



Driving Decision Making of Autonomous Vehicle According to Queensland Overtaking Traffic Rules

Hanif Bhuiyan^{1,2} · Guido Governatori³ · Andry Rakotonirainy² · Meng Weng Wong³ · Avishkar Mahajan³

Received: 3 March 2023 / Accepted: 1 August 2023 / Published online: 12 September 2023
© The Author(s) 2023

Abstract

Improving the safety of autonomous vehicles (AVs) by making driving decisions in accordance with traffic rules is a complex task. Traffic rules are often expressed in a way that allows for interpretation and exceptions, making it difficult for AVs to follow them. This paper proposes a novel methodology for driving decision making in AVs based on defeasible deontic logic (DDL). We use DDL to formalize traffic rules and facilitate automated reasoning, allowing for the effective handling of rule exceptions and the resolution of vague terms in rules. To supplement the information provided by traffic rules, we incorporate an ontology for AV driving behaviour and environment information. By applying automated reasoning to formalized traffic rules and ontology-based AV driving information, our methodology enables AVs to make driving decisions in accordance with traffic rules. We present a case study focussing on the overtaking traffic rule to illustrate the usefulness of our methodology. Our evaluation demonstrates the effectiveness of the proposed driving decision-making methodology, highlighting its potential to improve the safety of AVs on the road.

Keywords Autonomous vehicle · Driving decision · Overtaking · Defeasible deontic logic

1 Introduction

Over the last few decades, intelligent systems have been a widely accepted technology with various degrees of interaction. Whilst this technology has positive effects, there are also negative impacts. For instance, although we recognize the necessity

and benefits of advanced vehicle technology for society, road crashes remain a significant global public health concern due to the increasing number of fatalities and disabilities. Every day, more than 3,700¹ people worldwide die as a result of road crashes, with driver behaviour accounting for 90% of these incidents. In Australia, over 12,000² people died in road crashes from 2011 to 2020, with an average of 58 deaths per year in Queensland due to high speed from 2013 to 2017.³ Adhering to traffic rules can decrease the likelihood of fatalities and injuries.

Introducing Autonomous Vehicles (AVs) that can automatically make driving decisions according to traffic rules can address traffic violations [1–4]. However, it is challenging for AVs to adhere to traffic rules whilst making driving decisions. The existing regulatory framework does not clearly address how AVs fit in it [5]. The current traffic regulatory framework is made for humans. According to Leenes and Lucivero [6], the current traffic rule model for AVs may be incomplete for some scenarios on the road. For instance, the current traffic rules contain some vague terms (e.g., “can safely overtake,” “overtake when there is a clear view,” etc.) that are difficult for AVs to follow [7]. Furthermore, AVs may be unable to correctly obey the rules governing exceptions [7]. To address this issue, this paper proposes an automatic driving decision methodology to resolve the issues mentioned above.

This paper aims to develop a methodology to make driving decisions for autonomous vehicles (AVs) according to Queensland’s overtaking traffic rules. Overtaking traffic rules is particularly difficult, given their complex and diverse conditions. To effectively manage exceptions and vague terms, we applied defeasible deontic logic (DDL) [8] to formalize Queensland’s overtaking traffic rules. The reason behind to use DDL is that it has been successfully applied to formalizing normative frameworks in different domains and applications [9, 10], for a detailed overview of DDL in the legal domain see [11]. Witt, et al. [10] report very high (syntactic) variability of formalizing rules when they are done by a team of coders; moreover, they report that adopting a common naming convention and sharing a formalizing methodology greatly increase the agreement among the formalizing by the different coders. Also, fitting textual provisions in the patterns allows us to identify expressions that could have different syntactic structures but the same semantic meaning; such expressions will be formalized by the same atoms.

Our contributions to this work are as follows:

- We formalized the Queensland overtaking traffic rules using DDL.
- We verified the formalization of rules for AVs and formulated AV driving information into the machine-computable format provided by the simulator. We created an ontology knowledge base to make the machine-computable format of AV information.

¹ https://www.who.int/violence_injury_prevention/road_traffic/en/.

² <https://www.bitre.gov.au/statistics/safety>.

³ <https://streetsmarts.initiatives.qld.gov.au/speeding/factsheet>.

- Finally, we designed a reasoning engine to make driving decisions based on traffic rules. This reasoning engine requires the machine-computable format of AV driving information (ontology) and formalized traffic rules.

The proposed driving decision methodology is evaluated by an empirical experiment where the legality of overtaking manoeuvres by an AV is evaluated by humans (drivers and domain specialists). The experiment gives us a way to determine if the ‘vague terms’ and ‘exceptions’ were properly captured and handled. It is hypothesized that the proposed driving decision methodology will be effective if its outcome is close to the human legal perception. Two types of cases (clear and borderline) were designed to evaluate the methodology performance by covering all aspects of situations that humans face in real life whilst taking legal action. Humans can judge clear cases directly, but it is challenging for them to judge borderline cases. Ideally, in clear cases, a human could approach 100% accuracy, and in borderline cases, the accuracy should be close to 50%.

According to the experiment outcome, around 85% (clear cases) and 59% (borderline cases) degree of agreement is shown between humans and the proposed driving decision methodology. Based on this reasonable degree of similarity, it can be stated that the proposed methodology is promising for making legal driving decisions for Autonomous Vehicles regarding the current Queensland traffic rules.

2 Related Work

Autonomous vehicles (AVs) may soon be introduced into the current transport system. However, to achieve this aim, there are several challenges to resolve. Among them, one of the critical issues is how AV will make driving decisions according to the current traffic rules [7, 12–15]. To address this, AV development has to ensure that AV can make driving decisions according to the current traffic rules. Without this, safe AV operation cannot be confirmed in the current transport system [3, 16].

Traffic regulation consists of a set of rules that primarily aim to improve road safety by regulating the behaviour of road entities such as pedestrians, vehicles, bicycles, etc. Some fundamental traffic rules, such as “follow the speed limit,” “overtake when safe,” and “always keep clear intersection,” appear to be simple enough. Still, numerous subtle meanings and exceptions to these rules necessitate situational (surrounding and environmental) awareness and even (value) judgments on the driver’s part. Integrating such traffic rules into AV is challenging and requires an appropriate reasoning framework, including proper knowledge representation and an intelligent algorithm. Until now, various research efforts have focussed on driving decision making for AVs according to traffic rules. These studies are discussed in further detail below.

Rizaldi, et al. [7] used the Isabelle logic theorem to create a formal representation of traffic regulations that can be used to observe AV behaviour and confirm compliance with these regulations. The rules are transformed into Linear Temporal Logic (LTL) using High Order Logic (HOL), and a verified checker is applied

to validate the AV's compliance with the formalized rules. Data are assessed using discrete-time runs.

Zhang, et al. [15] proposed an expert system to formalize traffic rules to develop the driving knowledge base, which can play a significant role in making intelligent driving decisions for AV. A knowledge acquisition method (knowledge hypergraph) was applied to build traffic rules knowledge base. This method defined traffic rules concepts, quantitatively explained rule characteristics, and created a logical relation between rule terms, and thus accomplished the formalization based on the standardization representation of rules.

Morignot and Nashashibi [17] introduced an ontology-based decision-making methodology to ease traffic restrictions, such as crossing solid lanes on the road when required for safety. The suggested ontology depicts the AV and other vehicle information, the traffic infrastructure, and the traffic rules. The methodology illustrated how SWRL (Semantic Web Rule Language) rules could formalize traffic rules to make the driving decision. However, this proposed methodology was confined to one traffic scenario.

Another study by Zhao, et al. [18] introduced an ontology model for making a fast decision according to traffic rules at the intersection. Ontologies in this proposed method represent knowledge of the sensory data. Traffic rules were represented (encoded) through SWRL rules. However, the methodology was limited to working only on specific traffic rules.

Buechel, et al. [19] proposed a framework for traffic scenario modelling and decision making in uncontrolled intersections for AVs. This framework utilizes a traffic situation ontology (e.g., lane 1, lane 2, lane 3, lane 4) and traffic rules to improve the AV's situational awareness. Decision-making rules are derived from the traffic situation ontology and traffic rules.

Shadrin, et al. [20] designed an expert system to establish a formal representation of traffic regulations in specific scenarios to manage AVs. This system includes data processing algorithms, multidimensional databases, and a cognitive model of traffic objects and their interactions. The traffic rules are categorized into two sets, with one including traffic lights, road markings, and road signs and the other comprising approximately 800 traffic rules.

The Mivar system introduced in [21] can track vehicle actions in real-time and notify the driver about breaches of traffic regulations. The system is composed of three primary modules: a trajectory control system (which controls lane position, safe distance from other vehicles, etc.), a technical vision system (which provides a real-time view of the road situation), and a decision support system (DSS) that analyzes data received from other assisted devices. Algorithms are created based on this information to oversee the driver's compliance with traffic rules.

Alves et al. [22] introduced an intelligent agent-based architecture to model AV behaviour and thus make an ethical decision if required to violate traffic rules. This architecture consists of an ethical agent, which ensures the safety of road entities by monitoring AV behaviour. Traffic rule was formalized using the GWENDOLEN agent programming language. However, the proposed architecture was limited to specific traffic scenarios.

McLachlan et al. [23] proposed a method for deconstructing the traffic rule and representing it with the necessary needed information for the decision making of autonomous driving. Using legal vocabulary, the technique deconstructs road rules, which are then specified in structured English logic, Boolean Logic. The key points of this method are that the chronology of operation and processes of rules were represented through Boolean logic. The evaluation of this rule representation was conducted on 23 UK road rules. However, this methodology of deconstructing traffic rules works only for simple traffic laws, such as traffic lights, seat belt wearing, speed limit etc.

None of these researches addresses traffic rule vagueness and exceptions together, which can present challenges when making driving decisions. Our proposed approach addresses these issues by utilizing a DDL-based traffic rules formalizing mechanism that enables effective handling of exceptions and resolution of rule vagueness, thereby enabling AVs to make appropriate driving decisions. Moreover, none of the previous studies evaluated their proposed approach by comparing the results (reasoning outcome) with human evaluation. In this research, a robust human evaluation, including domain specialists (driver trainers), is conducted to evaluate the proposed driving decision methodology. This evaluation (qualitative and quantitative) has proved this proposed driving decision methodology's effectiveness.

3 Driving Decision Methodology

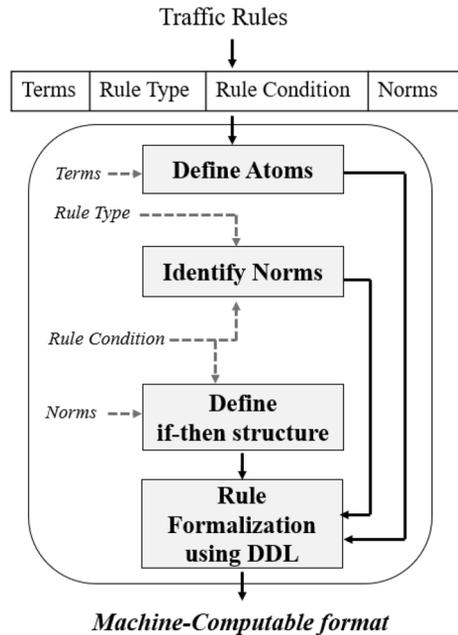
The proposed approach involves three interrelated modules that collaborate to allow autonomous vehicles to make driving decisions based on traffic rules. The first module concerns transforming traffic rules into a format that can be interpreted by machines. The second module concentrates on structuring AV information in a manner that is consistent with the formatted traffic rules. Lastly, the third module involves mapping and reasoning the traffic rules and AV data to establish the optimal driving decision. Each module is briefly explained below.

3.1 Traffic Rules Formalization

In this section, we present our methodology for formalizing traffic rules in a machine-computable format using DDL as its formal foundation. The proposed methodology consists of four steps, illustrated in Fig. 1: define atoms, identify norms, define if-then structures, and formalizing rules using DDL. These steps are done manually. The methodology's input is a set of traffic rules in natural language.

Define Atoms: To translate traffic rules into a machine-computable format, we start by defining atoms that are composed of terms that can be evaluated as either true or false. These atoms are based on the terms used in the rules, including the subject (s), predicate (p), property (pr), object (o), and qualifier (q) in the rule sentence. However, the structure of traffic rules can be heterogeneous, which creates variations in the atom structure. To address this issue, an empirical study of the Queensland Traffic Rules was conducted, and atoms were semantically defined

Fig. 1 Workflow for traffic rules formalization



based on five aspects: Subject-Predicate-Object, Subject-Predicate-Qualifier-Object, Subject-Property, Subject-Predicate-Object-Object, and Subject-Qualifier-Predicate-Object. For instance, the Queensland Traffic Rule, part 11 division 3 rule 142(1).b, states that “the vehicle is giving a right change of directional signal”. By applying our proposed method of defining atoms, we can represent this rule as S-P-O (Subject-Predicate-Object): vehicle_IsGiving_rightChangeOfDirectionSignal.

Atom vehicle_IsGiving_rightChangeOfDirectionSignal.

Identify Norms: Norms are the set of conditions that dictate the particular actions in rules, defining conditional terms and concepts within the rules. Each norm is represented by one or more rules that can be either constitutive or prescriptive. Constitutive rules establish the terms that are specific to legal documents, whereas prescriptive rules indicate the “mode” of behaviour using deontic modalities such as obligation, permission, and prohibition. Obligation mandates the subject to perform a particular action, whilst prohibition prohibits the subject from performing an action. Permission denotes the state of an action where the subject has no obligation or prohibition to act. The study at hand identifies norms based on both constitutive and prescriptive rules. For example, in the Queensland Traffic Rules, part 11, division 3, rule 141 states that “A driver must not overtake to the right of a vehicle if the vehicle is....”. Here “must not” (prohibition) is identified as a prescriptive norm within this statement. As an instance of constitutive rule, in Queensland Traffic Rule, part 11 division 3, rule 151A (5)

states: “end no filtering sign means a traffic sign”. Here, we can identify the norm “means”, which defines what is “end no filtering sign”.

Define if–then structure Traffic rules prescribe specific actions and behaviours for the subject to follow. Each rule consists of two parts: an antecedent or premise and a consequent or conclusion. Logical operators such as OR, AND, XOR, and NOR can be used to join multiple antecedents. For instance, Queensland overtaking rule 143 (1.a) states that, “A driver must not drive past, or overtake, to the left of a vehicle displaying a do not overtake turning vehicle sign, unless—if the vehicle is turning left and is giving a left change of direction signal”. We can represent the if–then structure of this rule by defining the atoms and specifying the norms as follows:

```

IF
    %% displaying a do not overtake turning vehicle sign %%
    vehicle_Display_doNotOvertakeTurningVehicleSign // atom
AND
    %% if the vehicle is turning left %%
    vehicle_IsTurningLeft // atom
AND
    %% is giving a left change of direction signal %%
    vehicle_IsGivingLeftChangeOfDirectionSignal // atom
THEN [OBL] // norms
    %% A driver must not drive past to the left of a vehicle %%
    NEG (driver_DrivePastToTheLeftOf_vehicle)

```

Rule Formalization using DDL: We consider the traffic system as a normative system composed of a set of clauses or norms to transform traffic rules into a specific format (interpretable by machine) using DDL. One or more rules express each norm in the form of “if...then” statements, where the antecedent describes the conditions of applicability, and the consequent describes the normative effect. For example:

$$\alpha_1, \dots, \alpha_n \Rightarrow \beta; \quad \text{where } \alpha_1, \dots, \alpha_n \text{ are the conditions and } \beta \text{ is effect.}$$

We categorize norms/rules as either constitutive or prescriptive. Constitutive rules establish the terminology used in the normative system, whereas prescriptive rules decide the normative effects that are enforced based on the applicable conditions. The normative effects are conveyed through deontic modalities like Obligation (O), Prohibition (F), and Permission (P). These modalities are modal operators. When a modal operator is applied to a statement, it transforms it into a new proposition by defining the “truth” of the original proposition. According to QLD Overtaking Traffic Rule 143 (2), the proposition: “driver_DrivePastToTheRightOf_vehicle”

signifies that the driver is passing a vehicle from its right side. Using the aforementioned deontic operators, we may differentiate the following claim:

- *PassRight/OvertakeRight* This is a factual statement that is true if the following vehicle approaches/overtakes from the right and false in all other circumstances (\neg *PassRight/OvertakeRight* is true).
- [O]*PassRight/OvertakeRight* This is a deontic expression, meaning that the following vehicle must pass/overtake on the right of the leading vehicle. The statement is true if the requirement to pass is in effect in a specific situation.
- [F] *PassRight/OvertakeRight* This is a deontic expression, meaning that the following vehicle is not allowed to pass/overtake on the right. The statement is true if the prohibition against passing is in effect in a specific situation.
- [P] *PassRight/OvertakeRight* This is a deontic expression, meaning the following vehicle can pass/overtake on the right. If the right to pass is given in a certain situation, the statement can be judged to be true.

We adopt the standard deontic logic relationships between these ([O], [P], [F]) deontic modalities, which can be exemplified as follows (taking the example of the “pass/overtake” atom):

$$\begin{aligned} [F] \text{ pass/overtake} &\equiv [O] \neg \text{pass/overtake} \\ [O] \text{ pass/overtake} &\equiv [F] \neg \text{pass/overtake} \\ [P] \text{ pass/overtake} &\equiv \neg [O] \neg \text{pass/overtake} \end{aligned}$$

A complete example of formalizing traffic rules using DDL is shown in Fig. 2. We used the Queensland (QLD) Passing or Overtaking Traffic Rules 143 (2) for this example.

QLD Traffic Rule 143: Passing or overtaking a vehicle displaying a do not overtake turning vehicle sign (2) A driver must not drive past, or overtake, to the right of a vehicle displaying a do not overtake turning vehicle sign if the vehicle is turning right, or making a U-turn from the centre of the road, and is giving a right change of direction signal, unless it is safe to do so.

Formalization of Traffic Rule 143 (2): The machine-computable format of Traffic Rule 143(2) is presented in Fig. 2, where we have combined atoms, norms, and if-then structures using the mechanisms described earlier and thus formalized the rules. Each rule is assigned a unique identifier, such as r143_2_i, r143_2_ii, etc. The priority of the formalized rules is illustrated at the bottom of this figure, where they are ordered according to the priority of traffic rules.

3.2 Ontology Knowledge Base

Ontologies are a structured approach to organizing knowledge, comprising concepts (classes) and relationships (properties) arranged hierarchically within a specific knowledge domain. They offer a comprehensive semantic network that enables seamless communication and information sharing between software and hardware agents by creating precise and comprehensive conceptual schemas.

Atom Initialisations:

Atom vehicle_IsTurningRight
Atom vehicle_IsMakingUturn
Atom vehicle_IsOn_centreOfRoad
Atom vehicle_IsStationary
Atom driver_IsDrivingOn_MultiLaneRoad
Atom vehicle_IsGivingRightChangeOfDirectionSignal
Atom vehicle_Display_doNotOvertakeTurningVehicleSign
Atom IsSafeToPassIn_MarkedLane
Atom IsSafeToOvertakeIn_MarkedLane
Atom IsSafeToPassToTheRightOf_vehicle
Atom driver_DrivePastToTheRightOf_vehicle
Atom driver_OvertakeToTheRightOf_vehicle

Rules:

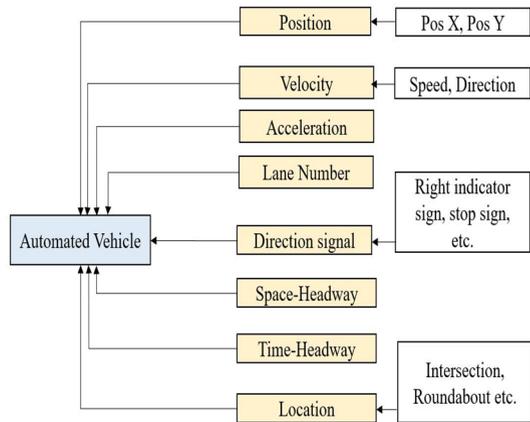
r143_2_i: vehicle_Display_doNotOvertakeTurningVehicleSign & vehicle_IsTurningRight
 & vehicle_IsGivingRightChangeOfDirectionSignal => [F] driver_DrivePastToTheRightOf_vehicle
 r143_2_ii: vehicle_Display_doNotOvertakeTurningVehicleSign & vehicle_IsTurningRight
 & vehicle_IsGivingRightChangeOfDirectionSignal => [F] driver_OvertakeToTheRightOf_vehicle
 r143_2_iii: vehicle_Display_doNotOvertakeTurningVehicleSign & vehicle_IsMakingUturn &
 vehicle_IsOn_centreOfRoad
 & vehicle_IsGivingRightChangeOfDirectionSignal => [F] driver_DrivePastToTheRightOf_vehicle
 r143_2_iv: vehicle_Display_doNotOvertakeTurningVehicleSign & vehicle_IsMakingUturn &
 vehicle_IsOn_centreOfRoad
 & vehicle_IsGivingRightChangeOfDirectionSignal => [F] driver_OvertakeToTheRightOf_vehicle
 r143_2_exception_i: vehicle_Display_doNotOvertakeTurningVehicleSign &
 IsSafeToPassToTheRightOf_vehicle & [P] driver_Overtake_vehicle => [P]
 driver_DrivePastToTheRightOf_vehicle
 r143_2_exception_ii: vehicle_Display_doNotOvertakeTurningVehicleSign &
 IsSafeToPassToTheRightOf_vehicle & [P] driver_Overtake_vehicle => [P]
 driver_OvertakeToTheRightOf_vehicle

 r_{143.2.exception.i} > r_{143.2.i}
 r_{143.2.exception.i} > r_{143.2.iii}
 r_{143.2.exception.ii} > r_{143.2.ii}
 r_{143.2.exception.ii} > r_{143.2.iv}

Fig. 2 Formalization of Queensland Overtaking Traffic Rule 143 (2)

The ability of an ontology to represent knowledge in machine-computable RDF (Resource Description Framework) data makes it a crucial feature as it defines the data model through conceptual statements [24]. This approach can close the gap between AV perception and knowledge processing, as demonstrated by previous studies [25, 26]. Thus, we utilize ontologies, a machine-computable knowledge base, in our work to represent AV information. Moreover, previous research shows that ontology is an effective tool for representing driving behaviour and road maps, which is significant for AV knowledge processing [25]. The machine-computable knowledge base is used as input to the reasoning engine, which

Fig. 3 A simple ontology design of AV behaviour information



analyzes the available data to determine the legal requirements for the AV in a particular situation.

The knowledge base is comprised of two ontologies: AV behaviour and AV environment ontology. The AV behaviour ontology is constructed using data on AV behaviour, such as acceleration, speed, direction signal, etc. A simple ontology design of AV behaviour is shown in Fig. 3. Meanwhile, the environment ontology is created using information about the road, such as road markings and type, as well as details about the surroundings of the AV, such as other vehicles. All of this data is gathered via the CARRS-Q advanced driving simulator.⁴ Moreover, ontologies are highly flexible and can be expanded effortlessly by incorporating new ideas, depending on the requirements, making them extremely adaptable. To create a road map in the simulator, we acquire road-related data from the QLD Transport and Main Roads website for Queensland, Australia.⁵

3.3 Reasoning

In this section, we will introduce the reasoning engine that enables AVs to make driving decisions based on traffic rules. The working flow diagram of the reasoning process is depicted in Fig. 4. The reasoning engine's input comprises atoms (derived from formalized traffic rules), formalized traffic rules, and a knowledge base, represented by a cloud shape in the diagram. The SPARQL Query Algorithm (Fig. 6) is triggered to extract relevant information from the knowledge base.

The reasoning engine (Fig. 4) works in four steps:

- i. The reasoning engine receives atoms and formalized rules from the traffic rule formalization section.

⁴ <https://research.qut.edu.au/carrsq/services/advanced-driving-simulator/>.

⁵ <https://www.tmr.qld.gov.au/Travel-and-transport/Maps-and-guides>.

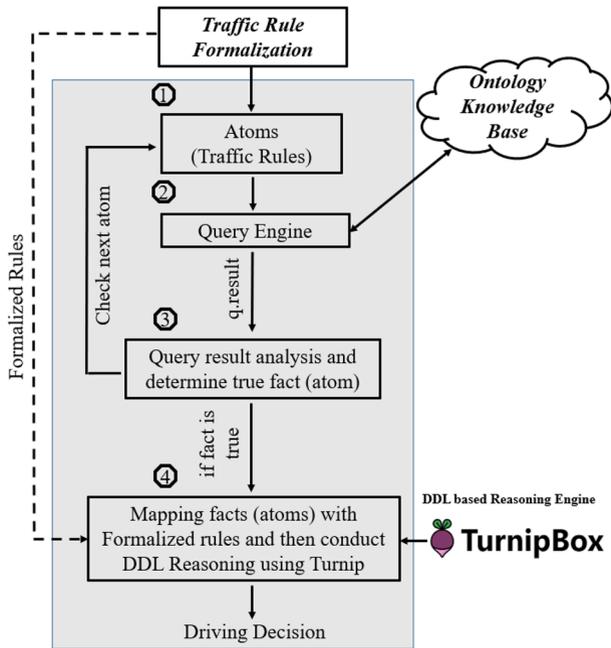


Fig. 4 Reasoning engine workflow

- ii. For each atom, the “query engine” retrieves corresponding information from the knowledge base (ontology) and sends it to the “determine true fact” step.
- iii. The “determine true fact” step analyses and verifies true atoms based on the received knowledge base information.
- iv. Selected rules and true atoms are sent to the DDL-based reasoner (TurninBox), and thus make driving decisions.

A brief illustration of these four steps (complete working procedure of the reasoning engine) is given below based on a case study: a long vehicle right overtaking manoeuvre (Fig. 5).

Working Procedure of Reasoning Engine (Case Study)

This section uses a case study (Fig. 5) to describe how the proposed driving decision methodology works. It is a 13.5 s (second) overtaking manoeuvre. The CARRS-Q advanced driving simulator is used to make this scenario. For every 0.05 s, the simulator provides information on these manoeuvres. This information includes all vehicles, roads, traffic information, etc. For this study, each 0.05-s interval is considered a timestamp.

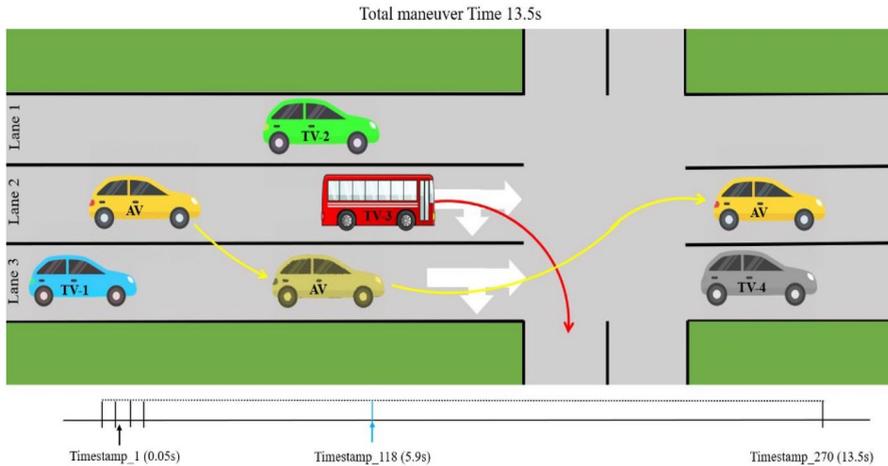


Fig. 5 A case study of a long vehicle right overtaking manoeuvre

In this scenario, the AV (yellow) plans to pass (overtake) the long red vehicle (TV-3, which indicates a right turn and displaying a “do not overtake turning vehicle” sign). Meanwhile, the green vehicle (TV-2) is moving in its lane, and the blue vehicle (TV-1) is also moving in its lane. AV must consider the blue vehicle (TV-1) whilst changing lanes to pass the long red vehicle. The Queensland (QLD) overtaking traffic rules specify that the AV must adhere to traffic rules 145, 144, 143 (2), 142, and 140 to overtake the red vehicle. According to the author understanding, the AV must deal with several vague terms and exceptions, such as “clear view of approaching vehicle,” “safe to do so,” and “must not overtake a vehicle to the left unless” to perform this manoeuvre.

We randomly selected a time slot (t_{118} : 18.09 s) to illustrate the reasoning behind the proposed driving decision-making methodology. The following section outlines the overall reasoning process employed for this particular time slot (t_{118} : 5.09 s), step-by-step.

1st Step (Atom): The Traffic Rules Formalization section outlines the fundamental components of Overtaking Traffic Rules 145, 144, 143 (2), 142, 140. Thirty-eight atoms were derived from these rules. Some atoms are shown in below Table 1.

2nd Step (Query Engine): The query engine is equipped with predefined SPARQL queries corresponding to each atom. These queries are generated manually based on an analysis of Queensland’s Overtaking traffic rules. The number of queries for each atom may differ. SPARQL is an efficient query language that facilitates access to ontology-based knowledge bases. To handle the ambiguity (vague terms, such as “safe distance”, “approaching traffic”, etc.) in rules, ontology and query methods were utilized to provide a deterministic and algorithmic approach. Queries were constructed using state-of-the-art techniques from traffic research to

Table 1 Atoms of QLD traffic rules 145, 144, 143 (right), 142, 140 (case study)

Atoms	
145	Atom driver_IsOvertaking_vehicle Atom driver_ReturnTo_markedLane
144	Atom driver_isDrivingOn_twoWayRoad Atom driver_HasPassed_anotherDriver
143 (right)	Atom vehicle_IsTurningRight Atom vehicle_IsMakingUturn Atom vehicle_IsOn_centreOfRoad Atom vehicle_IsStationary
142	Atom driver_OvertakeToTheRightOf_vehicle Atom vehicle_IsTurningRight Atom vehicle_IsGivingRightChangeOfDirectionSignal
140	Atom driver_HasClearViewOf_approachingTraffic Atom driver_CanSafelyOvertake_vehicle Atom driver_Overtake_vehicle

determine, for example, whether a given distance between two vehicles was safe. In this context, we utilize SPARQL queries to fetch AV behaviour and environmental information from the knowledge base. A simple example of a SPARQL query is shown below. SPARQL_Query_Algorithm (Fig. 6) is used to trigger these queries optimally.

Query: What are AV curvilinear coordinates in time (t)? (AV_Behaviour)

SPARQL:

```

prefix ab:<http://www.semanticweb.org/bhuiyanh/ontologies/2020/5/untitled-ontology-137#>
SELECT ?is_positionx ?is_positiony
WHERE
{
ab:vehicle-1 ab:is_positionx ?is_positionx.
ab:vehicle-1 ab:is_positiony ?is_positiony
}
    
```

Answer Example:

```

“285.551116943” ^^<http://www.w3.org/2001/XMLSchema#string>
“164.853302002” ^^<http://www.w3.org/2001/XMLSchema#string>
    
```

Algorithm 1: SPARQL Query Algorithm

Input: a (atom), Q (Queries), KB (Knowledge Base).
Output: Query Result.

```

1 Function SPARQL_Query_Algorithm(a, Q, KB):
2   Initialise:
   for atom a, making query set  $\mathbb{Q}=(q_1 \dots , q_n)$  using Q.
   if any query (q) has subquery then making subquery set  $\mathbb{S} = (s_1 \dots , s_n)$  using Q
   //for all of this initialization,  $n \in \mathbb{N}$ 
3   Function SQ_AV_B_OWL(a):
4     qb [] = a & Q // fetching queries for atom a from Q into qb
5     foreach q  $\in$  qb do
6       if q has no Subquery then
7         | Knowledge.Extract (q)
8       else
9         | Subquery (q)
10      end
11    end
12  End
13  Function SQ_AV_E_OWL(a):
14    qe [] = a & Q // fetching queries for atom a from Q into qe
15    foreach q  $\in$  qe do
16      if q has no Subquery then
17        | Knowledge.Extract (q)
18      else
19        | Subquery (q)
20      end
21    end
22  End
23  Function Subquery(q):
24    subq [] = q & S // fetching subqueries for query of q from S
    into subq
25    foreach s  $\in$  subq do
26      if s has no Subquery then
27        | Knowledge.Extract (s)
28      else
29        | Subquery (s)
30      end
31    end
32  End
33  Function Knowledge.Extract(q):
34    q  $\leftarrow$  KB
35    if q.result  $\neq$  NULL then
36      | return result
37    else
38      | break
39    end
40  End
41 End

```

Fig. 6 SPARQL_Query_Algorithm

There are three inputs for the SPARQL_Query_Algorithm: atom, queries, and Knowledge base. In line 2, all of the inputs are initialized into different sets. Lines 3–12 handle the queries for behaviour ontology. In line 4, qb fetches the queries for the atom. In line 6, it is checked whether there are any subqueries for the particular query? If the query has subqueries, then the Subquery function is called for that query (Line 9); otherwise, in line 7, the query is triggered in the knowledge base to retrieve information. Lines 13–22 handle queries for the environment ontology. In

line 14, it is checked whether the particular query has any subqueries or not. If the query has subqueries, then the Knowledge_Extract function is called (line 17); otherwise, the Subquery function is called (line 19). In lines 23–32, the Subquery function is executed. In the subquery function, it is checked whether the subquery has a nested subquery or not. If any subquery has a subquery, then the Subquery function is called recursively for that subquery. Lines 33–40 handle the knowledge retrieval from the knowledge base. In line 34, the query result assigns to the variable q. If the query result is NULL, then the process breaks and goes for the following query. Some examples of queries and their results are shown in Table 2.

3rd Step (Determining True Facts): This step involves determining the verifiable facts (atoms) that pertain to the driving action of the AV in the overtaking scenario depicted in Fig. 4. For each atom, data is retrieved using a query engine from the ontology knowledge base. For each atom, the query engine has pre-set SPARQL queries. The query results are subsequently compared with these answers, and based on the matching results, the atom is determined as true or false. For a simple instance, to verify the atom: “vehicle_IsOn_centreOfRoad”, two SPARQL Queries are triggered. Based on these queries answers, it is determined whether the

Table 2 Example of an atom, its corresponding queries and its verification conditions

<p>Atom vehicle_IsOn_centreOfRoad.</p>
<p>Query 1: How many lanes in the road? (AV_Environment)</p> <p>SPARQL:</p> <pre>prefix as:<http://www.semanticweb.org/bhuiyanh/ontologies/2020/5/untitled-ontology-139#> SELECT ?is_totallane WHERE { as:time as:is_totallane ?is_totallane. }</pre> <p>Answer Example: “3.0” ^^<http://www.w3.org/2001/XMLSchema#string></p>
<p>Query 2: What is the target vehicle (TV) Lane Number? (AV_Behaviour)</p> <p>SPARQL:</p> <pre>prefix ab:<http://www.semanticweb.org/bhuiyanh/ontologies/2020/5/untitled-ontology-137#> SELECT ?is_lanenumber WHERE { ab:vehicle-2 ab:is_lanenumber ?is_lanenumber }</pre> <p>Answer: “2.0” ^^<http://www.w3.org/2001/XMLSchema#string></p>
<p>Verification conditions for this Atom vehicle_IsOn_centreOfRoad.</p> <p>IF:</p> <p>TV lane number ≥ Floor (totallane / TV lane number) & TV lane number < totallane is True</p> <p>THEN: The atom is considered a True atom for the specific timestamp.</p> <p>ELSE: The atom is considered a False atom.</p>

atom is true/false. Atom, its corresponding queries and its verification condition is shown Table 2.

Some atoms are “black & white”, a simple query checking can verify whether the atom is true or false. For example, the “driver_IsDrivingOn_MultiLaneRoad” atom can be verified based on a query (“what is vehicle lane number?”) outcome. If the vehicle lane number is more than 1, the atom is considered true.

For some atoms, several queries and subqueries and their result required further computation to verify atoms. For instance, the atom “vehicle_CanBeSafelyOvertakenIn_markedLane” includes two queries, “Is AV in safe-distance?” and “Can AV safely change lane?”, both of which necessitate the identification of the AV’s safe distance and safe lane change. To address these queries, we reference the methodology presented in [27, 28], which provides a means of identifying safe lane changes and verifying safe distances for AVs. Once an AV has safely changed lanes, it must maintain a safe distance as specified by the safe distance condition.

The process of identifying which atoms are black and white and which need specialized queries is a manual one. Queries are manually written based on the available parameters (AV behaviours and corresponding rules) and the understanding of the meaning of the parameters.

Once all atoms are analyzed, the system determines the set of true atoms corresponding to the AV’s driving action in the particular time slot (t_118: 5.09 s). Some of the true atoms for this instance (case study: Fig. 5) are shown below.

- Atom** anotherDriver_IsDrivingOn_markedLane
- Atom** driver_HasReturnedTo_markedLane
- Atom** anotherDriver_IsDrivingOn_lineOfTraffic
- Atom** vehicle_IsGivingRightChangeOfDirectionSignal
- Atom** vehicle_IsTurningRight
-
-
-
- Atom** driver_IsDrivingOn_MultiLaneRoad
- Atom** vehicle_Display_doNotOvertakeTurningVehicleSign
- Atom** driver_CanSafelyOvertake_vehicle
- Atom** driver_OvertakeToTheRightOf_vehicle

4th step (mapping and reasoning in turnip): Turnip is a reasoning tool that operates on defeasible deontic logic. It takes facts (atoms), strict rules, defeasible rules, defeaters, superiority relations, and the modality of DL as input. A comprehensive overview of Turnip is beyond the scope of this study. We use Turnip to map and reason based on the formalized rules and atoms for our research purposes.

Table 3 Turnip reasoning for the case study

Facts (True Atoms)	Facts (True Atoms)
anotherDriver_IsDrivingOn_markedLane driver_HasReturnedTo_markedLane anotherDriver_IsDrivingOn_lineOfTraffic driver_OvertakeToTheRightOf_vehicle driver_CanSafelyOvertake_vehicle	anotherDriver_IsDrivingOn_markedLane driver_HasReturnedTo_markedLane anotherDriver_IsDrivingOn_lineOfTraffic driver_HasClearViewOf_approachingTraffic driver_OvertakeToTheRightOf_vehicle driver_CanSafelyOvertake_vehicle
Results	Results
[F] driver_OvertakeToTheRightOf_vehicle	[P] driver_OvertakeToTheRightOf_vehicle

In this step, the Turnip engine receives true facts (atoms) and formalized rules for the specific time slot (t_118: 5.09 s) of the overtaking case shown in Fig. 5. Turnip then proceeds to map and reason on these rules and facts, producing the result displayed in Table 3. The result of the reasoning indicates that, based on Queensland Overtaking Traffic Rules 145, 144, 143 (right), 142, and 140, the AV is prohibited ([F]) to overtake right, as there is no fact (atom) indicating that AV has a clear view of approaching traffic. However, if the AV had a clear view of approaching traffic (Table 3), then the AV had permission to overtake the vehicle from the right side.

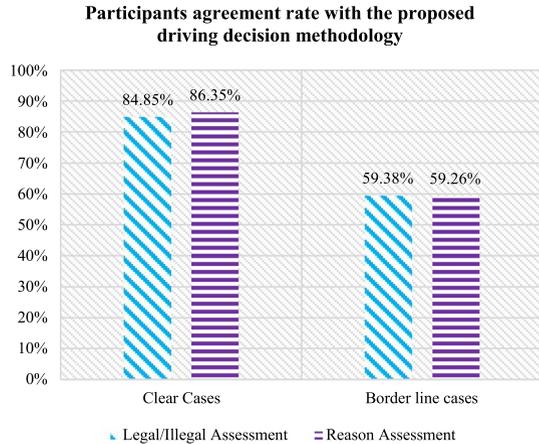
4 Experiment

The proposed methodology for making driving decisions was tested in a large-scale experiment where forty overtaking manoeuvres based on eight realistic Queensland overtaking traffic scenarios were evaluated. A set of five different overtaking manoeuvres was created for each overtaking traffic scenario. Two of these five manoeuvres were clearly legal or illegal, whilst the other three were more ambiguous and difficult to classify. Parameters for the ambiguous situations were set near the calculated threshold, whilst values for the clear cases were significantly different. For instance, if a safe distance of 10 m was determined, distances of 9 or 11 m would be borderline, whilst distances of 1 or 20 m would be considered clear cases. The reason for considering these two types of cases is to evaluate the proposed driving methodology’s performance (effectiveness) in all aspects of driving, both easy and complicated.

Initially, the proposed methodology was used to assess these manoeuvres, and then participants, including both general drivers and domain specialists, were asked to evaluate them. The effectiveness of the methodology was then determined by comparing the participants’ evaluations with the proposed methodology’s evaluation. The experiment was conducted in two aspects:

1. determining if each manoeuvre was legal or illegal, and
2. if a manoeuvre was illegal, identify the reason behind it.

Fig. 7 Performance of the proposed driving decision methodology



Thirty-two participants of varying ages evaluated the forty overtaking manoeuvres with a legitimate Australian driver's license. Eight driver trainers and twenty-four regular drivers comprised these thirty-two participants. No consideration was given to the participants' gender at any point. Since driver trainers need to know more about acceptable (lawful) vehicle passing techniques, they took part in this assessment as domain specialists. The study team personally sought out domain specialists via their driving schools and in-person General drivers. There were three different categories of general drivers as participants, which were:

- Experienced driver (+ 10 years of driving experience),
- Average experienced driver (5–10 years of driving experience) and
- Inexperienced driver (under five years of driving experience).

Each domain specialist evaluated twenty manoeuvres. They assessed ten manoeuvres in stage 1, then had a 10-min rest before completing another 10 manoeuvre assessments in stage 2. Each general participant reviewed ten manoeuvre videos. The participant was given manoeuvres based on randomly selected scenarios. Each participant completed the evaluation on their own. Participants were requested to refrain from talking about anything else but the examination.

Ten participants—six general participants and four domain specialists—evaluated each manoeuvre. They assess the manoeuvres from a set of video footage presented by a researcher from a computer screen. Participants double-checked each video and determined the manoeuvre's legality based on their familiarity with Queensland traffic rules. If they determined that a manoeuvre was illegal, they gave justification (reasons). The evaluation's outcome was written down on paper. After finishing all assessments, the researcher performed the required post-processing and analysis, and then recorded the findings. The effectiveness (performance) of the driving decision methodology is evaluated in light of this assessment.

Figure 7 illustrates the effectiveness of the driving decision methodology. The figure compares the agreement rates between participants and the methodology for

clear and borderline overtaking manoeuvres. The results show that, on average, for clear overtaking manoeuvres, there is an agreement of 84% and 86% on the legal/illegal assessment and reason identification, respectively, between the participants and the methodology. However, for borderline overtaking manoeuvres, the agreement rates for both legal/illegal decision and reason identification are almost the same at 59%. These borderline cases are intentionally created to test human perception of manoeuvres that are very close to the threshold between legal and illegal. The 50% agreement rate indicates that borderline cases are truly ambiguous. Based on the agreement rates for clear and borderline cases, it can be concluded that the proposed driving decision methodology for autonomous vehicles is promising.

After analyzing the assessment, it is seen that participants have different understandings of traffic rules. We do not have any information about why the participants made their choice. They provide their decision on what they believe and their perception (of manoeuvres seen in a computer simulation). The result might be more palpable if we could recruit participants by interviewing their knowledge of traffic rules and regulations. However, recruiting participants is a challenging task. It is difficult to find expected participants through interviews (examining). Moreover, assessment from video footage is a challenging task. Sometimes it might be hard to observe everything perfectly from video footage.

5 Conclusion and Future Work

This work demonstrates the feasibility of formalizing current traffic rules using Defeasible Deontic Logic (DDL) to enable Autonomous Vehicles (AVs) to follow traffic rules. By formalizing traffic rules, we can make driving decisions according to these rules and determine which rules require additional interpretation for AVs. Whilst some existing traffic rules require human interpretation, AVs need proper additional information to interpret these terms accurately. In this work, queries for each atom represent the necessary additional interpretation for traffic rules, allowing AVs to make accurate driving decisions according to traffic rules. This formalization mechanism could be valuable for technology contributors developing AVs and transport authorities to understand the interpretation of existing traffic rules for AVs.

We plan to experiment with the proposed driving decision methodology in the CARRS-Q level 4 AV in realistic test case scenarios. We also intend to expand the scope of this formalization mechanism to cover other traffic environments, such as lane changes, roundabouts, intersection crossings, and more.

The proposed driving decision methodology is not real-time. This work is simulation-based. For this work, we are not considering the computational performance. Based on the number of parameters (rules and AV behaviour), the computation time varies. In future, we plan to work in real-time.

Acknowledgements This research is supported by the National Research Foundation, Singapore under its Industry Alignment Fund – Pre-positioning (IAF-PP) Funding Initiative. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of National Research Foundation, Singapore.

Funding Open Access funding enabled and organized by CAUL and its Member Institutions.

Declarations

Conflict of interest On behalf of all authors, the corresponding author states that there is no conflict of interest.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Koopman, P., & Wagner, M. (2017). Autonomous vehicle safety: An interdisciplinary challenge. *IEEE Intelligent Transportation Systems Magazine*, 9(1), 90–96. <https://doi.org/10.1109/IMITS.2016.2583491>
2. Khorasani, G., Tatari, A., Yadollahi, A., & Rahimi, M. (2013). Evaluation of intelligent transport system in road safety. *International Journal of Chemical, Environmental and Biological Sciences (IJCEBS)*, 1(1), 110–118.
3. Bhuiyan, H., Governatori, G., Bond, A., & Rakotonirainy, A. (2023). Traffic rules compliance checking of automated vehicle maneuvers. *Artificial Intelligence and Law Journal*. <https://doi.org/10.1007/s10506-022-09340-9>
4. Leenes, R., & Lucivero, F. (2014). Laws on robots, laws by robots, laws in robots: Regulating robot behaviour by design. *Law, Innovation and Technology*, 6(2), 193–220.
5. Khorasani, G., Tatari, A., Yadollahi, A., & Rahimi, M. (2013). Evaluation of intelligent transport system in road safety. *International Journal of Chemical Environmental & Biological Sciences (IJCEBS)*, 1(1), 110–118.
6. Leenes, R., Lucivero, F. (2015). *Laws on robots, laws by robots, laws in robots: Regulating robot behaviour by design*. 2015, pp. 193–220.
7. Rizaldi, A., et al. (2017). Formalising and monitoring traffic rules for autonomous vehicles in isabelle/HOL. In *Integrated formal methods. Lecture notes in computer science* (vol. 10510, pp. 50–66). Springer. https://doi.org/10.1007/978-3-319-66845-1_4.
8. Governatori, G., Olivieri, F., Rotolo, A., & Scannapieco, S. (2013). Computing strong and weak permissions in defeasible logic. *Journal of Philosophical Logic*, 42(6), 799–829. <https://doi.org/10.1007/s10992-013-9295-1>
9. Governatori, G., Romeu, P.C., & de Koker, L. (2020). On the formal representation of the australian spent conviction scheme. In V. Gutiérrez-Basulto, T. Kliegr, A. Soyly, M. Giese, & D. Roman (Eds.), *Rules and reasoning* (pp. 177–185). Springer International Publishing.
10. Witt, A., Huggins, A., Governatori, G., & Buckley, J. (2021). Converting copyright legislation into machine-executable code: interpretation, coding validation and legal alignment. In *Presented at the 18th ICAIL, São Paulo, Brazil, 2021*. <https://doi.org/10.1145/3462757.3466083>.
11. Governatori, G., et al. (2021). Logic and the law: Philosophical foundations, deontics, and defeasible reasoning. *Handbook of Deontic Logic and Normative Reasoning*, 2, 655–760.
12. Collenette, J., Dennis, L. A., & Fisher, M. (2022). Advising autonomous cars about the rules of the road. [arXiv:2209.14035](https://arxiv.org/abs/2209.14035). <https://arxiv.org/abs/2209.14035>.
13. Prakken, H. (2017). On the problem of making autonomous vehicles conform to traffic law. *Artificial Intelligence and Law*, 25(3), 341–363. <https://doi.org/10.1007/s10506-017-9210-0>

14. Yu, W., Li, J., Peng, L.-M., Xiong, X., Yang, K., & Wang, H. (2022). SOTIF risk mitigation based on unified ODD monitoring for autonomous vehicles. *Journal of Intelligent and Connected Vehicles* (**no. ahead-of-print**).
15. Zhang, Z., et al. (2017). The visual representation and acquisition of driving knowledge for autonomous vehicle. *IOP Conference Series: Materials Science and Engineering*, 235, 012011. <https://doi.org/10.1088/1757-899x/235/1/012011>
16. Bhuiyan, H., Governatori, G., Islam, M. B., Bond, A., & Rakotonirainy, A. (2020). Traffic rules encoding using defeasible deontic logic. In *Legal knowledge and information systems: JURIX 2020: the thirty-third annual conference, 2020*. <https://doi.org/10.3233/FAIA200844>.
17. Morignot, P., & Nashashibi, F. (2012). An ontology-based approach to relax traffic regulation for autonomous vehicle assistance. <https://arxiv.org/ftp/arxiv/papers/1212/1212.0768.pdf>. Accessed 30 Dec 2022.
18. Zhao, L., Ichise, R., Sasaki, Y., Zheng, L., & Yoshikawa, T. (2016). Fast decision making using ontology-based knowledge base. In *Intelligent vehicles symposium (IV)* (pp. 173–178). <https://doi.org/10.1109/IVS.2016.7535382>.
19. Buechel, M., Hinz, G., Ruehl, F., Schroth, H., Gyoeri, C., & Knoll, A. (2017). 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). <https://doi.org/10.1109/IVS.2017.7995917>.
20. Shadrin, S. S., Varlamov, O. O., & Ivanov, A. M. (2017). Experimental autonomous road vehicle with logical artificial intelligence. *Journal of Advanced Transportation*, 2017, 2492765. <https://doi.org/10.1155/2017/2492765>
21. Aladin, D. V., Varlamov, O. O., Chuvikov, D. A., Chernenkiy, V. M., Smelkova, E. A., & Baldin, A. V. (2019). Logic-based artificial intelligence in systems for monitoring the enforcing traffic regulations. *IOP Conference Series: Materials Science and Engineering*, 534, 012025. <https://doi.org/10.1088/1757-899x/534/1/012025>
22. Alves, G. V., Dennis, L., & Fisher, M. (2021). An agent-based architecture with support to ethical decisions on a road traffic scenario. In *IROS workshop on building and evaluating ethical robotic systems 2021*. <https://personalpages.manchester.ac.uk/staff/louise.dennis/pubs/ADFers21.pdf>. Accessed 29 Dec 2022.
23. McLachlan, S., Neil, M., Dube, K., Bogani, R., Fenton, N., & Schaffer, B. (2021). Smart automotive technology adherence to the law:(De) constructing road rules for autonomous system development, verification and safety. *International Journal of Law and Information Technology*, 29(4), 255–295. <https://doi.org/10.1093/ijlit/eaac002>
24. Dean, A., & Jim, H. (2011). *Semantic web for the working ontologist: Effective modeling in RDFS and OWL*. Elsevier.
25. Zhao, L., Ichise, R., Mita, S., & Sasaki, Y. (2015). Ontologies for advanced driver assistance systems," 人工知能学会第二種研究会資料 (vol. 2015, no. SWO-035, p. 03). https://doi.org/10.11517/jsaisigtwo.2015.SWO-035_03.
26. Suryawanshi, Y., Qiu, H., Ayara, A., & Glimm, B. (2019). An ontological model for map data in automotive systems. In *Second international conference on artificial intelligence and knowledge engineering (AIKE)* (pp. 140–147). <https://doi.org/10.1109/AIKE.2019.00034>.
27. Rizaldi, A., Immler, F., & Althoff, M. (2016). A formally verified checker of the safe distance traffic rules for autonomous vehicles. In *NASA formal methods. lecture notes in computer science* (vol. 9690, pp. 175–190). Springer. https://doi.org/10.1007/978-3-319-40648-0_14.
28. Pek, C., Zahn, P., & Althoff, M. (2017). Verifying the safety of lane change maneuvers of self-driving vehicles based on formalized traffic rules. In *IEEE intelligent vehicles symposium (IV)* (pp. 1477–1483). <https://doi.org/10.1109/IVS.2017.7995918>.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Authors and Affiliations

Hanif Bhuiyan^{1,2}  · **Guido Governatori**³  · **Andry Rakotonirainy**²  ·
Meng Weng Wong³  · **Avishkar Mahajan**³ 

✉ Hanif Bhuiyan
hanif.bhuiyan@monash.edu
<https://research.monash.edu/en/persons/hanif-bhuiyan>

Guido Governatori
gvdgdo@gmail.com
<https://governatori.net/>

Andry Rakotonirainy
r.andry@qut.edu.au
<https://research.qut.edu.au/carrsq/staff/andry-rakotonirainy/>

Meng Weng Wong
mwwong@smu.edu.sg
<https://cclaw.smu.edu.sg/people/wong-meng-weng>

Avishkar Mahajan
avishkarm@smu.edu.sg
<https://cclaw.smu.edu.sg/people/wong-meng-weng>

¹ Monash University, Monash Data Futures Institute, Victoria, Australia

² Queensland University of Technology, CARRS-Q, Queensland, Australia

³ Centre for Computational Law, Singapore Management University, Singapore, Singapore