing elements of the environment, several works, presented in [3] have been organizing them as a taxonomy of elements and more recently [13], [14].

No work has yet explored how to use uncertainty from perception and situation assessment to compute the membership of the OD to the ODD. In order to

quantify risk, accurate uncertainty quantification and propagation from the perception systems to downstream systems is essential. Thus, like other systems, the ODD monitoring output should be representative of the uncertainty from the input.

## 3. ODD description

An ODD description language (see Figure 1) is presented to formally define an ODD textually. It represents the known limits of a driving system, independently of the system's architecture. This definition is based on a set of attributes of the driving environment (road network, road users, weather conditions, etc.), organized in the form of a taxonomy.

### 3.1. High and low level description of the ODD

An ODD description must contain the information needed to define all the operating conditions under which a system has been designed. Or, under Closed-World Assumption [15], it must contain all the operating conditions under which a system has not been designed for.

Even with automatic ODD identification methods, the ODD description should always be easily readable by non-technical users and experts. On the other hand, ODD monitoring requires the ODD description to be machine-readable. Thus, the role of ODD description is to allow for the most human-readable formal description that can be machine evaluated. A statement like "Can't drive faster than 30 km/h with low or worse visibility." will be written as follows:

```
reject target_speed > 30
and visibility <= low
```

As such language cannot be directly machine interpreted, a lower level description is required. Tools like Xtext or its heir Langium allow parsing high-level custom languages, called Domain Specific Language (DSL), and converting them in lower level languages or data structures. Here, the "low-level" ODD description will take the form of a JSON structure that contains all the information needed to

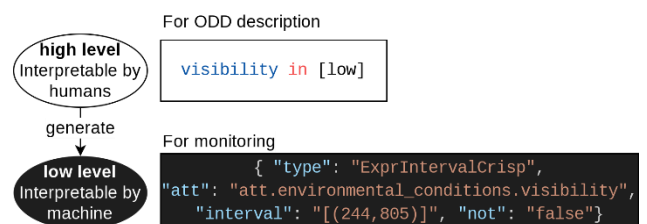


Figure 2: The two levels of ODD description. The high-level description goal is to be as readable as possible while being usable to generate a lower-level version that is directly machine-readable.

describe the ODD statements. In Figure 2, there is a sample of the human-readable "high-level" and its equivalent machine-readable "low-level" JSON description. In this paper, the simplest human-readable format will be used to illustrate ODD statements.

### 3.2. Syntax of the ODD description

Syntactically, an ODD is a list of statements, accepting or rejecting a type of OD. Two types of ODD descriptions could be used: permissive and restrictive. In a permissive description, every OD is accepted by default and each statement starts with "reject" to add additional constraints. We could have the following statements:

"Heavy rain or worse is not OK."

"Highways in low or worse visibility are not OK."

```
reject rain.intensity >= heavy
reject road_importance in [motorways]
visibility <= low
```

Restrictive descriptions are the opposite, no OD is accepted by default, and each statement adds a valid OD. In this paper, only permissive descriptions will be used. In statements, a type of OD is represented by a combination of expressions. Each expression verifies that an attribute in the environment belongs to an interval or a list of possible values.

The "low-level" ODD description is the same as the "high-level", except it is organized as a JSON structure. Each statement is composed of a status (reject or accept), and one or many expressions. The expressions are organized hierarchically as an abstract syntax tree (AST). This is a structure easy to evaluate during execution, used to save the operators to be applied between expressions and their priorities (see Section 4.4 for the evaluation). Each expression contains an OD attribute name, and the value, interval, fuzzy interval, or list to compare the OD attribute value to during execution.

### 3.3. Taxonomy of the ODD attributes

The ODD attributes taxonomy is a tree-like structure that allows the definition of the different attributes used in the ODD description and OD representations. This is the first step in creating a semantic relationship between the named elements of the environment and their machine representation. The second step is to have functions to observe/measure the said attribute as detailed in Section 5.1. In practice, this is a YAML file organizing OD attributes and their metadata in a tree structure. It is based on multiple works, including

existing taxonomies presented in [3]. Each attribute is unique and can be identified by its path in the taxonomy. When this is unambiguous, the name of the attribute or the last elements of the path can simply be used.

For example, `visibility` is the shorthand for:

```
att.environmental_conditions.visibility
```

Each attribute is enriched with different types of information. Firstly, descriptive information such as the name, description, and metric (for numerical attributes) is used to define as precisely as possible what the attribute represents and how it should be measured. For example, the difference can be made between the rain intensity detected directly by a rain sensor and that received by weather information. Next comes the type of attribute. This can be numerical values (visibility, distances, etc), single-choice groups (type of lane markings, road, etc), multiple-choice groups (type of road users in sight, etc), and Booleans (pedestrian interaction, etc). In this paper, only numeric attributes will be addressed. However, most of the techniques presented will also work with the other attribute types. Numeric attributes will have two other information items. Firstly, the maximum range that the value can take (e.g.  $[0, \infty]$  for a distance). Secondly, a list of categories, describing common presets for some attributes. For example, "visibility in [low]" will be equivalent to "visibility in [(244,805)]". The advantage of this approach is that the high-level ODD descriptions are simplified, improving usability while maintaining the benefits of numeric representations, such as order (poor < low < moderate < good).

## 4. ODD monitoring

A method is presented to estimate the degree of membership of a system's observed operational domain (OD) to its ODD (See Figure 1). This degree of membership, ranging from 0 to 1, is used to decide whether the system is within its ODD. It considers the uncertainty of the observations made. We also introduce the possibility of using fuzzy sets to model the numerical attributes of an ODD, as a tool to better represent the vagueness of real-world boundaries.

As mentioned in Section 3.2, an ODD is described as a set of statements, each of which adds a constraint (in the case of a permissive description) or a valid OD (in the case of a restrictive description). Each statement is composed of expressions, separated by Boolean operators. The value of an expression can be evaluated using a membership function, which will depend on the nature of the value and the interval.

### 4.1. Membership function definition

The membership value  $\mu_{att}$  an expression, represents the degree to which the value  $x \in \mathbb{R}$  of an attribute  $att$  belongs to a union of intervals  $I_{att} = \bigcup_{i=1}^n [a_i, b_i]$ . For numerical attributes representing measurements with no uncertainties, the membership function  $\mu_{att}: \mathbb{R} \rightarrow \{0,1\}$  is defined by:

$$\forall x \in \mathbb{R}, \begin{cases} \mu_{att}(x) = 1 & \text{if } x \in I_{att} \\ \mu_{att}(x) = 0 & \text{else} \end{cases} \quad [1]$$

4.2. Membership function for uncertain observations  
The particularity of live information is that it always comes with a level of uncertainty linked to its measurement. Here, uncertain values will be represented as a normal distribution. To determine how much the value lies within an interval, the sum of the areas under the curve is computed for each sub-interval within it.

Let the function  $f_{att}$  be the probability density function of the Normal distribution representing the value  $x$  of an attribute  $att$ . The membership function  $\mu_{att}: \mathbb{R}^2 \rightarrow [0, 1]$  is defined by:

$$\mu_{att}(x) = \sum_{i=0}^n \int_{a_i}^{b_i} f_{att}(x) dx \quad [2]$$

4.3. Membership function with fuzzy intervals

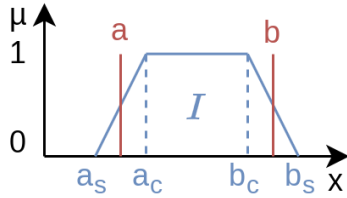


Figure 3: Fuzzy interval.  $a_c$  and  $b_c$  are the core of the interval, inside which the membership value is 1.  $a_s$  and  $b_s$  are the support of the interval, outside which the membership value is 0.  $\mu_I: \mathbb{R} \rightarrow [0,1]$  is the membership function.

To represent the vagueness of the real world, we introduce the possibility of defining fuzzy intervals in the ODD description. In classical set theory, an element either belongs or does not belong to a set. In fuzzy set theory [16], an element can belong to a set with a certain membership degree between 0 and 1. This membership degree is assigned using a membership function. Instead of a classical interval like  $[a, b]$ , it would be a trapezoidal fuzzy interval characterized by the quadruple  $[a_s, a_c, b_c, b_s]$  of real numbers [17]. Here, only trapezoidal fuzzy intervals are used, but this could extend to any kind of fuzzy interval (Figure 3).

For measurement values without uncertainty, the membership function  $\mu_{att}: \mathbb{R} \rightarrow [0, 1]$  of the value  $x$  of

an attribute  $att$  to an interval  $I_{att} = \bigcup_{i=1}^n [a_{s_i}, a_{c_i}, b_{c_i}, b_{s_i}]$  is defined by :

$$\mu_{att}(x) = \mu_{I_{att}}(x) = \max(\mu_I(x), \forall I \in I_{att}) \quad [3]$$

For uncertain measurement values, the membership function  $\mu_{att}: \mathbb{R}^2 \rightarrow [0, 1]$  is defined by:

$$\mu_{att}(x) = \sum_{i=0}^n \int_{b_{s_i}}^{a_{s_i}} \mu_{att}(x) \cdot f_{att}(x) dx \quad [4]$$

4.4. Uncertain statement evaluation using fuzzy logic  
Using the statement's expressions values just computed, the complete statement can be evaluated. As the measures feeding the expressions are uncertain, the resulting membership value will be between 0 and 1 (i.e. not classic booleans).

To combine these values using Boolean operators, fuzzy logic can be used. In fuzzy logic, values, instead of being true or false, are real numbers between 0 and 1, representing degrees of truth. This is an extension of classical logic, allowing the propagation of measurement uncertainty and the fuzziness of intervals to the result. See Table 1 for the fuzzy logic operators used.

The various expressions of the statement are organized in an abstract syntax tree. This is a tree representing the structure of the logical relationships between the expressions, defined in the high-level language using operators and parentheses. For real-time evaluation, once the membership values of the expressions have been computed, they are combined, taking into account this structure and the precedence of logic operations.

Boolean	Fuzzy	Symbol
NOT(x)	$1 - x$	$\neg x$
AND(x,y)	MIN(x,y)	$x \wedge y$
OR(x,y)	MAX(x,y)	$x \vee y$

Table 1: Fuzzy logic operators sorted by precedence.  $\neg$  has priority over  $\wedge$  which has priority over  $\vee$ .

As permissive descriptions have intervals describing values out of the ODD, an additional negation is applied on "reject" statements. This way, the ODD membership associated with any statement uniformly represents how much an OD is in an ODD. The final ODD membership is a conjunction of all the statement values:

$$\left. \begin{array}{l} \text{reject } A \\ \text{reject } B \end{array} \right\} \text{reject } A \text{ and } B$$

4.5. Time to exit the ODD (TTE)



One application of the membership computation is to use the membership value of the predicted attributes to determine how long the system will take to exit the ODD. The notion of time to exit the ODD (TTE) is introduced, representing the estimated time for the system's OD to leave the ODD. The TTE value simply corresponds to the minimum time before a membership falls below a threshold. This low threshold represents, under which membership values the system is considered "out-of-ODD". This threshold can be regarded as a hyperparameter that determines the prudence of the TTE estimation. Values close to 0 indicate that a system is only considered "out of the ODD" when it is almost certainly out. Inversely, values close to one correspond to a conservative estimation of the TTE. In the same way that the Time To Collision (TTC) can be used to make driving decisions, the TTE can be used as an input to make meta-decisions regarding when to use a given system.

## 5. Results

To illustrate the presented method, two simple use cases are used, including one inspired by an accident scenario. Potential applications for meta-decision will be discussed using the transition-of-control (ToC) fallback required in level 3 ADS as an example.

### 5.1. Experimental setup

The method presented has been tested in simulation using Carla Simulator [18] which will serve as the environment for the system (Figure 1). Since the measurements made directly in Carla are perfect, uncertainty will be added to the measurements. If we define  $x$  as the true value, the measured uncertain value  $\tilde{x}$  can be expressed as:

$$\tilde{x} \sim \mathcal{N}(x + \epsilon, \sigma_{\mu}^2) \quad [5]$$

Where  $\epsilon \sim \mathcal{N}(\mu_e, \sigma_e^2)$  is the measurement error, which is normally distributed with variance  $\sigma_e^2$  and bias  $\mu_e$ . It represents the discrepancy between reality and the measured value, i.e. random error and systematic error.  $\sigma_{\mu}^2$  is the measure uncertainty, the expected error of the measure. This is often estimated using the statistical dispersion of measured values against a ground truth. For the various use cases presented below, the error, uncertainty are available in Table 2.

Parameters	ego loc.	bus loc.	vis.
Measurement variance $\sigma_e^2$	0.5	0.5	10
Measurement bias $\mu_e$	0.5	0.5	0
Measurement uncertainty $\sigma_{\mu}^2$	2	2	10

Table 2: Error and uncertainty values for the attributes used in the two use cases. The attributes are the ego location, bus stop location, and visibility.

### 5.2. Statement evaluation from uncertain observations

To illustrate the statement evaluation from uncertain observations, let's take a simple use case in which the ego vehicle moves along a road approaching a bus stop. The ODD rule used by monitoring is, "The distance to the closest on-lane bus stop cannot be less than 20 m".

```
reject bus_stop_on_lane_distance
in [(0, 20)]
```

Figure 4 shows membership variation due to the measurement error from both ego and bus stop location being propagated to the computed distance.

With the time-based sliding windows (of 0.5 second) smoothing the result, and because the ego speed is almost constant, the curve is similar to a sigmoid from a cumulative distribution function. This is expected, as the membership values are computed from the area under the curve of the distance uncertainty distribution, and the distance decreases at an almost constant rate.

The average membership reaches a value of less than 0.5 about 0.2 seconds after the ground truth. This threshold corresponds to an estimate similar to the one that would be obtained by using only the mean of the measured distribution.

A confident threshold of 0.05 can also be considered. The system will be "out-of-ODD" 1 s after the ground truth but with greater confidence. Inversely, conservative thresholds greater than 0.5 will give early "out-of-ODD" signals.

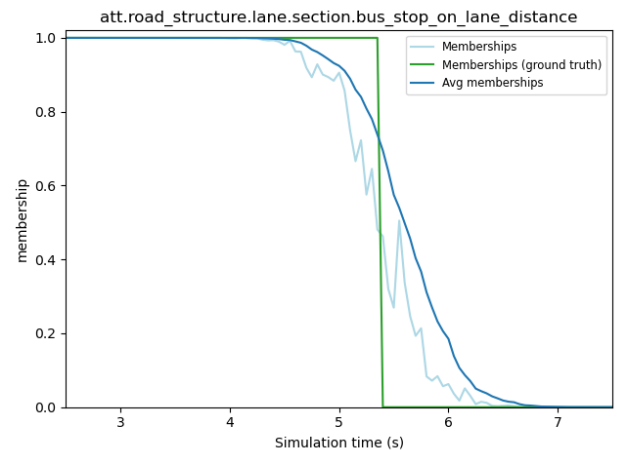


Figure 4: Value of the OD membership to the ODD over time. Memberships (light blue) is the membership evaluated from the uncertain measurement. Avg memberships (blue) is the same membership, averaged over the last 0.5 s. Memberships (ground truth) (green) is the membership evaluated from the real values.

Regarding the standard membership, for both the 0.5 and 0.05 thresholds, two "out-of-ODD" signals happened, while only one with the averaged membership, thus limiting the number of "out-of-ODD" alerts.

Using a trapezoidal fuzzy interval, a similar ODD statement could be:

```
reject bus_stop_on_lane_distance in
  [ (0, 0, 15, 25) ]
```

The second interval bound (15,25) means that, without considering uncertainty, the membership value will start to decrease 25 meters before the bus stop and reach 0 once closer than 15 meters. It can help to represent ODD description that can be subjective by nature. For example, the second interval bound (15,25) could be used to represent optimistic and pessimistic values. Combined with an adapted threshold, it can represent the prudence of the ODD exit estimation. A threshold close to 1 for a pessimistic estimation and close to 0 for an optimistic one. Thus, the time window where the ODD's membership decreases from 1 to 0 is wider (Figure 6). In this context, using fuzzy intervals is convenient to represent subjective or imprecise numeric ranges.

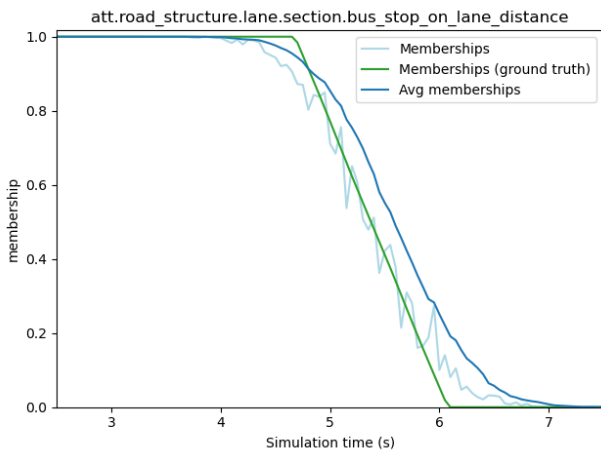


Figure 5: Value of the OD membership to the ODD over time. Similar to **Erreur ! Source du renvoi introuvable.** except that the rejected interval for the distance with the bus stop has a fuzzy bound between 15 and 25 meters instead of just 20 meters.

We saw here that in the presence of uncertain measured values, the membership computation provides a degree of truth for ODD exit, dependent on the localization uncertainty. The smoothed membership computation allows for a valid estimation of the true ODD membership while reducing the number of "out-of-ODD" alerts. Moreover, fuzzy intervals can be used to define an ODD description closer to the vagueness of the world. This result depends on two hyperparameters, the sliding windows average, and the ODD exit threshold.

### 5.3. Time to exit the ODD (TTE)

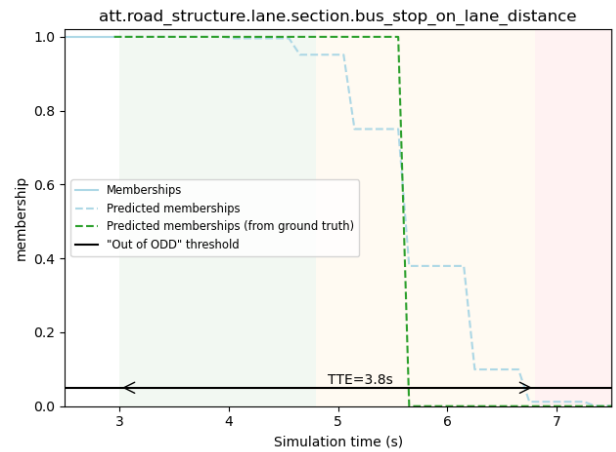


Figure 6: Time to exit the ODD computed from predicted distance between the vehicle and the bus stop.

Using the same use case, based on the predicted future distances to the bus stop, the time to exit the ODD (TTE) can be estimated. Thus, it depends on the expected trajectory of the vehicle and the uncertain location of the bus stop. The TTE is the time until the first membership value goes below a given threshold. In Figure 7 the threshold is 0.05 giving a TTE of 3.8 seconds. The prediction (light blue dashed line) has a stair shape because it is dependent on the vehicle location prediction from the local planner, which is discreet. Predicted memberships are the memberships computed from the predicted measured values (here the distance to the bus top). The predicted membership from ground truth (green dashed line) is the membership value of the predicted attribute values without uncertainty or error (no bus stop and ego localization uncertainty). The TTE is the time until the first membership value goes below the "Out of ODD" threshold.

A meta-decision system could decide to deactivate the driving system and go back to manual driving when the TTE is low. For level 3 ADS, this is called a transition-of-control (ToC) fallback. Then, it can decide to trigger a minimal risk maneuver (MRM) like an emergency stop when the TTE is too low, or the ODD is left. This is represented by the background colors of the figure: white is the past, in green everything is fine, in yellow (TTE=2s) the driving system deactivation is requested, in red (TTE=0s) a minimal risk maneuver (MRM) is requested.

Here, the times are given as an example to better illustrate the application. In practice, the expected time to hand over to the driver would likely be around 30 seconds. While this is not a problem when based on a priori information (road network, etc), it is not always possible to obtain this level of anticipation with dynamic information (presence of pedestrians, distances, etc).

#### 5.4. Multi expression statements

The second use case (Figure 8) is similar to a past ADAS accident [19]. In this accident, the perception system failed to recognize a stopped emergency vehicle in time. The emergency vehicle had emergency lights, during a foggy night, and the ego vehicle was cruising at high speed (87 kph). The emergency vehicle was finally detected 34 meters before impact. The perception failure was probably caused by a combination of factors, the flashing emergency lights, low luminosity and visibility being the main factors.

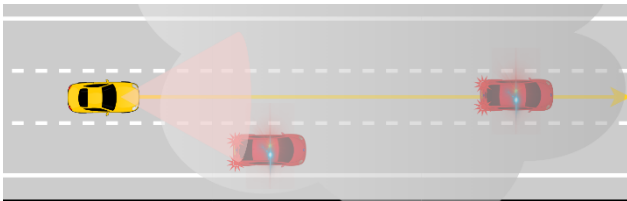


Figure 7: Emergency vehicles use case. The ego vehicle (yellow) is driving straight past the first stopped emergency vehicle (red, on the side), and then toward a second emergency vehicle (red, in the lane) stopped on the road with emergency lights. In this use case, the combination of low visibility and the flashing lights of the emergency vehicles on the trajectory would be detected too late by a perception system not adapted for this type of situation. Here, the simulated detection range was 20 meters.

In the use case we will be using, the difference is that only visibility is used as an aggravating factor for the perception capabilities. Visibility also progressively worsens as the vehicle enters the area. Knowing that the perception system is unable to correctly detect emergency vehicles in low visibility conditions, it would be possible to use the presence of other emergency vehicles, which would not otherwise be considered obstacles, as clues indicating an unsuitable domain of operation. This can then be described as a situation out of the ODD as:

```
reject vehicles.in_sight in
  [emergency_vehicles]
  and visibility <= low
```

In Figure 9, as the vehicle enters a low visibility area, the expression value associated with visibility (in dotted light blue) starts to increase. This is because, in a reject statement, intervals represent rejected values. Thus, being in the interval decreases the statement membership. The yellow area represents when the visibility value becomes low according to the ground truth (solid green line). However, the statement's membership does not decrease because the expression associated with emergency vehicles being in sight is still 0. It is only when the system also has the first emergency vehicle in sight (dotted red

line), that the ODD membership starts to rapidly decrease (red area). As soon as the conjunction of both the low visibility and the sight of the side emergency vehicle is

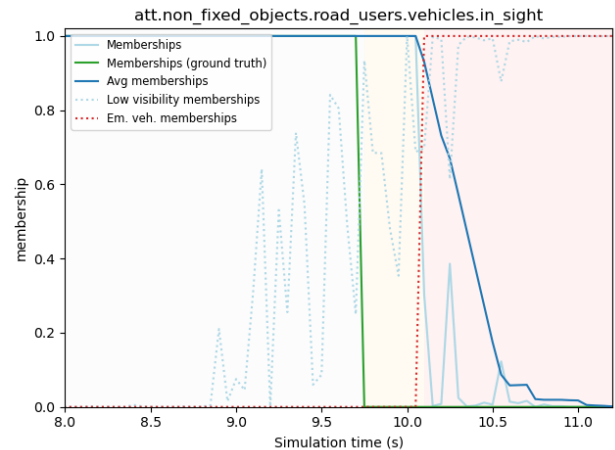


Figure 8: The ODD membership only decrease when all the expressions are fulfilled.

observed, the monitoring system detects that this situation is out of the ODD. Then, depending on the system's level of autonomy, a transition-of-control or MRM can be triggered to avoid risky situations. Of course, there will not always be an emergency vehicle on the side in this type of situation, but this can be generalized to any element that can be used as a clue for unsuitable operating conditions, like warning triangles, warning lights, etc.

With this example, we showed that ODD statements can be composed of multiple expressions, narrowing down the accepted or rejected situations to better correspond to the known limits of a system.

## 6. Conclusion

A method was presented to monitor live numerical ODD attributes (like dynamic elements, weather, road users, etc.), that often have uncertainty in their measures. The proposed odd monitoring system estimates the degree of ODD membership of a system, given uncertain observations of the OD. This degree of membership, ranging from 0 to 1, is used to decide whether or not the system operates within its ODD.

For this purpose, a formal way to describe an ODD was presented. It uses a high-level domain specific language (DSL) as a natural, human-readable format, which can be converted into a lower-level structured description, interpretable at runtime.

This description can be composed of multiple expressions, referencing driving environment attributes organized as a taxonomy.

Fuzzy intervals were also included in the ODD description and monitoring, as a tool to better represent the vagueness of real-world boundaries.



Finally, two use cases allowed to illustrate how ODD description and monitoring can be used to safeguard complex systems in simulation and estimate the time to exit the ODD (TTE) from predicted observations. In the future, we plan to test this approach with real driving systems in order to have more realistic attribute values, as well as identify relevant ODD descriptions specific to each system.

## 6. Acknowledgement

This work has been carried out within the SIVALab, a joint laboratory between Renault and Heudiasyc (Université de Technologie de Compiègne, CNRS), and funded by Renault and the Association Nationale de la Recherche et de la Technologie.

## 7. References

- [1] SAE International: "Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles", J3016\_202104, 2021.
- [2] T. Charmet, V. Cherfaoui, J. Ibanez-Guzman, and A. Armand: "Operational Design Domain Monitoring with Uncertain Measurements", *International Conference on Intelligent Transportation Systems (ITSC)*, Edmonton, Canada, 2024.
- [3] T. Charmet, V. Cherfaoui, J. Ibanez-Guzman, and A. Armand: "Overview of the Operational Design Domain Monitoring for Safe Intelligent Vehicle Navigation", *International Conference on Intelligent Transportation Systems (ITSC)*, Bilbao, Spain, 2023.
- [4] D. Wittmann, C. Wang, and M. Lienkamp: "Definition and identification of system boundaries of highly automated driving," *7. Tagung Fahrerassistenzsysteme*, 2015.
- [5] I. Colwell, B. Phan, S. Saleem, R. Salay, and K. Czarniecki: "An Automated Vehicle Safety Concept Based on Runtime Restriction of the Operational Design Domain", *Intelligent Vehicles Symposium (IV)*, Changshu, China, 2018.
- [6] M. Gyllenhammar *et al.*: "Towards an Operational Design Domain That Supports the Safety Argumentation of an Automated Driving System", *10th European Congress on Embedded Real Time Systems (ERTS)*, Toulouse, France, 2020.
- [7] Z. Alsayed, P. Resende, and B. Bradai: "Operational Design Domain Monitoring at Runtime for 2D Laser-Based Localization algorithms", *International Conference on Advanced Robotics (ICAR)*, 2021.
- [8] F. Woerter, A. Kreuz, A. Salvi, M. Dreiser, and G. Weiss: "Enhanced System Awareness as Basis for Resilience of Autonomous Vehicles", *International Workshop on Critical Automotive Applications: Robustness & Safety (CARs)*, 2021.
- [9] E. Schwalb, A. Richter, and D. Rohne: "Validating Autonomous Behaviors against Partially Specified Ambiguous Requirements", *International Conference on Systems, Man, and Cybernetics (SMC)*, 2022.
- [10] Iain Whiteside: "ASAM OpenXOntology Proposal Workshop 5 Ontologies and ODDS at Five", presented at the ASAM OpenXOntology

Proposal Workshop, Apr. 24, 2020. Accessed: Nov. 02, 2022. [Online]. Available:

<https://www.asam.net/conferences-events/detail/asam-openxontology-proposal-workshop/>

- [11] E. Schwalb, P. Irvine, X. Zhang, S. Khastgir, and P. Jennings: "A Two-Level Abstraction ODD Definition Language: Part II", *International Conference on Systems, Man, and Cybernetics (SMC)*, Melbourne, Australia, 2021.
- [12] ASAM: "OpenODD: Concept Paper." Accessed: Feb. 13, 2023. [Online]. Available: <https://www.asam.net/standards/detail/openodd>
- [13] International Organization for Standardization: "ISO 34503:2023 : Road Vehicles — Test scenarios for automated driving systems — Specification for operational design domain", 2023.
- [14] L. Mendiboure, M. L. Benzagouta, D. Gruyer, T. Sylla, M. Adedjouma, and A. Hedhli: "Operational Design Domain for Automated Driving Systems: Taxonomy Definition and Application", *Intelligent Vehicles Symposium (IV)*, 2023.
- [15] R. Reiter: "A logic for default reasoning", *Artif. Intell.*, vol. 13, no. 1, pp. 81–132, 1980.
- [16] L. A. Zadeh: "Fuzzy sets", *Inf. Control*, vol. 8, no. 3, pp. 338–353, 1965.
- [17] J. Fortin, D. Dubois, and H. Fargier: "Gradual Numbers and Their Application to Fuzzy Interval Analysis", *IEEE Trans. Fuzzy Syst.*, vol. 16, no. 2, pp. 388–402, 2008.
- [18] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun: "CARLA: An Open Urban Driving Simulator", *arXiv: arXiv:1711.03938*, 2017.
- [19] Wall Street Journal: "Watch: Exclusive Tesla Footage Suggests Reasons for Autopilot Crashes", *WSJ*, 2023. Accessed: Apr. 29, 2024. [Online]. Available: <https://www.wsj.com/video/series/in-depth-features/watch-exclusive-tesla-footage-suggests-reasons-for-autopilot-crashes/2FBEE1CA-56E1-4ACC-92B6-DED638B531CE>

## 8. Glossary

ADAS: Advanced Driver-Assistance Systems  
 ADS: Autonomous Driving Systems  
 DSL: Domain Specific Language  
 ML: Machine Learning  
 MRM: Minimal Risk Maneuver  
 ODD: Operational Design Domain  
 OD: Operational Domain  
 ToC: Transition-of-Control  
 TTE: Time to exit the ODD