

Using Ontologies for the Formalization and Recognition of Criticality for Automated Driving

LUKAS WESTHOFEN¹, CHRISTIAN NEUROHR¹, MARTIN BUTZ², MAIKE SCHOLTES³,
AND MICHAEL SCHULDES³

¹Institute of Systems Engineering for Future Mobility, German Aerospace Center (DLR) e.V., 26129 Oldenburg, Germany

²Bosch Corporate Research, Robert Bosch GmbH, 71272 Renningen, Germany

³ika, RWTH Aachen University, 52062 Aachen, Germany

CORRESPONDING AUTHOR: L. WESTHOFEN (e-mail: lukas.westhofen@dlr.de)

This work was supported by the German Federal Ministry for Economic Affairs and Climate Action within the Project “Verification & Validation Methods for Automated Vehicles in Urban Environments.”

ABSTRACT Knowledge representation and reasoning has a long history of examining how knowledge can be formalized, interpreted, and semantically analyzed by machines. In the area of automated vehicles, recent advances suggest the ability to formalize and leverage relevant knowledge as a key enabler in handling the inherently open and complex context of the traffic world. This paper demonstrates ontologies to be a powerful tool for a) modeling and formalization of and b) reasoning about factors associated with criticality in the environment of automated vehicles. For this, we leverage the well-known 6-Layer Model to create a formal representation of the environmental context. Within this representation, an ontology models domain knowledge as logical axioms, enabling deduction on the presence of critical factors within traffic scenarios. For executing automated analyses, a joint description logic and rule reasoner is used in combination with an a-priori predicate augmentation. We elaborate on the modular approach, present a publicly available implementation, and exemplarily evaluate the method by means of a large-scale drone data set of urban traffic scenarios.

INDEX TERMS Intelligent vehicles, safety, knowledge representation, inference mechanisms.

I. INTRODUCTION

PURSUING the feat of automated driving—i.e., taking human operators out of the control loop – has sparked scientific interest for several decades. Pushed by pioneers like Ernst Dickmanns as early as the 1980s, it peaked in prestigious research projects such as PROMETHEUS [1]. Despite focused industry efforts and assertive claims, the dream of full driving automation on urban roads has not been realized ever since.

Currently, systems at SAE Level 3 [2]—e.g., traffic jam assists – are developed only for highly restricted operational design domains (ODDs). This indicates that a key barrier lies within both the complexity and openness of less restricted driving contexts as well as their robust perception and subsequent semantic understanding. Consequentially, the

development of an automated driving system (ADS) mature enough to overcome these challenges can only be mastered by including stakeholders of a diverse set of disciplines.

It is hence imperative to identify a common conceptualization of the ODD across all parties involved in the system’s development and operation. Furthermore, this conceptualization must also be comprehensible for machines, as it forms the basis of the interaction for all subsystems of the ADS with the real world. While for restricted and highly regulated domains such as highways, a model may be feasibly constructed in a comparatively short time, an urban context has two properties impeding this approach. Firstly, it is highly *complex*, meaning that individuals in such scenarios can be of a vast amount of possible types, each of them possessing a multitude of potentially relevant properties possibly with uncountably large value ranges. Secondly, the urban context is *open*, meaning that new classes, properties, and behaviors of individuals can emerge rapidly without prior notice or

The review of this article was arranged by Associate Editor Abdulla Hussein Al-Kaff.

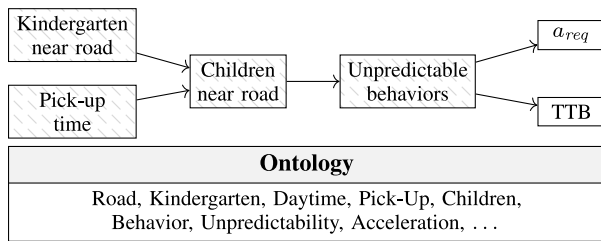


FIGURE 1. Exemplary interaction of criticality phenomena and metrics in traffic conflicts, based on an ontology. Criticality phenomena are hatched, metrics are white. TTB is the time to brake, a_{req} is the maximum required deceleration.

time to respond. Therefore, a central question arises: *How to construct a machine-comprehensible model of open and complex driving contexts?*

Earlier work suggests that such operational domains (ODs) can be structured through the identification and subsequent explanation of safety-relevant factors, called criticality phenomena [3]. This knowledge can then be constituted to a context model, in turn enabling semantic reasoning.

For example, such reasoning allows for an automated assessment of the situational risk. In the area of automated driving, risk evaluations are often performed using criticality metrics [4], e.g., to guide decisions of the driving automation towards states of minimal risk or to derive relevant test cases within a safety case. Essentially, such metrics measure ‘late’ factors that closely precede traffic conflicts, e.g., (predicted) small temporal and spatial distances. Naturally, the question arises: Can the previously derived criticality phenomena be simultaneously used to approximate the actual risk earlier?

Figure 1 sketches an example of how criticality phenomena and metrics interact. Here, the presence of kindergartens goes along with the presence of children alongside the driving area at pick-up times. The resulting possibility of traffic participants with unpredictable behaviors can be, statistically speaking, antecedent to an increased maximum required deceleration (a_{req}) or decreased time to brake (TTB). In total, the recognition and analysis of combinations of factors like the presence of kindergartens, children, and certain daytimes, can lead to a more timely and robust comprehension of the situational risk.

The work at hand addresses the two challenges directly visible in Figure 1: Firstly, building an underlying ontology and secondly, leveraging it to reason on criticality phenomena. The core of our work is based on the family of well-understood description logics (DLs) as a formalism to construct an ontology. Contemporary approaches suggest the use of ontologies for situation comprehension within ADSs, but lack a justification behind the selection and formalization of relevant concepts. As to bridge this gap, we propose to integrate the construction of the ontology into the verification and validation process of an ADS. More specifically, this ontology is iteratively refined by expressing the criticality phenomena using DL axioms and rules. Finally, the

approach enables us to analyze these factors using a statistical evaluation of their association with traffic conflicts as measured by criticality metrics.

To summarize, the *main contributions* of this work are

- 1) to provide a method that enables the
 - construction of an ontology based on a set of criticality phenomena,
 - formalization of such factors based on the ontology,
 - inference of their presence in traffic scenarios, and
- 2) an open-source implementation of the method, including
 - the Automotive Urban Traffic Ontology, and
 - an ontology-based tooling that recognizes criticality phenomena in traffic scenario data.

After presenting related work in Section II, Section III introduces the foundations of our approach, namely the methodical criticality analysis and the formalisms of description logics and rules. The method to both construct an ontology as well as to infer the presence of criticality phenomena (first contribution) is the topic of Section IV. The implemented ontology (second contribution) is portrayed in Section V, followed by an example on its iterative construction based on the formalization process in Section VI. To conclude contribution two, Section VII presents our publicly available tooling. An initial evaluation is provided in Section VIII, for which we have exemplarily chosen a drone data set, depicting the usage of this method for an in-depth semantic analysis of data.

II. RELATED WORK

For road traffic, conceptualizations of knowledge have been collected, aggregated, consolidated, and described since the regulation of the traffic domain. For example, in Germany, there exist harmonized taxonomies and guidelines on the type of objects as well as how they shall be constructed [5].

For automated systems sufficiently intelligent to navigate complex traffic scenarios, the subsequent step is making this aggregated knowledge accessible. A unified terminology of the basic models – scene, situation, scenario – was defined by Ulbrich *et al.* [6]. Within those scenarios, it was found that classes of entities could be structured along the so-called 6-Layer Model, a deliberately informal but comprehensive framework [7]. We base the work at hand on these natural-language definitions as well as their classes of entities and provide a formal implementation thereof. Going into more detail, Czarnecki *et al.* collected large amounts of structured albeit informal domain knowledge regarding the environment of automated vehicles [8], [9]. Furthermore, several standards present a structured domain model alongside, e.g., ASAM OpenSCENARIO [10]. These sources were considered during the development of our implementation.

Hence, an ontology with formal semantics can be constructed from structured knowledge. For automated driving, several groups have formalized their domain conceptualizations by means of ontologies to represent both knowledge

and data. This includes the RONNY ontology for intersection situations [11], the Automotive Global Ontology (AGO) [12], an Object-Oriented Framework (OOF) [13], and an approach of Core Ontologies (CO) [14]. Recently, the community initiated standardization efforts for the simulation domain [15]. Three of those ontologies are developed for specific use cases – intersection recognition for RONNY, closed world scenario descriptions for OOF, and automated decision making for CO – while the ontology presented for the work at hand aims to support various competencies. The AGO provides a knowledge organization framework focused on data labeling, but also includes a methodical approach to derive ontological concepts. This derivation is based on pre-existing data schemes to foster compatibility for the subsequent labeling use case. Hence, none of the ontologies considers its systematic integration into a safety case by means of a traceable identification and formalization of concepts w.r.t. the safety-relevant aspects of the OD. Rather, they represent the particular views and knowledge of the authors and knowledge sources.

Besides formalizing (shared) conceptualizations, research has leveraged the capabilities using logic reasoning on formalized ontologies to facilitate artificial intelligence functionality in ADSs. For a general semantic interpretation of perceived scenes, approaches have been presented as early as 2006 [16], based on the foundations of prior research in logics for artificial intelligence, including expert systems [17] as well as spatial and temporal logics for robotics [18], [19] in the 1970s to 1990s. Applications for automated driving can be traced back to using RONNY on the 2007 DARPA Urban Challenge vehicle AnnieWAY [11], which relies on a plain DL approach for scene inference on road geometries. We adopt a similar dogma, but also consider scenarios, i.e., the dynamical actions of traffic participants, in the inference process. Furthermore, our work specifically addresses the problem of lifting concrete domains to abstract ones such that input data becomes suitable for DL reasoning. Subsequent to RONNY, various ontology-based approaches for situation comprehension and planning components of automated vehicles were proposed. This includes a turning assistant for urban intersections [20] and context-aware speed adaption systems [21], [22]. Furthermore, ontological formalisms were profoundly leveraged by Hülsen *et al.* in order to interpret the driving context at urban intersections using a combined DL-rule approach [23]. Their application of ontology-based inference on A-Boxes is related to the evaluation presented in Section VII: They specifically leverage the open world assumption and rely on rules and description logics for situation analysis. Note, however, that the authors were developing their approach specifically for run time inference. Reasoning is hence only performed on abstract concepts within a reduced ontology. Furthermore, this inference approach is purely scene-based, whereas the work at hand aims to show how inference on large ontologies can be performed on complex scenario data at design time.

While these use cases relate to the situation comprehension of an ADS, our approach views ontologies as an inherent cornerstone of a safety case, which is in turn required for homologation. In the end, a shared ontological model between the stakeholders of the safety case will be necessary. For this purpose, it is hence insufficient to omit a justification for the selection of concepts, e.g., why to differentiate between a personal mobility device and a bicycle. This justification can then be used for verifying and validating the situation comprehension of the ADS. First steps have been made by Jatzkowski *et al.* who demonstrated how the relevancy of scene concepts can be derived from abstract system skills [24]. In order to methodically support a safety case, Bagschik and Menzel examined how a knowledge base can be leveraged for automated scene creation [25] which in turn can be used to derive test scenarios from keywords [26]. In such a scenario-based verification and validation process, it is essential to assign observed concrete scenarios to scenario classes, an approach coined as tagging in prior work [27]. Albeit all four approaches are methodically integrated into their respective verification and validation processes, they do not highlight the applicability of the underlying logical formalisms for automated reasoning, or leave this issue for future work. However, it is only due to a logical foundation that a) concepts become rigorously traceable and b) inference can be leveraged for, e.g., formal scenario analysis or consistency checks of the domain model (i.e., whether certain axioms are contradictory).

Finally, when moving from risk assessment at design time to dynamic risk assessment, early work shows that ontological reasoning can be employed to quantify risk at run time [28]. Here, a rudimentary and small ontology was used for demonstration purposes, leaving open the methodical construction of larger ontologies necessary to represent complex urban contexts and the handling of issues accompanied by the growth in size in terminological and assertional components.

III. PRELIMINARIES

A. CRITICALITY AND ITS PHENOMENA WITHIN THE OPEN CONTEXT

1) THE OPEN CONTEXT PROBLEM

An open and complex urban context poses great challenges when aiming to safely deploy an ADS [3], [29]. Therefore, the ISO/DIS 21448 instructs to reduce the set of unknown hazardous scenarios, which can lead to unsafe situations if they are not considered in the design process [30]. This requirement becomes harder to fulfill as the context grows in openness and complexity.

2) CRITICALITY ANALYSIS

It is therefore imperative to analyze the operational environment prior to system design, which was coined under the term *criticality analysis* [3]. Its goal is to identify and understand the OD elements that are – independent of the system’s realization – safety-relevant for the driving task.

Hence, it uncovers both potential unknown unknowns as well as increases the understanding of known unknowns. The analysis delivers evidences for OD coverage and decomposition by a) the creation of a finite catalog of abstract scenarios that can be used to identify both requirements on the system behavior and relevant test cases, b) an argumentation about the coverage of such a catalog w.r.t. the OD, and c) a downstream effort reduction by providing generic statements about the safety-relevant elements of the OD.

To this end, one is specifically concerned with identifying situations of high accident risk. Hence, criticality can roughly be understood as a function of the probability and severity of the occurrence of *any* accident given a traffic situation. We build upon the following definitions from prior work.

Definition 1 (Criticality) [3]: Criticality is the combined risk of the involved actors when the situation is continued.

Aspects of criticality are measured using criticality metrics:

Definition 2 (Criticality Metric) [4]: A criticality metric is a function $\kappa : \mathcal{S} \times \mathbb{R}^+ \rightarrow \mathcal{O}$ that measures for a given traffic scene $S \in \mathcal{S}$ at a time $t \in \mathbb{R}^+$ aspects of criticality on a predetermined scale of measurement $\mathcal{O} \subseteq \mathbb{R} \cup -\infty, +\infty$. Scenario level criticality metrics extend this definition from scenes to scenarios.

Criticality metrics measure late factors present in the chain of events leading to traffic conflicts. Consequentially, a measurement of factors preceding in turn increases in criticality metrics can be fruitful for purposefully generating evidences in a safety case, e.g., to identify factors of high relevance. We refer to these factors as criticality phenomena.

Definition 3 (Criticality Phenomenon) [3]: A criticality phenomenon is a single influencing factor, or a combination thereof, that is associated with increased criticality in a scene or scenario.

This work examines how a formalization of such phenomena can be leveraged for the construction of an ontology as well as for an analysis of criticality in traffic scenarios. The construction of such an ontology is inherently tied to a safety case, as elaborated in Section IV-A. A subsequent ontology-based analysis contributes, in turn, to the single steps of a verification and validation process, as illustrated in Section IV-B.

B. KNOWLEDGE REPRESENTATION AND REASONING

Knowledge representation and reasoning concerns formalisms that model expert knowledge as well as methods thereon that enable machines to reason about conceptualizations of the real world. We base our approach on those foundations, which are concisely introduced in the following.

1) ONTOLOGIES AND DESCRIPTION LOGICS

The term ontology originates from philosophy, denoting the discipline of studying things, their existence, and human conceptualizations thereof. Based on this, the fields of formal logic and computer science specify an ontology as a formalization of a conceptualization of a shared mental model of

a certain aspect of the real world (often called the *domain of discourse*).

Such a shared formal model of the domain of discourse enables multiple stakeholder to agree on a common and consistent terminological basis. This is specifically important considering the multiple points of failure in information exchange: Different symbols can stand for the same real-world things, and different real-world things can be referred to by the same symbols. Furthermore, a layer of indirection is introduced by the interpreter through first resolving symbols to certain concepts, which is eventually resolved to a (potentially unintendedly different) real-world object. Ontologies make those relations explicit: they clearly assign semantics to concepts which are unambiguously denoted by a set of symbols. They can also aid in making the ambiguities of natural language explicit. Having a consistent and commonly understood ontology therefore mitigates mismatches in the semantics of used symbols and concepts among all parties.

DLs is a family of formal logics to represent such ontologies as a decidable fragment of first order logic. At the core of DLs are so-called *concepts* (or classes), which are semantically defined as sets of individuals, and *roles* (or relations), which are, analogously, sets of tuples of individuals. Individuals are concrete, existing instances within the domain of interest [31]. All three can be referred to by their names, i.e., symbols.

Definition 4 (Vocabulary): A vocabulary is a triple $\mathbf{N} = (\mathbf{N}_R, \mathbf{N}_C, \mathbf{N}_I)$ of sets of *role names* \mathbf{N}_R , *concept names* \mathbf{N}_C , and *individual names* \mathbf{N}_I .

Definition 5 (Ontology): An ontology over a vocabulary \mathbf{N} is a tuple $\mathcal{O} = (\mathcal{T}, \mathcal{A})$, where

- \mathcal{T} is a finite set of terminological assertions (called *T-Box*), namely *general concepts inclusions* (GCIs) such as $C \sqsubseteq C'$ for two concept names $C, C' \in \mathbf{N}_C$ as well as *role inclusion axioms* (RIAs) such as $r \sqsubseteq r'$ for two role names $r, r' \in \mathbf{N}_R$, and
- \mathcal{A} is a finite set of *concept and role assertions* (CAs and RAs) (called *A-Box*), such as $C(x)$ and $r(x, x')$ for $x, x' \in \mathbf{N}_I$, $C \in \mathbf{N}_C$, and $r \in \mathbf{N}_R$.

Note that the possible logical operators that can be used to define the right- and left-hand sides of the GCIs and RIAs in \mathcal{T} are determined by the given DL fragment, consisting, e.g., of negation (\neg), intersection (\sqcap), union (\sqcup), and universal ($\forall r.C$) and existential ($\exists r.C$) role quantification. Semantically, the operators are defined in terms of an interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, where $\Delta^{\mathcal{I}}$ is the interpretation domain and $\cdot^{\mathcal{I}}$ an interpretation function assigning each operator a subset of $\Delta^{\mathcal{I}}$ or $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$. For example, we say that $C \sqsubseteq C'$ iff for all interpretations \mathcal{I} it holds that $C^{\mathcal{I}} \subseteq C'^{\mathcal{I}}$. $\top, \perp \in \mathbf{N}_C$ are unique names for the top (universally true) and bottom (universally false) concept with $\top^{\mathcal{I}} = \Delta^{\mathcal{I}} \wedge \perp^{\mathcal{I}} = \emptyset \forall \mathcal{I}$. The work omits further details and assumes some familiarity with DLs [31].

For our purposes, we additionally define the special functions $con : \mathcal{T} \rightarrow 2^{N_C}$, $rol : \mathcal{T} \rightarrow 2^{N_R}$, and $ind : \mathcal{A} \rightarrow 2^{N_I}$ which retrieve the set of employed roles, concepts, and individual names from the given GCI, RIA, CA, and RA respectively. In conjunction, we say $nms := rol \cup con \cup ind$. Furthermore, the set of T-Box axioms that involve a concept or a role X is defined as $axs(X) := \{\alpha \in \mathcal{T} \mid X \in nms(\alpha)\}$.

2) RULES

The modeled axioms of a DL ontology are often enhanced by rules in form of a set of Horn clauses [32].

Definition 6 (Rule Atom): For a given ontology O over a vocabulary N , a rule atom over a set of variables names V is an expression of the form

- $C(v)$,
- $\mathcal{r}(v, v')$,
- $v = v' \circ v''$ or $v \sim v''$ for $\circ \in \{+, -, \cdot, \div\}$ and $\sim \in \{<, \leq, =, \neq, \geq, >\}$,

where $C \in N_C$, $\mathcal{r} \in N_R$, and $v, v' \in V$, $v'' \in V \cup \mathbb{Q}$.

Definition 7 (Rule): For a given ontology O over a vocabulary N , a rule over variables V is a Horn clause of a set of antecedent rule atoms R_A over V and precedent rule atoms R_P over $V' \subseteq V$, i.e., of the form $\bigwedge_{r_a \in R_A} r_a \implies \bigwedge_{r_p \in R_P} r_p$. Semantically, they are defined by universal quantification over the variables of V .

3) REASONING

A reasoner can then deduct new assertions from the given ontology, and is therefore able to check propositions and answer queries. For example, it can deduce the axiom $C'(x)$ from the previously sketched axioms in Definition 5. Generally, we denote the entailment of an axiom α from an ontology O by $O \models \alpha$. Based on this, a subset of relevant decision problems is:

- a concept is inconsistent, i.e., $C \sqsubseteq \perp$,
- a concept subsumes another, i.e., $C \sqsubseteq C'$,
- two concepts are equivalent, i.e., $C \sqsubseteq C' \wedge C' \sqsubseteq C$ (concisely, $C \equiv C'$), and
- membership of a given individual to a concept, i.e., $C(x)$.

Furthermore, knowledge bases can be queried, i.e., retrieving all individuals $x : C(x)$. These reasoning inquiries are substantiated to our domain of discourse in Section IV-B.

IV. REASONING ABOUT CRITICALITY FOR OPEN CONTEXT SYSTEMS

A. ONTOLOGIES FOR STRUCTURING AN OPEN AND COMPLEX CONTEXT

The necessity of tackling the open context problem has been introduced in Section III-A, for which we propose to use ontologies as introduced in Section III-B. In its core, ontologies are based on the so-called *open-world assumption*, stating that if $O \not\models \alpha$ then $O \models \neg\alpha$ does not necessarily hold. Therefore, ontologies distinguish between the actual truth value of a statement and the information about

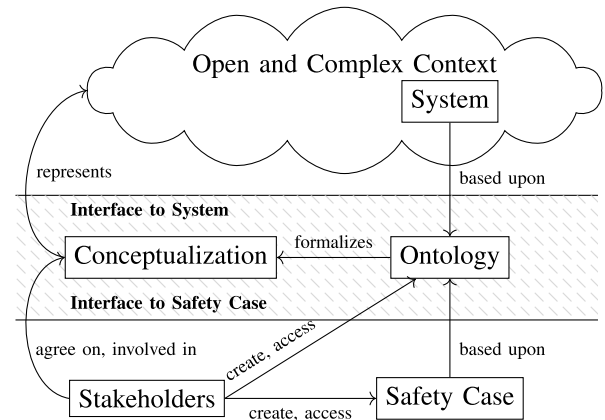


FIGURE 2. Schematic representation of the role of an ontology in developing and safeguarding complex systems in open contexts.

it that can be inferred from the current knowledge, mitigating the problem of incomplete knowledge about the open context. Here, the openness refers to the inherent existence of things possibly unknown at design time, e.g., a traffic participant may not be detected by a machine learning perception system due to its anomalous shape (such as costumes during carnivals). Although a reasoner may find that, based on the knowledge retrieved from the perception system, $O \not\models (\exists has_traffic_entity \cdot Traffic_Participant)(s_1)$ holds, it will not conclude that no traffic participant is present in the current scene s_1 . This can be then, for example, be propagated when querying the ontology whether it knows about the absence of vulnerable road users close to the road.

On a more general note, Figure 2 schematically depicts how an ontology functions as a central interface between the real world and the various entities that relate to the world.

In the life cycle of a complex system, those entities are manifold in appearance, ranging from persons, such as developers, engineers, and testers, up to artifacts, such as the safety case, data schemes, and even the system itself, all with diverse models of the world. A mismatch in the understanding of the context between one of those entities can be fatal. This issue specifically concerns data-focused applications as a well-founded safety case relies not solely on a single data source but rather on various recording modalities to support its claims, e.g., from test vehicles, naturalistic driving studies, or drone data from intersections. Their data schemes need to be aligned in order to avoid discrepancies between these evidences. For example, a safety case may demonstrate that a machine learning-based pedestrian perception has an acceptable False Negative Rate w.r.t. the overall positive risk balance. Everything else should then relate to precisely the same understanding of a pedestrian that was used for labeling the test data and hence for calculating the False Negative Rate.

Naturally, the questions arise how to reach an agreement on a common conceptualization as well as how to

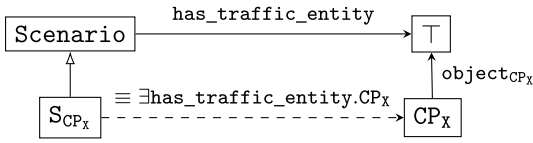


FIGURE 3. Model of the relevant concepts and relations of the competency questions. A solid edge with solid tip is a custom relation, a solid edge with open tip is the subsumption relation, and a dashed edge is a definition.

construct a formalization thereof. We propose to guide the creation of such a model along the identification of criticality phenomena. A methodical criticality analysis iteratively assembles a list of safety-relevant factors within traffic scenarios, although its completeness argumentation is based on two assumptions:

- 1) there exist finitely many criticality phenomena, and
- 2) their traces are recognizable within data.

The justification of these assumptions rests upon a comparison to human driving skills [3, Sec. IV-C]. The completeness assumption can then be justified as follows: By definition, each criticality phenomenon is associated with criticality. If we choose a suitable measurement principle (e.g., a suitable criticality metric) that can uncover their implications by finding related traffic conflicts, we are able to identify their traces in data. By analyzing whether this event can not be explained by a previously discovered phenomenon—i.e., we have found a new phenomenon –, it is possible to achieve a completeness based on the overall finiteness assumption [33]. This completeness can then be recognized by a reduced frequency of unexplained traffic conflicts. Further discussion on discovering and explaining phenomena are given by Neurohr *et al.* [3].

Concurrently to the criticality analysis, a shared ontology can be created from these factors, closing the gap in handling the open context at design time. Note that the completeness of the shared ontology is based on the validity of the justification of the two previously mentioned assumptions. Obviously, the ontology needs to be maintained during deployment. Furthermore, concepts irrelevant for safety, e.g., those regarding mobility or performance goals of the system, may be added.

B. COMPETENCY QUESTIONS

We now derive competency questions that aid in developing an ontology for structuring the open context.

1) PRELIMINARIES FOR COMPETENCIES

First, we introduce some high-level DL notations on the representation of criticality phenomena and scenarios. Those high-level classes and their relations are depicted in Figure 3.

We start with the concept of all scenarios, *Scenario*. Scenarios can have various member individuals *i*, denoted by the relation *has_traffic_entity*. Criticality phenomena are modeled as sets of such scenario entities. For each criticality phenomenon *X*, we denote its corresponding concept $CP_X \sqsubseteq \exists \text{has_traffic_entity}^{-1}.T$.

For example, the class of all fast vehicles can be defined as $CP_{\text{fast}} \equiv \text{Vehicle} \sqcap \text{Fast_Object}$. Whereas this example solely refers to one subject w.r.t. criticality, such phenomena can generally address one or more objects in addition to the subject. Occlusion, for example, has the occluded entity as its subject but also relates to the occluding entity as well as the entity for which the occlusion happens. Therefore, objects are referred to by the relation object_{CP_X} , mapping from CP_X to a number of objects.

Eventually, we are interested in decomposing the scenario space into a set of abstract scenarios covering the identified criticality phenomena. For a given CP_X , we therefore denote its scenario class by $S_{CP_X} \equiv \exists \text{has_traffic_entity}.CP_X$.

2) DERIVATION OF COMPETENCY QUESTIONS

Starting early in the design phase of ADSs, contemporary approaches suggest to cluster the scenario space into abstract scenario classes. This clustering can be formally analyzed to support the experts decomposing the OD. Relevant questions are:

- Whether a set of scenario classes can occur at the same time – if they are disjoint, e.g., risk estimations can be performed without explicitly considering their conjunction—i.e., $S_{CP_1} \sqcap \dots \sqcap S_{CP_n} \sqsubseteq \perp$ (T1).
- Whether a scenario class is subsumed by another, as to facilitate a hierarchical decomposition within the safety argumentation, i.e., $S_{CP_1} \sqsubseteq S_{CP_2}$ (T2).
- Whether the chosen decomposition is complete, i.e., $\text{Scenario} \sqsubseteq S_{CP_1} \sqcup \dots \sqcup S_{CP_n}$ (T3).

This decomposition can be supported by a counterexample-guided abstraction refinement as proposed by Neurohr *et al.* [3, Sec. V-A5]). For this, is necessary to assess whether a given scenario is member of a certain scenario class or an individual is or is involved in a criticality phenomenon—i.e., $\neg S_{CP_X}(s)$ (A1), $\neg CP_X(i)$ (A2), or $\exists \text{object}_{CP_X}^{-1}.T(i)$ (A3). Reasoning services on the ontology can support answering such questions formally by delivering counterexamples as to why these memberships do not hold. Furthermore, data recording and subsequent scenario mining is often used to identify relevant test scenarios. Given a real world traffic recording, we can find representative instances for our previously defined abstract scenario classes by asking a reasoner to query for A1 to A3. Finally, during system testing, answering A1 to A3 allows to check whether a concrete simulation or test run is an instance of a given test case. As one central element of a safety case, system testing can contribute valuable evidences for a necessary homologation prior to deployment.

However, even after successful homologation, constant monitoring of the ODD is required post-deployment, e.g., to detect changes in the assumptions on the ODD that were made in the pre-deployment safety case. The ontology can be used to formalize such assumptions on the ODD definition. An ontology-based monitor can then decide whether

TABLE 1. Competency questions of the proposed ontology. Questions with bold-faced identifiers are focused on in this work.

Reasoning	DL notation	Explanation	Example	ID
<i>A-Box</i>	$S_{CP_x}(s)$	Membership of a scenario s within a scenario class S_{CP_x}	Does the scenario have an occluded pedestrian?	A1
	$CP_x(i)$	Membership of an individual i within a criticality phenomenon CP_x	Is the pedestrian occluded?	A2
	$\exists \text{object}_{CP_x}^{-1}.\top(i)$	Membership of an individual i as an object of a criticality phenomenon CP_x	In the given scenario, is the ego vehicle causing an occlusion?	A3
	$\mathcal{O} \models \text{Scenario}(s)$	Consistency of an observed scenario s with the modeled knowledge	Is the perceived pedestrian with 40km/h consistent with the modeled knowledge?	A4
<i>T-Box</i>	$S_{CP_1} \sqcap \dots \sqcap S_{CP_n} \sqsubseteq \perp$	Compatibility of a set of criticality phenomena	Can the criticality phenomena Foggy Night & Sun Glare exist simultaneously?	T1
	$S_{CP_1} \sqsubseteq S_{CP_2}$	Abstraction relation of criticality phenomena	Is the concretization of Occluded Road User into Occluded Vehicle and Occluded Pedestrian correct?	T2
	$\text{Scenario} \sqsubseteq S_{CP_1} \sqcup \dots \sqcup S_{CP_n}$	Completeness of a set of scenario classes	Is the scenario space decomposition into right and left turn maneuvers complete?	T3

a) the observed scenario is consistent with the modeled knowledge about the world (i.e., the assumptions) and b) the observed or predicted future scenario still lies within the ODD. Otherwise, appropriate mechanisms have to be executed by the ADS. The first query reduces to the formal expression $\mathcal{O} \models \text{Scenario}(s)$ (A4), whereas the second one can be based on A1 to A3. Especially after incidents but also in joint operation with humans, e.g., for Level 3 functions, the system’s behavior needs to be explainable at runtime. If a vehicle chooses a behavior based on specific circumstances in its environment (i.e., assesses A1 to A3), it is advisable to investigate why this behavior was chosen. We can query a reasoner to derive a logical explanation as to why S_{CP_x} was deemed to be present.

Table 1 presents the derived seven competency questions – A1 to A4 and T1 to T3 – the proposed ontology was implemented against. Note that there also exist other possible competency questions, such as model finding – algorithmically generating a witness for the satisfiability of $\top \sqsubseteq \neg C$, e.g., a concrete scenario s for a given abstract scenario class S_{CP_x} . Such competency questions require different architectures and approaches in ontology design and were thus excluded.

Whereas the ontology has been implemented w.r.t. all competency questions, the evaluation of the ontology-based tooling as presented in Section VII and Section VIII is focused on the first three competency questions, A1 to A3. This allows, on one hand, to keep the ontology implementation generic and applicable to a variety of use cases, and, on the other hand, the practical evaluation focused and comprehensible.

C. METHODOLOGICAL FORMALIZATION AND RECOGNITION CRITICALITY

We have introduced the working hypothesis that by creating an ontology via a set of identified criticality phenomena we

are able to achieve an iterative completeness of the model w.r.t. the safety relevant factors. Furthermore, we established the practical relevance of the guiding competency questions regarding the safety assurance of ADSs within a thereby created model. In particular, we demonstrated the importance of answering competency question A1 – scenario membership. Based on these foundations, we propose a general method that

- 1) formalizes criticality phenomena within an iteratively refined ontology,
- 2) transforms data from various sources into the language of the ontology, and
- 3) utilizes DL & rule reasoners to infer the presence of criticality phenomena in such traffic scenarios

The process is depicted in Figure 4 where these steps are respectively marked by ①, ②, and ③.

In step ①, the T-Box \mathcal{T} is iteratively constructed by formalizing a list of criticality phenomena. This list is constantly growing by continuous conduction of a criticality analysis [3]. We therefore pick a criticality phenomenon CP_x and create a formalization of its semantics using appropriate rules or DL axioms. This semantics can be based on both existing concepts – C_1, \dots, C_n – as well as newly introduced concepts – C_k, \dots, C_1 . This necessitates the addition of C_k, \dots, C_1 to \mathcal{T} , in turn recursively formalizing them until the valid assumption can be made that so-called basic concepts can not be further specified. A validation of the assumption is required, i.e., it has to be argued why neither a rule nor a description logic axiom is suitable to represent the given concept.

If one aims to infer the presence of a formalized criticality phenomenon, these basic concepts are then required to be either directly quantifiable or computationally derivable from the data. In that case, they shall have a definition in natural language as to facilitate a correct observation. If

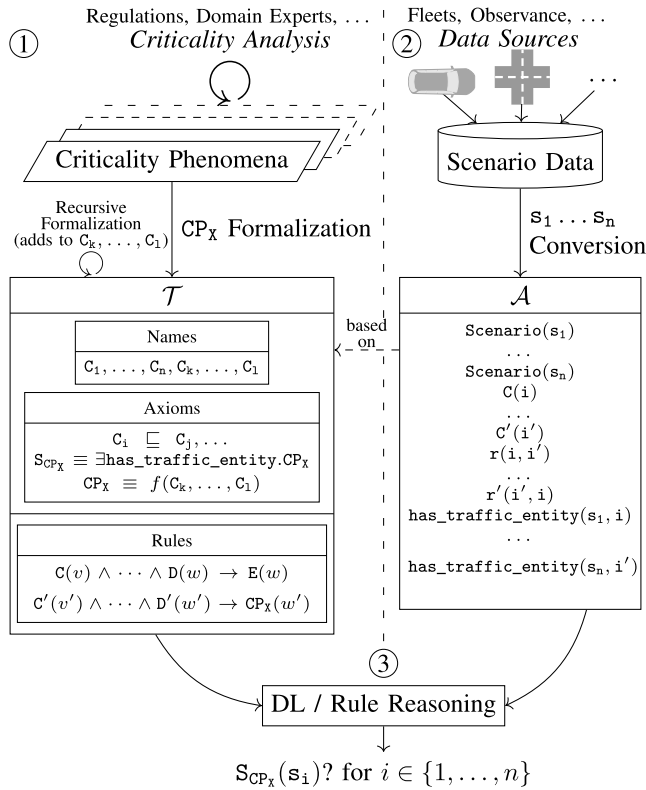


FIGURE 4. The general method of formalizing a natural language phenomenon and reasoning about its presence in data.

basic concepts are not observable in data (for example, if the recording modality does not allow to recognize certain features such as whether turning indicators are enabled), an analysis of the phenomena relying on those concepts can not be performed.

Hinging on the assumption of convergence of the criticality analysis, this process leads to an iterative completion of the ontology towards an exhaustive representation of all concepts that are deemed essential for identifying safety-relevant factors. This step is elaborated on in detail in the upcoming Section V – concerning the basic ontological framework – and Section VI – concerning the formalization of criticality phenomena within this ontological framework.

Step ② unifies the various scenario data sources by conversion into a \mathcal{T} -compliant A-Box \mathcal{A} . This creates scenarios s_1 to s_n with traffic entities i to i' that are represented using the concepts C, C', \dots and roles r, r', \dots of \mathcal{T} . A realization of this step is given in Section VII-A and Section VII-B.

Once the manifold data sources are convertible from their native data schemes, new facts can be inferred by performing reasoning on the ontology $\mathcal{O} = (\mathcal{T}, \mathcal{A})$, as indicated by ③. Such a reasoning pipeline has been prototypically implemented for the work at hand whilst taking into account various practical concerns, e.g., temporal reasoning performance and concrete domain abstraction. This finally answers competency question A1, i.e., for which

$i \in \{1, \dots, n\}$ $S_{CP_X}(s_i)$ holds, and, per definition, also A2 to A3, cf. Section VII-C.

V. THE AUTOMOTIVE URBAN TRAFFIC ONTOLOGY

Starting with step ① of the sketched method of Figure ??, this section presents the employed T-Box. It describes the conceptual foundations, general structure, idea, modeling, and implementation of an integrated set of suitable ontologies.

A. THE 6-LAYER MODEL AS A CONCEPTUAL FOUNDATION

As a tool for structured environment description, the well-known Layer Model can be utilized to create a formal model of the context. It was originally defined by Schuldt [34] and has been continuously refined for specific use cases, e.g., by Bagschik *et al.* for the instantiation of highway scenes [25]. In a recent work by Scholtes *et al.*, their ideas are examined in-depth and adapted for urban environments while keeping compatibility to the fundamental ideas behind the Layer Model [7]. Since Scholtes *et al.* give detailed definitions of the layers as well as justifications, guidance, and examples, it serves as a suitable basis for our use case – namely, the construction of a formal ontology implementing the Layer Model.

The work categorizes the environment, its entities, and their properties in six individual layers: Road Network and Traffic Guidance Objects (L1), Roadside Structures (L2), Temporary Modifications of the former (L3), Dynamic Objects (L4), Environmental Conditions (L5), and Digital Information (L6), encompassing the so-called 6-Layer Model (6LM). It thereby provides a generally usable, unbiased, and objective environment description, i.e., it functions as a clearly structured basis for scenario descriptions and their ontologies. While the 6LM provides a unique top-level structuring for the environment, an ontology details and formalizes the different entities, properties, and relations in a more comprehensive way.

B. ARCHITECTURE

In order to address the issue of integrating the 6LM into a formal ontology as necessitated by the process of Figure 4, we present the Automotive Urban Traffic Ontology (A.U.T.O.). In its core, it is a set of integrated ontologies with the goal to enable reasoning about criticality in urban traffic scenarios. Due to this scope, the 6LM as defined by Scholtes *et al.* for urban environments, is used as an architectural guiding pattern.

Architecturally, we separate each layer into a *core* and an *implementation*. Cores represent an ontological interface which other layers and applications of the ontology can access without being concerned with detailed domain knowledge, which in turn is given in a set of implementations. For example, the concept Lane is part of the Layer 1 core, but concrete specializations of lanes can be given in an implementation.

TABLE 2. Detached, related ontologies of the 6LM used within A.U.T.O.

Ontology	Competency	Example Concepts	Example Roles
Time	Temporal concepts	Intervals, Time Points	After, Before
Geometry	Geometries & shapes	Geometry, Rectangle	Contain, Disjoint
Physics	Physical properties	Dynamical object	Speed, Acceleration
Act	Activities over time	Actor, Event, Activity	Conduct, Participate
Communication	Message exchange	Sender, Receiver	Transmit, Receive
Perception	Perceiving Entities	Observer, Occlusion	Is Occluded For

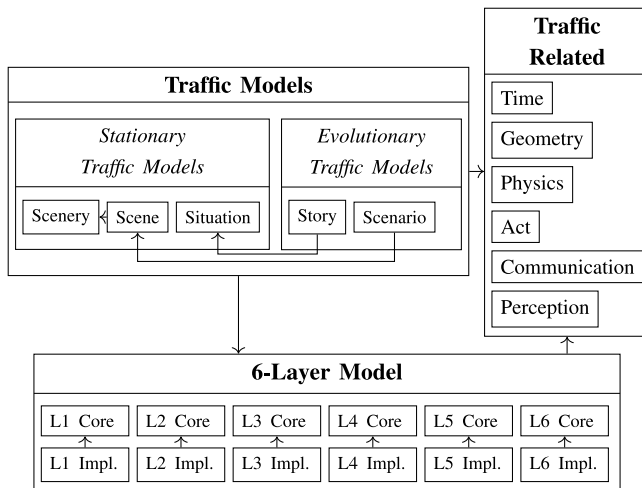


FIGURE 5. The modular architecture of the Automotive Urban Traffic Ontology.

Moreover, this work extends the 6LM by detaching conceptually separated parts which semantically span over multiple layers or are completely independent of the traffic domain. This modularity enables separation of concerns and re-use of concepts, e.g., for spatial properties (heights, distances, etc.).

Note that often, so-called Upper Level Ontologies are employed for the purpose of unification. We deliberately omit the use of such ontologies as we were not able to identify an ontology that satisfied the needs imposed by the competency questions. Namely, this concerns issues of suitable formalization to enable inferences by DL reasoners, documentation, implementation, comprehensibility, and modularity. This problem is currently addressed in the standardization project ASAM OpenXOntology [15], where an Upper Level Ontology is laboriously integrated into a traffic domain ontology.

We hence opt for a lean approach, where single, re-usable domain ontologies are imported, as displayed in Table 2.

On top of the related ontologies, we introduce a framework in which traffic entities of the 6LM are integrated into traffic models such as scenes and scenarios. Collectively, this conglomerate of ontologies yields the A.U.T.O. shown in Figure 5.

The diagram indicates the direction of the dependencies of the modules through arrows. Therefore, both the traffic models as well as the 6LM rely on the set of related ontologies. The former uses, e.g., temporal concepts to specify

scenarios, the latter uses, e.g., physical properties for traffic participants.

C. IMPLEMENTATION

We provide an open-source implementation¹ of the previously introduced ontology in OWL2 (i.e., the DL *SROIQ^(D)*).

The provided A.U.T.O. consists of 26 OWL2 files implementing the modular architecture of Figure 5. Each file represents one ontology module in which its respective domain concepts are defined. For example, the Physics ontology defines concepts such as *Spatial_Object*, *Dynamical_Object*, and *Non_Moving_Dynamical_Object* alongside their natural language definition. Here, a *Spatial_Object* is defined as ‘any physical object with a geometrical representation, e.g., a vehicle’. If expressible in DLs, the OWL2 files also specifies logical axioms implementing these natural language definitions. For example, the Physics ontology states $\text{Dynamical_Object} \sqsubseteq \text{Spatial_Object}$ and $\text{Non_Moving_Dynamical_Object} \equiv \text{Dynamical_Object} \sqcap \exists \text{has_velocity}.\{0.0\}$. Those logical T-Box axioms then allow the reasoner, in turn, to infer entailed statements on the A-Boxes for the subsequent evaluation – such as, whether a given vehicle is a *Non_Moving_Dynamical_Object*. For a visual excerpt of the T-Box of A.U.T.O., including the subsumption relation and other logical axioms, we reference to Figure 6.

We also rely on well-established ontologies, enabling compatibility to a wide range of tooling such as GraphDB:

- geometrical concepts are represented using OGC’s *GeoSPARQL* ontology,² which gives access to DE-9IM and RCC8 vocabularies [35], and
- temporal concepts are represented using W3C’s *OWL Time* ontology,³ which itself relies on an adapted version of Allen’s 13 temporal relations [19].

A Note on Temporal Identity: Some ontological approaches assume a perdurantistic dogma, i.e., things are inherently four dimensional entities in space-time. A.U.T.O. takes an endurantistic view: it specifies individuals in three dimensions and tracks temporal identity by the role *identical_to*.

VI. ONTOLOGY-BASED FORMALIZATION OF CRITICALITY PHENOMENA

In general, criticality phenomena are not fixed to the combined DL-rule approach proposed in this work. Various approaches exhibit differences in ease and specificity of the modeling process, computational efficiency of decidability problems, and availability of expertise. However, a common feature is their formalization being based on an implicit or explicit ontology. This section depicts the formalization

1. <http://github.com/lu-w/auto>
 2. <https://www.ogc.org/standards/geosparql>
 3. <https://www.w3.org/TR/owl-time>

process for our DL-rule approach as well as how the formalization can be leveraged for the construction of such an ontology.

A. THE FORMALIZATION PROCESS

Recall that the general method of Section IV-C started the formalization process (step ①) where a criticality phenomenon is formalized. We now explain this process in more detail, which is based on the previously introduced framework of A.U.T.O. and a catalog⁴ of criticality phenomena defined in natural language that was assembled during prior work [3]. We illustrate the formalization of the exemplary criticality phenomenon ‘*Bicyclist Illegitimately Riding Over Pedestrian Crossing*’ (CP 69) as a special case of *Misconduct* (CP 143).

The first step involves the collection and assembly of knowledge regarding the semantics of the criticality phenomenon, which relies on standards, laws, and domain experts. For our example, we deem as necessary the semantics of the relevant regulatory aspects, bicyclists, pedestrian crossings, and riding over. During the process, the stakeholders agree on

- 1) the relevant regulatory aspects being given by the German traffic regulations (StVO) stating that a bicyclist can not exercise the right of way indicated by a pedestrian crossing if the bicyclist is mounted,
- 2) a bicyclist as a traffic participant on a bicycle, and
- 3) a pedestrian crossing as a an area that is officially marked with corresponding white stripes.
- 4) riding over as the movement of an actor across an area

A suggestion of an (aspectual) semantics is then given by: “A bicyclist that in one scene is not on a drivable lane, and in a directly succeeding scene on a pedestrian crossing where the bicyclist is exercising a right of way is using this pedestrian crossing illegitimately.”

As temporal identities of traffic entities need to be tracked through different scenes, we express this using a rule-based approach such that individuals can be bound to variables:

$$\begin{aligned}
 & \text{Scene}(s_1) \wedge \text{Scene}(s_2) \wedge \text{has_traffic_entity}(s_1, b_1) \\
 & \wedge \text{has_traffic_entity}(s_2, b_2) \wedge \text{identical_to}(b_2, b_1) \\
 & \wedge \underline{\text{Bicyclist}}(b_1) \wedge \underline{\text{Bicyclist}}(b_2) \\
 & \wedge \text{directly_after}(s_2, s_1) \\
 & \wedge \underline{\text{Non_Driveable_Lane}}(w) \\
 & \wedge \text{has_traffic_entity}(s_1, w) \\
 & \wedge \text{intersects}(b_1, w) \wedge \underline{\text{Pedestrian_Crossing}}(c) \\
 & \wedge \text{has_traffic_entity}(s_2, c) \wedge \text{intersects}(b_2, c) \\
 & \wedge \text{Driveable_Lane}(l) \wedge \text{in_scene}(l, s_2) \wedge \text{intersects}(c, l) \\
 & \wedge \text{Driver}(d) \wedge \text{drives}(d, v) \wedge \text{in_scene}(d, s_2) \\
 & \wedge \text{intersects}(d, l) \wedge \text{has_intersecting_path}(v, b_2) \\
 & \rightarrow \text{CP}_{69}(b_2)
 \end{aligned}$$

4. https://github.com/lu-w/auto/blob/main/criticality_phenomena.owl

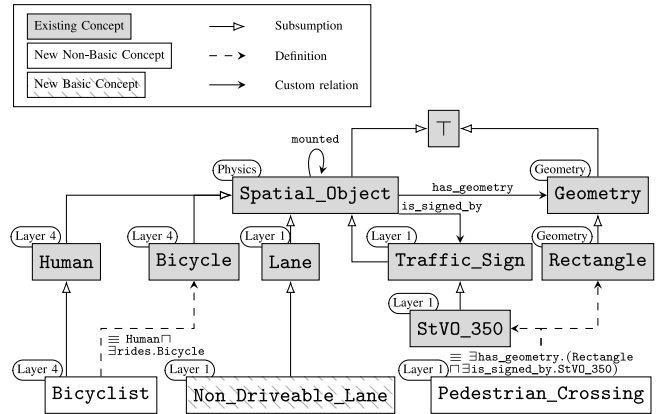


FIGURE 6. Derived concepts for the criticality phenomenon ‘Bicyclist Illegitimately Riding Over Pedestrian Crossing’ including their subsumption into relevant parts of the existing T-Box, as implemented in A.U.T.O.

Intuitively, this rule states that if a bicyclist is driving on a non-drivable lane (e.g., sidewalk) in one scene and on a pedestrian crossing in a subsequent scene in which also a traffic participant on an intersecting lane with an intersecting path, then CP_{69} holds for that bicyclist. More specifically, the identity of the bicyclist in the first scene (s_1) is referred to as b_1 , respectively as b_2 for the second scene (s_2). The scenes have b_1 and b_2 as their respective traffic entity (hence, $\text{has_traffic_entity}$) and are directly after each other (hence, directly_after). Furthermore, s_1 has a non-drivable lane w as its traffic entity which the bicyclist b_1 intersects. Similarly, the second scene s_2 has a pedestrian crossing c and driveable lane l that intersects c . Furthermore, s_2 has a driver d within s_2 on lane l that drives a vehicle v which in turn has an intersecting path with the bicyclist b_2 .

The formalization relies on a set of concepts from A.U.T.O. that are assumed to be already existent as well as suitably defined and axiomatized in the ontology, e.g., for temporal and spatial concepts. On the other hand, Bicyclist, Pedestrian_Crossing, and Non_Driveable_Lane (underlined) are, for the scope of the example, not present at the time of formalization and therefore need to be added to \mathcal{T} . The following integration of concepts and their subsumption is also visualized in Figure 6.

- $\text{Bicyclist} \equiv \text{Human} \sqcap \exists \text{rides.Bicycle}$
 - Human, Bicycle are basic concepts
 - rides \equiv mounted
 - * mounted is a basic concept (something physically and stably touches another spatial object)
- $\text{Pedestrian_Crossing} \equiv \exists \text{has_geometry. (Rectangle} \sqcap \exists \text{is_signed_by.StVO_350)}$
 - has_geometry (geometries of spatial objects), Rectangle (a two-dimensional shape), is_signed_by (which traffic signs apply to an individual), and StVO_350 (marker for pedestrian crossing) are basic concepts.

- `Non_Driveable_Lane`, a basic concept (a lane that is not intended to be driven on by any vehicle, e.g., a walkway)

Requiring that the necessary basic concepts can either be directly observed in the analyzed data or at least be derived therefrom, e.g., by means of machine learning algorithms such as image segmentation for `Non_Driveable_Lane`, we now have the ability to (aspectually) infer the presence of CP_{69} .

B. APPROACHING FORMALIZABILITY

Concepts and phenomena may be inherently unformalizable due to their vagueness (e.g., subjectivity and case-by-case decisions). To mitigate this problem, CP_X can be specified by:

- 1) An *exact representation*: if possible, we give an exact representation using an equivalence, i.e., $CP_X \equiv f(C_k, \dots, C_1)$ where f is a logical concept expression.
- 2) An *under approximation*: if an exact representation is not possible, we use an approximation. In DL, an under approximation can be specified as a GCI of the form $CP_X \sqsubseteq f(C_k, \dots, C_1)$. On top of DL, sufficient conditions for CP_X can be stated as a rule, i.e., $\bigwedge_{r_a \in R_A} r_a \rightarrow CP_X(v)$.
- 3) An *over approximation*: an over approximation is specified by a GCI $CP_X \sqsupseteq f(C_k, \dots, C_1)$. Combined with an under approximation, it allows for a two-sided bound.

For example, ‘Occluded Pedestrian’ (CP 157) can be exactly formalized assuming that `Pedestrian`, `Is_Occlusion`, and `is_occluded` are exactly formalized or basic concepts: $CP_{157} \equiv Is_Occlusion \sqcap \exists is_occluded.Pedestrian$. The phenomenon ‘Misconduct’ (CP 143) on the other hand has a broad scope. We hence model certain aspects of it, such as driving without vehicle lights turned on in night scenarios: $CP_{143} \sqsubseteq (Off_Light_Vehicle \sqcap \exists in_traffic_model.Night_Scenario) \sqcup CP_{69}$. Similarly, ‘Vulnerable Road User with Road Access’ (CP 114), contains the vague concept Road Access, which was over approximated in terms of `is_near`. It is formalized as $CP_{114} \sqsubseteq Vulnerable_Road_User \sqcap \exists is_near.Driveable_Lane$. Such approximations can be iteratively refined.

Besides those three formalizations, our prototype contains the definitions of in total 16 criticality phenomena, depicted along their primary classification within A.U.T.O. in Table 3. They were selected from a catalog of roughly 300 criticality phenomena described in natural language which originated during an initial iteration of the criticality analysis. Their formalization is detailedly presented in the Appendix.

VII. IMPLEMENTING THE CRITICALITY RECOGNITION

We present a Python implementation⁵ of the process of Section IV that is based on the ontology introduced in

5. <https://github.com/lu-w/criticality-recognition>

Section V. While the realization of step ① has already been unveiled, we now focus on the data and reasoning steps, i.e., ② and ③.

A. THE OMEGA-FORMAT

As motivated in Section I, the use of various data sources is of high value when analyzing the criticality within traffic scenarios. Methodically, our approach rests upon a unified data format that allows to represent traffic scenarios from different sources, including drones, measurement vehicles, and stationary cameras. This format therefore provides input for A.U.T.O. A-Boxes. For this, the OMEGA-format was chosen [36]. The data format is HDF5-based and designed to store reference data of urban automotive settings in an object-list structure together with map and weather information.

B. ONTOLOGY MAPPING

In order for Automotive Urban Traffic Ontology (A.U.T.O.) to harness the information provided by OMEGA-compliant data sets, we need to map the OMEGA-format onto the concepts provided by A.U.T.O., i.e., a realization of step ②. As the development of the OMEGA-format was performed in conjunction with the development of A.U.T.O., the format is naturally mappable to the concepts of the ontology.

For example, we translate each object of the OMEGA-type `reflector_guidingLamps` into a corresponding axiom within the ABox. The following specification of the OMEGA-format defines `reflector_guiding_lamps` as

“6.1 Type: `reflector_guidingLamps`: Any reflectors or lamps along the lane (reflectors and guiding lamps are assumed to be regularly placed, their exact position is not known.)”.

We create a new individual i as well as a corresponding boundary geometry i_{geom} and map the definition to i by:

$(System \sqcap \forall consists_of.(Reflector_Guiding_Lamp \sqcap \exists within.\{i_{geom}\}))(i)$.

C. AUTOMATED RECOGNITION OF CRITICALITY PHENOMENA

After creating an A-Box \mathcal{A} based on \mathcal{T} containing a set of n scenarios, the naïve option would be to run the reasoner directly on $\mathcal{O} = (\mathcal{T}, \mathcal{A})$. With this, two problems arise:

- 1) deriving abstract facts from concrete domains, such as performing complex trigonometrical calculations to check whether one geometry occludes another, and
- 2) exploding state spaces for large A-Boxes, especially when analyzing real world scenarios.

We resolve both issues within our implementation.

1) DERIVING ABSTRACT FACTS FROM CONCRETE DOMAINS

Here, we propose an *iterative augmentation approach* to lift A-Boxes from concrete to abstract domains. We provide a library where the user can easily specify definitions for

TABLE 3. Exactly (\equiv) or over (\supseteq) resp. under (\supset) approximately formalized criticality phenomena for the prototypical evaluation.

(L1) Road Network	(L2) Roadside Structures	(L4) Dynamic Objects	(L5) Environmental Conditions	Physics	Perception	Policy
Pedestrian Crossing / Ford \equiv	Building for Unpredictable Road Users near Road \supseteq	Pedestrian on Roadway \equiv Vulnerable Road User with Road Access \supseteq Passing of Parking Vehicle \equiv Intersecting Planned Paths \supset	Heavy Rain \equiv Freezing Temperatures \equiv	Small Distance \supset Strong Braking Maneuver \equiv High Relative Speed \equiv	Occlusion \equiv Occluded Pedestrian \equiv Occluded Traffic Infrastructure \equiv	Illegitimate Use of Pedestrian Crossing \supset Misconduct \supset

Algorithm 1 Inference by Lifting Concrete to Abstract Domains

Input: an ontology \mathcal{O} **Output:** \mathcal{O} including all inferred facts

```

1: procedure DO_INFERENCE
2:    $a = 1$ 
3:   while  $a > 0$  do
4:      $a = \text{DO\_AUGMENTATION}(\mathcal{O})$ 
5:      $\text{DO\_REASONING}(\mathcal{O})$ 
6:   end while
7:   return  $\mathcal{O}$ 
8: end procedure

```

arbitrary concepts C and (reified) roles r using Python code.⁶ This enables us to derive whether those concepts or roles are present by means of classical imperative programming. The library provides a high level interface that performs such an augmentation for each individual within \mathcal{A} of an ontology \mathcal{O} , denoted by $\text{DO_AUGMENTATION}(\mathcal{O})$, changing \mathcal{O} in-place and returning the number of performed augmentations.

Finally, the lifting of domains needs to be integrated with the reasoner, whose interface is denoted by $\text{DO_REASONING}(\mathcal{O})$. This integration is sketched by Algorithm 1.

This simple yet correct procedure yields an iterative detection of all possible inferences on \mathcal{O} . Note that efficiency can be improved by directly integrating the augmentation into the reasoning engine, which was left for future work.

2) EXPLODING STATE SPACES

The second issue concerns the sheer size of real world scenarios. As an example, a complex urban traffic scene may contain 80 individuals on Layer 4 and 200 individuals on Layers 1 and 2. Even at a sampling rate of 3 Hertz, this amounts to over $2.5 \cdot 10^5$ individuals in 30 seconds and a combinatorial size of $6.25 \cdot 10^{10}$ and $1.5625 \cdot 10^{16}$ for potential binary and ternary relations.

Upon closer inspection, many of those relations may not be logically possible. For example, spatial relations between a lane in scene one and a vehicle in scene two are irrelevant. Therefore, we safely use successive examination of the single scenes to correctly infer all non-temporal facts.

Algorithm 2 Temporal Inference

Input: an ontology $\mathcal{O} = (\mathcal{T}, \mathcal{A})$ with exactly one scenario s and scenes $s_1 \dots s_n$ in \mathcal{A} **Output:** \mathcal{O} including inferred facts for s

```

1: procedure DO_TEMPORAL_REASONING
2:   for  $i \in \{1, \dots, n\}$  do
3:      $\mathcal{A}_{s_i} = \text{SCENE}(\mathcal{A}, i)$ 
4:      $\text{DO\_INFERENCE}((\mathcal{T}, \mathcal{A}_{s_i}))$ 
5:   end for
6:    $\mathcal{A}_{merged} = \text{MERGE}(\mathcal{A}_{s_1}, \dots, \mathcal{A}_{s_n})$ 
7:    $C = \text{TEMPORAL\_CONCEPTS}(\mathcal{T})$ 
8:    $\mathcal{A}_{sliced} = \text{SLICE}(\mathcal{A}_{merged}, C)$ 
9:    $\text{DO\_INFERENCE}((\mathcal{T}, \mathcal{A}_{sliced}))$ 
10:   $\mathcal{A}_{res} = \text{MERGE}(\mathcal{A}_{merged}, \mathcal{A}_{sliced})$ 
11:  return  $(\mathcal{T}, \mathcal{A}_{res})$ 
12: end procedure

```

This still leaves open the question of handling temporality. For example, the previously introduced rule for criticality phenomenon CP_{69} argues over two directly succeeding scenes, which can not be recognized on a scene level. However, running the reasoner directly on the scenario-level A-Box leads to the discussed scalability issues. Analogously to program slicing [37], we suggest a temporal slicing approach.

Here, entities not relevant for temporal inference are sliced from the A-Box and re-added after finishing the temporal reasoning on the whole scenario, as shown in Algorithm 2. The algorithm relies on the following operators:

- $\text{SCENE}(\mathcal{A}, i)$, which reduces \mathcal{A} to an A-Box \mathcal{A}_{s_i} with only individuals of scene s_i , i.e., $\mathcal{A}_{s_i} := \{\alpha \in \mathcal{A} \mid \exists n \in \text{ind}(\alpha) : (\exists \text{has_traffic_entity}^{-1}. \{s_i\})(n)\}$.
- $\text{MERGE}(\mathcal{A}_1, \dots, \mathcal{A}_n)$, whose result is roughly equal to $\bigcup_{i \in \{1, \dots, n\}} \mathcal{A}_i$, but special temporally constant individuals (e.g., lanes) are not merged as to avoid duplicates. Temporal identity is added by the $\text{MERGE}(\cdot)$ algorithm using the previously introduced role identical_to .
- $\text{TEMPORAL_CONCEPTS}(\mathcal{T})$, which identifies a set $\mathbf{N}_{\mathcal{T}}$ of concepts and roles that are (possibly indirectly) used in an axiom involving a concept from a set of temporal concepts T , e.g., $T = \{\text{Interval}, \text{Point}\}$. Formally, $\mathbf{N}_{\mathcal{T}} = \{C \in \mathbf{N}_C \cup \mathbf{N}_R \mid D \in T, k \geq 0, \exists \alpha_0 \in \text{axs}(C), \exists \alpha_2, \dots, \alpha_{k-1} \in \mathcal{T}, \alpha_k \in \text{axs}(D) : \bigwedge_{i=0}^{k-1} (\text{nms}(\alpha_i) \cap \text{nms}(\alpha_{i+1}) \neq \emptyset)\}$. In the case that \mathcal{O} contains rules, $\mathbf{N}_{\mathcal{T}}$ can be defined analogously.

6. <https://github.com/lu-w/owlready2-augmentator>

- $\text{SLICE}(\mathcal{A}, \mathcal{T})$, which creates an A-Box $\mathcal{A}_{\text{sliced}}$ with only individuals that relate to concepts in \mathcal{T} , i.e., $\mathcal{A}_{\text{sliced}} := \{\alpha \in \mathcal{A} \mid \exists i \in \text{ind}(\alpha) : \exists C \in \mathcal{T} : C(i)\}$.

Intuitively, the algorithm takes as input an A-Box \mathcal{A} with one scenario consisting of n scenes. It creates an A-Box \mathcal{A}_{res} where all temporal facts—i.e., those on a scenario-level—are inferred. This concerns, for example, the rule for CP_{69} , which can only be applied when multiple scenes are present in the analyzed A-Box. Firstly, it infers all entailed facts on each scene using Algorithm 1. Afterwards, it merges all scenes into a single A-Box $\mathcal{A}_{\text{merged}}$. Based on the formalized criticality phenomena in \mathcal{T} , it also derives a set of temporal concepts C which are necessary to infer facts about the presence of temporal criticality phenomena. Those are then used to create a sliced A-Box $\mathcal{A}_{\text{sliced}}$ in which only individuals of concepts in C remain. This sliced A-Box is then considerably smaller than the original A-Box and therefore suitable for the inference sketched in Algorithm 1. Finally, all individuals that were previously sliced away are re-added to the A-Box \mathcal{A}_{res} .

Since criticality phenomena have been modeled as axioms within \mathcal{T} , Algorithm 2 eventually yields an inferred A-Box \mathcal{A} in which the presence of criticality phenomena is documented.

VIII. EVALUATION AND RESULTS

We evaluate the implementation on the inD data set [38], showing how the reasoner can answer the competency questions A1 to A3 based on the exemplary set of formalized phenomena and our implementation of Algorithm 1 and Algorithm 2. Albeit the main contribution of this work being the development of the method and tooling, we present a first evaluation for a semantic data analysis use case. Recall that we have previously argued that a verification and validation process relies on various recording modalities, including fleet monitoring, test vehicles, and drone data. As to exemplarily examine one of these relevant data sources, we have selected a large-scale drone data set which can act as a precursor to a rigorous investigation of a variety of data sources.

A. THE IND DATA SET

The inD data set consists of naturalistic trajectories of more than 11 500 road users, including vehicles, bicyclists, and pedestrians, at four urban intersections in Aachen, Germany. The utilization of drones in the creation of the data set retains naturalistic driving behavior by being inconspicuous and aids the accuracy of positioning and object dimensions. Its distinctive perspective allows to draw conclusions about the behavior of traffic participants in certain situations—e.g., bicyclists at pedestrian crossings – which in turn can be constructively used as assumptions for designing, implementing, and safeguarding an ADS. Moreover, and in contrast to an in-vehicle perspective, we are able to assume a holistic view on the traffic scenario, i.e., tracking of ‘behavior chains’ where the decisions of traffic participants influences each other through transitive propagation. As an example of such

a chain, consider an occluded pedestrian which leads to a strong braking maneuver of an intersecting vehicle that in turn leads to a potential rear-end collision of a following vehicle.

B. REASONING RESULTS

In order to demonstrate the practicality and utility of our approach, we evaluate of the method on inD in three steps:

- Firstly, we present an intuitive example depicting the inference results for a single, representative traffic conflict.
- Secondly, analyze four scenarios on each intersection.
- Lastly, we compare the inferences of our tooling against two criticality metrics as determined by prior work [39].

1) AN EXEMPLARY TRAFFIC CONFLICT

We identified a simple yet representative traffic conflict within the inD data set.⁷ Here, a passenger car approaches the intersection at the Frankenberger Park in Aachen, Germany, from the east while a pedestrian is occluded behind parking vehicles. The passenger car is forced to a substantial braking maneuver by the appearance of the pedestrian. The criticality phenomena within this conflict were inferred automatically by our tooling and are represented visually in Figure 7.

The inferences show that the pedestrian and vehicle are indeed bilaterally occluded by parking vehicles while having a high relative speed during the first seconds of the scenario. Furthermore, the driver passes these parking vehicles while the pedestrian has access to the road. This leads to the occurrence of the more concrete criticality phenomenon Small Distance. Note that, albeit a harsh braking, the criticality phenomenon Strong Braking Maneuver was not detected, as the vehicle has been decelerated just below the threshold of 4.61 m/s^2 . This showcases an inherent limitation of the binary identification approach, where values just below thresholds are deemed irrelevant albeit possibly still influencing the overall criticality.

As an illustration of the aforementioned holistic view on traffic scenarios, Figure 8 shows a bar chart of the amount of criticality phenomena per scene for the exemplary scenario.

It is evident that occluded traffic infrastructures – such as lane markings – make up a large amount of the overall criticality phenomena. Furthermore, we find a considerable number of small distances and intersecting planned paths, as is to be expected in urban intersection scenarios. Overall, the amount of inferred criticality phenomena per scene is quite high, raising the issue of the specificity of the formalizations.

2) EVALUATION OF MULTIPLE SCENARIOS

We present an evaluation of in total 16 scenarios – four scenarios for each of the four inD intersections – with a mean duration of 11.27 ± 3.68 seconds covering a total span of 180.28 seconds with 361 unique traffic participants. The scenarios were converted into A.U.T.O. A-Boxes at a

7. Namely, scenario number 94 in recording number 18.

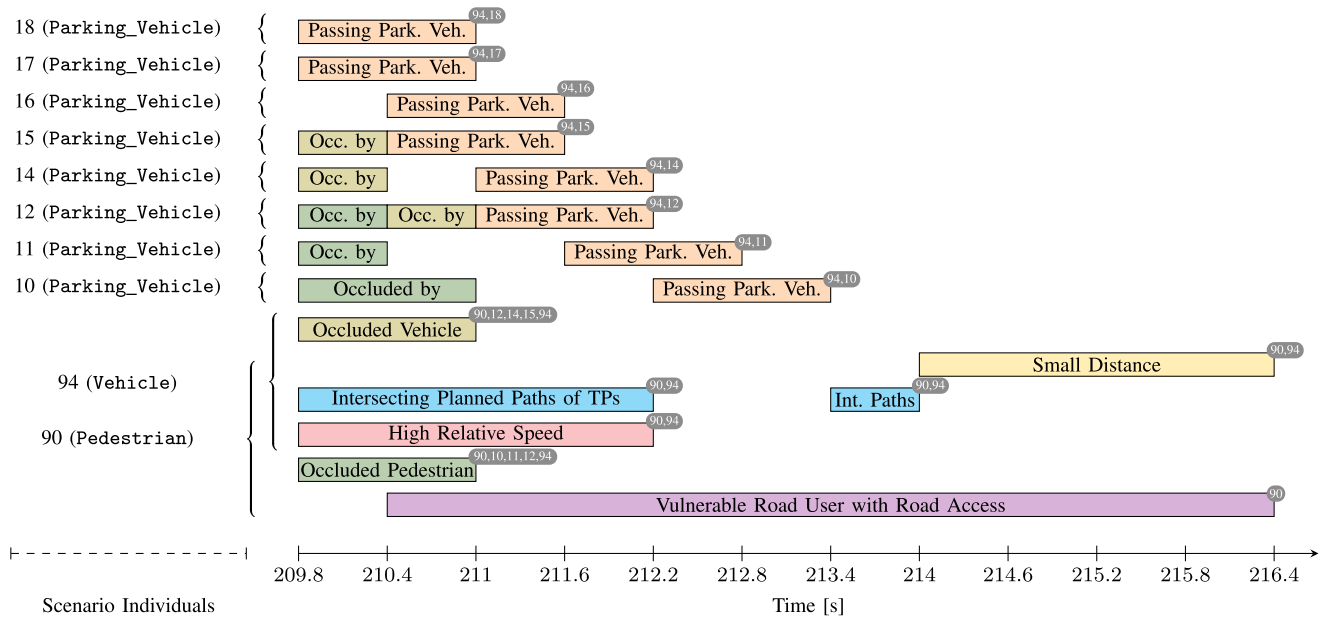


FIGURE 7. The inferred criticality phenomena over time of a 6.6 second scenario with one moving vehicle, one pedestrian, and eight parking vehicles. The subject (and objects) are annotated by gray boxes on each criticality phenomenon. Occ. by = Occluded by and Passing Park. Veh. = Passing of Parking Vehicles.

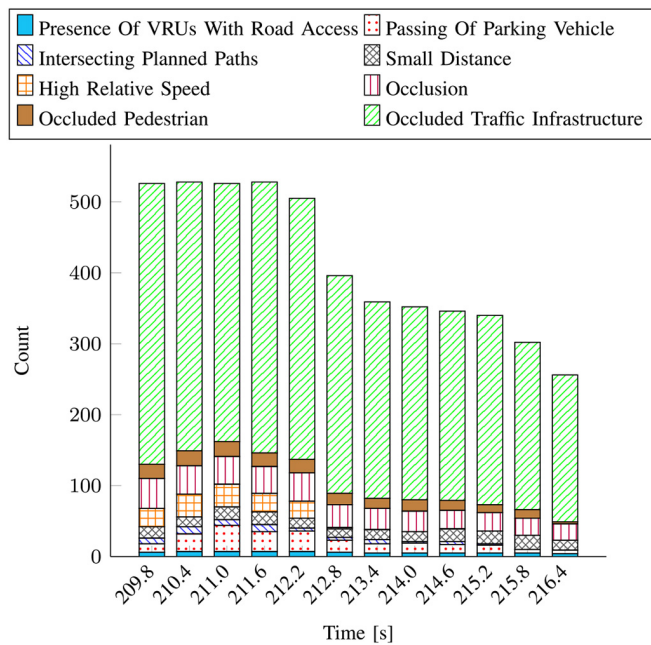


FIGURE 8. Identified criticality phenomena within the exemplary scenario.

rate of 1 Hertz, leading to 183 scenes. Due to the large scale of the data set, we reduced the evaluation to scene-level criticality phenomena, i.e., performing only Algorithm 1. The overall number of inferred criticality phenomena amounts to 114922, resp. 628 per scene. If this number is again taken relative to the involved traffic participants, we find that there exist roughly 1.74 criticality phenomena per scene per traffic participant.

A detailed break-down of the inferred criticality phenomena per scene per traffic participant into the single

TABLE 4. Inferred criticality phenomena (CP) relative to the number of traffic participants (TP) and scenes over the selected 16 scenarios, arranged by intersection (Int.).

CP per TP per Scene	Int. 1	Int. 2	Int. 3	Int. 4
High Relative Speed	0.380	0.438	0.584	1.99
Intersecting Planned Paths	0.058	0.066	0.050	0.047
Occluded Pedestrian	0.811	1.332	0.377	0.291
Occluded Traffic Infrastructure	21.781	13.127	47.658	32.076
Occlusion	1.629	1.628	1.076	0.978
Presence of VRUs With Road Access	0.257	0.180	0.094	0.191
Small Distance	0.278	0.513	0.338	0.271
Strong Braking Maneuver	0.003	0	0	0

phenomena classes for each of the intersections is presented in Table 4.

For example, on intersection two, there exist 1.332 occluded pedestrians per traffic participant in each scene. The value is considerably lower for the other intersections, indicating a higher pedestrian occurrence or occlusion potential.

To further show the capabilities of a semantic analysis, we demonstrate how a naïve analysis of the temporal relation of two criticality phenomena can be performed – namely Intersecting Planned Paths and Occlusion – on passenger car-pedestrian constellations. We first formulate two hypotheses:

H_1 occlusions and intersecting paths are, independent of their temporal constellation, associated between passenger cars and pedestrians, and

H_2 in the cases where both occur, it is more likely to observe intersecting paths after occlusions than before.

We test H_1 by means of a simple contingency table as depicted in Table 5. This table displays the amount of cases of all four combinations of the occurrence of

TABLE 5. Amount of the criticality phenomena for all pairwise combinations of passenger cars and pedestrians.

	Occlusion	No Occlusion
Intersecting Paths	7	1
No Intersecting Paths	846	532

the two criticality phenomena between passenger cars and pedestrians.

From this table, we compute the phi coefficient as

$$\Phi = \frac{n_{11}n_{00} - n_{10}n_{01}}{\sqrt{(n_{10} + n_{11}) \cdot (n_{00} + n_{01}) \cdot (n_{00} + n_{01}) \cdot (n_{01} + n_{11})}},$$

where n_{ij} indicates the number n in the i -th row and j -th column of the contingency table. For the analyzed criticality phenomena, this yields $\Phi = 0.04$. With the phi coefficient taking values from $[-1, 1]$ in the dichotomous case, we find a minor yet positive correlation between the variables.

As to test H_2 , we first perform a simple analysis of the differences of the starting times of the intervals in which the criticality phenomena occur, taking into account all combinations of traffic participants. Since the association has been examined by Φ , we can now focus on only the cases in which both small distances and occlusions occur. Our results indicate that, on average, small distances start 0.87 seconds after occlusions, with a standard deviation of 5.83 seconds.

3) COMPARISON AGAINST CRITICALITY METRICS

Besides comparing the relations between criticality phenomena themselves, one can also identify their relation to measured criticality as computed by criticality metrics. Here, our hypothesis states that an increase of a criticality metric is often preceded by a set of relevant criticality phenomena. This methodical approach obviously rests upon the assumption of a specific and sound formalization of criticality phenomena as well as upon employing valid criticality metrics.

Prior work has already examined various criticality metrics for 22 traffic scenarios of inD for left-turn and passing maneuvers of vehicle pairs [39]. We restrict our exemplary comparison to the analysis of one scenario and two metrics, namely scenario number 449 in recording number 28 and the Time To Collision (TTC) and Required Acceleration (a_{req}). Similar to Figure 7, the plot of this scenario is presented in Figure 9 but now includes the metrics' values over time.

In the example, a minimum TTC of 0.6 s and a_{req} of -11.8 m/s^2 is given, with the metrics starting a linear approach of their minimum values at second 1044.7. After second 1046.7, the passing maneuver is finished. Our tooling observes that before this critical encounter, an occlusion took place between both traffic participants. Furthermore, we also identified high relative speeds during the passing phase of both vehicles.

A note on risk quantification: Within a rigorous safety case, this relation needs to be examined statistically on populations of representative scenarios, i.e., how strong is the

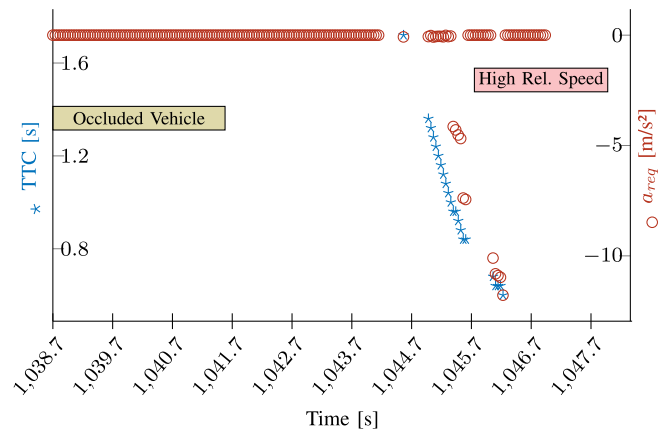


FIGURE 9. The computed criticality metrics and inferred criticality phenomena for the exemplary scenario between both traffic participants over time.

association between certain phenomena and measured criticality across the OD. Depending on this strength, certain phenomena may be treated differently during the argumentation and evidence generation. For example, an argument as to why the ADS sufficiently mitigates the risk induced by occluded pedestrians may be elaborated on more extensively than those in which a lane marking is occluded – depending on the associational findings and subsequent causal analyses. After this initial examination of the OD, safety standards require the development of a system design as well as a subsequent hazard analysis and assessment of risks. This assessment can be based upon prior knowledge that originated from the analysis of criticality metrics and phenomena. Since criticality phenomena represent factors that impact the safety in traffic in general, it is likely that such phenomena will also be relevant for the hazard analysis of the given system. The quantification provided by the aforementioned associational analysis of criticality can be used as an initial estimate for the risk assessment, although the specific system design may influence the actual risk. For example, an ADS could be designed in a way that effectively mitigates risks induced by occluded pedestrians. Nevertheless, an argument as to why this specific criticality phenomenon is handled sufficiently well by the system is mandatory.

IX. CONCLUSION AND FUTURE WORK

We presented a first step towards a structured method to obtain a formalized definition of an OD, incorporated into the framework provided by the criticality analysis. This ontological OD definition has shown to be well-suited for reasoning over formalized criticality phenomena, thereby being a central element in a verification and validation strategy.

Obviously, future work can address the extension of the knowledge base – formalizing a wider variety of criticality phenomena –, the data sources – analyzing data generated by test vehicles or simulation runs –, a rigorous data analysis – studying the presence of (conjunctions of) criticality phenomena and their relation to traffic conflicts –

as well as a practical evaluation of the competency questions A4 and T1 to T3. An in-depth semantic analysis of large-scale intersection scenarios over a variety of criticality phenomena enables insights into the complex operating contexts of urban ADSs prior to system design. Furthermore, a first evaluation demonstrated the method’s feasibility on a drone data set. However, its transferability to live settings under real-time constraints, e.g., for monitoring of ODD boundaries defined via criticality phenomena, is yet to be examined. Incorporating a vehicle perspective (instead of the actor-agnostic view of drones) implicates the necessity of theoretically and practically addressing incomplete and uncertain knowledge about the environment originating from vehicle sensor data. Eventually, the results of this method can be used for safeguarding specific system designs. Future work has to address a rigorous analysis of the relation between criticality phenomena and criticality metrics as well as how these results can be used for the system’s hazard analysis and risk quantification.

Theoretically, we based the presented approach on DLs as well as rules and identified them as a suitable formalism to represent and reason on abstract knowledge about road traffic.

For this theoretical aspect of the work, it remains an open question how to formalize knowledge about concrete domains directly in DLs. For the examined application of A-Box reasoning, we have tackled this challenge by means of augmentations that annotate existing spatio-temporal relations, but methodically, an integration into the logical framework is preferable. In the case of more abstract T-Box reasoning, it is not only preferable, but also mandatory to incorporate spatio-temporal semantics. For example, Allen’s calculus states that $\text{meets} \circ \text{during} \sqsubseteq \text{overlaps} \sqcup \text{during} \sqcup \text{starts}$, which is not a valid RIA in $\mathcal{SROIQ}^{(D)}$. In general, $\mathcal{SROIQ}^{(D)}$ has been shown not to be able to represent the semantics of such calculi [35], i.e., relational composition tables, due to the lack of, e.g., role negations. At least for RCC8, an alternative formalization using GCIs was proposed [40] and prototypically implemented [41], although the work seems to be abandoned. Therefore, theoretical advances need to address the incorporation of spatio-temporal semantics into the reasoning process, e.g., by integrating qualitative reasoning frameworks.

We furthermore presented an implementation of the proposed method and evaluated a prototype on the large and multi-faceted A-Boxes of the inD drone data set.

Practically, we found that reasoners, albeit mature, currently diminish in viability, leaving many of the available tools unmaintained. Another practical aspect concerns the extensive space and time consumption of the PELLET reasoner when running on the large A-Boxes of inD, and hence potential performance improvements by, e.g., extending the idea of A-Box slicing. Alternatively, one refrain from using DL reasoners for inferring temporal properties. Scenarios can still be sliced, but analyzed with a solver for a scenario language.

TABLE 6. Employed geometrical notation and functions.

Notation	Semantics
p	A point in two-dimensional Euclidean space
p_i	The centroid’s position of the geometry g_i
d_i	The diameter of the geometry g_i (m)
$p_i^{fl}, p_i^{fr}, p_i^{br}, p_i^{bl}$	The front left, right and back left, right points of the geometry g_i (assuming a given yaw θ_i)
$\text{dist}(p_1, p_2)$	The euclidean distance between two points (m)
$\text{area}(g_i)$	The area of the geometry g_i (m ²)
$\text{points}(g_i)$	The set of points of the geometry g_i
$\text{ray}(p, \theta)$	A ray (half-line) starting from p with angle θ
$\text{circle}(p, r)$	A circle around p with radius r
$\text{centroid}(g_i)$	The centroid point of the geometry g_i
$\text{polygon}(P)$	The polygon spanned by a tuple of points P

Lastly, the work has two assumptions: First, we assumed crisp terminological concepts within the T-Box. Second, the A-Box was considered perfectly representing ground truth.

The first issue ignores that the OD model represents inherently fuzzy knowledge of humans—e.g., ‘heavy rain’ may not have a crisp boundary. Representations of knowledge need to acknowledge vagueness within the T-Box and incorporate it into the reasoning process. The second issue concerns the applicability when presented with uncertain inputs. Here, one has to propagate those perception uncertainties along the inferences made by the reasoner. For a meaningful contribution to verification and validation, it is therefore imperative to both regard vagueness at design time and uncertainties at run time.

APPENDIX EVALUATED CRITICALITY PHENOMENA

We now display the formalization of the examined criticality phenomena as introduced in Table 3. Employed concepts are only traced one layer down their recursive formalization. Basic concepts are marked by ^b. Concepts that are augmented by an algebraic or first order logic formalization are indicated by := instead of the DL operators \equiv , \sqsubseteq , or \sqsupseteq . For this, we mix first-order predicate logic with DL and algebraic operations. Semantically, this is to be understood as a first order logic formula over the corresponding algebraic or geometric theory where the DL fragments, e.g., $\exists \text{has_traffic_entity}\{A_i\}$, represent sets of individuals on which we use set operators. These formulae are implemented using the augmentation library. We rely on some geometrical notation, listed in Table 6. We restrict the geometries to two dimension in Euclidean space and assume a single coordinate system.

Overall, the algebraic and geometric formalizations are based on the notation depicted in Table 7.

For an additional overview, Table 8 presents a listing of all employed target values for criticality phenomena where thresholds have been applied or used within their concepts.

a) *Pedestrian Crossing / Ford (CP 117)*: CP 117 is defined as the disjunction of pedestrian crossings and fords.

$\text{CP}_{117} \equiv \text{Pedestrian_Crossing}^b \sqcup \text{Pedestrian_Ford}^b$

TABLE 7. Employed mathematical identifiers including their corresponding DL representation (in A.U.T.O.).

Notation	Semantics	A.U.T.O. representation
A_i	A traffic participant with identifier no. i	Traffic_Participant(A), (\exists identifier $^{-1}.\{A\}$)(i)
O	A spatial object	Spatial_Object(O)
g_i	The geometry of A_i described as a polygon (tuple of points)	hasGeometry
θ_i	The yaw angle ($^\circ$) of A_i	has_yaw
θ_i^{max}	The assumed maximum yaw angle ($^\circ$) of A_i	has_maximum_yaw
ω_i^{max}	The assumed maximum yaw rate ($^\circ/s$) of A_i	has_maximum_yaw_rate
s_i	The speed (scalar, m/s) of A_i	has_speed
a_i	The acceleration (scalar, m/s 2) of A_i	has_acceleration

TABLE 8. Target values for each criticality phenomenon (CP).

CP ID	Concept	Target Value(s)	Unit
113	is_near	4	m
295	Heavy_Rain	10	mm/h
295	Extremely_Heavy_Rain	50	mm/h
294	Freezing_Air	0	$^\circ$ C
91	has_intersecting_path	8 (τ_1)	s
91	has_intersecting_path	3 (τ_2)	s
150	relevant_area	1	s
103	has_acceleration	-4.61, -3.3	m/s 2
163	high_relative_speed	0.25	-

b) *Building for Unpredictable Road Users near Road (CP 181)*: CP 181 is under-approximated by the (not exhaustive) union of retirement homes, schools, and kindergartens.

Kindergarten $^b \sqcup$ Retirement_Home $^b \sqcup$ School $^b \sqsubseteq$ CP $_{181}$

c) *Pedestrian on Roadway (CP 82)*: CP 82 is a pedestrian that geometrically intersects with a driveable lane.

CP $_{82} \equiv$ Pedestrian \sqcap \exists intersects.Driveable_Lane b

with Pedestrian \equiv Human \sqcap \nexists drives.Vehicle, and intersects(a, b) := $\exists g_1 \in (\exists$ hasGeometry $^{-1}.\{a\}) \wedge \exists g_2 \in (\exists$ hasGeometry $^{-1}.\{b\}).points(g_1) \cap points(g_2) \neq \emptyset$.

d) *Vulnerable Road User with Road Access (CP 113)*: CP 113 is a vulnerable road user near some driveable lane.

CP $_{113} \equiv$ VRU \sqcap \exists is_near.Driveable_Lane b

with VRU \equiv Bicyclist \sqcup Pedestrian, and is_near(a, b) := dist(a, b) < 4m.

e) *Passing of Parking Vehicle (CP 91)*: CP 91 is a temporal CP and therefore defined as a passing activity that is conducted by a driver overtaking parking vehicles.

CP $_{91} \equiv$ Pass \sqcap \exists conducted_by b .Driver \sqcap
 \exists has_participant b .Parking_Vehicle

with Driver \equiv \exists drives.Vehicle
 and Parking_Vehicle \sqsupseteq Vehicle \sqcap
 Non_Moving_Dynamical_Object \sqcap
 \exists intersects(Parking_Lane \sqcup Parking_Space).
 The passing activity is augmented using finite look back

from the current scene s_i , searching for a scene s_j and scenes s_k between s_i and s_j satisfying the passing constraint.

Pass(A_i, O) :=

$$O \in (\text{is_in_proximity}.\{A_i\}) \wedge O \in (\text{behind}.\{A_i\}) \wedge$$

$$\exists s_i \in (\exists \text{has_traffic_entity}.\{A_i\}).$$

$$O \in (\exists \text{in_traffic_model}.\{s_i\}) \wedge \exists s_j \in (\exists \text{before}.\{s_i\}).$$

$$\exists A_j \in (\exists \text{in_traffic_model}.\{s_j\} \sqcap \exists \text{identical_to}.\{A_i\}).$$

$$\exists O_j \in (\exists \text{in_traffic_model}.\{s_j\} \sqcap \exists \text{identical_to}.\{O\}).$$

$$O_j \in (\exists \text{is_in_proximity}.\{A_j\}) \wedge O_j \in (\exists \text{in_front_of}.$$

$$\{A_j\}) \forall s_k \in (\exists \text{after}.\{s_j\} \sqcap \exists \text{before}.\{s_i\}).$$

$$\exists A_k \in (\exists \text{in_traffic_model}.\{s_k\} \sqcap \exists \text{identical_to}.\{A_i\}).$$

$$\exists O_k \in (\exists \text{in_traffic_model}.\{s_k\} \sqcap \exists \text{identical_to}.\{O\}).$$

$$O_k \in (\exists \text{is_in_proximity}.\{A_k\}) \wedge$$

$$O_k \in (\nexists \text{in_front_of}.\{A_k\}) \wedge O_k \in (\nexists \text{behind}.\{A_k\}).$$

Due to mixing of universal, existential, and negated existential quantification, this formalization can not be depicted as a rule.

f) *Intersecting Planned Paths (CP 293)*: Paths are represented as rays, for which we check whether they intersect in a given threshold (τ_1) and how close it is (τ_2).

CP $_{91} \equiv$ Dynamical_Object $^b \sqcap$

\exists has_intersecting_path.Dynamical_Object b

with

has_intersecting_path(A_1, A_2) := $\exists p \in \mathbb{R}^2$.

$$p \in \text{ray}(p_1, \theta_1) \cap \text{ray}(p_2, \theta_2) \wedge \frac{\text{dist}(p_1, p)}{s_1} + \frac{\text{dist}(p_2, p)}{s_2} < \tau_1$$

$$\wedge \left| \frac{\text{dist}(p_1, p)}{s_1} - \frac{\text{dist}(p_2, p)}{s_2} \right| < \tau_2$$

where $\tau_1 := 8$ s and $\tau_2 := 3$ s.

g) *Heavy Rain (CP 295)*: CP 295 is defined as a rain with a precipitation intensity of at least 10 mm/h.

CP $_{295} \equiv$ Extremely_Heavy_Rain \sqcup Heavy_Rain

with rules Rain(r) \wedge has_precipitation_intensity(r, x) $\wedge x > 50 \rightarrow$ Extremely_Heavy_Rain(r) and Rain(r) \wedge has_precipitation_intensity(r, x) $\wedge x \geq 10 \wedge x < 50 \rightarrow$ Heavy_Rain(r).

h) *Freezing Temperatures (CP 294)*: We define CP 294 as an air individual that has a temperature of below 0 $^\circ$ C.

CP $_{294} \equiv$ Freezing_Air

with Air(a) \wedge has_temperature(a, t) $\wedge t < 0 \rightarrow$ Freezing_Air(a).

i) *Small Distance (CP 83)*: We identify any individual that has a small distance to some spatial object as CP 83.

CP $_{150} \equiv$ \exists object $_{CP150}$.Spatial_Object

with object $_{CP150}$ (A_1, A_2) := relevant_area(A_1) \cap relevant_area(A_2) $\neq \emptyset$. The relevant area is defined

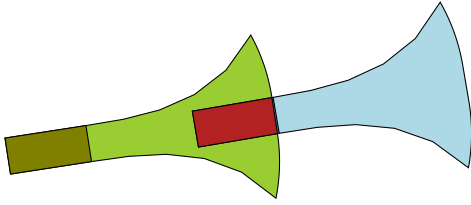


FIGURE 10. The relevant areas of two vehicles in a following scenario.

as the set of points that can be realistically reached:

$$\text{relevant_area}(A_i) := \{x \in \mathbb{R}^2 \mid \exists \omega \in [-\omega_i^{\max}, \dots, \omega_i^{\max}]. \\ \exists t < t_H. \exists p \in \{p_i^l, p_i^r\}. \text{pos}_\omega(p, s_i, \theta_i, t) = x\}$$

where $\text{pos}_\omega(p_i, s_i, \theta_i, t) := p_i + s_i \cdot t \cdot (\cos(\theta_i^t), \sin(\theta_i^t))$ and $\theta_i^t := \begin{cases} (\theta_i + \text{sgn}(\omega) \cdot \theta_i^{\max} \cdot t - \frac{\theta_i^{\max 2}}{2\omega}) & \text{if } |\omega \cdot t| > \theta_i^{\max} \\ (\theta_i + \frac{\omega \cdot t^2}{2}) & \text{else} \end{cases}$ and t_H is a time horizon (set to 1 s). Figure 10 shows two relevant areas and their intersection (in red).

j) *Strong Braking Maneuver (CP 103)*: Strong braking maneuvers are inferred based on the driven vehicle type.

$$\text{Motorized_Road_Vehicle}^b(v) \\ \wedge \text{has_acceleration}^b(v, a) \\ \wedge a < -4.61 \rightarrow \text{CP}_{103}(v) \\ \text{Bicycle}^b(v) \wedge \text{has_acceleration}^b(v, a) \\ \wedge a < -3.3 \rightarrow \text{CP}_{103}(v)$$

k) *High Relative Speed (CP 163)*: CP 163 is a vehicle with high relative speed, i.e., $\text{CP}_{163} \equiv \exists \text{object}_{\text{CP}_{163}}. T$ with

$$\text{object}_{\text{CP}_{163}}(A_1, A_2) := \frac{\|v_1 - v_2\|_2}{\min(s_{\text{rule}}^{\text{max}}, s_{\text{max}}^{A_1})} \geq 0.25$$

where $s_{\text{max}}^{\text{rule}}$ is the maximum allowed speed and $s_{\text{max}}^{A_1}$ is the estimated maximum speed capability of A_1 .

l) *Occlusion (CP 25)*: Occlusions are modeled as n -ary relations, allowing multiple occluding objects. CP 25 is the reified occlusion relation, i.e., $\text{CP}_{25} \equiv \text{Is_Occlusion}$.

Our two-dimensional approach is visualized in Figure 11. We over approximate occlusions by assuming a circular field of view with some visibility range r for an analyzed subject, i.e., $\text{fov}(A_i) := \{x \in \mathbb{R}^2 \mid \|c_i - x\| \leq r\}$. c_i is the point where the sensing of A_i is assumed to be located. For simplicity,

$$c_i := \text{centroid}(g_i) + (\cos(\theta_i), \sin(\theta_i)) \cdot \frac{\|p_i^l - p_i^r\|_2}{4}.$$

For a given opaque entity of non-zero height with its polygon exterior g and a subject A_i , the left- and rightmost points and relative angles from the viewpoint of c_i are:

$$\begin{aligned} & \bullet p_{g,i}^l := \text{argmin}_{p \in g} \theta_{\text{rel}}(c_i, p) \\ & \bullet p_{g,i}^r := \text{argmax}_{p \in g} \theta_{\text{rel}}(c_i, p) \\ & \bullet \theta_{g,i}^l := \min_{p \in g} \theta_{\text{rel}}(c_i, p) \\ & \bullet \theta_{g,i}^r := \max_{p \in g} \theta_{\text{rel}}(c_i, p) \end{aligned}$$

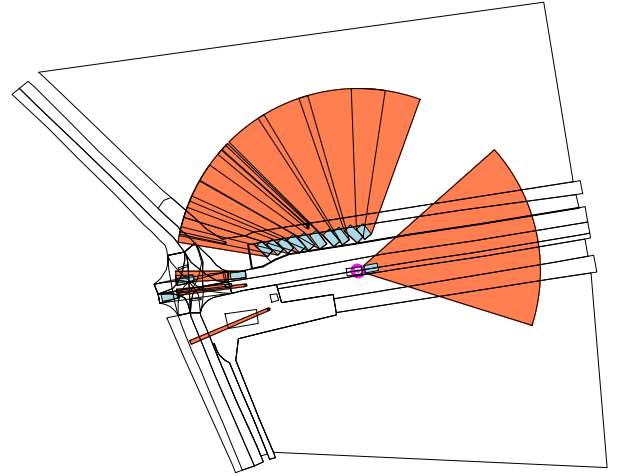


FIGURE 11. Visualization of occlusions on an inD scene. Occluded areas are orange, occluding objects are blue, and the observer is a purple dot.

where $\theta_{\text{rel}}(c_i, p) := \text{arctan2}(p_{i,1} - c_{i,1}, p_{i,0} - c_{i,0})$. For the edge case that $c_{i,1} \in [\min_{p_i \in g} p_{i,1}, \max_{p_i \in g} p_{i,1}]$ and $\exists p \in g. \theta_{\text{rel}}(c_i, p) \in [270, 360)$ and $\exists p \in g. \theta_{\text{rel}}(c_i, p) \in [0, 90)$, we note that min and max need to be switched in the definitions.

The circular segment spanned by g onto the boundary of the field of view is $\text{circular_segment}(c_i, r, \theta_{g,i}^l, \theta_{g,i}^r) := \{r \cdot (\cos(\theta), \sin(\theta)) + c_i \mid \theta \in [\theta_{g,i}^l, \theta_{g,i}^r]\}$. The exterior of g 's occluded area for A_i 's viewpoint is then $\text{occluded}(A_i, g) := \text{polygon}(\text{circular_segment}(c_i, r, \theta_{g,i}^l, \theta_{g,i}^r) \frown (p_{g,i}^r, p_{g,i}^l)) \cup \text{polygon}(g)$, with \frown denoting tuple concatenation. For a geometry g_O of a spatial object O and the subject actor A_i , we denote the rate of occlusion of O through g_1, \dots, g_n by⁸

$$\delta(A_i, g_O, g_1, \dots, g_n) \\ := \frac{\text{area}(g_O) \cap \bigcap_{j=1}^n \text{area}(\text{occluded}(A_i, g_j))}{\text{area}(g_O \cap \text{fov}(A_i))}.$$

If for an Observer(A) and Spatial_Object(O) it holds that there are $\{O_1, \dots, O_n\} \sqsubseteq (\text{Spatial_Object} \sqcap \neg\{A, O\})$ with $\delta(A, O, O_1, \dots, O_n) = d > 0 \wedge \bigwedge_{k=1}^n \exists h \in (\exists \text{has_height}^{-1}. \{O_k\}). h > 0$ then it there exists an individual $\text{Is_Occlusion}(N)$ with

- $\text{is_occluded_for}(N, A)$,
 - $\text{is_occluded_by}(N, O_1), \dots, \text{is_occluded_by}(N, O_n)$,
 - $\text{is_occluded}(N, O)$,
 - $\text{has_occlusion_rate}(N, d)$.
- m) *Occluded Pedestrian (CP 160)*: This phenomenon is:

$$\text{CP}_{160} \equiv \text{Is_Occlusion} \sqcap \exists \text{is_occluded.Pedestrian}$$

n) *Occluded Traffic Infrastructure (CP 265)*: Similarly:

$$\text{CP}_{265} \equiv \text{Is_Occlusion} \sqcap \\ \exists \text{is_occluded.Traffic_Infrastructure}$$

8. This is not perfectly accurate, but it suffices for our case.

o) *Illegitimate Use of Pedestrian Crossing (CP 69)*: The formalizing rule is detailedly presented in Section VI-A.

p) *Misconduct (CP 143)*: We base CP 143 on CP 69 as well as not turning on the vehicle lights in night scenarios.

$$CP_{143} \sqsupseteq (\text{Off_Light_Vehicle} \sqcap \exists \text{in_traffic_model}^b. \text{Night_Scenario}) \sqcup CP_{69}$$

with $\text{Off_Light_Vehicle} \equiv \text{Vehicle} \sqcap \nexists \text{consists_of}(\text{Headlight} \sqcap \text{Active_Lamp})$
 and $\text{Night_Scenario} \equiv \text{Scenario} \sqcap \exists \text{has_traffic_entity.Night_Environment}$.

ACKNOWLEDGMENT

The authors would like to thank the consortium for the successful cooperation.

REFERENCES

- [1] E. D. Dickmanns *et al.*, “The seeing passenger car ‘VaMoRs-P,’” in *Proc. Intell. Veh. Symp.*, 1994, pp. 68–73.
- [2] *Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles*, SAE Int. Standard J3016, 2021.
- [3] C. Neurohr, L. Westhofen, M. Butz, M. H. Bollmann, U. Eberle, and R. Galbas, “Criticality analysis for the verification and validation of automated vehicles,” *IEEE Access*, vol. 9, pp. 18016–18041, 2021.
- [4] L. Westhofen *et al.*, “Criticality metrics for automated driving: A review and suitability analysis of the state of the art,” 2021, *arXiv:2108.02403*.
- [5] R. Baier, *Richtlinien für Die Anlage von Stadtstraßen: RASt 06*. Cologne, Germany: FGSV Verlag, 2007.
- [6] S. Ulbrich, T. Menzel, A. Reschka, F. Schuldt, and M. Maurer, “Defining and substantiating the terms scene, situation, and scenario for automated driving,” in *Proc. IEEE 18th Int. Conf. Intell. Transp. Syst.*, Sep. 2015, pp. 982–988. [Online]. Available: <http://ieeexplore.ieee.org/document/7313256/>
- [7] M. Scholtes *et al.*, “6-layer model for a structured description and categorization of urban traffic and environment,” *IEEE Access*, vol. 9, pp. 59131–59147, 2021.
- [8] K. Czarnecki, “Operational world model ontology for automated driving systems—Part 1: Road structure,” Waterloo Intell. Syst. Eng. (WISE) Lab, Univ. Waterloo, Waterloo, ON, Canada, Rep., 2018, doi: [10.13140/RG.2.2.29991.65447](https://doi.org/10.13140/RG.2.2.29991.65447).
- [9] K. Czarnecki, “Operational world model ontology for automated driving systems—Part 2: Road users, animals, other obstacles, and environmental conditions,” Waterloo Intell. Syst. Eng. (WISE) Lab, Univ. Waterloo, Waterloo, ON, Canada, Rep., 2018, doi: [10.13140/RG.2.2.11327.00165](https://doi.org/10.13140/RG.2.2.11327.00165).
- [10] *OpenSCENARIO 2.0.0*, Assoc. Stand. Autom. Meas. Syst., Hoehenkirchen, Germany, Dec. 2021.
- [11] B. Hummel, “Description logic for scene understanding at the example of urban road intersections,” Ph.D. dissertation, Institut für Mess- und Regelungstechnik mit Maschinenlaboratorium, Universität Karlsruhe, Karlsruhe, Germany, 2009. [Online]. Available: <https://publikationen.bibliothek.kit.edu/1000015554>
- [12] I. Urbietta, M. Nieto, M. García, and O. Otaegui, “Design and implementation of an ontology for semantic Labeling and testing: Automotive global ontology (AGO),” *Appl. Sci.*, vol. 11, no. 17, p. 7782, 2021.
- [13] E. De Gelder *et al.*, “Towards an ontology for scenario definition for the assessment of automated vehicles: An object-oriented framework,” *IEEE Trans. Intell. Vehicles*, early access, Jan. 25, 2022, doi: [10.1109/TIV.2022.3144803](https://doi.org/10.1109/TIV.2022.3144803).
- [14] L. Zhao, R. Ichise, S. Mita, and Y. Sasaki, “Core ontologies for safe autonomous driving,” in *Proc. CEUR Workshop Proc.*, vol. 1486, 2015, pp. 1–4.
- [15] *OpenODD 1.0.0*, Assoc. Stand. Autom. Meas. Syst., Hoehenkirchen, Germany, Dec. 2021.
- [16] B. Neumann and R. Möller, “On scene interpretation with description logics,” in *Cognitive Vision Systems*. Heidelberg, Germany: Springer, 2006, pp. 247–275.
- [17] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*. London, U.K.: Pearson Educ., 2002.
- [18] A. G. Cohn, B. Bennett, J. Gooday, and N. M. Gotts, “Qualitative spatial representation and reasoning with the region connection calculus,” *Geoinformatica*, vol. 1, no. 3, pp. 275–316, 1997.
- [19] J. F. Allen, “An interval-based representation of temporal knowledge,” in *Proc. IJCAI*, vol. 81, 1981, pp. 221–226.
- [20] M. Boudra, M. D. Hina, A. Ramdane-Cherif, and C. Tadj, “Architecture and ontological modelling for assisted driving and interaction,” in *Proc. ACCENTS*, 2015, pp. 270–283.
- [21] A. Armand, D. Filliat, and J. Ibañez-Guzman, “Ontology-based context awareness for driving assistance systems,” in *Proc. IEEE Intell. Veh. Symp.*, 2014, pp. 1–6.
- [22] L. Zhao, R. Ichise, S. Mita, and Y. Sasaki, “An ontology-based intelligent speed adaptation system for autonomous cars,” in *Semantic Technology (Lecture Notes in Computer Science (Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics))*, 2015, pp. 397–413. [Online]. Available: http://link.springer.com/10.1007/978-3-319-15615-6_30
- [23] M. Hülsen, J. M. Zöllner, and C. Weiss, “Traffic intersection situation description ontology for advanced driver assistance,” in *Proc. IEEE Intell. Veh. Symp.*, 2011, pp. 993–999.
- [24] I. Jatzkowski, T. Menzel, A. Bock, and M. Maurer, “A knowledge-based approach for the automatic construction of skill graphs for online monitoring,” in *Proc. IEEE Intell. Veh. Symp. (IV)*, 2021, pp. 142–149.
- [25] G. Bagschik, T. Menzel, and M. Maurer, “Ontology based scene creation for the development of automated vehicles,” in *Proc. IEEE Intell. Veh. Symp. (IV)*, Jun. 2018, pp. 1813–1820. [Online]. Available: <https://ieeexplore.ieee.org/document/8500632/>
- [26] T. Menzel, G. Bagschik, L. Isensee, A. Schomburg, and M. Maurer, “From functional to logical scenarios: Detailing a keyword-based scenario description for execution in a simulation environment,” in *Proc. IEEE Intell. Veh. Symp. (IV)*, 2019, pp. 2383–2390.
- [27] E. de Gelder and O. O. D. Camp, “Tagging real-world scenarios for the assessment of autonomous vehicles,” 2020, *arXiv:2012.01081*.
- [28] M. A. Mohammad, I. Kaloskampis, Y. Hicks, and R. Setchi, “Ontology-based framework for risk assessment in road scenes using videos,” *Procedia Comput. Sci.*, vol. 60, no. 1, pp. 1532–1541, 2015. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S1877050915024278>
- [29] A. Poddey, T. Brade, J. E. Stellet, and W. Branz, “On the validation of complex systems operating in open contexts,” 2019, *arXiv:1902.10517*.
- [30] *Road Vehicles—Safety of the Intended Functionality*, ISO Standard ISO/DIS 21448, Dec. 2021.
- [31] F. Baader, D. Calvanese, D. L. McGuinness, D. Nardi, and P. F. Patel-Schneider, *The Description Logic Handbook: Theory, Implementation and Applications*. Cambridge, U.K.: Cambridge Univ. Press, 2003.
- [32] M. Krötzsch, *Description Logic Rules*, vol. 8. Heidelberg, Germany: IOS Press, 2010.
- [33] W. Damm and R. Galbas, “Exploiting learning and scenario-based specification languages for the verification and validation of highly automated driving,” in *Proc. IEEE/ACM 1st Int. Workshop Softw. Eng. AI Auton. Syst. (SEFAIAS)*, 2018, pp. 39–46.
- [34] F. Schuldt, “Ein Beitrag für den methodischen test von automatisierten Fahrfunktionen mit Hilfe von virtuellen Umgebungen,” Ph.D. dissertation, Von der Fakultät für Elektrotechnik, Technischen Universität Carolo-Wilhelmina zu Braunschweig, Braunschweig, Germany, 2017.
- [35] F. Hogenboom, B. Borgman, F. Frasinicar, and U. Kaymak, “Spatial knowledge representation on the semantic Web,” in *Proc. IEEE 4th Int. Conf. Semantic Comput.*, 2010, pp. 252–259.
- [36] M. Scholtes, M. Schuldes, H. Weber, N. Wagener, M. Hoss, and L. Eckstein, “OMEGAFormat: A comprehensive format of traffic recordings for scenario extraction,” in *Proc. FAS-Workshop*, 2022, pp. 195–205.
- [37] M. Weiser, “Program slicing,” *IEEE Trans. Softw. Eng.*, vol. SE-10, no. 4, pp. 352–357, Jul. 1984.
- [38] J. Bock, R. Krajewski, T. Moers, S. Runde, L. Vater, and L. Eckstein, “The inD dataset: A drone dataset of naturalistic road user trajectories at German intersections,” in *Proc. IEEE Intell. Veh. Symp. (IV)*, 2019, pp. 1929–1934.

- [39] P. Schneider, M. Butz, C. Heinzemann, J. Oehlerking, and M. Woehrl, "Towards threat metric evaluation in complex urban scenarios," in *Proc. IEEE Int. Intell. Transp. Syst. Conf. (ITSC)*, 2021, pp. 1192–1198.
- [40] Y. Katz and B. C. Grau, "Representing qualitative spatial information in OWL-DL," in *Proc. Workshop OWL Exp. Directions CEUR*, vol. 188. Galway, Ireland, Nov. 2005, pp. 1–5.
- [41] M. Stocker and E. Sirin, "PelletSpatial: A hybrid RCC-8 and RDF/OWL reasoning and query engine," in *Proc. OWLED*, vol. 529, 2009, pp. 1–10.

LUKAS WESTHOFEN received the B.Sc. and M.Sc. degrees (with Hons.) in computer science from RWTH Aachen University in 2015 and 2019, specializing on the topics of probabilistic programs and software verification. He currently works with German Aerospace Center (DLR e.V.) Institute of Systems Engineering for Future Mobility, where his research focuses on developing methods to establish confidence in the safety of automated vehicles. More specifically, his interests include the formalization of knowledge as well as its exploitation for safeguarding automated driving functions.

CHRISTIAN NEUROHR received the B.Sc. and M.Sc. degrees in mathematics from Technische Universität Kaiserslautern, Germany, in 2011 and 2013, respectively, and the Ph.D. (Dr. rer. nat.) degree from Carl von Ossietzky Universität Oldenburg, Germany, in 2018. After a short period as a Visiting Researcher with the MAGMA Group, University of Sydney, he started his occupation as a Postdoctoral Researcher with the German Aerospace Center (DLR e.V.) Institute of Systems Engineering for Future Mobility, where he is working in the area of scenario-based verification and validation of automated vehicles. Specifically, he is the Scientific Coordinator of the sub-project "Criticality Analysis" within the Project VVMethods.

MARTIN BUTZ received the Ph.D. (Dr. rer. nat) degree in mathematics from the University of Regensburg in 2012. Since then, he is working as a Research Engineer in the field of formal methods and software architecture with Corporate Research of Robert Bosch GmbH, Renningen, Germany. The main focus of his current research is verification and validation of autonomous driving functions, in particular, the application of formal modeling techniques for open context problems.

MAIKE SCHOLTES received the B.Sc. and M.Sc. degrees in computational engineering science from RWTH Aachen University, Germany, in 2015 and 2017, respectively, and the Ph.D. degree from the Institute for Automotive Engineering (ika), RWTH Aachen University. During semesters abroad she, *inter alia*, studied with the University of California at San Diego, San Diego. After working one year in the area of driving simulation with fka SV Inc. in the Silicon Valley, she started her Ph.D. degree. She works on the assessment of automated driving and ADAS with a focus on machine perception. She received the Dean's List Award for her M.Sc. degree.

MICHAEL SCHULDES received the B.Sc. and M.Sc. degrees in electrical engineering from RWTH Aachen University, Germany, in 2016 and 2019, respectively, where he is currently a Research Assistant with the Institute for Automotive Engineering. His research interests include assessment of autonomous driving functions with a focus on data-driven scenario-based approaches.