Robot Task Planning using Semantic Maps

Cipriano Galido [a,*] Juan-Antonio Fernández-Madrigal [a]
Javier González [a] Alessandro Saffiotti [b]

[a] *Dept. of System Engineering and Automation, University of Málaga, Spain*
[b] *AASS Mobile Robotics Lab, Örebro University, Sweden*

Abstract

Task planning for mobile robots usually relies solely on spatial information and on shallow domain knowledge, like labels attached to objects and places. Although spatial information is necessary for performing basic robot operations (navigation, localization, and obstacle avoidance), the use of deeper domain knowledge is pivotal to endow a robot with higher degrees of autonomy and intelligence. In this paper, we focus on semantic knowlege, and show how this type of knowledge can be profitably used for robot task planning. We start by defining a specific type of *semantic maps*, which integrate hierarchical spatial information and semantic knowledge. We then proceed to describe how these semantic maps can improve task planning in three ways: enriching the planning domain, relaxing unachievable goals, and improving the efficiency of the planner in large domains. Finally, we show several exepriments that demonstrate the effectiveness of our solutions in a doman involving robot navigation in a domestic environment.

Key words: Task planning, Robot maps, Mobile robotics, Knowledge representation, Cognitive robotics

1 Introduction

In an autonomous robot, task planning is used to plan a sequence of high-level actions that allows the robot to perform a given task. Task planning usually requires that several types of knowledge are encoded in the planner in a suitable way. These include causal knowledge, that is knowledge about the

* Corresponding author. Department of System Engineering and Automation. University of Málaga, Campus de Teatinos. E-29071 Málaga, Spain. Email: `cipriano@ctima.uma.es`.

effects of the robot's actions, and world knowledge, that is knowledge about the objects in the world, their properties and their relations. For example, given the task to fetch a bottle of milk, a task planner might produce a sequence of actions like "Go to the kitchen, dock to the fridge, open the fridge, perceptually acquire the milk bottle, grasp the bottle, close the fridge". To do so, the planner needs to know, e.g., that milk is stored in a fridge and that it is in the kitchen.

Knowledge about the structure and the current state of the world is usually encoded in the form of a *map*. The problem of how to represent, build, and maintain a robot map has been one of the most active areas of research in robotics in the last two decades, and very valuable solutions to this problem are now available [1]. However, most of this work has focused on representations of the spatial structure of the environment, like metric maps [2,3], topological maps [4,5], or appearance-based maps [6]. This kind of maps are needed at the level of navigation planning and execution, but they do not contain the more qualitative types of information needed to perform task planning. For instance, a metric map may represent the shape of a room, but it does not indicate whether this room is an office, a kitchen, or a bedroom – in fact, it does not even indicate that the given shape is a room. In most practical cases, this type of knowledge, which we call *semantic*, is encoded by hand in the domain description of the task planner using some *ad-hoc* language.

This tendency is now changing, and the field of autonomous robotics is witnessing an increasing interest in so-called *semantic maps*, which integrate semantic domain knowledge into traditional robot maps [7–10]. The significance of these maps is that they can provide a mobile robot with deduction abilities (apart from basic skills like navigation, localization, etc.) to infer information from its world even when it has not been completely sensed. The use semantic knowledge, thus, may enable a robot to perform in a more intelligent and autonomous manner. In the previous example, if the robot does not know where the kitchen is, but it has previously observed a microwave at a certain area, it can deduce that such an area is a kitchen, i.e. the place to go for accomplishing the "fetch a milk bottle" task.

In response to this tendency, a few recent works have addressed some issues related to the construction and usage of semantics maps [7,11,12] (see section 2). However, there are still open questions to be solved, including: how these maps can be automatically acquired, how semantic knowledge can be integrated with other types of knowledge in the maps (metric, topological, etc), and how it can be profitably used by the robot to plan and execute tasks.

This paper touches on all the questions, and focuses in particular on the last one last one: if a robot is endowed with an explicit representation of semantic information about its domain, how can a task planner profit from this ability? We explore this question in a constructive way. First, we propose our own *semantic map*, which integrates a spatial hierarchy of objects and places with a semantic hierarchy of concepts and relations. The integration comes from linking elements from the spatial hierarchy to elements at the semantic hierarchy in a general framework. We use this tool to engage in three case studies, referring to three different ways of using semantic information:

(1) For exploiting the semantic structure in order to plan at levels, namely considering general concepts instead of instances. This permits the planner to construct a plan in spite of the lack of knowledge about the existence of a particular element.
(2) For exploiting semantic inference in order to deduce properties of the world elements managed by the planner. For example, the robot can deduce that a visited area is a kitchen since it has perceived a fridge inside.
(3) For exploiting semantic constraints in order to improve the efficiency of planning. Semantics can be used to report the classes of object not involved in the plan that solves a task. Therefore, instances of such irrelevant classes can be discarded before planning.

We complement the above study by reporting two series of illustrative experiments. These experiments demonstrate that the use of semantic knowledge in task planning may endow a robotic system with increased ability to solve tasks with no human intervention (autonomy), to cope with situations that were not explicitly accounted for in its design (robustness).

This paper partly builds upon our previous work on semantics maps and task planning [7,13]. It goes much beyond that work, however, in presenting the following novel contributions: (1) a definition of a specific type of semantic maps that bridges the traditions in robot maps and in knowledge representation; (2) an analysis of different uses of semantic maps in task planning; and (3) an experimental validation of the hypothesis that semantic maps endow a robot with more autonomy and robustness.

The rest of this paper is organized as follows. In the next section we review some related works. Section 3 presents our own semantic map. Section 4 describes different ways to use this semantic map in task planning. Finally, Section 5 reports the illustrative experiments and Section 6 concludes.

2 Related Work

Mapping is probably one of the most important issues addressed in the mobile robotic literature. Its paramount relevance stems from the necessity of having a proper representation of the environment to enable a robot to plan and execute its tasks. In the last decades, a relevant part of research in robotics has focused on robot navigation and localization, and therefore, on maps specifically designed for such tasks. Roughly speaking, those maps are classified into two main representative groups: *metric* and *topological.* Metric maps [2,3] register and represent geometric features of the environment, while topological ones [4,5] represent distinctive points (qualitative, symbolic locations) and their topological relations. Given that both types of representations exhibit advantages and limitations, they can be combined into *hybrid maps* [14–16]. Moreover, hybrid maps, as presented in [17], are general enough to permit the combination of a set of maps of different nature.

Metric and topological maps are sufficient to provide a robot with its most basic functionality: navigate. Today, however, mobile robots are becoming increasingly used in complex scenarios, such as human assistance and entertainment applications [18,19]. In these applications, robots are supposed to possess certain social and cognitive skills on the top of basic navigation ability. Social skills, for instance, require the ability to interact with humans using a human-like language. This requires that the robot is endowed with a representation of the environment that involves human concepts and their semantic relations, in addition to mere metric or topological information.

Recently, the so-called *semantic maps* [20,12,21] have come up to capture the human point-of-view of robot environments, making possible high-level and more intelligent robot development and also human-robot interaction. [1] In order to maintain the usual abilities of the robot, i.e. navigation, semantic maps are normally combined with metric and/or topological representations into a hybrid map [7,21]. Some of these works have considered semantic information for improving both human-robot interaction and the general performance and autonomy of mobile robots by inferring information from semantics. One of the forerunners of this research line is [7], which presents a hybrid semantic map arranged into a two-hierarchical structure that enables a mobile robot to represent rooms and simple objects, e.g. a stove, and to infer implicit information such as "this place must be a kitchen, since I have seen a stove". Subsequent works [11,12,23,21] have gone into this topic in more depth, developing more extensive semantic relations and robust mechanisms for the semi-automatic construction of maps. However, none of them exploits all the valuable possibilities that semantic information provides: they use it to merely classify areas according to their properties (e.g. corridor or room) or to the objects recognized inside (e.g. I have found a TV set in this area, so this is a living room).

[1] The need to include semantic information into the robot representation of the environment was firstly considered in [22].

While most of the existing work on semantic map has addressed the ability of a robot to manage human-like concepts, semantic maps may also have other uses. One of the few works that deal with these other uses is the one presented in [13], in which semantic knowledge is used to improve the efficiency of robot task-planning, by discarding those instances of general classes of objects which are irrelevant for the task at hand. In this paper we present a comprehensive study of the utility and usage of semantic information for robot task planning that also complements that study.

3 The Semantic Map

Our study presupposes that a mobile robot has enough abilities to perceive and map the environment as well as to access and exploit semantic knowledge about it. In this section, we describe how we represent and maintain this knowledge (spatial and semantic) in the form of a semantic map.

3.1 Overview

Our semantic map comprises two separate but tightly interconnected parts: a *spatial box*, or S-Box, and a *terminological box*, or T-Box. Roughly speaking, the S-Box contains factual knowledge about the state of the environment and of the objects inside it, with special emphasis on spatial knowledge. The T-Box contains general semantic knowledge about that domain, giving meaning to the entities which are present in the spatial box in terms of general concepts and relations.

This structure is reminiscent of the structure of hybrid knowledge representation (KR) systems [24], which are now dominant in the KR community. In these systems, the knowledge base consists of a terminological component, called T-Box, that contains the description of the relevant concepts in the domain and their relations; and an assertional component, called A-Box, storing concept instances and assertions about those instances. Our semantic map goes one step further by extending the assertional component to be not simply a list of individuals and facts about these individuals; it also associates these individuals to sensor-level information, and it is endowed with a spatial structure — hence the name S-Box. The representation used in the S-Box borrows from the rich tradition of map representation in robotics [1].

The structure of our semantic map generalizes the structure of other semantic maps encountered in the robotic literature [11,12,21]. In those approaches, elements in a spatial map (usually metric or topological) are linked to labels that

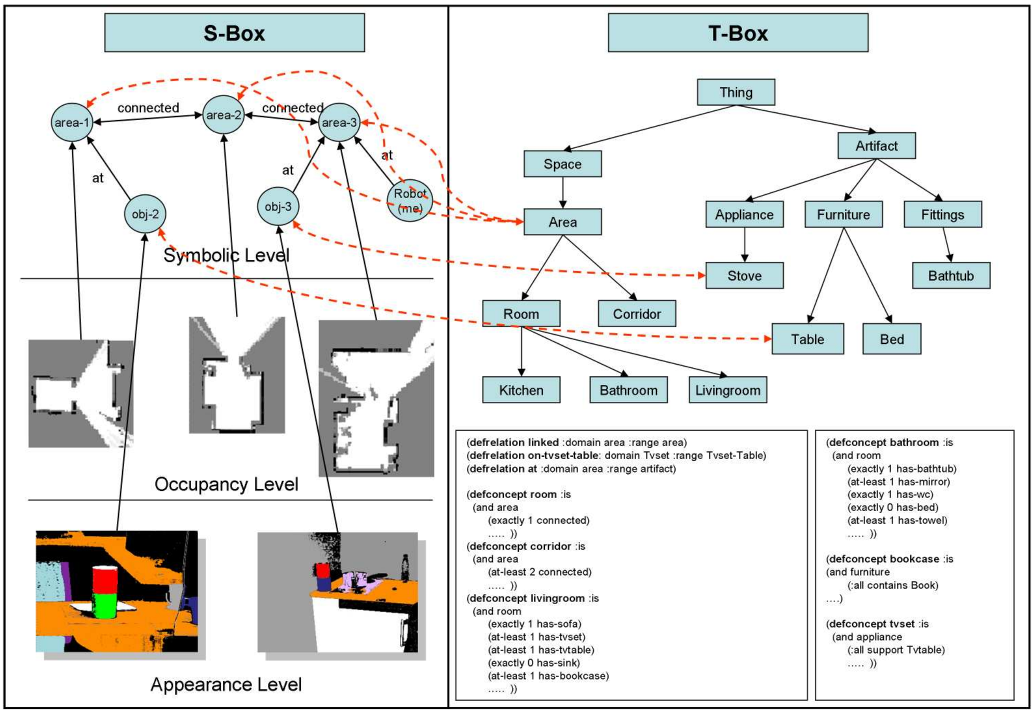

Fig. 1. An example of our semantic map. See explanation in the text.

specify their semantic category. Our semantic map representation is richer, in that semantic labels are part of a full ontology in a KR system, which specifies the relation between semantic categories, and can be used by the robot to perform inference, reasoning, and planning.

Figure 1 shows a simple example of a semantic map. The S-Box and the T-Box have a hierarchical structure. The hierarchy in the T-Box is a direct consequence of the fact that the semantic knowledge represented in it is a taxonomy. For the S-Box, the use of a hierarchical spatial representation is a convenient and common choice in the robotic literature [25,26,22,27] for improving computational efficiency and in particular for dealing with large-scale environments. Of course one could also consider a flat spatial map as a special case of the hierarchical one; this would not cause any essential change in our proposal. In general, we emphasize that in our framework the S-Box can be substituted by any other spatial representation.

3.2 The spatial knowledge

The S-Box represents factual spatial information about the robot environment, including morphology of the space, position and geometry of objects, position of specific places of interest, sensor signatures perceived from those places, topology of connectivity among areas and places, appearance of objects and landmarks, etc. We organize this information hierarchically with different types of knowledge at different levels.

The ground level of the spatial hierarchy, called *appearance level*, contains sensor signatures perceived from the environment plus information about the robot position from where the sensor information is gathered. The sensor signatures considered in our work are images of recognized objects and laser scans of distinctive places (not shown in the figure).

The next level, the *occupancy level*, represents areas of the environment as the result of fusing a number of scans from the appearance level. Map fusion is performed in our work following [28], that considers each scan as a node in a graph whose arcs measure the overlap between scans. The partition of this graph through a recursive minimum normalized cut produces groups of strongly connected nodes from which the map of the area they represent is obtained.

These first two levels (appearance and occupancy) are geometrically linked to indicate the particular area of the space where a sensor signature is perceived. They also can be linked to the T-Box, if the mapping or perception processes are able to classify the spatial entities according to some predefined categories, e.g. rooms, places, and objects.

In our system, the mapping process is only able to perform a very basic classification between areas, gateways, and objects. objects are further classified as sofa, bed, table, sink, stove, TV-set, cup, glass, etc. In the experiments reported later in this paper, we simplify classification by placing unique colored markers on relevant objects. This classification, that can be seen as part of an anchoring process [29], is the fundamental point in the connection between spatial and semantic knowledge. The assumption that the mapping process is able to classify the entities in the map is not restrictive: in the extreme case of a mapper with no classification ability, all entities can be linked to highly general concepts like “Space” or “Object”, or even “Thing”, although this would of course limit the benefits of using semantic knowledge.

The upper level of the spatial hierarchy is the *symbolic level*, that maintains a symbolic representation of the space on which task planning is performed. This level contains a graph that represents the percepts stored at the lower levels (as nodes) and their operational relations (modeled as edges). Nodes are directly created from the percepts, while different types of edges that model different relations, e.g., “connected”, “near”, “at”, etc., are created according to operations related to the geometrical information of the percepts. Notice that this symbolic level represents the topology of the space, including information about the objects found in the environment.

Upon this symbolic level, higher levels can be constructed by grouping (abstracting) symbols following a certain criteria. Arranging the symbolic information into a hierarchical structure can be useful, for instance, to improve the efficiency of task planning [30]. Some works have addressed the automatic construction of symbolic hierarchies to achieve efficiency in mobile robot path-planning [26] and in robot general task-planning [25].

3.3 The semantic knowledge

The semantic knowledge stored in the T-Box consists of general knowledge about the types of the entities in the domain, and how they are related. These are respectively represented in terms of *concepts* and *relations* in the tradition of description logics in knowledge representation [24]. Concepts and relations are structured into a hierarchy, often called an ontology, which provides an abstract description of the entities in the domain, and gives meaning to the terms used in the S-Box. For example, the perceptual signature of an object denoted by the term `area-22` can be associated to the concept `Kitchen` in the ontology. In addition, the T-Box supports inference by exploiting the structure of the ontology. For instance, if the ontology represents the fact that `Kitchen` is a specialization of the concept `Room` and that any `Room` has at least one door, we can infer that `area-22` has a door — although this door has not been explicitly asserted or observed yet.

In practice, we use the Loom knowledge representation system to represent semantic knowledge. In Loom, one defines concepts in relation to other concepts using the primitive `defconcept`. For instance,

```
(defconcept Kitchen
  :is (:and Room (:some has-fitting Sink) (:some has-appliance Stove)))
```

defines a kitchen to be a type of room characterized by having at least one fitting of class `Sink` and one appliance of class `Stove`. The definition

```
(defconcept Office
  :is (:and Room (:all occupant Employee)))
```

defines an office as a room whose occupants are all employees. Figure 1 shows a fragment of the Loom ontology used for the experiments in this paper.

Loom provides a convenient interface to assert and query knowledge using two primitives: `tell` and `ask`, respectively. In addition, Loom provides a number of other functions to update or query the knowledge base: for example, the `retrieve` primitive can be used to find instances that belong to a given concept or that satisfy a given formula, like in

```
  (retrieve ?x (:about ?x Office (:at-least 2 occupant)))
```

which will return all the instances that can be inferred to belong to the class `Office` and to have at least two persons in the role of `occupant`.

3.4 Integrating spatial and semantic knowledge

The basis for the integration of the spatial factual knowledge contained in the S-Box and the semantic general knowledge contained in the T-Box is given by the links between the named entities in the S-Box and the concepts and relations in the T-Box. These links are created during the acquisition of the S-Box by exploiting its classification ability, e.g., a local gridmap may be associated to the concept `Area` and a visual snapshot may be associated to the concept `Table`. In practice, whenever a new entity of a given type is created in the S-Box, it is given a unique name and this name is asserted in LOOM to belong to the given class. For instance, suppose that, during map building the robot creates a new local gridmap in S-Box for a new area. This grid would be called, say, `area-2`, and would be linked to the concept `Area` by calling the LOOM function

```
(tell (Area area-2))
```

In Figure 1 the links so created are indicated by the red dotted lines.

These links alone, however, simply associate labels to entities in the S-Box. The real power of the semantic map comes from the fact that these labels are connected in a full ontology of the domain, and that this ontology can be used to infer new properties of the entities in the S-Box. In practice, the symbolic information in the semantic map is accessed by posting queries LOOM, which perform inferences based on both the assertional knowledge in the S-Box and the semantic knowledge in the T-Box. For instance, to know the instances of the concept `Kitchen`, we issue the LOOM query

```
(retrieve ?x (Kitchen ?x))
```

Notice that there is no area in the map which is explicitly linked to the concept `Kitchen`, since the map builder does not have the ability to discriminate kitchens. LOOM, however, returns `(area-3)` as answer to the above query. In fact, `area-3` has been asserted to be an `Area`, and since it has only one filler of the `connected` role it can be deduced to be a `Room`. Moreover, an object of the class `Stove` has been observed `at` this room, and therefore `area-3` can be further classified as an instance of the class `Kitchen`. In the next section we will see more examples of properties that can be derived by reasoning in the T-Box.

In the reverse direction, the synergies between the T-Box and the S-Box allow the robot to ground the semantic symbols in actual sensor-based entities that can be used for navigation. Continuing the example above, if the robot is given the task, in human-like terms, to go to the kitchen, then the fact that the corresponding symbol `area-3` (which was deduced to be an instance of a kitchen) is linked to an occupancy grid allows the robot to give perceptual meaning to it, which can be used by the navigation routines.

4 Using the Semantic Map in Planning

We now focus on the semantic side of the map and on its benefits for classical robot task planning (STRIPS-like planners) [2]. Although the spatial hierarchy can also provide benefits for planning [30], we will only consider a flat symbolic level, given that spatial planning and the other low-level operations of the robot are not the main concerns of this work, but it is the exploitation of semantics for robot task-planning.

Before the following sections go into deeper detail, we enumerate the advantages of using our hybrid semantic map for robot task planning that we investigate below.

(1) Semantics can be used for generalization (i.e.,induction). Typically task planning works on logical predicates (goals, preconditions, and postconditions) that refer only to particular elements of the world (particular instances of objects, places, etc.). It is possible to use semantics for providing the planner with the capability of using generalized elements too (classes of objects or places). This ability exhibits valuable advantages for solving plans in absence of a complete knowledge of the environment (see section 4.1).
(2) Semantics can be used for enlarging or enriching the state space where planning is carried out. For example, for creating new connections between states or enriching state descriptions. This can help in situations where the absence of that knowledge precludes the construction of a plan (as shown in section 4.1 and 4.2). It can be also used for proposing automatic goals when spatial and semantic information disagree, as discussed in section 4.3.
(3) Semantics can be used for improving task planning efficiency. It can helps the system to detect the relevant classes of objects for the task at hand. In this manner, those classes not involved in the semantic plan (as well as all their instances) can be discarded when planning (see section 4.4).

[2] Probabilistic planning is out of the scope of this work [31].

4.1 Adding new implicit knowledge by semantics

When considering classical STRIPS-like planners, the initial state of the world contains all the information that is available to the planner to achieve the goal at hand. The lack of information within this state may prevent the system to find a plan. Semantics can be useful for enlarging and completing the initial state of the world by including deduced knowledge from the commonsense database (the semantic source). In the limit case, that initial state could be enriched with the entire closure of the semantics (or by applying all the considered domain rules). However, that may not be possible in general due to the amount of information and the computation burden involved.

A way of bounding the deduced information to be added to the initial state is to examine the goal to be reached, retrieving the concepts to which the symbols of the goal belong to from the semantic side, and planning then in semantics (that is, searching for a plan that involves only semantic categories). The categories in the semantic plan can be used to produce new knowledge about the current state of the world by examining if the initial state entails instances of all of them. If not, new instances can be included in the state (since for a category to exist, some instance of it must exist in the world) along with all their relations deduced from semantics. Finally, the definitive plan can be constructed by the planner in the state space according to this bounded semantic closure of the initial state.

This process permits the planner to find a solution for a goal even when some involved world elements have not been sensed before. Let us consider the following example: a servant robot is working in an apartment-like environment and is commanded to approach to the TV-set. If the robot has never seen the TV-set before, a traditional planning system would notify the lack of a plan to achieve the goal at hand. However, semantic information may be used to infer the probable location of a TV-set, and thus, to find a solution. Let the initial state of the robot (partially described from the map shown in figure 1), considering the assertions produced by the map building process, be inserted in LOOM as:[3]

```
(tellm (Kitchen area1) (Corridor area2) (Livingroom area3)
       (connected area1 area2) (connected area2 area3)
       (at robot area3))
```

It is clear that by only considering the spatial information no plan can be generated to approach a TV-set (there is no information about TV-sets in the initial state). However, assuming the semantic taxonomy shown in figure 1, and performing planning on general classes of objects instead of on particular in-

[3] The `tellm` primitive is used in LOOM to make multiple assertions at once.

stances, the goal is translated into "approach TV-set", which can be solved at the semantic side (by a conventional planner adapted to this semantic domain) as ((MOVE Kitchen Corridor) (MOVE Corridor Livingroom) (APPROACH TV-Set)). For this generic plan to be executed it is needed that particular instances of the involved classes, e.g. Kitchen, Corridor, Livingroom, and TV-Set exist, and that those instances fulfill the pre- and post-conditions of the actions of the resultant plan, namely (linked Kitchen Corridor), (at TV-Set Livingroom), etc.

In our example, no instances belonging to the class TV-Set are present in the initial state, so a Skolem instance, let say `obj1` should be created and added to the initial state plus the inferred relation that binds it as a TV-set located at a Livingroom. Thus, the initial state is now completed with:

```
(tellm (TV-set obj1) (at obj1 area3))
```

Notice that, although TV-set and TV-Table are also related in the semantic box, the TV-Table class does not participate in the semantic plan, and thus, no phantom instances have been considered for it.

Through the new initial state, a planner can now construct an instantiated plan, executable by the robot. It is worth to note that all the information included by semantics into the initial state must be validated during execution by the agent, that is, its actual existence must be confirmed. When the agent has to carry out an atomic operation of the plan, that operation may include semantic-generated information (elements and/or predicates) both in the preconditions and in the postconditions, thus the robot should account for enough sensor capabilities to check out if the preconditions of each atomic action hold before it is executed.

4.2 Inferring new properties through semantics

In the previous example, new instances of general classes were created into the spatial map through a partial closure of semantics. Apart from this, semantic information can be also used for the generalization of existing elements. In this manner, a particular symbol from the symbolic level not previously identified as a semantic concept by the map building process, may become an instance of a certain class, inheriting its properties, if it fits on the given semantic description. For instance, considering the previously commented semantic map and initial state completed with:

```
(tellm (at obj2 area4) (at robot area1)
       (connected area2 area4)
       (Bathtub obj2))
```

we can infer that `area4`, which is not an instance of any class is actually a `Bathroom`. That is, given the semantic information represented in figure 1, the LOOM query

```
(retrieve ?x (Bathroom ?x))
```

will provide the planner with `area4`. Through this inferred generalization (property) of the symbol `area4`, the planning system can solve now plans like "go to the bathroom" or planning on implicit knowledge derived from the fact that `area4` is actually a bathroom, for example "approach the washbasin" even when no washbasin has been previously sensed.

4.3 Inferring new goals

In addition to extend the set of achievable goals as shown above, semantic information can also be used to provide a robotic system with automatic goals that should be executed. This can be done by regularly checking whether the anchored symbols fulfill the semantic relations in the semantic box; if not, the corresponding predicate(s) to be added or deleted from the world state can be considered as goals to be planned and executed in order to keep the consistency of the spatial and semantic representations.

For instance, in the case of a servant robot and considering the semantic knowledge that imposes that towels are always at the bathroom, the goal `(at obj3 area4)` would be automatically generated if the the world initial state includes:

```
(tellm (towel obj3) (at obj3 area3))
```

which would make the robot find and execute a plan to bring the towel back to its place.

this usage of semantic information to automatically generate goals provides robots with a higher degree of autonomous and intelligent operation, and can be seen as a simple kind of motivational architecture [32,33].

4.4 improving planning efficiency

semantic information can also be used for improving planning efficiency, issue which is rarely considered in robotics in spite of its relevance when a robot has to deal with large amounts of information. moreover, with the help of semantics, some intractable problems under other planning approaches can become tractable [13].

more precisely, the planning process can be enhanced to check, at the semantic box, which categories are strictly needed for planning the task at hand. the categories not involved for solving the task provide a valuable hint about world information that can be discarded (instances of such categories). thus, the search space of planning can be largely reduced by discarding irrelevant elements before the spatial planning executes, reducing the computational effort.

consider the case of a servant robot that is commanded to take a particular book located on a bookcase at the living room. In a normal situation, the planning system should cope with thousands of objects with their relations and tenths of distinctive places for navigation, which could prevent the planning process from achieving a successful result. Semantic information can certainly helps the planning process to reduce the search space, by initially generating a semantic plan (as commented in section 4.1). For this example and following the maps from previous sections, the semantic plan obtained would be: (MOVE Kitchen Corridor) (MOVE Corridor Livingroom) (APPROACH Bookcase) (TAKE Book). The semantic concepts involved in this plan are the only relevant for the task at hand. Going back to the spatial information, all the instances of non-relevant concepts, e.g. Appliance, Fittings, Household, etc. can be ignored reducing thus the search space. Section 5 reports experiments that confirm that task planning efficiency can be improve using semantic knowledge.

5 Experiments

We now proceed to empirically demonstrate the utility of semantics knowledge, when connected to spatial knowledge inside a semantic map, in the different situations described above. We present two series of experiments. The goal of the first series is to prove that, by using semantic knowledge, a robot can solve task for which it would otherwise not find a solution. This claim is proved constructively, by showing actual situations where this is the case. These experiments have been run on a real robot. The goal of the second series of experiment is to prove that semantic knowledge can improve the efficiency of task planner. This claim is proved statistically, by solving a large number of planning problems in increasingly larger domains. These experiments have been run in simulation, since actual execution of the plans was not relevant.

Fig. 2. Two view of the experimental environment. Left: the robot Astrid in the living room facing a sofa. Right: the robot at the entrance of the kitchen. The colored markers used to detect a sofa, a TV-set and a table are clearly visible.

5.1 Experimental setup

For the real-robot experiments, we used an Activemedia PeopleBot robot called Astrid, equipped with a PTZ color camera and a laser range finder. The robot was placed in a home-like environment constructed inside one of our university building. Local gridmaps were built from laser data using the approach by Blanco and colleagues [28]. Since reliable object recognition is not the focus of this paper, we simplified this problem by tagging each relevant object with a colored marker. A color-based segmentation algorithm [34] together with geometric constraints was used to recognize the markers, which uniquely identify classes of objects. Figure 2 shows two views of the used environment, in which some of the markers are visible.

The T-Box in the semantic map was implemented using Loom, and task planning was done using PTLplan [35]. The ontology in the T-Box and the domains in the planner were hand-coded in these experiments. Figure 1 in section 3 above shows a fragment of the coded ontology.

5.2 Planning on implicit knowledge

The first series of experiments was aimed at demonstrating the ability to use implicit knowledge, inferred through the use of semantic knowledge, in planning. The goal were given to the robot in human-meaningful terms (e.g., go to the fridge). This series consisted of four phases.

In the first phase, the robot explored the environment and built a corresponding semantic map as explained in section 3. Local occupancy grids were built from laser data, while (marked) objects were recognized from camera images based on color segmentation. All local grids were asserted, via the `tell` primitive, to be instances of the general class `Area`. Recognized objects were classified according to the color pattern in their marker, and they were asserted to be instances of the corresponding concept. The position of each detected object relative to the robot was estimated from the observed size of the object, and it was then converted to a position in the local occupancy grid. The link between the object and the grid was included in LOOM by asserting an `at` relation between the corresponding instances. To keep the experiment simple, the robot was tele-operated and its self-localization was based just on odometry. The left side of Figure 1 in section 3 above is a partial view of the S-Map built after exploration. The names of the objects and of the areas are automatically generated identifiers.

In the second phase, the robot was standing in `area-1` (the entrance area) and it was been given the goal to go to the kitchen. This goal was entered into PTLplan as the goal formula

```
(exists (?x) (and (Kitchen ?x) (at me ?x)))
```

that is, the robot should be at a place which is classified as kitchen. If PTLplan only looked at the spatial box, as in most existing systems, then it could not find any plan to satisfy this goal since in Figure 1 there is no place which has been explicitly marked as being of class `Kitchen`. Using semantic knowledge, however, we could classify `area-2` as an instance of the class kitchen, since it contains a stove and stoves are only found in kitchens. In practice, when PTLplan tries to instantiate the variable `?x` in the first conjunct of the goal, it sends LOOM the query

```
(retrieve ?x (Kitchen ?x))
```

obtaining the answer

```
(area-2)
```

To satisfy the (instantiated) second conjunct `(at me area-2)`, then, PTLplan exploits the `connected` links in the spatial hierarchy, and it generates the simple 1-action plan `((MOVE AREA-1 AREA-2))`. This plan can then be executed by the ThinkingCap to produce actual physical motion, since the symbols `area-1` and `area-2` denote concrete entities in the spatial hierarchy.

In the third phase, the robot is given the goal to go to the children bedroom. In our test scenario, both `area-4` and `area-5` have been classified as bedrooms, but not enough objects have been observed in these areas to allow LOOM to further classify one of them as a children-bedroom. This means that the call `(retrieve ?x (ChildrenBedroom ?x))` returns zero instances.

In these cases, we exploit the semantic knowledge in LOOM to convert the lack of instances to a problem of partial observability as follows. We ask

Loom to retrieve all instances of the parent concept(s) of a children-bedroom: in this case there is only one parent concept, `Bedroom`, whose instances are `(area-4 area-5)`. Then, we assume that some of the distinctive properties that turn a bedroom into a children bedroom may have not been observed, and therefore generate a plan to go into each bedroom and gather more observations. PTLplan is able to reason about observation actions and generate a conditional plan that depends on the outcome of these actions. In our case, it generates the following plan:

```
((MOVE AREA-1 AREA-4)
 (OBSERVE-ROOM)
 (COND
   ((IS-A-CHILDREN-BEDROOM AREA-4 = T) :SUCCESS)
   ((IS-A-CHILDREN-BEDROOM AREA-4 = F)
    (MOVE AREA-4 AREA-1)
    (MOVE AREA-1 AREA-5)
    (OBSERVE-ROOM)
    (COND
      ((IS-A-CHILDREN-BEDROOM AREA-5 = T) :SUCCESS)
      ((IS-A-CHILDREN-BEDROOM AREA-5 = F) :FAIL)
))))
```

The action `OBSERVE-ROOM` moves inside a room while collecting data from the vision system until most of the room has been covered.[4] The predicate `(IS-A-CHILDREN-BEDROOM AREA-4)` is implemented by a corresponding call to the Loom subsystem: `(ask (ChildrenBedroom area-4))`. If the observation action has resulted in the observation of some discriminative elements (e.g., a toy), then this call will succeed and the goal to be at the children-bedroom will be satisfied. Otherwise the plan will proceed to explore the other room, and it will succeed or fail depending on the result of that exploration.

In the fourth phase, the robot is given the goal to go near the fridge, which is expressed by:

```
(exists (?x) (Fridge ?x) (near me ?x))
```

No fridge was observed in the exploration phase, and then there is no object in the spatial map which is classified as fridge. Therefore, the direct Loom query

```
(retrieve ?x (Fridge ?x))
```

[4] A smarter strategy would be to extract from the semantic knowledge base the distinctive elements to be observed in order to classify a bedroom as a children bedroom, e.g., a toy, and to use this information to parametrize the vision system or to only observe likely places for toys. This strategy was not attempted in our experiments.

returns no instances. In this situation, we asks LOOM to retrieve all the instances of any place which has at least one fridge, by issuing the following query:

```
(retrieve ?x (and (Place ?x) (about ?x (at-least 1 has-fridge))))
```

The LOOM subsystem returns the single instance (area-2), since this has been classified as being a kitchen, and the semantic knowledge base includes information that a kitchen must include a fridge. At this point, we create a new Skolem symbol FRIDGE-0 to denote the inferred (but unseen) fridge, and asserts it to be of class Fridge and to be inside area-2. Since the fridge has not been observed, we need to strengthen the original goal to require that the fridge be anchored (seen) before we approach it:

```
(exists (?x) (Fridge ?x) (anchored ?x) (near me ?x))
```

From this goal, and given the new Skolem instance, PTLplan generates the following conditional plan:

```
((MOVE AREA-1 AREA-2)
 (OBSERVE-ROOM)
 (COND
   ((ANCHORED FRIDGE-0 = F) :FAIL)
   ((ANCHORED FRIDGE-0 = T) (APPROACH FRIDGE-0) :SUCCESS)
))
```

According to this plan, the robot navigates to the kitchen, explores it, and if FRIDGE-0 has been seen (anchored), then it approaches it. If the fridge is not seen, the plan fails since, although the existence of a fridge can be inferred, no perceptual entity for it can be created in the spatial hierarchy and therefore no data is available for navigation.

5.3 Planning on a large domain

In order to test the use of semantics for improving task planning in large and/or complex scenarios we have performed a number of simulated experiments, using the Metric-FF planner for obtaining both, the semantic and the spatial plans. As commented, the former yields the relevant categories (and therefore, the useful instances) to be considered, while the latter becomes the final solution to the task.

We have considered scenarios in which the number of objects and places within the spatial box have been gradually increased from 100 to 5000, and random "pick up an object" tasks were planned following three different strategies: 1) Considering all the spatial information in a flat structure, 2) Considering

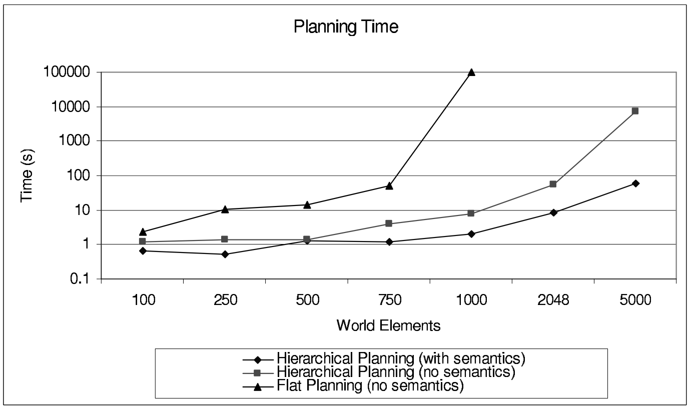

Fig. 3. Task Planning comparison. Average time for planning five random tasks with random variations of the complexity of the simulated environment.

all the spatial information but arranged into two symbolic levels (following [30]), and 3) Considering only the relevant spatial information provided by the semantic plan.

Figure 3 shows the average planning time for a set of random tasks varying the complexity of the environment (number of elements) for case (i) planning only at the symbolic level, case (ii) using a 2-layered symbolic structure (no semantics), and case (iii) exploiting semantics. Although the behavior of each of the three planning strategies follows an exponential trend (which is usual in planning processes), the chart of figure 3 clearly demonstrates the benefits of using semantic information for planning. Also notice that considering semantics shortens the planning time, which proves that it actually alleviates the combinatorial explosion of the search involved in planning by discarding unnecessary objects for the task.

6 Conclusions

This paper has surveyed the application of semantics to robot task planning processes. A well-founded semantic map representation has been proposed to properly integrate spatial knowledge, modeled through the usual techniques found in the robotic literature, and common-sense (semantic) knowledge, modeled through approaches from the AI community. The proposed semantic map has been utilized to enable a mobile robot to plan and execute tasks that could not be completed without the help of semantics.

Among the different usages of semantics for planning, we remark in this paper the possibility of planning on general concepts instead of on particular instances. This provides the planning system with valuable benefits like, for instance, enriching the state space with non-sensed information, deducing automatic goals for the robot, and also improving planning efficiency. Our experiments have revealed the clear utility of the use of semantics in the task planning process of mobile robotics applications.

Finally, we emphasize that our semantic map can be seen as a general framework on which any other spatial or knowledge representations can fit. Our short-term goal is to enlarge the semantic knowledge considered here and deal with other applications of semantics identified in the paper, namely using semantics to improve the sensorial abilities of the robot.

References

[1] S. Thrun, Robotic mapping: A survey, in: G. Lakemeyer, B. Nebel (Eds.), Exploring Artificial Intelligence in the New Millenium, Morgan Kaufmann, 2002.

[2] M. Asada, Map building for a mobile robot from sensory data, IEEE Trans. on Systems, Man, and Cybernetics 37 (6) (1990) 1326–1336.

[3] S. Thrun, D. Fox, W. Burgard, Probabilistic mapping of an environment by a mobile robot, in: Proc. IEEE Int. Conf. on Robotics and Automation, 1998.

[4] H. Choset, K. Nagatani, Topological simultaneous localization and mapping (SLAM): Toward exact localization without explicit localization, IEEE Trans. on Robotics and Automation 17 (2) (2001) 125–137.

[5] B. Kuipers, Y. Byun, A qualitative approach to robot exploration and map-learning, in: Workshop on Spatial Reasoning and Multi-Sensor Fusion, Charles, IL, 1987, pp. 390–404.

[6] B. Kröse, O. Vlassis, R. Bunschoten, Y. Motomura, A probabilistic model for appearance-based robot localization, Image and Vision Computing 19 (6) (2001) 381–391.

[7] C. Galindo, A. Saffiotti, S. Coradeschi, P. Buschka, J. Fernández-Madrigal, J. González, Multi-hierarchical semantic maps for mobile robotics, in: Proc of the IEEE/RSJ Int Conf on Intelligent Robots and Systems (IROS), Edmonton, CA, 2005, pp. 3492–3497.

[8] A. Nüchter, O. Wulf, K. Lingemann, J. Hertzberg, B. Wagner, H. Surmann, 3D mapping with semantic knowledge, in: RoboCup Int. Symposium, 2005.

[9] D. Meger, P. Forssen, K. Lai, S. Helmer, S. McCann, T. Southey, M. Baumann, J. Little, D. Lowe, B. Dow, Curious george: An attentive semantic robot, in: IEEE IROS Workshop: From sensors to human spatial concepts, 2007, pp. 390–404.

[10] A. Ranganathan, F. Dellaert, Semantic modeling of places using objects, in: Robotics: Science and Systems Conf., 2007.

[11] G. Kruijff, H. Zender, P. Jensfelt, H. Christensen, Situated dialogue and spatial organization: What, where... and why?, Int. Journal of Advanced Robotic System (To appear).

[12] O. Mozos, P. Jensfelt, H. Zender, M. Kruijff, W. Burgard, From labels to semantics: An integrated system for conceptual spatial representations of indoor environments for mobile robots, in: IEEE ICRA Workshop: Semantic Information in Robotics, 2007.

[13] C. Galindo, J. Fernandez-Madrigal, J. Gonzalez, A. Saffiotti, Using semantic information for improving efficiency of robot task planning, in: IEEE ICRA Workshop: Semantic Information in Robotics, 2007.

[14] K. Kouzoubov, D. Austin, Hybrid Topological/Metric Approach to SLAM, in: IEEE ICRA, New Orleans (LA), USA, 2004, pp. 872–877.

[15] S. Thrun, A. Bücken, Integrating grid-based and topological maps for mobile robot navigation, in: 13th National Conf. on Artificial Intelligence, Portland, address=Portland, Oregon, pages=944–951, 1996.

[16] N. Tomatis, I. Nourbakhsh, R. Siegwart, Hybrid simultaneous localization and map building: a natural integration of topological and metric, Robotics and Autonomous Systems 44 (2003) 3–14.

[17] P. Buschka, A. Saffiotti, Some notes on the use of hybrid maps for mobile robots, in: Proc of the 8th Int Conf on Intelligent Autonomous Systems (IAS), Amsterdam, NL, 2004, pp. 547–556.

[18] W. Burgard, A. Cremers, D. Fox, D. Hahnel, G. Lakemeyer, D. Schulz, W. Steiner, S. Thrun, Experiences with an interactive museum tour-guide robot, Artificial Intelligence 114 (1–2).

[19] R. Simpson, Smart wheelchairs: A literature review, Rehabilitation Research and Development 42 (4) (2005) 423–436.

[20] P. Beeson, M. MacMahon, J. Modayil, A. Murarka, B. Kuipers, B. Stankiewicz, Integrating Multiple Representatios of Saptial Knowledge for Mapping, Navigation, and Communication, Interaction Challenges for Intelligent Assistants, AAAI Spring Symposium Series, 2007.

[21] H. Zender, P. Jensfelt, O. Martinez-Mozos, G.-J. Kruijff, W. Burgard, An integrated robotic system for spatial understanding and situated interaction in indoor environments, in: AAAI-07, Integrated Intelligence Track, 2007.

[22] B. Kuiper, Modeling spatial knowledge, in: S.Chen (Ed.), Advances in Spatial Reasoning, Vol. 2, The University of Chicago Press, 1990, pp. 171–198.

[23] S. Vasudevan, S. Gachter, V. Nguyen, R. Siegwart, Cognitive maps for mobile robots– and object based approach, Robotics and Autonomous Systems 55 (2007) 359–371.

[24] F. Baader, D. Calvanese, D. McGuinness, D. Nardi (Eds.), The Description Logic Handbook, Cambridge University Press, 2007.

[25] C. Galindo, J. Fernandez-Madrigal, J. Gonzalez, Multiple Abstraction Hierarchies for Mobile Robot Operation in Large Environments, Studies in Computational Intelligence, Vol. 68. Springer Verlag, 2007.

[26] J. Fernandez-Madrigal, J. Gonzalez, Multi-Hierarchical Representation of Large-Scale Space, Int. Series on Microprocessor-based and Intell. Systems Eng., vol 24, Kluwer Academic Publishers, Netherlands, 2001.

[27] B. Lisien, D. Morales, D. Silver, G. Kantor, I. Rekleitis, H. Choset, The hierarchical atlas, IEEE Trans. on Robotics 21 (3) (2005) 473–481.

[28] J. Blanco, J. Gonzalez, J. Fernandez-Madrigal, Consistent observation grouping for generating metric-topological maps that improves robot localization, in: IEEE International Conference on Robotics and Automation (ICRA), 2006.

[29] S. Coradeshi, A. Saffiotti, An introduction to the anchoring problem, Robotics and Autonomous System 43 (2-3) (2003) 85–96.

[30] C. Galindo, J. Fernandez-Madrigal, J. Gonzalez, Hierarchical task planning through world abstraction, IEEE Trans. on Robotics 20 (4) (2004) 667–690.

[31] S. Thrun, W. Burgard, D. Fox, Probabilistic Robotics, MIT Press, Cambridge, MA, 2005.

[32] A. Stoytchev, R. Arkin, Incorporating motivation in a hybrid robot architecture, Journal of Advanced Computational Intelligence and Intelligent Informatics 8 (3) (2004) 100–105.

[33] M. Quoy, P. Laroque, P. Gaussier, Learning and motivational couplings promote smarter behaviors of an animat in an unknown world, Robotics and Autonomous Systems 38 (3–4) (2002) 149–156.

[34] Z. Wasik, A. Saffiotti, Robust color segmentation for the robocup domain, in: Proc of the Int Conf on Pattern Recognition (ICPR), Quebec City, Quebec, CA, 2002.

[35] L. Karlsson, Conditional progressive planning under uncertainty, in: Proc of the Int Joint Conf on Artificial Intelligence (IJCAI), Seattle, USA, 2001, pp. 431–438.