Intelligence & Robotics

Research Article

Planning robotic agent actions using semantic knowledge for a home environment

Rodrigo Bernardo[1,2], João M. C. Sousa[1], Paulo J. S. Gonçalves[2]

[1]IDMEC, Mechanical Engineering Department, Instituto Superior Técnico, University of Lisbon, Lisbon, Lisbon 1049-001, Portugal.
[2]Industrial Engineering Department, Polytechnic Institute of Castelo Branco, Castelo Branco, AV. Empresário, Castelo Branco 6000-767, Portugal.

Correspondence to: Dr. Rodrigo Bernardo, IDMEC, Mechanical Engineering Department, Instituto Superior Técnico, University of Lisbon, Av. Rovisco Pais, 1 1049-001 Lisbon, Portugal. E-mail: rodrigo.f.bernardo@tecnico.ulisboa.pt

Abstract

Autonomous mobile robotic agents are increasingly present in highly dynamic environments, thus making the planning and execution of their tasks challenging. Task planning is vital in directing the actions of a robotic agent in domains where a causal chain could lock the agent into a dead-end state. This paper proposes a framework that integrates a domain ontology (home environment ontology) with a task planner (ROSPlan) to translate the objectives coming from a given agent (robot or human) into executable actions by a robotic agent.

Keywords: Ontologies, autonomous robotics, planning, knowledge representation,semantic maps

1. INTRODUCTION

Robots are increasingly present in environments shared with humans and highly dynamic environments[1,2]. It is therefore imperative to study and develop new techniques so that robots can effectively move, locate themselves, detect objects, perform tasks, *etc.* in places that can change rapidly, in an autonomous way. More complex methodologies require systems capable of deliberating quickly and effectively. Several studies point to the need for knowledge as way to address this challenges[3,4]. Therefore the formal conceptualization of the robotics domain is an essential requirement for the future of robotics, to design robots that can autonomously perform a wide variety of tasks in a wide variety of environments[5].

Robots need to efficiently create semantic models of their environment (semantic maps). One way that has proved to have shown great value in representing the information of the environment where robots work, is through semantic maps. These combine semantic, topological, and geometric information into a compact representation[6,7]. Existing semantic maps need to evolve from task-specific representations to models that can be dynamically updated and reused in different tasks. This is one of the major limitations of these approaches. Moreover, the ontologies developed to date are not reusable, being one of the major limitations in this strategy. Ontologies should move towards a more homogeneous structure and easy interchangeability between different structures in order to be reusable[8]. Recently, robots are pouring into home environments, and thus the need to communicate with humans is increasing. The tasks of robots are not only to navigate in an accurate geometrical space but also to understand the indoor environment and share common semantic knowledge with people. Consider the task of fetching a cup of coffee. If a robot had only a representation of the environment through a metric map, it would have to search in a crude way all over the environment until it found the cup. If more knowledge were added through semantics to the robot, such as the probability of the cup being in each room, the search could be guided from locations with high probability to locations with lower probability. In short, with the evolution of systems and artificial intelligence (AI), ontologies become a great solution to make domain knowledge explicit and remove ambiguities, enable machines to reason, and facilitate knowledge sharing between machines and humans, focusing on a new generation of intelligent and integrated technologies for smart manufacturing.

Currently, in robotics, the most used middleware is the ROS (https://www.ros.org). This is the standard middleware for the development of robotic software, allowing the design of modular and scalable robotic architectures. There is a framework in ROS called ROSPlan (http://kcl-planning.github.io/ROSPlan/) that provides a collection of tools for AI planning, namely ROSPlan. It has a variety of nodes which encapsulate planning, problem generation, and plan execution. It possesses a simple interface and links to common ROS packages. To date, this framework does not yet have an adequate interface for semantic queries, thus lacking a general standardized framework for working with ontologies, natively supporting symbolic logic and advanced reasoning paradigms. In this sense, the paper proposes a framework that integrates a domain specific home environment ontology with a task planner (ROSPlan), translating the objectives coming from another agent (robot or human) into executable actions by the robotic agent. Two reasoning systems for task planning were developed, which are based on ontologies The first system uses the MongoDB (https://www.mongodb.com/) database, while the second system uses the domain specific ontology home environment that is proposed in this paper.

The paper is structured as follows. In the next section, the related work is reviewed. The design methodology section introduces the reasoning systems presented in this work. In the results section, the structure of the ontology is presented, along with the results obtained from the different proposed reasoning systems. In the validation and discussion section, the developed ontology is validated and discussed. Finally, the conclusions section presents the conclusions of the work and the future way forward.

2. RELATED WORK

Ontologies are a powerful solution for acquiring and sharing common knowledge. Ontologies represent a common understanding in a given domain, promoting semantic interoperability among stakeholders, because "sharing a common ontology is equivalent to sharing a common world view"[3]. All concepts in an ontology must be rigorously specified so that humans and machines can use them unambiguously, empowering robots to autonomously perform a wide variety of tasks in a wide variety of environments.

Depending on their level of generality, different types of ontologies can be identified[9]; among many types, we can identify as the main ones:

- An upper or general ontology (upper ontology or foundation ontology) is a model of the common objects that are generally applicable to a wide variety of domain ontologies. There are several higher ontologies standardized for use, such as SUMO (suggested upper merged ontology)[10], Cyc ontology[11], BFO (basic formal ontology)[12], and DOLCE (descriptive ontology for linguistic and cognitive engineering)[13].
- Domain ontologies (domain ontology or domain-specific ontology) model a specific domain or part of the world (*e.g.*, robotic[14], electronic, medical, mechanical, or digital domain).
- Task ontologies describe generic tasks or activities[15].
- Application ontologies are strictly related to a specific application and used to describe concepts of a particular domain and task.

In the next subsections, the related work is reviewed about semantic maps, ontologies for semantic maps, and some applications of knowledge representation for robotic systems.

2.1" Semantic maps

The daily challenges have drive the research for automated and autonomous solutions to enable mobile robots to operate in highly dynamic environments. For this purpose, mobile robots need to create and maintain an internal representation of their environment, commonly referred to as a map. Robotic systems rely on different types of maps depending on their goals. Different map typology's have been developed such as metric and topological maps, which are generally 2D representations of the environment[16], or hybrids (a combination of the previous two)[17,18]. There are also maps with 3D representation (sparse map, semi-dense map, and dense map). Metric and topological maps only contain spatial information[19]. A fundamental requirement for the successful construction of maps is to deal with uncertainty arising, from errors in robot perception (limited field of view and sensor range, noisy measurements, *etc.*), from inaccurate models and algorithms, *etc.*

To get around this limitation, semantic maps were developed to add additional information, such as instances, categories, and attributes of various constituent elements of the environment (objects, rooms, *etc.*)[5-7]. These provide robots with the ability to understand beyond the spatial aspects of the environment, the meaning of each element, and how humans interact with them (features, events, relationships, *etc.*). Semantic maps deal with meta information that models the properties and relationships of relevant concepts in the domain in question, encoded in a knowledge base (KB).

2.2" Ontologies for semantic maps

One of the tasks to be solved in mobile robot navigation is the acquisition of information from the environment. In the field of semantic navigation, information includes concepts such as objects, utilities, or room types. The robot needs to learn the relationships that exist between the concepts included in the knowledge representation model. Semantic maps add to classical robotic maps spatially grounded object instances anchored in a suitable way for knowledge representation and reasoning. The classification of instances through the analysis of the data collected by sensors is one of the biggest challenges in the creation of semantic maps (*i.e.*, to give a richer semantic meaning to the sensor data)[20,21].

In the last decade, several papers have appeared in the literature contributing different representations of semantic maps. Kostavelis *et al.*[5] summarized the significant progress made on a broad range of mapping approaches and applications for semantic maps, including task planning, localization, navigation, and human–robot interaction. Semantics has been used in a diverse range of applications. Lim *et al.*[22] presented an approach for unified robot knowledge for service robots in indoor environments. Rusu *et al.*[23] developed a map called Semantic Object Maps (SOM), which encodes spatial information about indoor household environments, in particular kitchens, but in addition it also enriches the information content with encyclopedic and common sense knowledge about objects, as well as includes knowledge derived from observations. Galindo *et al.*[24] proposed an approach for robotic agents to correct situations in the world that do not conform to the semantic model by generating appropriate goals for the robot. In short, it combines the use of a semantic

map with planning techniques in Planning Domain Description Language (PDDL) that converts the goals into actions (moving the robot, picking and dropping an object, *etc.*). Wang *et al.* [25] employed the relationships among objects to represent the spatial layout. Object recognition and region inference are implemented by using stereo image data. Vasudevan *et al.* [26] created a hierarchical probabilistic concept-oriented representation of space, based on objects. Diab M *et al.* [27] introduced an interpretation ontology to identify possible failures which occur during automatic planning and the execution phase; this ontology aims to improve planning and allow automatic replanning after error. Balakirsky *et al.* [28] proposed an ontology-based framework that allows a robotic system to automatically recognize and adapt to changes that occur in its workflow and dynamically change the details of task assignment, increasing process flexibility by allowing plans to adapt to production errors and task changes.

In short, semantic maps enable a robot to solve reasoning problems of geometric, topological, ontological, and logical nature, in addition to localization and path planning [29]. Formal conceptualization of the robotics domain is an essential requirement for the future of robotics, in order to be able to design robots that can autonomously perform a wide variety of tasks in a wide variety of environments.

2.3" Applications in robotic systems

Different research groups have used semantic knowledge in the area of robotics. Semantic knowledge allows a clear dialog between all stakeholders involved in the life cycle of a robotic system and enables the efficient integration and communication of heterogeneous robotic systems. These facilitate communication and knowledge exchange between groups from different fields, without actually forcing them to align their research with the particular view of a particular research group [30].

One of the most recent advances in the field of robotics can be denoted by analyzing the KnowRob project, where researchers aimed to enable a robot to answer different types of questions about possible interactions with its environment, using semantic knowledge [31,32]. For example, they developed an ontology that allows the robot to start an assembly activity, with incomplete knowledge. It identifies the missing parts, having the ability to reason about how the missing information can be obtained [32]. KnowRob employs the DUL foundational ontology, which is a slim version of the Descriptive Ontology for Linguistic and Cognitive Engineering (DOLCE). DUL and DOLCE have a clear cognitive bias, and they are both well established in the knowledge engineering community as foundational ontologies. However, DUL does not define very specific concepts such as fork or dish. These concepts are needed for our robots that do everyday activities [8]. There are also other relevant works that aim at the standardization of knowledge representation in the robotics domain, such as IEEE-ORA [33], ROSETTA [34], CARESSES [35], RoboEarth [36], RoboBrain [37], RehabRobo-Onto [38], and OROSU [39].

All of the above work already represents promising advances in the use of semantics in robotic systems; however, it lacks the ability to perform advanced reasoning and relies heavily on *ad hoc* reasoning solutions, significantly limiting its scope. A general standardized framework for working with ontologies is needed, natively supporting symbolic logic and advanced reasoning paradigms.

The next sections present the reasoning frameworks, with particular emphasis on the domain specific ontology, home environment. The proposed ontology is designed to be easily reusable in different environments of a house, as well as by different robotic agents.

3.   DESIGN METHODOLOGY

The focus of the developed ontology is to enable robotic agents to interact with elderly people within a home environment. The robots are to assist the elderly people to manage and better perform their daily lives, and

thus to longer live an independent life in their known surroundings. They are also there to help better maintain social contacts, which is known to have a very positive influence on the mental and physical health of elderly people. An initial and extensible list of tasks that these robots are eventually supposed to perform includes the following activities:

- Help elderly people out of bed and or the couch.
- Serve the breakfast.
- Supply elderly people medicine.
- Bring books or operate media (entertainment).
- Make up the bedroom.
- Play games.
- Adjust settings: shades, light, and heat.
- Serve drinks.
- Assist during bathing.
- Clean the rooms.

When developing the ontology, several concepts were searched in databases such as dictionaries on the web. The search was carried out on specific sections on the different concepts of a house (https://www.enchantedlearning.com/wordlist/house.shtml, https://dictionary.cambridge.org/pt/topics/buildings/houses-and-homes/). It was also extracted from the documentation of the project RoCKIn@Home challenge (http://rockinrobotchallenge.eu/home.php, http://rockinrobotchallenge.eu/RoCKIn_D2.1.1.pdf), concepts and typical tasks for a domestic robot. The concepts were chosen in order to characterize the simplified environment of a house, based on a smart-home environment, built on the robotics laboratory of Instituto Politécnico de Castelo Branco. A challenge arose in the choice of concepts due to the great complexity of objects that can be found in a given room. To simplify the process, the concepts were defined for characterizing the simplified environment that is found in the laboratory.

3.1" Knowledge Engine

Figure 1 depicts the global knowledge engine conceptual framework, designed to achieve the main objective of the paper: the integration of a domain specific home environment ontology, with a task planner (ROSPlan), transforming the goals coming from the reasoning into executable actions by the robotic agent. The framework have three main parts: reasoning (which includes the ontologies), planning, and the robot. These parts are presented in the remainder of the paper.

A domain specific home environment ontology, aligned with a MongoDB database, encapsulate the important concepts of the domain to be considered (space of a house, objects, *etc.*). Indeed, the main benefit of a domain ontology is to set standard definitions of shared concepts identified in the requirement phase and to define appropriate relations between the concepts and their properties[40]. The ontology contain concepts of Core Ontology for Robotics and Automation (CORA), with the representation of fundamental concepts of robotics and automation[41].

The ontology model is based on the concepts and relationships between different entities, and then aligned with the MongoDB database. The basic concept of the reasoning process is based on the premises that: a relational database contains both the entities of the conceptual hierarchy and the instances of the physical hierarchy, this information is stored in lists, and these lists are related to each other, as in the entity–relationship model of the environment[42,43].

The domain specific ontology home environment is defined with the Protégé software. Protégé version 5.5.0 was used[44]. The the domain ontology was verified through version 1.4.3 of HermiT Reasoner to ensure that it is free of inconsistencies[45]. Protégé is a free, open-source editor for developing the ontologies produced by
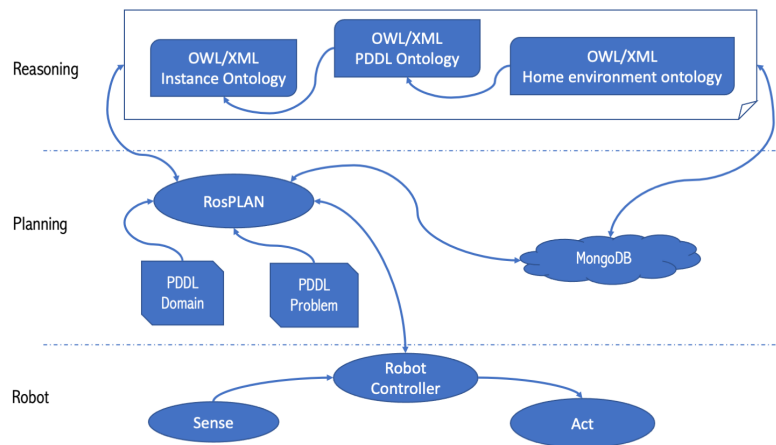
Figure 1. Global System, containing a Reasoning section that is based on an ontology, a Planning section that is based on the Planning Domain Definition Language (PDDL), a relational database (MongoDB) that is queried using the ontology and ROSPlan, and a Robot section that is based on robot controller, as well as its sensing and acting devices.

Stanford university. It is a java-based application (multi-platform), with plugins such as ontoViz to visualize the ontologies. The backbone of protégé is that it supports the tool builders, domain specialists, and knowledge engineers.

3.1.1. Domain specific ontology home environment
Figure 2 shows the main relationships of the developed domain-specific home environment ontology, which was designed for an agent to interpret and interact with its surrounding environment. In this case, the environment is a house, with special focus on the internal environment.

The developed model is subdivided into three main classes: *Home_lab*, *Information*, and *Objects*. The class *Home_lab* is subdivided into class *Environment*, which is subdivided into internal and external environment. The *Indoor_environment* is further subdivided into *Hall*, *Corridor*, and *Rooms*. The class *Rooms* is subdivided into the possible rooms types of an house (bathroom, bedroom, kitchen, *etc.*). The class *Information* is subdivided into *Information_Object*. The *Objects* class is subdivided into *Types_Devices* that are subdivided into *Sensors* and *Actuators*. The main class *Objects* is further subdivided into internal and external objects that contemplate the objects that can be found in a home environment. Finally, the *Objects* class is subdivided into *PhysicalAgent*, which is subdivided into the agents that can appear in the environment as *human* or *robot*, the latter further subdivided into the different types of robots.

Figure 2 presents the hierarchical class where the main concepts defined in the ontology are visible as:

- **Environment:** The surroundings or conditions in which an agent, person, animal, or plant lives or operates.
- **Indoor environment:** Environment situated inside of a house or other building.
- **Corridor:** A long passage in a building from which doors lead into rooms.
- **Hall:** The room or space just inside the front entrance of a house or flat.
- **Rooms:** Space that can be occupied or where something can be done (kitchen, bedroom, etc).
- **Outdoor Objects:** Used to describe objects that exist or appear outside a home.
- **Objects:** Any physical, social, or mental object, or a substance. Following DOLCE, objects are always participating in some event (at least their own life), and are spatially located (defined by: http://www.ontologydesignpatterns.org/ont/dul/DUL.owl).
- **Outdoor environment:** Environment situated outside of a house or other building.
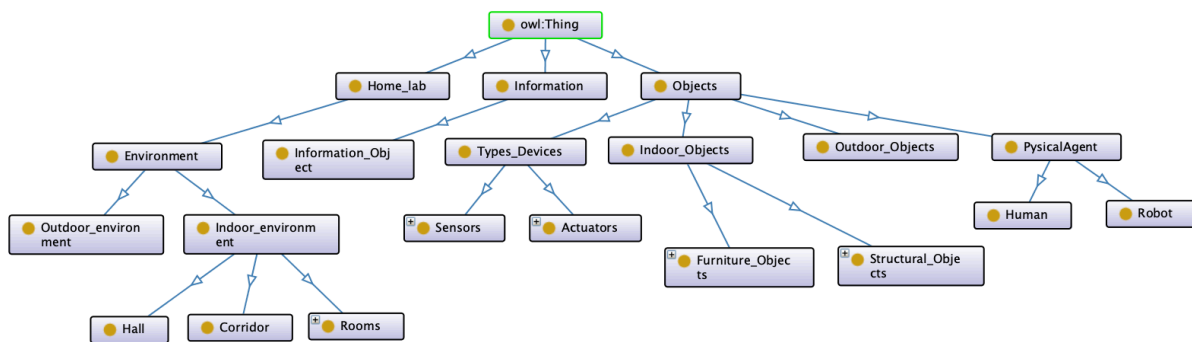
Figure 2. Snapshot of the main ontology (relations *is-a*).

- **Information:** Facts provided or learned about something or someone.
- **Information_Object:** They are messages performed by some entity[46]. They are ordered (expressed in accordance with) by some information encoding system (*e.g.*, sensors present in the agent). They can express a description (the ontological equivalent of a meaning/conceptualization), they can be about any entity, and they can be interpreted by an agent.
- **Indoor Objects:** Used to describe objects that exist or appear inside a home.
- **Outdoor Objects:** Used to describe objects that exist or appear outside a home.
- **Physical Agent:** Any agentive Object, either physical (*e.g.*, a whale, a robot, or an oak tree) or social (*e.g.*, a corporation, an institution, or a community) (defined by: http://www.ontologydesignpatterns.org/ont/dul /DUL.owl).
- **Mobile robot:** Robot that is able to move in the surrounding (locomotion) (*i.e.*, autonomous mobile robot and autonomous mobile and manipulator robot).
- **Not mobile robot:** Robot that is not able to move in its surroundings (*i.e.*, robot arm).
- **Types Devices:** A collection of properties that define different components and behaviors of a type of device (actuators, sensors, *etc.*).

The properties of objects and the spatial relationships between them represent the characteristics of the environment and the spatial arrangement, respectively. Several *Object properties* were created to relate the different concepts, so that an agent can characterize its surrounding environment:

- *ObjectProperty*:
  - *belowOf* ;
  - *isMember*;
  - *isObjectOf*;
  - *isPartOf*;
  - *LeftOf*;
  - *onTopOf*;
  - *RightOf*.
- *LocationProperty*:
  - *isConnectedTo*.
- *AgentProperty*:
  - *isGoingTo*;
  - *isIn*.

The object properties were defined to make explicit the relationships between concepts. The properties *belowOf*, *LeftOf*, *onTopOf*, and *RightOf* were created to define the relationships between the different concepts of *Indoor_Objects* and *Outdoor_Objects*. Through these, the agent can identify the disposition of objects in the environment, creating relationships about them (*e.g.*, based on Figure 3, if an agent has to guide an elderly person to the chair that is in the room, it knows that it is on the right side of the bed). The property *isConnectedTo*

is a transitive and symmetric property, which correlates the different concepts of the *Environment* according to the environment in which the agent is inserted. The object properties *isGoingTo* and *isIn* are defined in order to correlate the *AgentProperty* concept with the *Environment*. (*e.g.*, the robot *isIn* the living room, but it *isGoingTo* the bedroom). The object property *isPartOf* is a symmetric property that is used for the agent to link a given instance of the *Objects* class with an *Environment*. This allows the agent to know which objects are in a room. It differs from the object property *isObjectOf* because in this property the agent is sure that the object exists in the environment. Finally, the *isObjectOf* concept was defined to relate the concepts: Objects to the Environment. Through this, the objects that can be found in each zone of the environment are defined. (*i.e.*, in a room there can be an object of the type bed, chair, television, carpet, *etc.*). Thus, an agent can search for an object by the place with the highest probability of it being found (*i.e.*, if the agent has to find a frying pan, it knows that this object is commonly in a kitchen).

3.2" PDDL Planning Agent

The Planning Domain Definition Language (PDDL) describes problems through the use of predicates and actions. The problems in PDDL are defined in two parts, a domain and a problem file. This language has undergone different modifications in order to make it capable of dealing with more complex tasks [47-49]. The ROSPlan framework was used to perform the planning tasks [50]. ROSPlan is a high-level tool that provides planning in the ROS environment; it generates the PDDL problem, the plan, the action dispatch, the replanning, *etc*. Different action interfaces have been written in C++ to control the Autonomous Manipulator Mobile Robot (AMMR) (*i.e.*, base, arm, and gripper). These interfaces are constantly listening for action PDDL messages. In addition, the MongoDB database was used for semantic memory storage (locations, robots, home objects, goal parameters, *etc.*).

The POPF planner (https://nms.kcl.ac.uk/planning/software/popf.html), a forwards-chaining temporal planner, was used. After the plan was generated, the interface actions interconnect the plan with the lower level control actions, allowing the robotic agent (AMMR) to complete the plan. During execution, if an action fails due to changes in the environment, the planning agent reformulates the PDDL problem by re-planning.

4.   RESULTS

This section discusses the main results obtained by applying the proposed framework. For a better understanding, the results are divided into two subsections: The first subsection refers to the validation results of the home environment ontology, where the main reasoning techniques were presented and how they can be used. In the second subsection, the results of the reasoning system through MongoDB are presented. The way the robot performs a set of tasks, in a real environment, is also presented.

4.1" Validation home environment ontology

The home environment ontology contains a vast number of concepts regarding the home environment as well as the different objects that may be present in a given room. For example, if a given agent is in a room for the first time, based on the objects it observes, through its sensors, it can categorize the space based on the knowledge represented on the ontology. Based on the ontology, if the agent sees objects such as knives, pots, and pans, then it infers that it must be in a kitchen. In this situation, the agent will identify the room and create all the relations of the objects it detects in the environment.

Knowledge reasoning techniques can infer new conclusions and thus help to plan dynamically in a non-deterministic environment. In the presented application, spatial reasoning and reasoning based on relations are used [51]. Spatial reasoning is mostly used, when it is done through reasoning on the ontology hierarchy and spatial relations therein, allowing to predict the exact spatial location of an object in the environment. This prediction is obtained using a set of asserted facts and axioms on the ontology. Reasoning over the ontology
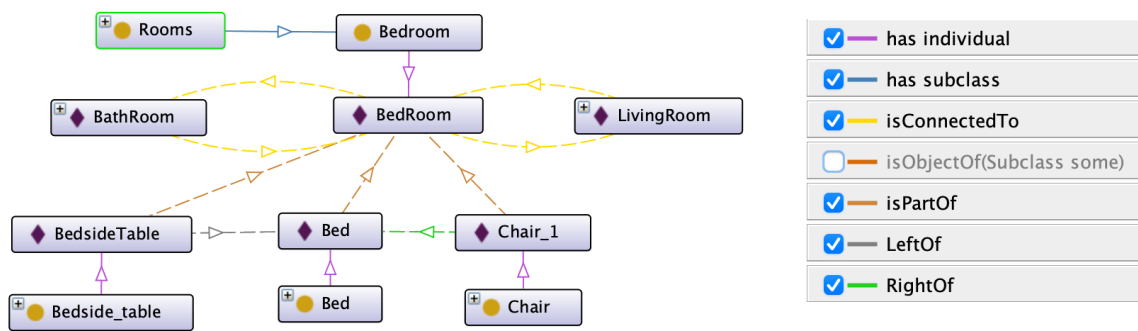
Figure 3. Example of the ontology being used to instantiate the house objects belonging to the instance *"Bedroom"*, present in the home environment.
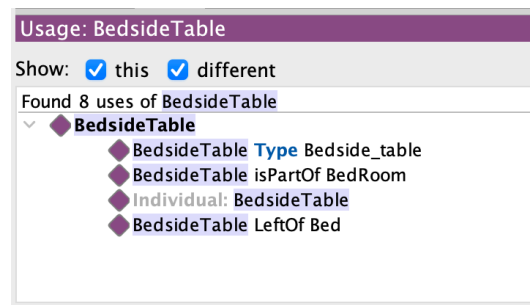

Figure 4. Example of the knowledge representation of an object in the environment of a house that has been instantiated.

relations is used to help with inferring new conclusions (*e.g.*, If B is a subclass of A and C is a subclass of B, then C is a subclass of A can be inferred as transitivity holds for the subclass property).

The object property hierarchy view displays the asserted and inferred object property hierarchies. From the knowledge base built, all existing components from the environment are instantiated. That is, the individuals are based on their type (Figure 3). This allowed testing the functionality of the ontology. Figure 3 depicts that each room was instantiated to the corresponding *Rooms* subclass (*e.g.*, the *BedRoom* instance is related to the *Bedroom* class). Several relations can be drawn from the instances that represent the environment, such as which objects belong to the *BedRoom* instance. Although the relations of the *BathRoom* and *LivingRoom* instances do not appear in Figure 3, these can be obtained, based on the created instances present in the ontology, in order to represent both rooms.

As previously stated, having the environment of a house in an ontological knowledge oriented database is of special interest, for example, to know where each object belongs. In fact, as depicted in Figure 4, the information about the BedsideTable is completely available to the user, using a simple logic description. A query to the ontology will retrieve useful information, for example where the object is attached. This issue is further discussed in the next subsection.

Performing logical description reasoning actions to obtain valuable data for the robot's reasoning can be done, as presented in Figure 5, namely information about objects, rooms, and their relations defined in the ontology. For example, what type of object is instantiated as Bed? In which rooms is it present? What is on its left? What is on its right? Using the ontology and description logic queries, it is straightforward to obtain the following
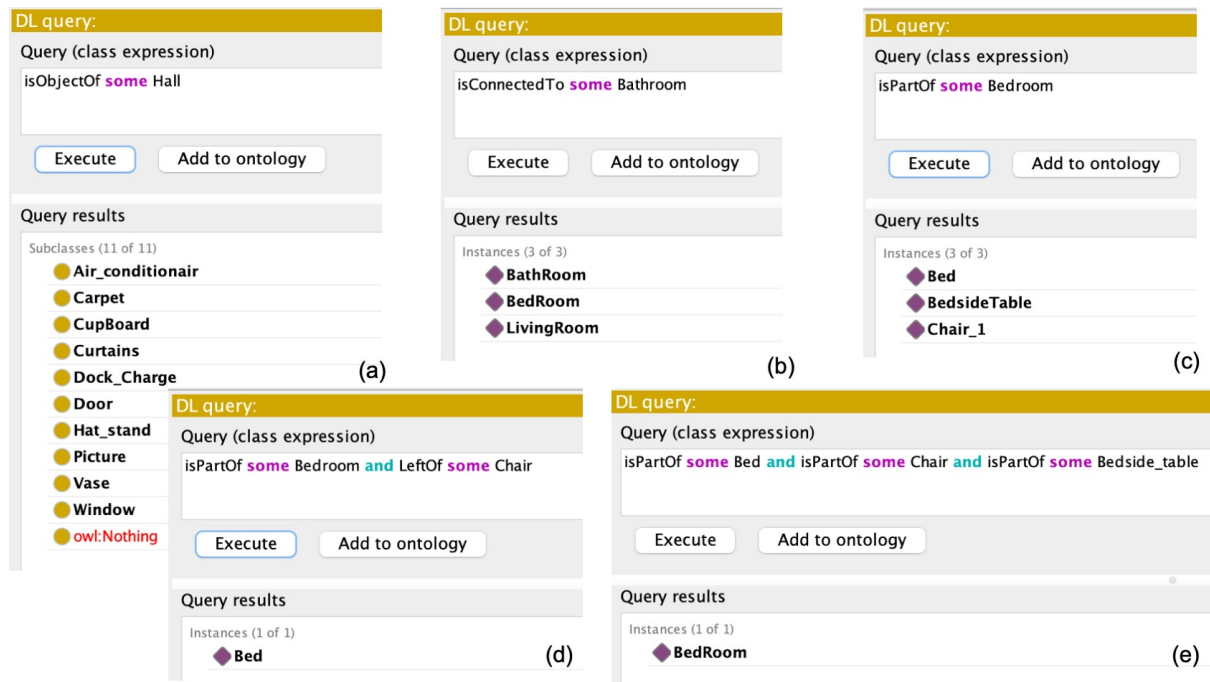
Figure 5. Reasoning using the ontology. (a) Query about the classes of objects that can be found in a class Hall. (b) Query about the instance(s) connected to instance of class Bathroom. (c) Query on the instance(s) that are part of the class Bedroom. (d) Query about the instance(s) that are part of class BedRoom and that are on the left of instances of class Chair. (e) Query to discover the location (room where the agent is in) based on what the agent observes.

information:

- Classes of *Objects* that can be found in a certain instance of room (*e.g.*, in the Hall in Figure 5a).
- Connectivity relationship between instances of the *Rooms* class referring to an environment (Figure 5b).
- Instance of objects present in an instance of room (Figure 5c).
- Recognize which instance(s) of the *Objects* class belong to a particular instance of a *Rooms* class and are to the left of an instance of the *Chair* class (Figure 5d).
- Recognize which instance of the *Rooms* class belong to a particular instance(s) of a *Objects* class (*e.g.*, the robotic agent can locate itself (know in which room it is), based on the objects it observes) (Figure 5e).

Through the ontology developed, one or more agents are able to locate themselves more efficiently in the environment. When the agent is lost, it can identify the room where it is, based on the objects it observes (Figure 5e). Observing Figure 3, if the robot recognizes a *Bed*, a *BedsideTable*, and a *Chair_1*, it knows that it is in a *Bedroom*. The robotic agent will be able to perform a search in an optimized way for an object. It does not need to perform a massive search for all the rooms; *e.g.*, it knows which are the rooms in which there is a higher probability of finding a fridge, teapot, *etc.*

4.2"　Validation the reasoning system with MongoDB

To test the system based on the MongoDB database, a problem was outlined, for the agent to execute/solve (Figure 6). An AMMR is used, composed of a mobile base and a robotic arm of the Universal Robotics UR3, equipped with a RobotIQ 2f-140 gripper, the whole system runs with the middleware: ROS (Figure 7a). Figure 7b depicts the layout of a simple home environment, an apartment for elderly people, created under the EUROAGE project [52], which is in the robotics laboratory of the Polytechnic Institute of Castelo Branco.

Initial conditions were established, such as the location of the AMMR (dock), the world coordinate at which the robot arm is located (p0), and the location of the object in the environment (LivingRoom), as well as its

```
( d e f i n e  ( p r o b l e m  t a s k )
( : d o m a i n  AMMR)
( : o b j e c t s
 L i v i n g R o o m  BedRoom  –  Rooms
 r o b o t _ b a s e  –  r o b o t
 p0  p1  p2  p3  –  p _ g o a l
 obj1  dock  –  O b j e c t s
 r o b o t i q  –  g r i p p e r
)
( : i n i t
 ( a t  r o b o t _ b a s e  dock )
 ( a t  – a r m  p0 )
 ( g r i p p e r _ i s _ f r e e  r o b o t i q )
 ( a l l  obj1  p3  r o b o t _ b a s e  L i v i n g R o o m )
 ( d o c k e d  r o b o t _ b a s e )
 ( d o c k _ a t  dock )

)
( : g o a l  ( a n d
 ( a t  obj1  p2  r o b o t _ b a s e  BedRoom  )
 ( d o c k e d  r o b o t _ b a s e )
) )
)
```

Figure 6. Task definition in PDDL.

(a) Robotic agent (AMMR)
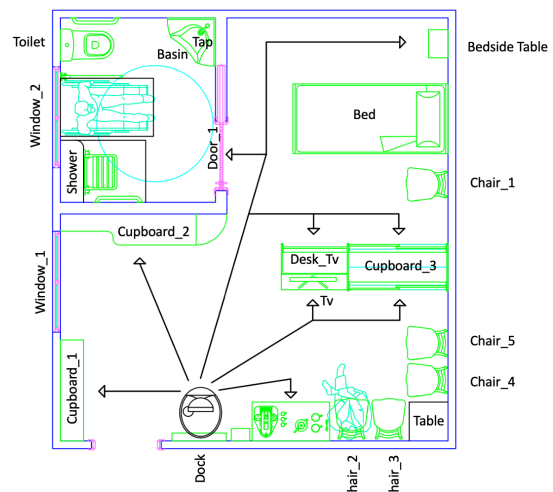
(b) Environment

Figure 7. Robotic agent (AMMR) and simplified environment of a house, created in the framework of the EUROAGE project.

world coordinate (p3). The AMMR aims to leave the dock and pick up an object (obj1) that is in the living room, at the coordinates of world (p3). After the object is grabbed, the AMMR should take it to the bedroom and drop the object at the coordinates of world (p2). After the pick and place tasks are completed, the AMMR should return to the dock. This task definition is depicted in Figure 6.

Using ROSPlan, the generated plan is visible in Table 1. The right column presents the time of each durative action.

Figure 8, presents an excerpt of the global view of the ROS nodes and topics used by the system. It is possible to verify how the connections between them occur. It is visible in Figure 8a that, when the planner (ROSPlan)

Table 1. Example: generated plan

Global Time	Action and respective objects used	Time of action
0.000	(undock robot_base dock)	[5.000]
5.001	(localise robot_base)	[10.000]
15.002	(open robotiq robot_base)	[2.000]
17.002	(move_base dock LivingRoom robot_base)	[5.000]
22.003	(move_ur3 p0 p3 LivingRoom robot_base)	[5.000]
27.003	(pick obj1 p3 robotiq robot_base LivingRoom)	[2.000]
29.003	(move_base LivingRoom BedRoom robot_base)	[5.000]
34.003	(move_ur3 p3 p2 BedRoom robot_base)	[5.000]
39.003	(drop obj1 p2 robotiq robot_base BedRoom)	[2.000]
41.003	(dock robot_base dock)	[5.000]

(a) Execution of the action *undock*

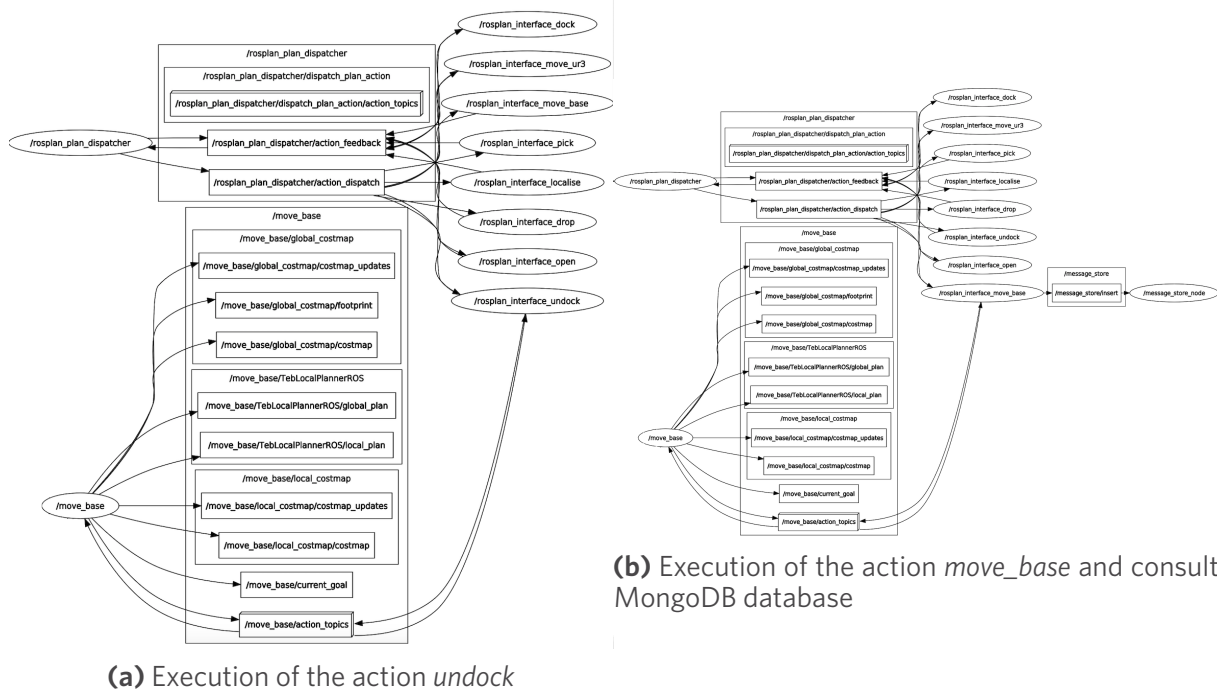**(b)** Execution of the action *move_base* and consult MongoDB database

Figure 8. Execution of a plan by the robotic agent (AMMR).

requests that the agent should leave the dock, the action responsible for the task, */rosplan_interface_undock*, is triggered. This action in turn communicates with the action */move_base* that communicates with the actuators (motors) to move the robot. Figure 8b shows that, when the robot is requested to move to a certain room, */rosplan_interface_move_base* is activated, which in turn consults the MongoDB database to know where the room is located in the map. Then, the robot moves in the environment, after calling */move_base action*.

5. CONCLUSION AND FUTURE WORK

The use of ontologies has become a great solution, and one of the paths to follow in the future to make domain knowledge explicit and eliminate ambiguities, allow machines to reason, and facilitate knowledge sharing between machines and humans. In this work, a structured ontology is presented to be used by robotic agents in order to assist them in their deliberation tasks (interaction with the environment and robot movement). It is imperative to endow robotic agents with semantic knowledge. Several approaches in the literature show advantages in systems using databases such as MongoDB, pointing to their speed of response compared to ontology-based systems. This paper introduces a framework that combines both, in terms of concepts and their implementation in real robotic systems.

The proposed framework improves the problem specification in PDDL based on the updated information coming from the ontology, making the generated plan more efficient. For example, if an external agent (robot or human) launches a task for the robotic agent to collect a certain object and transport it to a specific location, the ontology will be queried. As such, the problem in PDDL is written with specific information, such as the relations between objects, relations between objects and the environment, the current location of the robot, and so on.

The developed home environment ontology was validated by performing successful queries to it, using a standard reasoner in Protégé. The concepts within the home environment ontology, used to define the MongoDB database, were experimentally validated, for semantic reasoning in the the home environment of the laboratory. Moreover, the ROSPlan, together with the developed interface actions, was shown to be a very efficient approach, in interfacing the low-level control with the semantic reasoning of the robot agent.

In future work, the developed domain ontology will be aligned with upper ontologies, *e.g.*, DOLCE. Further developments will be pursued to speed up the ontology-based approach, by exploring ways to make querying more efficient by solving the limitation presented in other works [43], where it is pointed out that these solutions are slower than the pure database approaches.

DECLARATIONS

Authors' contributions
Implemented the methodologies presented and wrote the paper: Rodrigo Bernardo
Developed the idea of the proposed framework: Rodrigo Bernardo, Paulo J. S.Gonçalves
Managed and supervised the research project: João M. C. Sousa , Paulo J. S.Gonçalves
All authors have revised the text and agreed to the published version of the manuscript.

Availability of data and materials
The main data supporting the results in this study are available within the paper. The raw datasets here reported will be available upon request.

Financial support and sponsorship

Conflicts of interest
All authors declared that there are no conflicts of interest.

Ethical approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Copyright

REFERENCES

1. Burke JL, Murphy RR, Rogers E, Lumelsky VJ, Scholtz J. Final report for the DARPA/NSF interdisciplinary study on human-robot interaction. *IEEE Trans Syst , Man, Cybern C* 2004;34:103–12.

2. Thrun S. Toward a framework for human-robot interaction. *Human–Computer Interaction* 2004;19:9–24. [DOI: 10.5898/JHRI.3.2.Beer]

3. Olszewska JI, Barreto M, Bermejo-Alonso J, et al. Ontology for autonomous robotics. In: 2017 26th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN). IEEE; 2017. pp. 189–94.

4. de Freitas EP, Olszewska JI, Carbonera JL, et al. Ontological concepts for information sharing in cloud robotics. *J Ambient Intell Human Comput* 2020:1–12.

5. Kostavelis I, Gasteratos A. Semantic mapping for mobile robotics tasks: A survey.*Robotics and Autonomous Systems* 2015;66:86–103.

6. Hanheide M, Göbelbecker M, Horn GS, et al. Robot task planning and explanation in open and uncertain worlds. *Artificial Intelligence* 2017;247:119–50.

7. Toscano C, Arrais R, Veiga G. Enhancement of industrial logistic systems with semantic 3D representations for mobile manipulators. In: Iberian Robotics conference. Cham: Springer International Publishing; 2017. pp. 079/4: 0

8. Olivares-Alarcos A, Beßler D, Khamis A, Goncalves P, Habib MK, et al. A review and comparison of ontology-based approaches to robot autonomy. *The Knowledge Engineering Review* 2019;34.

9. Guarino N. Formal ontology in information systems: Proceedings of the first international conference (FOIS'98), June 6-8, Trento, Italy. vol. 46. IOS press; 1998.

10. Niles I, Pease A. Towards a standard upper ontology. In: Proceedings of the international conference on Formal Ontology in Information Systems-Volume 2001; 2001. pp. 2–9.

11. Lenat D, Guha R. Building large knowledge-based systems: Representation and inference in the CYC project. *Artificial Intelligence* 3; ; 5–83–75/850

12. Arp R, Smith B, Spear AD.   Building ontologies with basic formal ontology.   Mit Press; 2015.

13. Masolo C, Borgo S, Gangemi A, Guarino N, Oltramari A. Wonderweb deliverable d18: Ontology library. Technical report, ISTC-CNR; 2003.

14. Prestes E, Carbonera JL, Rama Fiorini S, et al. Towards a core ontology for robotics and automation. *Robotics and Autonomous Systems* 2013;61:1193–204.

15. Balakirsky S, Schlenoff C, Rama Fiorini S, et al. Towards a robot task ontology standard. In: International Manufacturing Science and Engineering Conference. vol. 50749. American Society of Mechanical Engineers; 2017. p. V003T04A049.

16. Efficient integration of metric and topological maps for directed exploration of unknown environments. *Robotics and Autonomous Systems* 2002;41:21–39.

17. Sim R, Little JJ. Autonomous vision-based robotic exploration and mapping using hybrid maps and particle filters. *Image and Vision Computing* 2009;27:167–77.

18. He Z, Sun H, Hou J, Ha Y, Schwertfeger S. Hierarchical topometric representation of 3D robotic maps.*Auton Robot* 4243–67–977/930

19. Niloy A, Shama A, Chakrabortty RK, et al. Critical design and control issues of indoor autonomous mobile robots: A Review.*IEEE Access* 4243–; –5755: /920

20. Liu J, Li Y, Tian X, Sangaiah AK, Wang J.*Toy drys yo can ' a acegus ' data-an ontolo y ' app roach 0 'Sensors (Basel)* 423; –3; –33; 50

21. Günther M, Wiemann T, Albrecht S, Hertzberg J. Model-based furniture recognition for building semantic object maps. *Artificial Intelligence* 2017;247:336–51.

22. Lim GH, Suh IH, Suh H. Ontology-based unified robot knowledge for service robots in indoor environments.*IEEE Trans Syst , Man, Cybern A* 4233–63–6; 4/72; 0

23. Rusu RB, Marton ZC, Blodow N, Holzbach A, Beetz M. Model-based and learned semantic object labeling in 3D point cloud maps of kitchen environments. In: 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE; 2009. pp. 3601–8.

24. Galindo C, Saffiotti A. Inferring robot goals from violations of semantic knowledge. *Robotics and Autonomous Systems* 2013;61:1131–43.

25. Wang T, Chen Q. Object semantic map representation for indoor mobile robots. In: Proceedings 2011 International Conference on System Science and Engineering. IEEE; 2011. pp. 309–13.

26. Vasudevan S, Siegwart R. Bayesian space conceptualization and place classification for semantic maps in mobile robotics. *Robotics and Autonomous Systems* 2008;56:522–37.

27. Diab M, Pomarlan M, Beßler D, et al. An ontology for failure interpretation in automated planning and execution. In: Iberian Robotics conference. Springer; 2019. pp. 381–90.

28. Balakirsky S. Ontology based action planning and verification for agile manufacturing. *Robotics and ComputerIntegrated Manufacturing*

2015;33:21–28.

29. Garg S, Sünderhauf N, Dayoub F, et al. Semantics for robotic mapping, perception and interaction: A Survey."*FNT in Robotics*"4242⇉ <
'''''''''3/4460

30. Manzoor S, Rocha YG, Joo SH, et al. Ontology-based knowledge representation in robotic systems: a survey
oriented toward applications."*Applied Sciences*"4243⇌33‑65460

31. Tenorth M, Beetz M. KnowRob: A knowledge processing infrastructure for cognition-enabled robots. *The International Journal of Robotics Research* 2013;32:566–90.

32. Beßler D, Pomarlan M, Beetz M. Owl-enabled assembly planning for robotic agents. In: Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems; 2018. pp. 1684–92.

33. Schlenoff C, Prestes E, Madhavan R, et al. An IEEE standard ontology for robotics and automation. In: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE; 2012. pp. 1337–42.

34. Stenmark M, Malec J. Knowledge-Based Industrial Robotics. In: SCAI; 2013. pp. 265–74.

35. Bruno B, Chong NY, Kamide H, et al. The CARESSES EU-Japan project: making assistive robots culturally competent. In: Italian Forum of Ambient Assisted Living. Springer; 2017. pp. 151–69.

36. Waibel M, Beetz M, Civera J, et al. Roboearth."*IEEE Robot Automat Mag*"4233⇌3: ‑8; /: 40

37. Saxena A, Jain A, Sener O, et al. Robobrain: Large-scale knowledge engine for robots. arXiv preprint arXiv:14120691 2014.

38. Dogmus Z, Erdem E, Patoglu V. RehabRobo-Onto: Design, development and maintenance of a rehabilitation robotics ontology on the cloud."*Robotics and Computer-Integrated Manufacturing*"4237⇌55‑322/; 0

39. Gonçalves PJS, Torres PMB."*Mpqy ngf i g'tgr tgugpvcvkqp"cr r nkgf "q 'tqdqvke"qtvj qr gf ke"uwti gt{'0Robotics and Computer-Integrated Manufacturing*"4237⇌55< 2/; 0

40. Bayat B, Bermejo-Alonso J, Carbonera J, et al. Requirements for building an ontology for autonomous robots. *IR*"4238⇌65‑68; /: 20

41. IEEE standard ontologies for robotics and automation. *IEEE Std 18722015* 2015:1–60.

42. Crespo J, Barber R, Mozos O. Relational model for robotic semantic navigation in indoor environments."*J Intell Robot Syst*"4239⇌ 8‑839 '''''''''/5; 0

43. Crespo J, Barber R, Mozos O, BeBler D, Beetz M. Reasoning Systems for Semantic Navigation in Mobile Robots. In: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE; 2018. pp. 5654–59.

44. Protégé. Protégé; Access May, 2020. Available from: https://protege.stanford.edu.

45. Reasoner H. Hermit Reasoner; Access May, 2020. Available from: http://www.hermit-reasoner.com.

46. Gangemi A, Borgo S, Catenacci C, Lehmann J. Task taxonomies for knowledge content. *Metokis Deliverable D* 2004;7:2004.

47. Fox M, Long D. PDDL2. 1: An extension to PDDL for expressing temporal planning domains."*jair*"4225⇌42‑83/3460

48. Edelkamp S, Hoffmann J. PDDL2. 2: The language for the classical part of the 4th international planning competition. Technical Report 195, University of Freiburg; 2004.

49. Gerevini A, Long D. Plan constraints and preferences in PDDL3. Technical Report 2005-08-07, Department of Electronics for Automa- tion …; 2005.

50. Cashmore M, Fox M, Long D, et al. Rosplan: Planning in the robot operating system. In: Proceedings of the International Conference on Automated Planning and Scheduling. vol. 25; 2015.

51. Gayathri R, Uma V. Ontology based knowledge representation technique, domain modeling languages and planners for robotic path planning: A survey. *ICT Express* 2018;4:69–74.

52. Gonçalves PJS, Lourenço B, Santos S, Barlogis R, Misson A. Computer vision intelligent approaches to extract human pose and its activity from image sequences."*Electronics*"4242⇌ ‑37; 0