An Ontology to Classify Marine Encounters According to COLREG

Nicola Sabatino[1,*], Daniele Porello[2,†] and Raphael Zaccone[1,†]

[1]*Dipartimento di Ingegneria Navale, Elettrica, Elettronica e delle Telecomunicazioni, Università degli Studi di Genova, Italy*
[2]*Dipartimento di Antichità, Filosofia, Storia, Geografia, Università degli Studi di Genova, Italy*

Abstract

In recent years, there has been a rapid increase in interest in computer-governed vessels, commonly referred to as Maritime Autonomous Surface Ships (MASSs), within the global maritime cluster. As MASSs are expected to enhance operational efficiency at sea and reduce the need for human operators in hazardous working environments, both industry and academia are investing an increasing amount of resources into this emerging technology. The first prototypes of computer-governed vessels are already operating at sea. The development and construction of MASSs present challenges, with navigation safety being a key concern. As they operate alongside crewed vessels, ensuring they function safely and pose no threat to seafarers, yachtsmen, or passengers is essential. The best practices for the safe operation of vessels are defined by the *Convention on the International Regulations for Preventing Collisions at Sea* (COLREG), making it an ideal template for a MASS operating system. However, COLREG rules are designed for human interpretation and cannot be directly processed by a computer agent, with some containing ambiguous provisions. To address these issues, we present an ontology for the key COLREG rules based on an top-level ontology (DOLCE), to enhance semantic clarity and interoperability. The aim is twofold: to provide a common framework for interpreting navigation data in relation to COLREG, and to establish a knowledge base that classifies marine encounter scenarios. As a result, we show that ontological reasoning can accurately classify maritime encounters and suggest the required behaviour of vessels according to COLREG.

Keywords

Maritime Autonomous Surface Ship, COLREG, Collision Avoidance, Ontology Web Language, Top-level Ontology, Descriptive Ontology for Linguistic and Cognitive Engineering

1. Introduction

The worldwide maritime sector has shown a growing interest in computer-governed vessels in recent years. The International Maritime Organization (IMO) [1], i.e., the regulatory authority of the United Nations (UN) for the global maritime industry, has published several documents to guide the future development of computer-governed ships, referred to by the IMO as Maritime Autonomous Surface Ships (MASSs), cf. [1], [2], [3]. Following these regulatory works, many actors in the global maritime industry have started the trials of MASS prototypes, cf. [4]. Although the first MASS prototypes are starting trials at sea, there are many challenges still to overcome before they can enter full commercial use, one of the most important being collision avoidance, cf. [5].

The collisions at sea have always been a severe threat to the safety of navigation and still represent a fair share of all the accidents occurring at sea, cf. [6]. For this reason, in 1972, the IMO published the Convention on the International Regulations for Preventing Collisions (COLREG), cf. [7], entering into force in 1977, a collection of mandatory rules describing all the actions and measures that shall be taken to avoid collisions between vessels. For this reason, the COLREG makes an ideal template for the development of MASS's operative systems.

Proceedings of FOIS 2025 Satellite events co-located with the 15th International Conference on Formal Ontology in Information Systems (FOIS 2025), September 10-12, 2025, Catania, Italy
[*]Corresponding author.
[†]These authors contributed equally.
nicola.sabatino@edu.unige.it (N. Sabatino); daniele.porello@unige.it (D. Porello); raphael.zaccone@unige.it (R. Zaccone)
0009-0009-0147-8843 (N. Sabatino); 0000-0003-3655-0218 (D. Porello); 0000-0002-5489-4498 (R. Zaccone)

[1]see https://www.imo.org/

The characteristics of COLREG in relation to MASS research have inspired some work in very recent years related to the concept of COLREG classification, i.e., the ability to classify a marine encounter situation into one of the situation categories defined by the regulation. The most basic form of COLREG classification can be easily achieved via simple consideration of the relative courses of two vessels, relying on linear algebra. Such simple approaches can be applied effectively to detecting and classifying collisions from real vessel navigation data, cf. [8], or as part of collision avoidance algorithms, cf. [9]. Moreover, linear algebra-based classification methods can be extended to temporal awareness by using predictive models, cf. [10]. More sophisticated COLREG classification approaches have also been proposed, e.g., Discrete-Event Systems theory, cf. [11]. A number of recent articles make use of semantic technologies to model related aspects, such as navigation rules [12, 13], communication about marine encounters [14], and ship behaviour [15].

In this paper, we design an ontology of COLREG[2] to enable accurate automatic classification of encounter scenarios and generation of COLREG-compliant prescriptions. Once the data of the vessels are provided, ontological reasoning allows the classification of the correct scenario, which triggers the relevant prescription for the course of action. With respect to the current literature, this paper provides the following contribution: i) we place the COLREG ontology within the rich semantic environment of the *Descriptive Ontology for Linguistic and Cognitive Engineering* (DOLCE) [16, 17]; ii) we include encounters between sailing vessels and between vessels of different categories; iii) we model COLREG rules regarding lights and shapes; iv) we propose an ontological model of a large amount of the COLREG.

The remainder of this paper is organised as follows. Section 2 presents the structure and core principles of COLREG. Section 3) presents and motivates the COLREG ontology. Section 4 describes the classifier of marine encounters based on the COLREG ontology. Finally, Section 5 discusses future developments.

2. A Brief Introduction to the COLREG

COLREG is a convention published by the IMO and must be enforced by each UN member. The main topic discussed in COLREG is collisions at sea, and the purpose of the convention is to establish good practices to reduce the risk of collisions between vessels. COLREG is divided into five parts, each associated with a letter from *A* to *E* in alphabetical order, each divided into several rules. Parts *A* and *E* define the applicability of COLREG and its exemptions, while parts *B*, *C* and *D* define the operational requirements. Most of the rules modelled by this ontology are from parts *B* and *C*.

An important concept in COLREG is vessel categories. COLREG divides vessels into the following categories (cf. COLREG, Rule 3):

- *Power-Driven Vessel:* any vessel propelled by machinery;
- *Sailing Vessel:* any vessel propelled solely by sails;
- *Vessel Engaged in Fishing:* any vessel involved in fishing activities and using fishing equipment which restricts her manoeuvrability;
- *Vessel Not Under Command:* any vessel which is unable to manoeuvre;
- *Vessel Restricted in Her Ability to Manoeuvre:* any vessel engaged in working activities which restrict her ability to manoeuvre;
- *Vessel Constrained by Her Draught:* any vessel which, because of her draught with respect to the depth of the waters in which she is sailing, is restricted in her ability to deviate from her course.

According to the categories of vessels at risk of collision and their respective courses, COLREG identifies four categories of marine encounter situations (cf. COLREG, Rules 12, 13, 14, 15):

- *Sailing Vessel Encounter:* encounter situation involving two sailing vessels in collision route with one another;

[2]Released at https://w3id.org/colreg-ontology

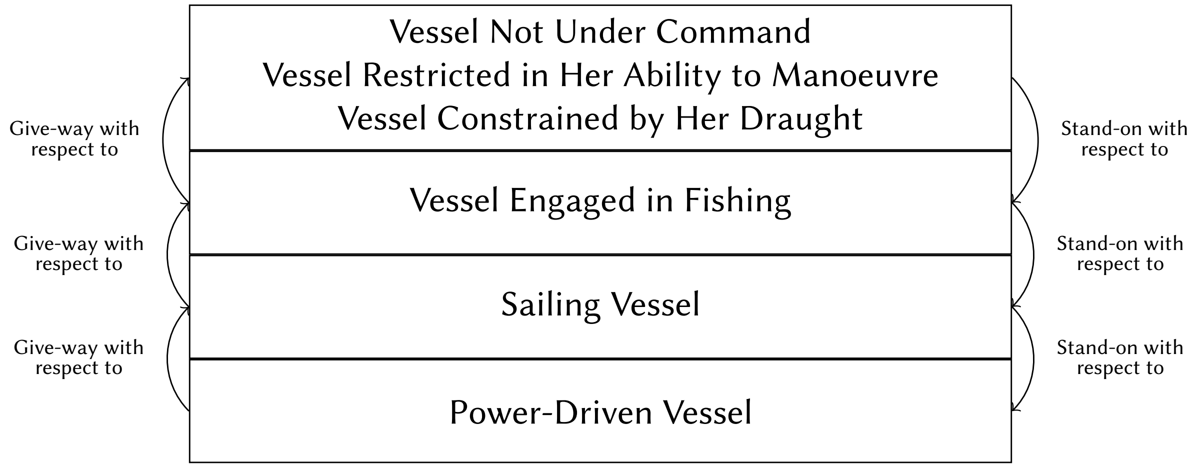


Figure 1: Hierarchy of vessel categories

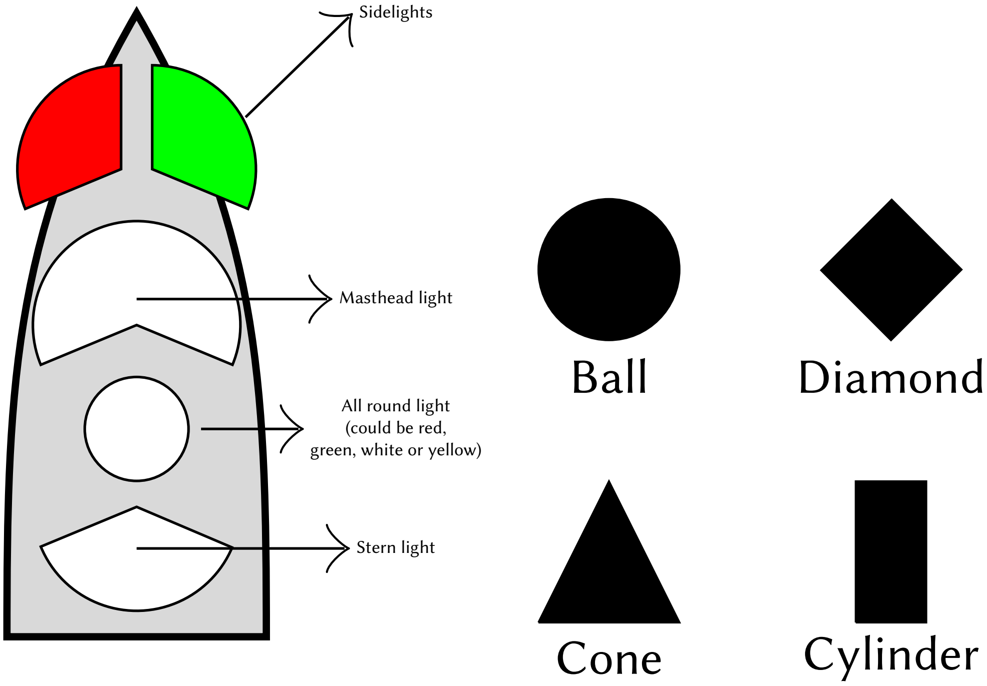
- *Overtaking:* encounter situation between two generic vessels where one is gaining on the other;
- *Head-on:* encounter situation between two power-driven vessels which are meeting on reciprocal, or nearly reciprocal, courses;
- *Crossing:* encounter situation between two power-driven vessels having crossing courses.

The rules describing these situation categories also define the criteria to assign the behaviour that each vessel taking part in the situation is required to follow. COLREG defines two types of behaviours: stand-on, i.e., the vessel is allowed to keep her course, and give-way, i.e., the vessel is required to alter her course to avoid the collision. COLREG defines also the behaviours that vessels belonging to different categories have to comply with by introducing a hierarchy between the vessel categories. For example, a power-driven vessel has to give way to a sailing vessel, a sailing vessel to a vessel engaged in fishing, etc. The rationale behind this hierarchy is that the most manoeuvrable vessel has to give way. We summarise the hierarchy between vessel categories in Figure 1. Moreover, COLREG describes a set of lights and shapes that any vessel is required to exhibit in certain conditions. Such devices are arranged onboard vessels to allow an observer to infer the vessel category and course in limited visibility or in the absence of navigation sensors. Figure 2 shows the lights and shapes defined by COLREG.

A COLREG classifier aims to identify the stand-on and give-way vessel in a marine encounter scenario. The ontology described in this paper follows the logical schema described above to perform the classification: firstly, we assign the vessel categories to the vessels taking part in the scenario; secondly, we determine the relative positions and direction of motions; thirdly, we identify the situation category, and finally, we assign the behaviours.

3. The COLREG Ontology

The methodology we followed to design the ontology involved a close examination of the COLREG document to elicit the main classes, relations, and constraints. In particular, the COLREG document contains explicit definitions of several terms (cf. Rule 3, 'General Definitions') which we manually rendered into OWL as a first step in the formalisation process. Secondly, to provide semantics for the newly introduced COLREG classes and relations, we proposed categorising them within the framework of a top-level ontology. Therefore, the ontological modelling choices are justified by aligning the COLREG domain ontology with this top-level ontology. As a primary validation step, we established the consistency and concept satisfiability of the domain ontology integrated with the top-level ontology. Moreover, we tested the effectiveness of the ontology to classify maritime encounters, cf. Section 4. A comprehensive empirical validation of the ontology and the classifications of maritime encounters is left for future work.



(a) Lights (b) Shapes

Figure 2: Lights and shapes defined in COLREG

We designed the COLREG ontology by situating it within the environment of a (foundational) top-level ontology, namely DOLCE, [16, 17]. DOLCE is an ISO standard top-level ontology for information technology[3]. Originally, DOLCE was developed in first-order modal logic (QS5). A subsequent simplification of the ontology in Common Logic[4] has been developed, and several OWL versions of DOLCE have been proposed. To design the COLREG ontology, we use the recent OWL2 version of DOLCE, termed DolceBasic, cf. [18].[5]

Thus, the COLREG ontology imports DolceBasic. The primary motivation for embedding a domain ontology within the context of a top-level ontology is to make use of the definitions of the general classes and relations, thereby providing a rich and clear semantic environment to develop the domain ontology. In our application, we seek a balance between the expressivity of the semantics of the COLREG elements and the efficiency of the ontology in achieving its objective, namely, automatically classifying the scenarios listed in the COLREG. Therefore, several modelling choices shall depend on the intended application, as we will see.

We briefly summarise the intended meaning of the categories of DOLCE used by the COLREG ontology (Figure 3), while we refer to [17, 18] for a general presentation of DOLCE and DolceBasic. Endurant (or, simply, objects) and Perdurant (or, simply, events) are distinguished by their mode of persistence in time. Endurants have all their parts present at every moment they exist, whereas perdurants can only be partially present at a time when they are present. For instance, all the parts of a sailing vessel are present whenever the vessel itself is present. In contrast, the parts of an encounter event between two ships may not all be present at a specific moment, e.g., at the instant when one ship has just approached another, the earlier phases of the encounter have already passed.

Besides the general categories, DOLCE defines several general relations, including: the (binary) *parthood* between perdurants (or abstract) (e.g. a crossing of two vessels is part of their encounter),

[3]See https://www.iso.org/standard/78927.html

[4]Also an ISO standard for information technology, see https://www.iso.org/standard/66249.html

[5]For the documentation about DOLCE, DOLCE in Common Logic, and DolceBasic, cf. https://github.com/appliedontolab/DOLCE. The repository also contains a tutorial on the use of DolceBasic.

the ternary *temporary parthood*, meaning that an endurant is temporary part of another endurant at certain time (e.g. a sail is part of a sailing vessel at a certain time), the ternary *participation*, an endurant participates to a perdurant at a certain time (e.g. a sailing vessel participates to a vessel encounter at a certain time).

To address the well-known expressivity limitations of OWL, cf. [19, 20], a number of simplifications of DOLCE have been made in DolceBasic, specifically, by restricting it to binary relations, cf. [18]. Temporalised relations, such as temporary parthood and participation, are in fact ternary relations, as they relate two entities and a time parameter. DolceBasic contains the original binary relations of DOLCE, plus the *constant* binary versions of the temporalised relations of DOLCE. This means that the temporalised relations in DolceBasic are provided with a specific intended meaning, cf. [18]: constantParticipantOf(x, y) means that the endurant x participates to perdurant y, for the whole presence of y, while constantPartOf(x,y) means that whenever y is present, x is a temporary part of y. For the intended application to modelling COLREG scenarios, the constant version amounts to assuming that temporary parthood and participation remain stable throughout the designated time window of the represented scenarios. This constrain is at work in the subsequent modelling choices.

The COLREG ontology is publicly available at the permanent link https://github.com/Sabbus/COLREG-Ontology/blob/b080067d207c6c0d838c974dda16e3fb9891ab96/colreg_ontology.owl.[6] We start by discussing the taxonomy of classes in COLREG that are added as subclasses of DolceBasic, cf. Figure 3. The two main classes are **Vessel** and **Situation**. Vessels include the types of ships described in COLREG, while situations describe the types of encounters that are regulated by the COLREG rules.

The class **Vessel** is categorised under the class AgentivePhysicalObect of DolceBasic. This means that they are endurants, i.e. they persist through time, they have temporal and spatial locations, they possess (constant) temporary parts. They are 'objects' in that they have a unity criterion, as opposed to physical objects without clear individuation (e.g. the AmountOfMatter, such as sand, iron, or water). They are 'agentive' in the sense that we can directly ascribe attitudes, such intentions and decisions, or actions, to the vessels themselves. This ascription of agency is motivated by the intention to align with the terminology of the COLREG, which often ascribes attitudes, actions, and responsibility to the vessel itself, rather than attributing them to the persons in charge. Moreover, as we are modelling a MASS, the ascription of agency, in functional terms, is quite standard for artificial agents as well, in a Dennettian sense. More specifically, vessels are a type of *artifacs*. DolceBasic lacks the class of artifacts which is however understood as a subclass of physical objects. For the intended application to the classification of scenarios, this level of abstraction is sufficient. For an analysis of artifacts in DOLCE, cf. [22] and [17].

The class **Situation** includes the encounter situations described in the COLREG. Scenarios are perdurants, as they unfold over time. Moreover, in our ontology we restrict scenarios to have only two vessels as participants. This is not stated explicitly in COLREG, however, the regulation describes only two-vessel encounters. This aspect is important when determining the classification of scenarios among perdurants, i.e., whether they are subclasses of Event or Stative.

In DOLCE, Event and Stative are distinguished by the way they behave with respect to mereological sums. Events are *anti-cumulative* (cf. [16], Dd58), that is, the mereological sum of two perdurants of the same type is not a perdurant of that type (e.g., the mereological sum of two separated ship races is not a ship race), whereas stative are *cumulative* (cf. [16], Dd57), i.e., the sum of two perdurants of the same type is a perdurant of that type (e.g., the sum of two states of laying is a state of laying). The sum of two separated instances of **Situation** might involve more than two vessels as participants, so it cannot be of the type **Situation**, which are constrained to two vessels. For this reasons, we decided to classify **Situation** under the class Event. Moreover, Event is divided into Achievement, which are mereologically atomic, with no proper parts (e.g., reaching a harbour), and Accomplishment, which have proper parts (e.g., navigating from one location to another). Therefore, the encounters described by the COLREG are classified under Accomplishment. The class **Situation** is partitioned into subclasses

[6] DolceBasic is expressed in $\mathcal{SRIF}$, a sublanguage of OWL2, cf. [18]. The COLREG ontology adds qualified cardinality restrictions, obtaining the logic $\mathcal{SRIQ}$, cf. [21]. In the same repository the classifier of maritime encounters and a number of tests are stored.

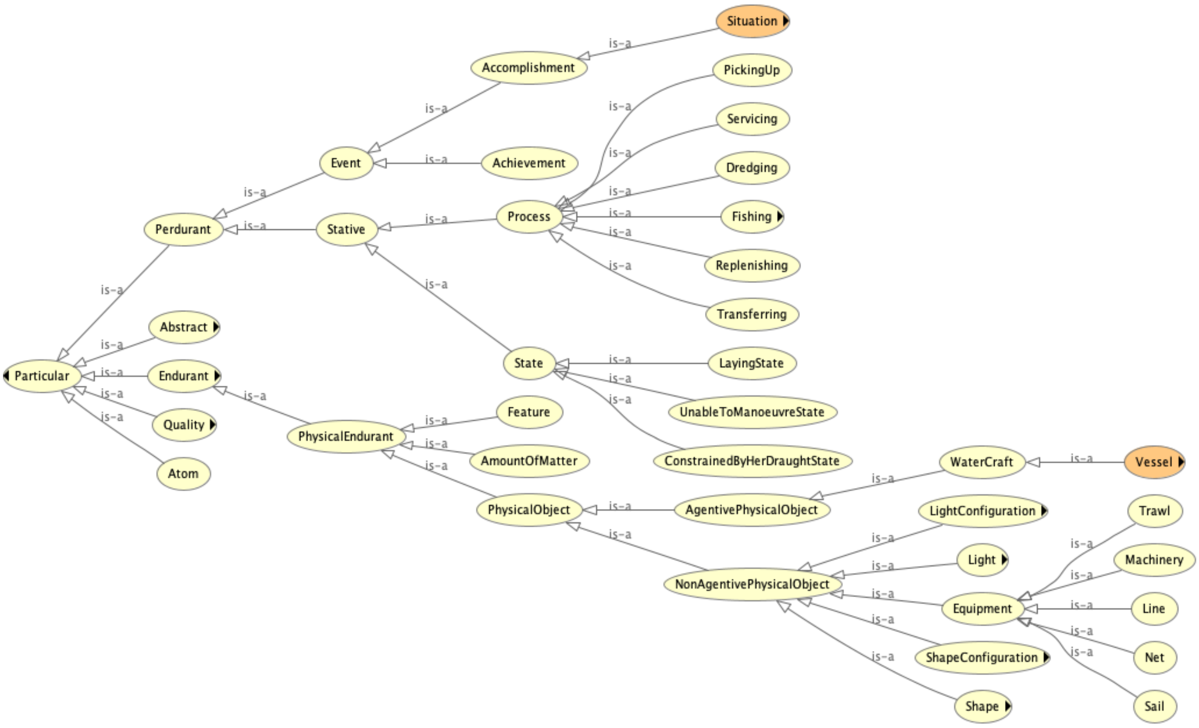


Figure 3: An excerpt of COLREG in DolceBasic

that list the situations described by the COLREG. We assume that the types of scenarios are disjoint. This is motivated to simplify the application of this ontology, that is, to enable the unambiguous automatic classification of scenarios.

Under the class NonAgentivePhysicalObect, we included a number of classes (e.g. **Machinery**) that are useful to define the types of vessels according to the COLREG. For analogous reasons, we included a number of classes under State (e.g., **LayingState**) and Processes (e.g., **Servicing**). Moreover, we added a number of domain-specific object properties, see Section 3.4, and data properties, Section 3.5. Finally, the classes or the axioms of the ontology are annotated with the reference to the pertinent COLREG Rule.

3.1. COLREG: the class Vessel

Vessels are a type of **WaterCraft**, according to COLREG, Rule 1 (a). The subclasses of **Vessel** are depicted in Figure 3. Two important classes require clarification: **OwnShip** and **TargetShip**. These classes enables the perspective of the ship taking actions in an encounter scenario. In ontological terms, these classes represent *roles* of vessels, i.e., a vessel takes on the role of the "own" or "target" ship, given a certain scenario. For an analysis of roles in DOLCE, we refer to [23] and [17].[7] To express the role-based nature of the classes, we constrained the elements of those classes to depend on an encounter situation: every own or target ship is defined with respect to exactly one situation. This view appears somewhat restrictive, as in principle the same vessel could serve as either an own ship or target ship of different situations. However, the intended application of the ontology is to classify one scenario at a time, so allowing the same vessel to participate to distinct scenarios would complicate the implementation of the classifier. It is interesting to note that this restriction amounts to view own ships and target ships as *qua*-entities, defined by a *basis*, the vessel, and a *gloss*, the property of participating to a situation, cf. [24]. The *gloss*, in this case, specifies the situation in which the vessel participates as own (target) ship. According to the identity conditions of *qua* entities, a vessel participating as own ship to scenario

[7]Roles are not present in DolceBasic, so we only characterise the aspect of dependence on a scenario. By means of axioms such as "**OwnShip** is equivalent to inverse (**hasOwnShip**) exactly 1 **Scenario**", where the object property **hasOwnShip** relates scenarios with their unique own ship, see also Section 3.4.

s_1 and a vessel participating as own ship to scenario s_2, when $s_1 \neq s_2$, are distinct, since the glosses express distinct properties, cf. [24], p. 100. Moreover, *qua* entities can inherit properties of the basis or the gloss and, in this case, own and target ships inherit the physical object nature of the the vessel basis.

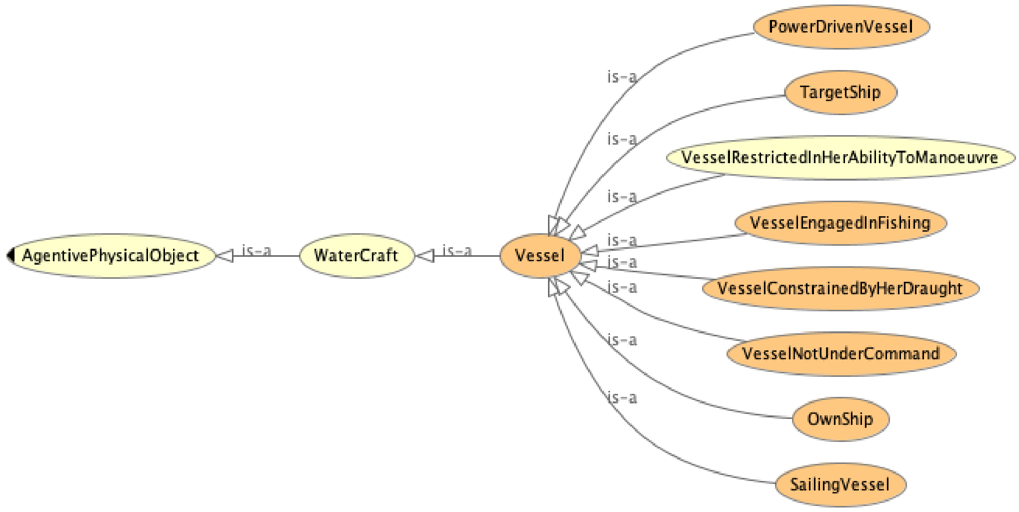


Figure 4: The class **Vessel**

The other subclasses of **Vessel** define the types of vessels listed in Section 2. These classes are placed as disjoint from each other in the ontology so that ambiguity in the classification of the situation is avoided although such disjointness is not explicitly stated in COLREG. For instance the class **PowerDrivenVessel** is defined by those vessels that are "propelled by some machinery" (cf. COLREG, rule 3, (b)): we included the class **Machinery** to classify the ship engines, as non-agentive physical objects, and we have introduced the object property **propelledBy** as a sub-relation of the inverse of (constant) proper parthood, to enable the inference from "if x is propelled by y, then y is a (constant) proper part of x".

3.2. COLREG: the class Situation

The class **Situation** includes events that have exactly two participants: an own ship and a target ship. That is, each scenario is associated to a unique own ship, by means of the **hasOwnShip** functional object property, and to an unique target ship, by **hasTargetShip**. Those object properties are subrelations of the inverse of **constantParticipantOf** to enable inferences such as "if the scenario x has own ship y, then y is a constant participant of the event x". So own and target ships are constant participant of their encounter situation. Thus, the ontology also infers that the encounter situation **specificallyDependsOn** the own (target) ship. The subclasses of **Situation** include the scenarios listed in Section 2.

As we mentioned, the scenarios are disjoint. This is motivated as a simplification to facilitate the design of the classifier of scenarios. Notice that the objective of the COLREG is to norm situations without ambiguity, to clearly select the course of action to be applied, also for this reasons scenarios are exclusive.A particular type of situation described in the ontology is the **DifferentVesselEncounter.** This situation is not explicitly defined in COLREG and it is used to classify all those situations different by the four listed in Section 2, i.e., all those situations in which two vessels of different categories are in risk of collision without one trying to overtaking the other.

3.3. The class Light and Shape

According to the COLREG, many encounter situations can be classified based on the light configuration that a vessel exhibits, expecially in distress situations. Thus, we have included the class **Light**, including several types of lights (e.g. **GreenSideLights**). Moreover, we have introduced a class of

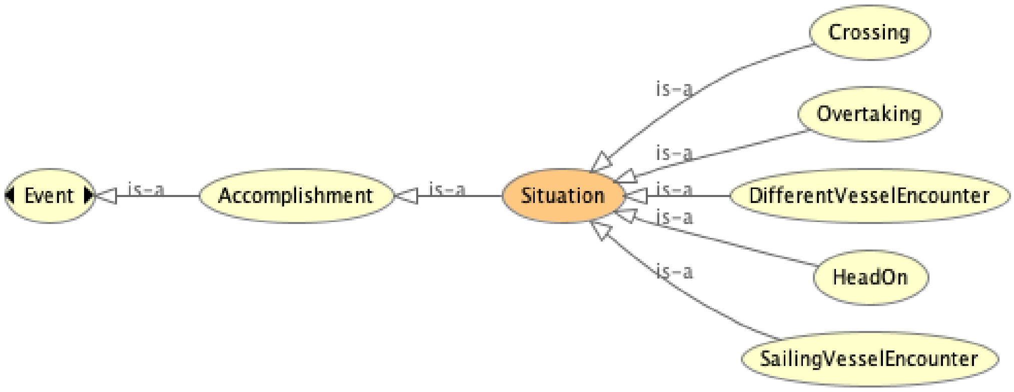


Figure 5: The class **Situation**

LightConfiguration that includes the possible configurations of lights that a vessel can have (e.g. **LightConfigurationForVesselNotUnderCommandDistancing**). **Light** and **LightConfiguration** are NonAgentivePhysicalObjects, which can be (constant) proper parts of a vessel. The various types of **LightConfiguration** are modelled by constraining the types of lights that can compose them, by means of the object property of DolceBasic constantAtomicPartOf.[8] That is, in our rendering, a light configuration has a number of (constant) atomic parts, which are the lights that compose the configuration (in principle, a light configuration is the mereological (constant) sum of the lights that compose it). We could have "defined" each type of configuration by listing exactly the lights that are parts of the configuration (by using cardinality restrictions such as *exactly*, *min*, and *max*). However, due to the computational complexity of these restrictions, this move would result in a very inefficient performance from the reasoner (specifically Pellet)[9]. Therefore, to associate a light configuration with its parts, we introduce a data property (**hasStringValue**) that associates a light configuration with the string of values corresponding to its lights.

A similar discussion applies to the classes **Shape** and **ShapeConfiguration**, which are intended as well as NonAgentivePhysicalObjects. Notice that by "shape", term used in the COLREG, we intend those specific parts of the ships with certain shapes, which are used as diurnal signals (thus, they are not to be categorised within Feature of dolce).

3.4. Object properties

As we discussed in the previous paragraph, the object properties **hasOwnShip** and **hasTargetShip** associate each situation with their unique own and target ship, they are subproperties of **hasVessel**, which is in turn a subproperty of the inverse of constantParticipantOf. The object property **engagedIn** relates a vessel to an activity (in this case, a Process, such as **Fishing**, or a State, such as **Laying**) and it is a subproperty of constantParticipantOf. The object property **fasterThan** relates the own ship and the target ship, according to the data about the relative speed of the two vessels.

The object properties **hasAhead**, **hasBehind**, **hasOnPortside**, and **hasOnStarboard** relate the own and the target ships according to their relative positions. The object properties **hasBowInSight**, **hasLightsInSight**, **hasPortsideInSight**, **hasShapesInSight**, **hasStarboardInSight**, and **hasSternInSight** relate the two vessels when the own ship has a certain part (e.g. bow, lights, shapes, portside) of the target ship in sight. We could have introduced at least a partial semantics for these relations, by means of subproperty chains, but we chose not to in order to avoid overburdening the reasoner

[8] They are atomic, in this rendering, for a technical reason, i.e. constantAtomicPartOf is not transitive, so it can be used in axioms such as "**LightConfiguration** SubClassOf (inverse (constantAtomicPartOf) some **Light**) and (inverse (constantAtomicPartOf) only **Light**)". Moreover, we have no need to introduce proper parts of lights within the COLREG.

[9] We ran our tests on a laptop with Fedora Linux 40 (Workstation Edition), an Intel Core i9-14900HX × 32 processor and 16GB of RAM. The classifier, using Pellet, takes about 1 second per test to respond without adding cardinality restrictions. When cardinality restrictions are included to define light configurations, the response time increases to about 1 minute per test.

during classification. We will discuss how these object properties operate to classify scenarios within the SWRL rules in 3.6.

3.5. Data properties

We introduced a number of data properties to associate the values that are required to perform the classifications of the encounter situations under their appropriate type. For instance, **hasHeading** associates a vessel to a value in sexagesimal degrees (a decimal number between 0° and 360°), representing the direction in which the vessel's bow is pointed with respect to the true north. The data properties **hasRelativeBearingWithRespectToOwnShip** and **hasRelativeBearingWithRespectToTargetShip** associate the target ship and the own ship respectively to a value between -180° and 180°. They represent the angle between the bow of the own (target) ship and the segment connecting the two vessels. These properties are used to identify the relative motion between two vessels.

The data property **hasBehaviour** associates a vessel with the actions that she has to take in a given situation according to the COLREG. Actions are here represented as data, i.e. as strings being either "alter_course" or "keep_course". We decided not to categorise behaviours or actions within DolceBasic, at least in this version of the ontology, as a simplification move. In principle, they could be categorised as perdurants, to which the relevant ship may participate. However, due to the limitations of DolceBasic concerning time and change, it is quite challenging to represent the case where a vessel takes actions before or after a certain event occurs (e.g. a ship alters her course after expecting a possible collision). Moreover, viewing actions as a type of event may be appropriate for actions that have already been performed, but not for those that are merely prescribed —that is, actions that may or may not occur. To capture this aspect, a deontic modality would be required on top of the standard ontology, but this falls outside the scope of OWL.

As we will see, **hasBehaviour** is of great importance, as it provides the output of the classifier of encounter situations.

3.6. Semantic Web Rule Language (SWRL)

In this ontology, SWRL rules are used to implement the actual classification mechanism as it is described at the end of Section 2.[10] The ontology contains many SWRL rules, but in this section, only a small number of rules are discussed which are representative of the classification mechanism that we implemented.

Formula 1 shows the criteria to identify the target ship as a power-driven vessel by the arrangement of her lights. From this equation it is also possible to infer the direction of motion of the target ship with respect to the own ship.

$$\begin{aligned}&\textbf{Situation}(?s) \land \textbf{hasOwnship}(?s, ?o) \land \textbf{hasTargetShip}(?s, ?t)\\ &\quad \land \textbf{hasLightsInSight}(?o, ?l) \land \textbf{hasStringValue}(?l, \text{“masthead\_light\_red\_sidelight”})\\ &\qquad \rightarrow \textbf{PowerDrivenVessel}(?t) \land \textbf{hasPortsideInSight}(?o, ?t)\end{aligned} \tag{1}$$

Formula 2 shows the identification of the position of the target ship relatively to the own ship. This rule makes use of the concept of relative bearing between the target ship and the own ship.

$$\begin{aligned}&\textbf{Situation}(?s) \land \textbf{hasOwnship}(?s, ?o)\land\\ &\quad \textbf{hasTargetShip}(?s, ?t) \land \textbf{hasRelativeBearingWithRespectToOwnShip}(?t, ?b)\land\\ &\qquad \mathsf{lessThanOrEqual}(?b, 112.5^\circ) \land \mathsf{greaterThan}(?b, 5^\circ) \rightarrow \textbf{hasOnStarboard}(?o, ?t)\end{aligned} \tag{2}$$

Finally the classification of the situation and the assignment of the behaviours is performed by specific rules of which Formula 3 is an example regarding the classification of a crossing situation.

[10] For the usage of SWRL, cf. https://www.w3.org/submissions/SWRL/

$$\textbf{Situation}(?s) \land \textbf{hasOwnShip}(?s, ?o) \land \textbf{hasTargetShip}(?s, ?t) \land \textbf{PowerDrivenVessel}(?o) \land \textbf{PowerDrivenVessel}(?t) \land \textbf{hasPortsideInSight}(?o, ?t) \land \textbf{hasOnStarboard}(?o, ?t) \rightarrow \textbf{Crossing}(?s) \land \textbf{hasBehaviour}(?t, \text{“keep\_course”}) \land \textbf{hasBehaviour}(?o, \text{“alter\_course”}) \quad (3)$$

4. Classification of marine encounters

Together with the ontology, we developed a simple software which implements the automated reasoning required to perform the classification of marine encounter scenarios. We refer to this software as the COLREG *classifier*. The COLREG classifier is distributed as a Python package and the source code can be found in the same repository of the ontology together with installation instructions.[11]

The classifier uses the *Owlready2* library, cf. [25], which provides API to the *Pellet* reasoner, cf. [26]. In implementing this classifier we decided to define custom JSON schemas to represent input and output data. There is a previous work in literature that proposes a JSON schema to described encounter situations, cf. [27], however the schema proposed in that paper conveys much more information than required by this classifier and, in some cases, even lacks some. The logic implemented is very simple and is intended to be just a proof of concept of an actual classifier. First, a file describing the encounter scenario is passed to the software. Such file describes a JSON object which contains the following keys:

- *name:* the name of the scenario described by the file;
- *own-ship:* own ship of the scenario;
- *target-ship:* target ship of the scenario;
- *wind-direction:* the direction from which the wind blows.

The JSON objects associated with the keys *own-ship* and *target-ship* contains the following keys:

- *name:* name of the vessel;
- *sog:* short for speed over ground, is the speed of the vessel;
- *heading:* the direction of the vessel's bow;
- *category:* the category to which the vessel belongs among those described by COLREG;
- *bearing-of-other-vessel:* the relative bearing of the other vessel.

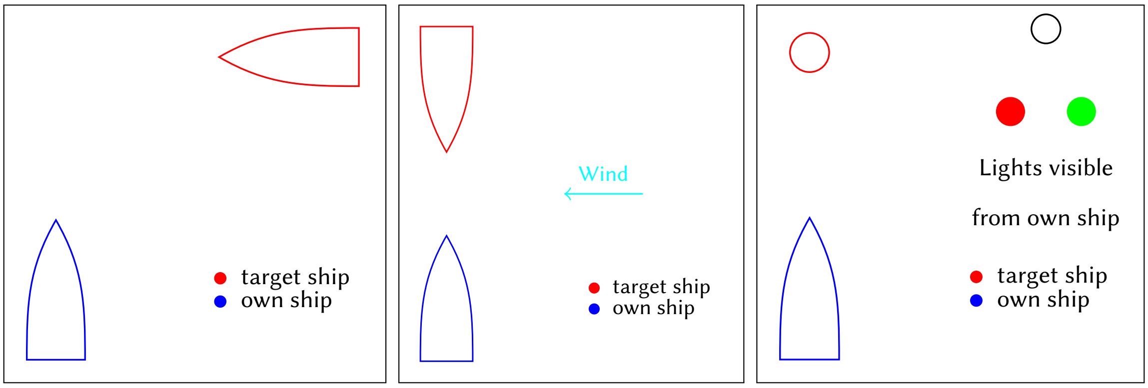
Among those keys, only *name* and *heading* are mandatory for the target ship, while the own ship is required to have *name*, *heading*, *sog*, *category* and *bearing-of-other-vessel*. Besides those described, the *own-ship* object may presents the following additional keys:

- *lights-in-sight:* an array of target ship's lights visible from the own ship;
- *shapes-in-sight:* an array of target ship's shapes visible from the own ship.

The data contained in the scenario file are then translated into OWL assertion axioms and imported in the COLREG ontology. Assertion axioms populate with instances certain classes, object or data properties of the COLREG ontology. The scenario's, own ship's and target ship's names are used to create individuals that are instances of the **Situation**, **OwnShip** and **TargetShip** respectively. The key *category* is used, if present, to make the own ship and the target ship instances also of the specified vessel category. Then instances of the **hasOwnShip** and the **hasTargetShip** properties are created to assign the newly instantiated own ship and target ship to the situation. Finally, all other information present in the JSON file is used to create instances of the data properties described in Section 3.5.

At this point Pellet is invoked by owlready2 to perform the classification and finally all relevant information are retrieved from the ontology and dumped to a result file, describing a JSON object containing the following keys:

[11] https://github.com/Sabbus/COLREG-Ontology/



(a) Scenario A (b) Scenario B (c) Scenario C

Figure 6: Test scenarios

Table 1
Input parameters

	Scenario A	Scenario B	Scenario C
ψ_{OS}	0°	0°	0°
$\beta_{TS,OS}$	45°	0°	0°
Category	Power-driven vessel	Sailing vessel	Power-driven vessel
ψ_{TS}	270°	180°	Unknown
$\beta_{OS,TS}$	-45°	0°	Unknown
Category	Power-driven vessel	Sailing vessel	Unknown
Wind	Unknown	90°	Unknown

- *behaviour:* one for the own ship and one for the target ship, it is either the string *alter_course* or *keep_course*;
- *category:* one for the own ship and one for the target ship, it is the inferred vessel category if it was not already specified in the scenario file;
- *situation-category:* the inferred situation category of the scenario.

We tested the COLREG classifier on 42 marine encounter scenarios which can be found in the repository. In this section, we present a selection of scenarios for the sake of brevity. In particular, we show the classification of the three scenarios shown in Figure 6. Scenario A represents two power-driven vessels with crossing courses, with the own ship having the target ship on her staboard. Scenario B shows two sailing vessels on reciprocal courses, with the own ship having the wind on her starboard side, the target ship on her portside. Finally, scenario C shows the encounter between a power-driven vessel (the own ship) and a target ship of unknown category and unknown heading; from the own ship it is possible to spot the lights shown in the figure. The parameters of each scenario are summarised in Table 1, where ψ_{OS} and ψ_{TS} indicate the heading of the own ship and target ship respectively and $\beta_{TS,OS}$ and $\beta_{OS,TS}$ indicate the relative bearing of the target ship from the own ship perspective and the relative bearing of the own ship from the target ship respectively. Once the classification software is launched, it correctly recognises scenario A as a crossing situation, with the own ship having to alter her course and the target ship being allowed to keep her course (cf. COLREG, Rule 15); scenario B as a sailing vessel encounter with the own ship having to alter her course, while the target ship is allowed to keep going (cf. COLREG, Rule 12); scenario C as a head-on situation between two power-driven vessel (cf. COLREG, Rules 14, 23) with both vessels having to alter their courses (cf. COLREG, Rule 14). Table 2 summarises the results.

Table 2
Output of the classification

	Scenario A	Scenario B	Scenario C
Situation	Crossing	Sailing vessel encounter	Head-on
Own ship category	Power-driven vessel	Sailing vessel	Power-driven vessel
Behaviour	alter_course	alter_course	alter_course
Target ship category	Power-driven vessel	Sailing vessel	Power-driven vessel
Behaviour	keep_course	keep_course	keep_course

5. Conclusions and Future Developments

In this paper, we showed the feasibility of the use of ontologies as part of an autonomous marine navigation operative system to ensure safe and compliant collision avoidance capabilities. Moreover, we proved DolceBasic and SWRL to be expressive enough to model a fair share of the rules of a complex regulation such as colreg. Future developments of this work should address the problem of scenario classification and behaviour assignment in encounter situations involving more than two vessels. Another limit that should be addressed is the absence of temporal relationship in the ontology presented in this work. A temporal formalisation of colreg rules would be able to achieve desirable capabilities such as dynamic classification and hysteresis to avoid sudden changes in the classification output for edge cases. Finally, we plan to evaluate the ontology and the prescriptions provided by the classifier using competency questions posed to a community of experts.

Declaration on Generative AI

The authors have not employed any Generative AI tools.

References

[1] I. M. Organization, Outcome of the regulatory scoping exercise for the use of maritime autonomous surface ships (MASS), 2021. URL: https://wwwcdn.imo.org/localresources/en/MediaCentre/HotTopics/PublishingImages/Pages/Autonomous-shipping/MSC.1-Circ.1638%20-%20Outcome%20Of%20The%20Regulatory%20Scoping%20ExerciseFor%20The%20Use%20Of%20Maritime%20Autonomous%20Surface%20Ships...%20(Secretariat).pdf, accessed on 02/01/2025.

[2] I. M. Organization, Interim guidelines for MASS trials, 2019. URL: https://wwwcdn.imo.org/localresources/en/MediaCentre/HotTopics/Documents/MSC.1-Circ.1604%20-%20Interim%20Guidelines%20For%20Mass%20Trials%20%28Secretariat%29.pdf, accessed on 02/01/2025.

[3] I. M. Organization, 109th session of the maritime safety comitee, 2024. URL: https://www.imo.org/en/MediaCentre/HotTopics/Pages/Autonomous-shipping.aspx, accessed on 02/01/2025.

[4] M. O. Lines, World's first successful sea trial of autonomous sailing on a commercial container ship voyage, 2022. URL: https://www.mol.co.jp/en/pr/2022/22007.html, accessed on 06/01/2025.

[5] A. Felski, K. Zwolak, The ocean-going autonomous ship—challenges and threats, Journal of Marine Science and Engineering 8 (2020). URL: https://www.mdpi.com/2077-1312/8/1/41. doi:10.3390/jmse8010041.

[6] R. Puisa, L. Lin, V. Bolbot, D. Vassalos, Unravelling causal factors of maritime incidents and accidents, Safety Science 110 (2018) 124–141. URL: https://www.sciencedirect.com/science/article/pii/S0925753518304545. doi:https://doi.org/10.1016/j.ssci.2018.08.001.

[7] I. M. Organization, Convention on the international regulations for preventing collisions at sea, 1972, 1972. URL: https://treaties.un.org/pages/showDetails.aspx?objid=08000002800fcf87, accessed on 02/01/2025.

[8] M. Martelli, S. Žuškin, E. Cellerino, R. Zaccone, Ship collision detection and classification employing ais data, in: ISOPE International Ocean and Polar Engineering Conference, ISOPE, 2024, pp. ISOPE–I.
[9] R. Zaccone, A dynamic programming approach to the collision avoidance of autonomous ships, Mathematics 12 (2024). doi:10.3390/math12101546.
[10] J. Gleeson, M. Dunbabin, J. J. Ford, Colreg scenario classification and compliance evaluation with temporal and multi-vessel awareness for collision avoidance systems, Ocean Engineering 313 (2024) 119552.
[11] P. N. Hansen, D. Papageorgiou, M. Blanke, R. Galeazzi, M. Lützen, J. Mogensen, M. Bennedsen, D. Hansen, Colregs-based situation awareness for marine vessels-a discrete event systems approach, IFAC-PapersOnLine 53 (2020) 14501–14508.
[12] S. Zhong, Y. Wen, Y. Huang, X. Cheng, L. Huang, Ontological ship behavior modeling based on colregs for knowledge reasoning, Journal of Marine Science and Engineering 10 (2022) 203.
[13] C. Zhou, K. Wen, J. Zhao, Z. Bian, T. Lu, M. Ko Ko Latt, C. Wang, Ontology-based method for identifying abnormal ship behavior: A navigation rule perspective, Journal of Marine Science and Engineering 12 (2024) 881.
[14] P. Hatlas-Sowinska, M. Wielgosz, Ontology based approach in solving collision situations at sea, Ocean Engineering 260 (2022) 111941.
[15] Y. Wen, Y. Zhang, L. Huang, C. Zhou, C. Xiao, F. Zhang, X. Peng, W. Zhan, Z. Sui, Semantic modelling of ship behavior in harbor based on ontology and dynamic bayesian network, ISPRS International Journal of Geo-Information 8 (2019) 107.
[16] C. Masolo, S. Borgo, A. Gangemi, N. Guarino, A. Oltramari, WonderWeb Deliverable D18, Technical Report, CNR, 2003.
[17] S. Borgo, R. Ferrario, A. Gangemi, N. Guarino, C. Masolo, D. Porello, E. M. Sanfilippo, L. Vieu, DOLCE: A descriptive ontology for linguistic and cognitive engineering, Applied Ontology 17 (2022) 45–69. doi:10.3233/AO-210259.
[18] D. Porello, L. Vieu, W. Terkaj, S. Borgo, F. Compagno, E. M. Sanfilippo, Dolce in owl: The core theory, in: 8th Workshop on Foundational Ontology (FOUST VIII)-Joint Ontology Workshops (JOWO)-Episode X: The Tukker Zomer of Ontology, and satellite events, CEUR Workshop Proceedings, 2024.
[19] I. Horrocks, O. Kutz, U. Sattler, The even more irresistible SROIQ, in: P. Doherty, J. Mylopoulos, C. A. Welty (Eds.), Proceedings, Tenth International Conference on Principles of Knowledge Representation and Reasoning, Lake District of the United Kingdom, June 2-5, 2006, AAAI Press, 2006, pp. 57–67. URL: http://www.aaai.org/Library/KR/2006/kr06-009.php.
[20] C. M. Keet, O. Kutz, Orchestrating a network of mereo(topo)logical theories, in: Proceedings of the 9th Knowledge Capture Conference, K-CAP '17, Association for Computing Machinery, New York, NY, USA, 2017. URL: https://doi.org/10.1145/3148011.3148013. doi:10.1145/3148011.3148013.
[21] Y. Kazakov, Riq and sroiq are harder than shoiq, in: Proceedings of the Eleventh International Conference on Principles of Knowledge Representation and Reasoning, 2008, pp. 274–284.
[22] S. Borgo, L. Vieu, Artefacts in formal ontology, in: Philosophy of technology and engineering sciences, Elsevier, 2009, pp. 273–307.
[23] C. Masolo, L. Vieu, E. Bottazzi, C. Catenacci, R. Ferrario, A. Gangemi, N. Guarino, et al., Social roles and their descriptions., in: KR, 2004, pp. 267–277.
[24] K. Fine, Acts, events and things, in: Sixth International Wittgenstein Symposium, Kirchberg-Wechsel (Austria), 1982, pp. 97–105.
[25] J.-B. Lamy, Owlready: Ontology-oriented programming in python with automatic classification and high level constructs for biomedical ontologies, Artificial intelligence in medicine 80 (2017) 11–28.
[26] E. Sirin, B. Parsia, B. C. Grau, A. Kalyanpur, Y. Katz, Pellet: A practical owl-dl reasoner, Journal of Web Semantics 5 (2007) 51–53. URL: https://www.sciencedirect.com/science/article/pii/S1570826807000169. doi:https://doi.org/10.1016/j.websem.2007.03.004, software Engineering and the Semantic Web.

[27] T. A. Pedersen, C. Vasanthan, K. Karolius, Ø. Engelhardtsen, K. P. Houweling, A. Jørgensen, Generating structured set of encounters for verifying automated collision and grounding avoidance systems, Journal of Physics: Conference Series 2618 (2023) 012013. URL: https://dx.doi.org/10.1088/1742-6596/2618/1/012013. doi:10.1088/1742-6596/2618/1/012013.