Towards a Congruent Interpretation of Traffic Rules for Automated Driving – Experiences and Challenges

Lukas Westhofen[1] [0000−0003−1065−4182], Ingo Stierand[1], Jan Steffen Becker[1], Eike Möhlmann[1], and Willem Hagemann[1]

German Aerospace Center (DLR) e.V., Institute of Systems Engineering for Future Mobility, Oldenburg, Germany
{lukas.westhofen,ingo.stierand,jan.becker,
eike.moehlmann,willem.hagemann}@dlr.de

Abstract. The homologation of automated driving systems for public roads requires a rigorous safety case. Regulations of the United Nations demand to demonstrate the compliance of the developed system with local traffic rules. Hence, evidences for this have to be delivered by means of formal proofs, online monitoring, and other verification techniques in the safety case. In order for such methods to be applicable traffic rules have to be made machine-interpretable. However, that pursuit is highly challenging. This work reports on our practical experiences regarding the formalization of a non-trivial part of the German road traffic act. We identify a central issue when formalizing traffic rules within a development process, coined as the congruence problem, which is concerned with the semantic equality of the legal and system interpretation of traffic rules. As our main contribution, we delineate potential challenges arising from the congruence problem, hence impeding a congruent yet formal interpretation of traffic rules. Finally, we aim to initiate discussions by highlighting steps to partially address these challenges.

Keywords: Automated Driving · Traffic Rules · Formalization.

1 Introduction to Traffic Rules for Automated Driving

Automated driving systems (ADSs) are anticipated to take over a large part of the tasks that are currently performed by human drivers. Vehicles shall recognize their environment and perform maneuvers without the need of human intervention. This holds especially for critical situations within the operational design domain. The task of recognizing a dynamic environment, interpreting this in the context of traffic rules, and in particular reacting accordingly, is performed by human beings with a high fault tolerance, as long as they are attentively following what is happening. Due to ever extending operational domains, new risk potentials arise, which must be analyzed before the release of the ADS.

Already in the early stages of the development of an ADS, the question arises as to which requirements must be met to be approved for participation in road traffic. This includes technical requirements (resources, real-time, ...), control requirements (longitudinal and lateral control of vehicle dynamics, ...), and behavioral requirements (appropriate indication of actions, ...). At least for the top-level requirement of traffic rule compliance, this question can be answered with certainty: the United Nations regulation number 157, which is concerned with the approval of automated lane keeping systems, states that '*the activated system shall comply with traffic rules relating to the dynamic driving task in the country of operation*' [22, clause 5.1.2]. This raises two important questions:

1. How ADSs can be designed to comply with the traffic rules, and
2. how it can be proven that they do so?

Traffic rules are written by legal experts to be interpreted again by humans. Therefore, developing solutions for the first question is hindered by their inherently and intentionally vague character. Scenario-based approaches are anticipated to approach the second question [14]. However, due to the high number of test cases, even with scenario-based testing not every test run can be presented to a legal expert for evaluation. Thus, an automated assessment is necessary not only for a correct implementation but also for verification. This directly results in the need for a machine-interpretable understanding of these rules.

A promising way to make knowledge machine-interpretable is the use of formal methods for specification and verification. These allow to analyze the consistency of requirements, enable formal proofs by means of model checking, and evaluate runs of a system during operation. The question therefore arises as to how a formal understanding of traffic rules and case law can be realized. Specifically, the following issue must be clarified: *How can we ensure and demonstrate that the implemented semantics correctly reflects the traffic rule interpretation by the legal expert?* The work at hand contributes to this core issue by

1. an *experience report* on formalizing a part of the German road traffic act,
2. a *formal framework* and *key challenges* distilled from our practical experience, inhibiting a sound and complete rule interpretation, and,
3. potential directions of *future work* for the resolution of these challenges.

We highlight the second point as our main contribution. Our experience indicates solutions therefor to be a prerequisite for the homologation of ADSs.

2 Related Work

The formalization of legal texts, such as the British nationality act [20], is not a new endeavor. These early approaches already included applications on traffic rules in 1991 [6]. Since ADS technology was still in its infancy, the focus was on the support of legal experts and teaching aspects.

Recent advances in ADSs gave updraft to the formalization of traffic rules. We are largely concerned with methodical issues. Related work includes a method for formalizing traffic rules into a defeasible deontic logic [3]. Here, a special focus lies on the identification of terms, the definition of atoms based on these terms, the type of the sentence and the eventual premise-consequence structure of the formalized rule. Similarly, Costescu delineates the necessity of having a legal analysis prior to the actual traffic rule formalization process as to achieve a common understanding between engineers and legal experts [5]. The key observation is that the natural language character of traffic rules is preventing a straightforward formalization. Moreover, the German research project VVMethods investigates, among others, methods for the derivation of behavioral requirements from traffic rules [18]. It highlights that a thorough legal analysis is required, adopting methods for the creation of legal opinions and its associated mindset.

Apart from methodical considerations, most work focuses on formalization techniques. Notably, work has been driven by the Technical University of Munich and partners. First steps were performed by Rizaldi et al. [17], where monitors were derived from linear temporal logic (LTL). Buechel et al. use an ontology-based approach with a joint description logic and rule reasoner [4]. Work since then focused on metric temporal logic, both for highways [13] and intersections [12]. Traffic rule formalization within a development process was examined by Esterle [7]. The thesis analyzes exemplary rules, albeit without apparent legal support, understating the interplay of different expert domains. Formalization is done by defeasible LTL rules, and observer automata are evaluated on realistic data. Besides (derivatives of) LTL, related work relies on a multi-lane spatial logic [19], a defeasible deontic logic [23], and answer set programming [11].

Technical advances have been pursued as well, such as LegalRuleML [1], which partially addresses the challenges – e.g. exceptions – later presented. Due to the large amount of prior work on the technicalities of traffic rule formalization, we refrain from detailing formalisms and focus on the open methodical gaps.

3 An Experience Report

In joint work together with an Original Equipment Manufacturer, a law firm, and BTC Embedded Systems AG (www.btc-embedded.com), we developed an approach to formalize German traffic act (StVO) rules relevant for highway driving. The approach was realized, resulting in 111 formalized StVO rules. An implementation of observers for the formalized rules has been applied to simulated traffic scenarios. The challenges raised in this paper originate in this work.

The taken approach is depicted in Fig. 1. In a first step, the terms and rules of the relevant part of the StVO were assessed and, where ever possible, clarified from a legal perspective. This provided the basis for the formalization, which consists of two layers. The base layer is established by a description logic ontology containing the basic terms (concepts and roles) used in the StVO, like 'vehicle' and 'in front of' [2]. We also call them 'observable entities', as these are assumed to be entities recognized by a suitable perception chain. The second layer is established by a definition of predicates and function terms, which take the ontology as their domain. Along this, a light-weight fragment of duration

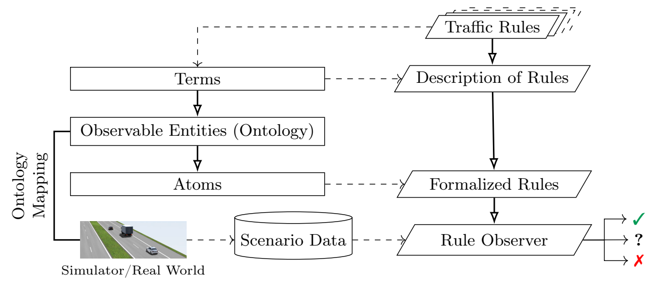

Fig. 1. Approach for the formalization and observer implementation of traffic rules.

calculus has been defined in order to allow specifying temporal terms. Finally, the traffic rules were formalized based on those *atoms* using the same language.

As the subsequent section illustrates, traffic rules contain vague terms (e.g. 'to endanger'). Many terms have therefore been defined in a sufficient and necessary version. Hence, each rule has been formalized in two versions: One to safely detect the satisfaction of the traffic rule, and one for the safe detection of violations.

Finally, a selected subset of the rules was manually implemented in the BTC Embedded Platform (EP) [21]. For this, a set of traffic scenarios has been created, simulated, and logged. A mapping of the relevant observable entities within the legal ontology to the parameters observable in the ontology of the simulator has been defined. The mapped simulation logs have been fed to the generated observers in order to detect rule satisfactions and violations, respectively.

3.1 Example

In order to illustrate our approach, let us examine the exemplary rule of section four, paragraph one, sentence one of the StVO: *'The distance to a preceding vehicle shall be, generally, large enough to allow stopping behind this vehicle even if it is suddenly braked.'* We also refer to this rule as the *distance rule*.

We identified the following relevant terms: *distance*, *preceding*, *vehicle*, *generally*, *large enough*, *stopping behind*. For those, legal experts delivered a (recursive) definition. For example, explicating a *large enough* distance involves the *stopping distance*, which in turn relies on the *braking distance*, *speed*, and the driver's *reaction time*. Moreover, it relies on the *minimal stopping distance*, which is dependent on various complex factors such as road surfaces, tire types, weather conditions, and temperatures, allowing only a partial definition. The set of relevant atoms and their observability assumptions are depicted in Fig. 2.

Omitting details due to conciseness, we can assemble the safe satisfaction as

$$\texttt{sat}_{4.1.1} \coloneqq \forall f_1, f_2 \in \texttt{Vehicle} : \texttt{preceding}(f_1, f_2) \wedge \texttt{generally}_n(f_1, f_2) \implies \texttt{enough\_distance}(f_1, f_2).$$

We introduced the atom `generally` to model exceptions, both in a sufficient and a necessary condition. This concerns e.g. situations of driving in a convoy

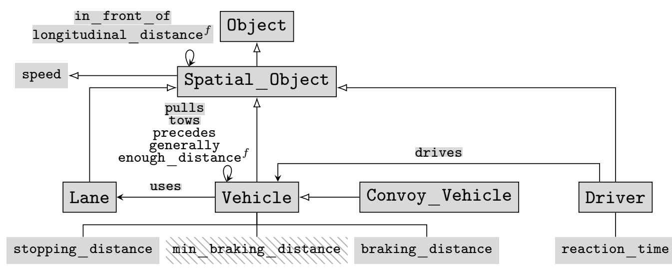

Fig. 2. Ontology and atoms including the relevant terms for the exemplary distance rule. Observable entities are denoted by a gray background, partially formalized atoms are hatched. A directed edge is a binary relation where an open tip denotes the subsumption relation. Properties are indicated by solid lines and functions by superscript ^{f}.

or during starting after waiting in traffic. This leads to the aforementioned two versions of the rule; one including the necessary condition of `generally` for a safe satisfaction and one including the sufficient condition of `generally` for a safe violation of the rule. In the example of a convoy, a partially formalized necessary condition is $\texttt{generally}_n(f_1, f_2) \coloneqq f_1, f_2 \notin \texttt{Convoy\_Vehicle}$. A partial formalization of a sufficient condition is, e.g., $\texttt{generally}_s(f_1, f_2) \coloneqq f_1, f_2 \notin \texttt{Convoy\_Vehicle} \wedge f_1.\texttt{speed} > 10\text{m/s}$. This can be used to safely detect violations of the safe distance, where $\texttt{vio}_{4.1.1}$ evaluates to true only if it is violated:

$$\texttt{vio}_{4.1.1} \coloneqq \exists f_1, f_2 \in \texttt{Vehicle} : \texttt{preceding}(f_1, f_2) \wedge \texttt{generally}_s(f_1, f_2) \wedge \neg\texttt{enough\_distance}(f_1, f_2).$$

Observers are created by translation to EP Universal Patterns, where observable quantities, e.g. `speed`, and atoms, e.g. `generally`, are represented as EP macros.

3.2 Results

In summary, over 160 traffic rules from the StVO have been identified as relevant for highway driving. 69% of these rules were formalized as described above, while 26% defines particular terms and thus are covered at one of the two layers. 4% of the rules are rather explanations and covered otherwise by the formalization. The remaining rules define priorities. For rule formalization, more than 600 terms were needed, from which half have been defined as observable entities, and the other half as atoms. Less than 60 atoms could not be (completely) formalized, where half of them were identified as vague legal concepts. This led to the overall result that 62% out of the formalized rules were *completely* formalized. The remainder depends on not or only partially formalized terms. Simulation scenarios for 26 rules have been created, where both satisfaction and violation of the corresponding rule have been checked by handcrafted observers.

4 Lessons Learned and Future Research Directions

The described practical experience was accompanied by various challenges. We aim to extrapolate our lessons learned to enable, in the end, elaborations on future research directions. We structure the discussion of our report as follows:

- Firstly, we introduce the central problem – called *congruence problem* – whose resolution we distilled as an abstract key goal during our activities;
- secondly, we present challenges that arise from the congruence problem;
- and finally, we sketch potential research directions for these challenges.

4.1 The Congruence Problem

In a nutshell, the *congruence problem* is concerned with the equivalence of semantics between a legal interpretation and the system's implementation. Before in-depth elaborations, we introduce the framework in which we place its definition.

The Semiotic Triangle as a Foundational Model As we are concerned with the equivalence in understanding of stakeholders, the *semiotic triangle* as introduced by Ogden and Richards [15] is a natural fit. It models how humans express conceptualizations of their world and is concerned with

1. symbols, which are identifiers in a given language,[1]
2. concepts, which are semantic interpretations of the symbols, and
3. referents, which are the entities in some world referred to by a concept.

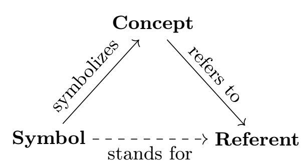

Fig. 3. The semiotic triangle, relating symbols, concepts, and referents.

Their relation is depicted in Fig. 3. Here, a symbol, e.g. 'vehicle', symbolizes an (intended) concept, e.g. 'a machine destined for transportation'. Communicating the symbol 'vehicle' invokes in the receiver the concept of a vehicle. This concept in turn maps to a set of things, called referents. This set can include, e.g., the bicycle in front of your house. Note that the mapping is not necessarily explicit. Because concepts are learned, humans with different background may use the same symbol for different concepts, or use different symbols for the same concepts. We now formally introduce the notion of a semiotic triangle.

[1] In our experience report, we referred to terms, which are defined as symbols that occur in the natural language specification of the rule set.

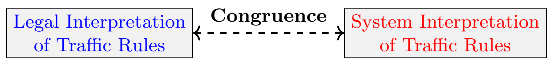

Fig. 4. Congruence between legal and system interpretation of rules

Definition 1 (Semiotic Triangle). *A semiotic triangle is a triple* $\mathsf{T} = (\mathsf{S}, \mathsf{R}, \mathsf{C})$, *where* S *is the set of all symbols,* R *the set of referents, and the conceptualization* C *a function* $\mathsf{C} : \mathsf{S} \to 2^{\mathsf{R}}$, *assigning every symbol a set of referents.*

We emphasize that referents can be anything that can be referred to; this includes imaginary objects as well as traffic rules and scenarios. Furthermore, we highlight that C may not be perfectly explicated. The interpretation of traffic rules is especially affected by this, as traffic rules are designed by and for humans. They exploit, e.g., inherent vagueness of concepts in order to reduce the degree of required explicitness, keeping the rule set concise yet comprehensible.

The Congruence Problem The eventual goal of the system's development is the implementation of some valid interpretation of the relevant legal concepts, a circumstance we call *congruence*. This relation is depicted in Fig. 4. Of course, this view is simplifying as there may be dissensions also among legal experts. Though this work assumes a harmonized legal interpretation, our general idea may be applied to consensus-building as well. The goal is to maximize congruence of legal and system interpretations of traffic rules, i.e. implement a system that judges a scenario (1) as conform to a traffic rule if and only if a legal expert does, and conversely (2) as violating a traffic rule if and only if a legal expert does.

Our practical experience shows that perfect congruence will not be achievable for traffic rules. Rather, we are interested to maximize the cases where the system judges a scenario as conform implies that the legal expert would have done so as well. This is due to an automated legal judgement of all scenarios being unrealistic; despite, we are interested in maximizing legal compliance of the system. Based on Definition 1, we are now ready to formally define congruence.

Definition 2 (Congruence of Semiotic Triangles). *The semiotic triangles* T_1 *and* T_2 *over symbol sets* S_1 *and* S_2 *with* $\mathsf{S}_1 \cap \mathsf{S}_2 \neq \emptyset$, *and concepts* C_1 *of* T_1, C_2 *of* T_2, *are* congruent *if for every* $\mathsf{s} \in \mathsf{S}_1 \cap \mathsf{S}_2$ *it holds that* $\mathsf{C}_1(\mathsf{s}) = \mathsf{C}_2(\mathsf{s})$.

Congruence hence requires that each shared symbol stands for the same set of referents. A visual interpretation is depicted in Fig. 5, assuming that both parties refer to a shared set of referents, i.e., to the same universe.

The definition is motivated by the implication that, if the semiotic triangles of the involved parties are congruent, their interpretation of the traffic rules and all involved symbols has to be congruent as well, therefore achieving the goal sketched in Fig. 4. In our practical experience, we identified congruence as a key step required for correctly formalizing, implementing, and verifying traffic rules. It is only due to congruence that subsequent discussions around technical means, e.g. suitable logics as presented in Sect. 2, become substantiated.

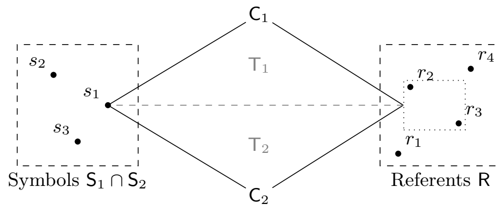

Fig. 5. Visualization of the definition of congruence between two semiotic triangles.

Up to this point, we have introduced both semiotic triangles – a formal framework on comprehension and communication – as well as congruence – a central problem in constructing a system that obeys traffic rules. The problem of establishing congruence is non-trivial. We emphasize this by understanding its emergence in the development process using the framework of semiotic triangles.

Emergence of the Congruence Problem Our report relates to the development of a safety-critical ADS by means of a state-of-the-art development process, i.e. compliant to ISO 26262 [9] and ISO 21448 [10]. A significant portion of the safety case will be concerned with establishing congruence between the legal and system interpretation of traffic rules. Obviously, a system development process involves a variety of relevant parties apart from the legal expert and the system. This includes requirement engineers, software programmers, test designers, certification authorities, and even those interacting with the product during operation. Each is equipped with a semiotic triangle, bidirectionally trying to achieve congruence with other stakeholders' triangles, such as those up- and downstream but also on the opposite development phase. This harmonization chain along a representative development scheme is depicted in Fig. 6.

A prototypical development process is displayed as a bridge between the legal expert and the system. Initially, the legal expert interprets laws and acts s.t. there arises a set of requirements on the system. They are written using the symbols and conceptualization of the legal expert and systematically transformed into a verified and validated system along a complex, iterative sequence of steps. For example, initially, legal requirements are decomposed by a requirements engineer into item-level and technical requirements which can then be used in the design phase. In the end, the system thus implements neither the traffic rules nor the interpretation of the legal expert directly but rather concepts that originated through the alignment of various semiotic triangles. This leads to the core issue:

How can the congruence of the semiotic triangles of the system and the legal expert be maximized in a development process?

A solution is the inductive definition of concepts based on other concepts.

A Decompositional Approach for Congruence: A natural way of aligning semiotic triangles of expert groups is to ask the experts for definitions of the relevant legal symbols. This is based on the *compositionality principle* of symbols [16].

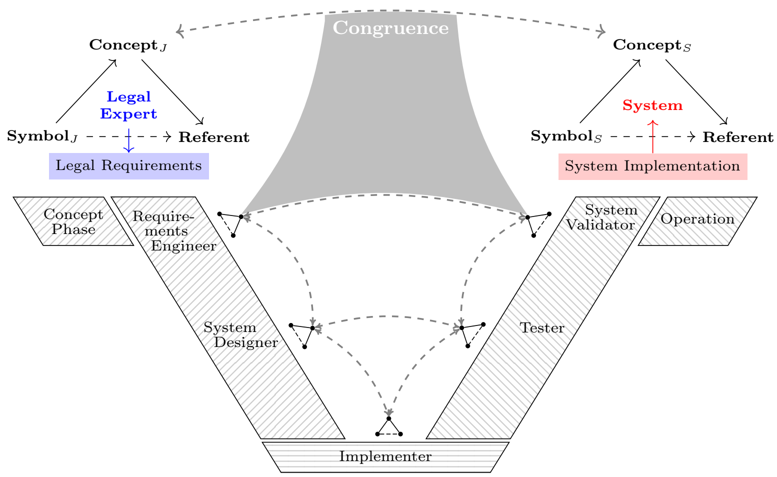

Fig. 6. The emergence of congruence along a representative development process.

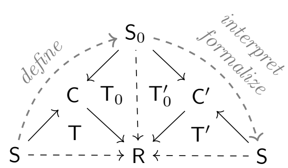

Fig. 7. An approach for congruence by compositionality.

For example, we can use the symbols *distance to*, *preceding*, and *vehicles* to construct a new symbol *distance to preceding vehicles*. In general, symbols can have different contextual semantics, so it is not possible to reconstruct the semantics of a sentence from the non-contextual semantics of its atoms as obtained by syntactic decomposition. However, we assume a clear context with no syntactic ambiguity, justified by the observation that traffic rules are intentionally written this way. Under this assumption, the compositionality principle states that the meaning of a sentence is unambiguously inferred from the contextual meaning and the ordering of contained symbols. We can inductively build definitions from an initial symbol set S_0 symbolizing the same concepts for the first (C) and the second stakeholder (C'). As Fig. 7 shows, by having congruent semiotic triangles T_0 and T'_0, and S_0 for definitions of new symbols $\mathsf{S} \setminus \mathsf{S}_0$, the semiotic triangles T and T' are also congruent. The two-level approach of Sect. 3 exploits this idea.

4.2 Key Challenges in Addressing the Congruence Problem

We introduced the congruence problem as a central issue. This sections substantiates its hardness by challenges that need to be solved as to mitigate its effect. We inferred them from our practical insights, but we do not claim completeness.

Alignment A development process involves various cross-domain experts, including lawyers, logicians, and engineers (cf. Fig. 6). Alignment is the establishment of congruence between these parties as to achieve an overall congruence of the legal and system's interpretation. This challenge was already experienced during our work, where comparatively small teams collaborated. It becomes even more pressing when involving entire companies along automotive supply chains.

Example 1 *For the distance rule, we identified the symbol* vehicle*. An alignment has to be reached as to what a vehicle entails. Whereas a machine learning engineer may learn a perception component on a data set with only wheeled vehicles, a legal expert may assume that a vehicle may not need to have wheels. Thus, the system may exhibit illegitimate behavior near rail vehicles.*

In the framework of semiotic triangles, alignment includes explicating the set vocabulary of relevant legal symbols S, and then agreeing on a definition of $\mathsf{C}(s)$ for each $s \in \mathsf{S}$ and each involved stakeholder in the development process.

Observability In our experience report, we relied on decomposing terms into sub-terms until one assumes observability by the perception chain. The question arises: how do we agree on a justification for this observability assumption? This requires in-depth stakeholder discussions during the design and implementation phase, e.g. when selecting suitable perception technologies from suppliers.

Example 2 *We assumed vehicles to be observable based on a given natural language explanation for the concept of a* vehicle*. We now have to justify this, or, alternatively, need to ask the legal experts for a detailed definition for the concepts used in the definition of a vehicle, such as* machine *and* transportation*.*

Formally, the challenge of observability asks for a justification of the assumption that it is sufficient to give an informal explication of $\mathsf{C}(s)$ for an $s \in \mathsf{S}$.

Vagueness Vagueness can be understood as the circumstance that concepts necessary for situation assessment do not have a clear truth condition. This implies that their set of referents is not *crisp* but rather *fuzzy*. For traffic rules, a common source of fuzziness are vague legal concepts. This can lead to the use of non-exhaustive enumerations of examples in definitions, which, during the development process, hinders the alignment. Thus, this challenge is located even prior to system development, namely within the traffic rules themselves.

Example 3 `generally` *is a vague legal term as exceptions can not be exhaustively explicated, which, legally, indicates the necessity of case-by-case decisions. However, there are special cases such as platooning that can be safely excluded.*

In the framework of semiotic triangles, a concept $\mathsf{C}(s)$ of some $s \in \mathsf{S}$ is vague if a referent $r \in \mathsf{R}$ exists for which the truth value of the statement $r \in \mathsf{C}(s)$ can not be exactly identified even if all necessary information is perfectly present.

Uncertainty If we assume a crisp membership function $\mathsf{C}(s)$, we could, in theory, clearly label referents by their symbols, yielding an evaluation of a scenario with absolute certainty. Although, for this to hold, we have to assume that we

1. considered all parts of the vehicle's environment needed for concept evaluation during its design and built a perception system able to recognize them,
2. knew every legal detail and the perception is perfectly certain, and
3. created a non-probabilistic environment model based on a certain perception.

Those assumptions correspond to *ontological*, *epistemic*, and *aleatoric* uncertainties [8]. In the development process of Fig. 6, we find ontological and aleatoric uncertainty arising mostly during system design and implementation. Epistemic uncertainty can be located in both the early design phase – a lack of knowledge of the designer – or the operation phase – a lack of knowledge of the system. In practice, these assumptions are unrealistic due to physical and economical limitations; thus, to achieve congruence, we have to consider such uncertainties.

Example 4 *For the distance rule, ontological uncertainty is the neglect of weather conditions when formalizing the stopping distance. Subsequently, epistemic uncertainty is the reception of uncertain information about the weather condition from a rain sensor of the ADS. Aleatoric uncertainty exists if we estimate a probability distribution of the current coefficient of friction based on this data.*

Ontological uncertainty requires a relevant symbol s to be not considered when collecting the legal symbols S, and subsequently developing the perception system. Epistemic uncertainty exists if there is some $r \in \mathsf{R}$ for which the truth value of the statement $r \in \mathsf{C}(s)$ can not be exactly identified due to some theoretically crisp information not being known at design- or runtime. Aleatoric uncertainty can be seen as a concept function $\mathsf{C}(s)$ that answers $r \in \mathsf{C}(s)$ only with a certainty $p \in [0, 1]$ due to unmeasurable random factors influencing the evaluation.

Interrelations Terminologically, we encounter *synonyms*, different symbols with the same meaning, and *polysemy*, similar symbols with different meanings.

Example 5 *Note that* distance *is a homograph. It may refer to a spatial, temporal, or even social distance. A legal analysis clarifies that it is to be interpreted spatially. Here, a synonym for spatial distance, e.g. gap, can be helpful.*

For some symbols $s \neq s'$, $\mathsf{C}(s) = \mathsf{C}(s')$ indicates a synonym. Polysemy can be seen as $\mathsf{C}(s) \neq \mathsf{C}(s')$ for some $s \approx s'$, where $\approx$ denotes linguistic similarity.

Interrelations of concepts – including rules – are often present in traffic rules. Traffic rules are especially affected by *priorities and exceptions* as well as *inconsistencies*. Even different rule sets can interact, e.g. by lex specialis. First of all, the challenge arises how to identify those interrelations inherent to traffic rules. The subsequent handling of priorities is then necessary to congruently reflect their semantics. Assume, e.g., the rules $\varphi_1 := a \implies b$ and $\varphi_2 := c \implies d$, not necessarily taken from the same rule set. A thorough legal analysis delivers us with the fact that these rules interrelate by $a \wedge c$ being satisfiable but $b \wedge d$ being unsatisfiable. If $a \wedge c$ holds, the implied behavior is unclear, as the system can not be compliant with both b and d. Prioritization can resolve this ambiguity by e.g. assigning φ_1 a higher precedence, therefore implying that b should follow in case $a \wedge c$ holds. Exceptions arise when the constraint of the higher prioritized rule (e.g. φ_1) is implied by the general case, i.e. if $a \implies c$.

Inconsistencies arise due to incompatible concepts. For the above sketch, without prioritization, it would have followed that φ_1 and φ_2 are contradictory and thus unsatisfiable. This is undesired during formalization and implementation – it indicates formalization errors and can lead to unspecified behavior.

Example 6 *We gave a necessary and sufficient version for the distance rule using* $\texttt{generally}_n(f_1, f_2) \coloneqq f_1, f_2 \notin \texttt{Convoy\_Vehicle}$ *and* $\texttt{generally}_s(f_1, f_2) \coloneqq f_1, f_2 \notin \texttt{Convoy\_Vehicle} \wedge f_1.\texttt{speed} > 10m/s$*. For these to be valid,* $\texttt{generally}_s \implies \texttt{generally}_n$ *has to hold. Imagine we made a modeling error by means of* $\texttt{generally}_s(f_1, f_2) \coloneqq f_1.\texttt{speed} > 10m/s$*. Then,* $\texttt{generally}_s \wedge \neg\texttt{generally}_n$ *is satisfiable but inconsistent with* $\texttt{generally}_s \implies \texttt{generally}_n$*.*

Generally, a priority is an arbitration mechanism that, for some $r \in \mathsf{R}$ satisfying two (contradictory) concepts, decides which concept r is assigned to. Inconsistency, on the other hand, is often based on the reduction to unsatisfiability. This arises when there is some $s \in \mathsf{S}$ with $\mathsf{C}(s) = \emptyset$. This may be due to a contradiction with another concept or a contradiction within its own concept function. It can also arise from more subtle forms of contradictions such as sufficient and necessary conditions conflicting with the first implying the latter.

Traceability We are concerned with traceability of the symbols and their concept functions arising during the congruence procedure. As our practical experience shows, this allows to iteratively detect and resolve reasons for misinterpretations during alignment, hence facilitating congruence. Without traceability support, keeping track of the large amount of traffic rules and their artifacts is almost impossible. Traceability thus affects the entire process of Fig. 6.

Example 7 *A definition of* vehicle *was used for labeling training data of a machine learning component. During testing, we find that the ADS does not hold enough distance around rail vehicles. Traceability allows to pinpoint the mismatch between the machine learning engineer's and legal expert's concept. We can now align the semiotic triangles of the engineer and legal expert.*

Traceability is formally concerned with the storage and accessibility of all artifacts C and S of all semiotic triangles involved in the system life cycle.

4.3 Solutions to be Developed in Future Work

We close the discussion by sketching open or partially examined ends, as to identify future research directions. To that effect, we propose investigations of the following topics, some of which the related work of Sect. 2 has touched on:

– Methods for alignment and uncertainty reduction in development processes.
– Suitable formalisms to express concepts as validly as possible, e.g. ternary, fuzzy, temporal, probabilistic, defeasible, and deontic logics.
– Tools for the formal specification of traffic rules within these formalisms, including the handling of complex interrelations, such as priorities.
– Verification technology demonstrating the system's adherence to traffic rules, which is suitable for the usage in a rigorous, ISO-compliant safety case.
– Means of ontology mapping for e.g. mapping simulator on legal ontologies.
– Traceability tooling that overarches all processes, where all artifacts concerning traffic rule compliance are represented and interlinked.

5 Conclusion

We presented a report of experience collected during the formalization of a non-trivial subset of the StVO. Based on this experience, we extrapolated the congruence problem and connected it to a typical development process. The formal framework of semiotic triangles was utilized to delineate which challenges arise when addressing congruence. Their practical relevancy was highlighted using the exemplary distance rule taken from the StVO. We finally sketched how some of these challenges may be addressed in future research and industrial efforts.

Acknowledgements We thank Tom Bienmüller and Tino Teige from *BTC Embedded Systems AG* for support and discussions during the formalization. This work was partially funded by the *Federal Ministry of Education and Research* (BMBF) as part of *MANNHEIM-AutoDevSafeOps* (reference no. 01IS22087C).

References

1. Athan, T., Boley, H., Governatori, G., Palmirani, M., Paschke, A., Wyner, A.: OASIS LegalRuleML. In: Proc. 14th Int. Conf. on Artificial Intelligence and Law (ICAIL 2013). pp. 3–12 (2013)
2. Baader, F., Calvanese, D., McGuinness, D., Patel-Schneider, P., Nardi, D., et al.: The description logic handbook: Theory, implementation and applications. Cambridge university press (2003)
3. Bhuiyan, H., Olivieri, F., Governatori, G., Islam, M.B., Bond, A., Rakotonirainy, A.: A methodology for encoding regulatory rules. In: MIREL@ JURIX (2019)
4. Buechel, M., Hinz, G., Ruehl, F., Schroth, H., Gyoeri, C., Knoll, A.: Ontology-based traffic scene modeling, traffic regulations dependent situational awareness and decision-making for automated vehicles. In: 2017 IEEE Intelligent Vehicles Symposium (IV). pp. 1471–1476. IEEE (2017)
5. Costescu, D.M.: Keeping the autonomous vehicles accountable: Legal and logic analysis on traffic code. In: Conference Vision Zero for Sustainable Road Safety in Baltic Sea Region. pp. 21–33. Springer (2018)
6. Den Haan, N., Breuker, J.: A tractable juridical KBS for applying and teaching traffic regulations. Legal knowledge-based systems. JURIX **91**, 5–16 (1991)

7. Esterle, K.: Formalizing and Modeling Traffic Rules Within Interactive Behavior Planning. Ph.D. thesis, Universität München (2021)
8. Gansch, R., Adee, A.: System theoretic view on uncertainties. In: 2020 Design, Automation & Test in Europe Conference & Exhibition (DATE). pp. 1345–1350. IEEE (2020)
9. International Organization for Standardization: ISO 26262: Road vehicles – Functional safety. Standard, Geneva, Switzerland (2018)
10. International Organization for Standardization: ISO 21448: Road vehicles – Safety of the intended functionality. Standard, Geneva, Switzerland (2022)
11. Karimi, A., Duggirala, P.S.: Formalizing traffic rules for uncontrolled intersections. In: 2020 ACM/IEEE 11th International Conference on Cyber-Physical Systems (ICCPS). pp. 41–50. IEEE (2020)
12. Maierhofer, S., Moosbrugger, P., Althoff, M.: Formalization of intersection traffic rules in temporal logic. In: 2022 IEEE Intelligent Vehicles Symposium (IV). pp. 1135–1144 (2022)
13. Maierhofer, S., Rettinger, A.K., Mayer, E.C., Althoff, M.: Formalization of interstate traffic rules in temporal logic. In: 2020 IEEE Intelligent Vehicles Symposium (IV). pp. 752–759. IEEE (2020)
14. Neurohr, C., Westhofen, L., Henning, T., de Graaff, T., Möhlmann, E., Böde, E.: Fundamental considerations around scenario-based testing for automated driving. In: 2020 IEEE Intelligent Vehicles Symposium (IV). pp. 121–127. IEEE (2020)
15. Ogden, C.K., Richards, I.A.: The meaning of meaning: A study of the influence of thought and of the science of symbolism. Harcourt, Brace & World, Inc. (1923)
16. Riemer, N.: Meaning in the empirical study of language, p. 1–44. Cambridge Introductions to Language and Linguistics, Cambridge University Press (2010)
17. Rizaldi, A., Keinholz, J., Huber, M., Feldle, J., Immler, F., Althoff, M., Hilgendorf, E., Nipkow, T.: Formalising and monitoring traffic rules for autonomous vehicles in Isabelle/HOL. In: International conference on integrated formal methods. pp. 50–66. Springer (2017)
18. Salem, N.F., Haber, V., Rauschenbach, M., Nolte, M., Reich, J., Stolte, T., Graubohm, R., Maurer, M.: Ein Beitrag zur durchgängigen, formalen Verhaltensspezifikation automatisierter Straßenfahrzeuge. preprint arXiv:2209.07204 (2022)
19. Schwammberger, M., Alves, G.V.: Extending urban multi-lane spatial logic to formalise road junction rules. In: Farrell, M., Luckcuck, M. (eds.) Proceedings Third Workshop on Formal Methods for Autonomous Systems, FMAS 2021, Virtual, October 21-22, 2021. EPTCS, vol. 348, pp. 1–19 (2021)
20. Sergot, M.J., Sadri, F., Kowalski, R.A., Kriwaczek, F., Hammond, P., Cory, H.T.: The british nationality act as a logic program. Communications of the ACM **29**(5), 370–386 (1986)
21. Teige, T., Bienmüller, T., Holberg, H.J.: Universal pattern: Formalization, testing, coverage, verification, and test case generation for safety-critical requirements. In: Workshop Methoden und Beschreibungssprachen zur Modellierung und Verifikation von Schaltungen und Systemen. Albert-Ludwigs-Universität Freiburg (2016)
22. UNECE, Geneva, Switzerland: UN Regulation No. 157: Uniform provisions concerning the approval of vehicles with regard to ALKS (2022)
23. Villata, S., et al.: Traffic rules encoding using defeasible deontic logic. In: Legal Knowledge and Information Systems: JURIX 2020: 33rd Annual Conference. vol. 334, p. 3. IOS Press (2020)