Robot, tell me what you know about...?: Expressing robot's knowledge through interaction

Raquel Ros, E. Akin Sisbot, Séverin Lemaignan, Amit Pandey and Rachid Alami
CNRS - LAAS, 7 avenue du Colonel Roche, F-31077 Toulouse, France
Université de Toulouse, UPS, INSA, INP, ISAE, LAAS, F-31077 Toulouse, France
Email: {rrosespi, slemaign, easisbot, akpandey, rachid}@laas.fr

Abstract— **Explicitly showing the robot's knowledge about the states of the world and the agents' capabilities in such states is essential in human robot interaction. This way, the human partner can better understand the robot's intentions and beliefs in order to provide missing information that may eventually improve the interaction. We present our current approach for modeling the robot's knowledge from a symbolic point of view based on an ontology. This knowledge is fed by two sources: direct interaction with the human, and geometric reasoning. We present an interactive task scenario where we exploit the robot's knowledge to interact with the human while showing its internal geometric reasoning when possible.**

I. Introduction

"Why is this robot doing this now?!" This is the typical question that at some point a user asks herself when interacting with a robot. And most probably, not only naive users, but also robot designers when working with their robots. Understanding and knowing the robot's internal knowledge and reasoning states is fundamental to improve any type of interaction. Feedback is specially crucial when a problem occurs, or when the robot makes an unexpected decision. Ideally, this feedback should be given through a natural dialogue where the robot explains its decisions and actions. In order to have robots capable of reasoning on their own internal states to naturally communicate with their human partners, several supportive mechanisms should be considered.

In this work we introduce several mechanisms and their connection applied to a face-to-face interactive task. In this task the human asks the robot about its knowledge on objects in the environment and about its reasoning on the agents' abilities in the world. Thus, we present an approach for modeling the robot's knowledge based on an ontology (Sec. II) and a geometric reasoner that transforms geometric world information into symbolic descriptions (Sec. III). A decisional reasoner interprets the human query (entered through keyboard) in order to identify the referred object (Sec. IV) and then queries the robot's knowledge about it to answer the human. Human queries are limited to a fix vocabulary and a specific format (interpretation of natural language is out of the scope of this work). The reply is at least given by spoken language (and written on the screen), and if possible, by visual feedback from the robot's internal 3D environment model.

This research was supported by a Marie Curie Intra European Fellowship and the European Community's Information and Communication Technologies within the 7th European Community Framework Programme under grant agreements no. [220368], ARBI and [215805] CHRIS projects.

II. Knowledge Representation

We believe that the knowledge model of a robot should include a comprehensive model of the roles, relationships and context of objects in the environment, as well as beliefs and intentions of other agents. Moreover, this understanding must rely on a formal encoding that requires high expressivity while remaining well suited for machine processing in order to be used by the robot.

We thus propose the use of ORO (the "OpenRobot Ontology" server), a central knowledge repository that stores, manages, processes and exposes knowledge for the robot from a symbolic point of view. It internally relies on RDF-derivate OWL Description Logics to formally represent statements on the world as triples `<subject> <predicate> <object>`. It uses two open-source libraries: *Jena* for storage and manipulation of statements and *Pellet* first-order logic reasoner to classify, apply rules and compute inferences on the knowledge base [1].

ORO defines an initial *upper* ontology for human-aware robotics called *OpenRobots Commonsense Ontology*. This initial ontology contains a set of concepts, relationships between concepts and rules and defines the "cultural background" of the robot, i.e. the a priori known concepts. Currently, this commonsense knowledge is focused on the requirement of human-robot interactions in everyday environments, but contains as well generic concepts like `thing`, `object`, `location` and relationships between those. The common-sense ontology design relies heavily on the standard OpenCyc upper ontology for the concepts naming, thus ensuring a good compatibility with other knowledge bases. Figure 1 illustrates a simple example with some concepts.

Besides simply storing and reasoning about knowledge, ORO offers several useful features for human-robot interaction. One advantage offered by the ORO architecture is that independent cognitive models for each agent can be maintained. When the robot interacts with a new agent, a separate RDF triple storage is created to store the robot's knowledge about the agent's perception. For instance, in the case of perspective taking, we compute the visibility and spatial information about the world from each agent point of view, and store it in their own cognitive models. Having

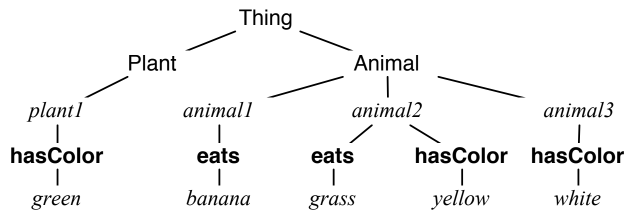
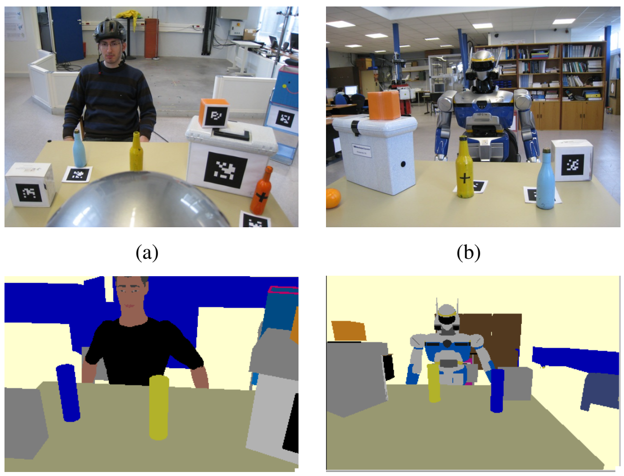


Fig. 1. Ontology example. Names with first capital letter correspond to classes; bold names, to properties; and italic names, to instances.



(a) (b) (c) (d)

Fig. 2. Visual perspective taking for each agent in the scenario.

separate cognitive models allows us to store and reason on different models of the world.

III. Geometrical Reasoning

This section describes different reasoning mechanisms to provide an abstraction layer to the decisional layer on top of the geometrical description of the environment.

To model the environment we use the software platform Move3D [2]. The kinematic structures of the human and the robot, as well as their positions and objects' positions are integrated into this platform to maintain a coherent model of the real environment. It also allows us to view the visual perspective of the agents in the world by modeling their visual sensors (eyes for humans, cameras for robots).

We divide the geometrical reasoning mechanisms in two groups: perspective taking descriptors and symbolic location descriptors. The first set corresponds to information obtained when reasoning from an agent point of view, while the second one corresponds to global descriptors independent of the agents in the environment. All this information is stored in the ontology, which in turn may infer additional information as we explain next. Moreover, the information concerning specific agents, i.e. perspective taking descriptors, is stored in each agent's cognitive kernel in ORO allowing the decisional level to reason about each agents' beliefs about the world.

A. Perspective Taking Descriptors

1) Visibility: Visual perspective taking refers to the ability for visually perceiving the environment from other's point of view. This ability allows us to identify objects or regions that are visible/hidden for/from others, which is essential for referring to things when interacting with others. For example, if the robot refers to an object, it should try to ensure that the human is able to see it in order to facilitate interaction. On the contrary, if the human refers to an object, based on the context, she could refer to a visible one (e.g. "take this ball") or to an invisible one (e.g. "find the ball").

We are currently able to compute "visibility" from an agent point of view for objects in the environment [3] and zones or regions around the agent [4]. An object or a region is visible for an agent if, while performing a minimum effort (i.e. only turning the head or standing, if possible), the object or region are within the agent's field of view and there are no occlusions in between.

2) Spatial: Spatial perspective taking refers to the qualitative spatial location of objects (or agents) with respect to a frame of reference (eg. the keys on my left). Based on this frame, the spatial description of an object varies. Humans mix perspectives frequently during interaction [5], i.e. they do not maintain a consistent perspective through a conversation. Therefore, the robot has to be able to understand and compute descriptions of objects based on different frames of reference to follow the interaction with its human partner.

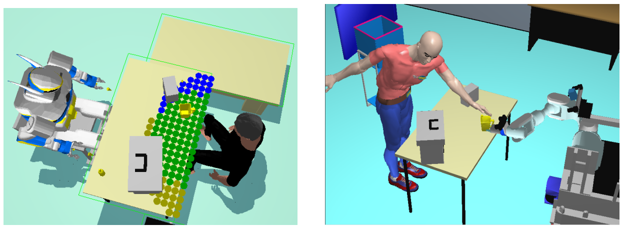
In this work, we use two types of the frames of reference: egocentric (from the robot perspective) and addressee-centered (from the human perspective). Thus, given an object and the referent, we can compute the spatial locations by dividing the space around the referent into n regions based on arbitrary angle values relative to the referent orientation. For example, for $n = 4$ we would have the space divided into front, left, right and back. Further subdivisions can be computed if we would like to represent distinctions among distances, e.g. near and far.

3) Reachability: An object or a region is reachable if there is a collision free posture for the agent where the end-effector is at the center of the object or region with a given tolerance. A valid posture includes moving the upper-body or standing, if possible.

This ability allows the robot to estimate the agent's capacity to reach an object, which is fundamental for task planning. For example, if the human asks the robot to give her an object, the robot must compute a transfer point where the human will be able to get the object. Figure 3 illustrates the reasoning results for reaching regions and an object.

B. Symbolic Location Descriptors

Symbolic location descriptors allow the robot to compute spatial relations between objects in the environment. The system infers symbolic relations between objects from its 3D geometric world representation. In this work we propose the use of three basic symbolic relations between each pair of objects. However, their inverse relations can be automatically computed at the symbolic level, i.e. through inference



(a) (b)

Fig. 3. (a) Reachable points from the human perspective when bending: yellow, blue and green points correspond to left hand, right hand and both hands respectively. (b) Human and robot posture for reaching the cup.

based on OpenRobots Commonsense Ontology, enlarging the symbolic descriptions knowledge easily.

- *IsIn*: indicates if an object (or an agent) is inside of another object. Its inverse relation corresponds to *Contains*.
- *IsOn*: indicates if an object (or an agent) is placed on top of another object. Its inverse relation is *IsUnder*[1].
- *IsNextTo*: tests if an object (or an agent) is next to another object. It has no inverse relation, but symmetric.

IV. Finding the referent

Given partial (or complete) description of an object (list of attribute-value pairs), the robot is able to identify the referred object the following way. First it obtains all objects that fulfill the initial description. Based on the result it either succeeds (obtains one single object), fails (no object with that description could be found) or obtains several objects. In this latter case, a new descriptor is added to the initial description and the process starts over again. Failure occurs when the description does not match any object from the robot's knowledge. This could be because the robot's knowledge is incomplete (the human refers to an unknown descriptor or descriptor value) or due to inconsistent information (human's and robot's beliefs are different).

In order to automatically add a new descriptor (attribute-value pair), the reasoning engine must find the best discriminant for the current list of objects being evaluated. If found, the robot asks the human for its value. Discriminants are descriptors that allow a maximum discrimination among a set of individuals (eg. color, type, location, etc.). We distinguish two types of discriminants. *Complete* discriminants are those attributes (or properties) whose values can uniquely identify those individuals. However, they are not always available. First, because two or more individuals may share the same value, and second, because not all individuals may share the same properties. Thus, *partial* discriminants are those properties that "better" split the set of individuals in different subsets based on some criteria. In the task we propose in this work we only make use of complete discriminants, although partial discriminants may be useful for other tasks, as the Spy Game introduced in [6].

[1]We consider that there is a physical contact between both objects, although the English definition of under does not necessarily imply it.

The algorithm to find discriminants has the following steps (to better follow it we show an example corresponding to the ontology shown in Fig. 1). We search a discriminant for the following individuals: $plant_1, animal_2$ and $animal_3$. First we obtain the direct properties for all the individuals, i.e. we do not consider all the hierarchy of properties. In the example, we only take the most direct class for $plant_1$, i.e. the class `plant` (and not the class `thing`). Next, we compute the number of individuals per property and the number of different values for that property. If there is more than one different value for the property (in other words, if not all individuals have the same value), then we consider that property as a potential discriminant. Finally, we sort the list of potential properties following two criteria: number of individual occurrences (i.e. the most individuals are covered by that property, the better) and values occurrences (i.e. the more distinct values, the better). The best discriminant corresponds to the first element of the sorted list. If several properties are equal, return all of them. In our example, the algorithm would return the property `hasColor`.

V. Application

We have designed an interactive task that exploits the robot's knowledge while using the different mechanisms presented in this work. The scenario for the task consists in a face-to-face interaction around a table with objects. The human may ask the robot the following questions:

1) Where is the object_description?: The robot indicates the location of the object based on spatial perspective taking and symbolic location descriptors.

2) Is the object_description visible?: The robot computes the visibility of the object from both agents' perspectives (robot and human) and indicates whether the object is visible or not. If it is, it also indicates if it is directly visible (within the agent's current FOV) or if the object is visible by turning the head (out of FOV). The view of the agents is displayed in the screen at the same time (Figures 2c and 2d).

3) Is the object_description reachable?: The robot computes the object's reachability from both agents' perspectives (robot and human) and indicates whether the object is reachable or not. The screen displays the movement of the agent for reaching the computed posture (Figure 3b).

From the human query, the decisional reasoner extracts the description of the referred object. The description can be either the id (eg. YELLOW_BOTTLE) or a set of attributes about the object (eg. yellow bottle). In the first case, if the id corresponds to a known object, the robot can directly answer the question. On the contrary, if the object is unknown, then a learning phase takes place, where the human describes recursively the type of the object until a known type is reached. Figure 4 shows an example of the learning process where the robot asks for the type of object until recognizing a known type. In the second case, where attributes describing the object are given, the reasoning engine has to figure out the id of the object. However, the information provided by the human may be incomplete or ambiguous. Therefore, the robot must ask the human additional information until

human	(Shows an object to the robot).
robot	(Identifies an unknown object.)
	What's the name of the object?
human	mist
robot	What kind of object is mist?
human	laptop
robot	What is a laptop?
human	computer
robot	What is a computer?
human	object
robot	Now I know what mist is.

Fig. 4. Learning process example where the robot learns that mist is a computer.

human	Where is the bottle?
robot	*(Available object ids: BLUE_BOTTLE, YELLOW_BOTTLE and ORANGE_BOTTLE.)*
	(Complete discriminants: hasColor)
	Which color is the object?
human	Yellow.
robot	The YELLOW_BOTTLE is located:
	- from my perspective: FRONT
	- from your perspective: LEFT
	- on HRP2TABLE.
human	Is the orange bottle visible?
robot	*(Available object ids: ORANGE_BOTTLE)*
	The ORANGE_BOTTLE is visible for me (out of FOV) and not visible for you.
human	Is the white box on my left reachable?
robot	*(Available object ids: SPACENAVBOX)*
	The SPACENAVBOX is reachable for me and reachable for you.

Fig. 5. Interactive task example.

identifying the object [7], [6]. Figure 5 illustrates different interaction scripts where the robot finds out the referred object and answers the human query.

VI. Related Work

Few work has been developed within the robotics field regarding perspective taking reasoning, which is fundamental for enhancing social interaction ([8],[9],[10]). Trafton et al. use both visual and spatial perspective taking for finding out the referent indicated by a human partner [10]. In [11], they also design a robot that is able to play hide and seek a child does. The strategy they model in the robot is to find those places that are not visible for the human partner. Berlin et al. [12] present their work focused on using visual perspective taking skills for learning from a human teacher. A teacher classifies objects in a given way. The robot then learns the classification function based on the teacher's visual perception of the world. Johnson and Demiris [13] apply visual perspective taking for action recognition. In their work, a robot who has complete visual access of the environment observes another robot with partial access performing a task. The first robot can recognize the task performed by the second robot because it is able to reason about its partial perception. The most significant work for computing reachability has been introduced by Zacharias et al. [14], but only from the robot point of view and not the human, as we do in our work.

The novelty of our work is that we combine (1) different geometric reasoning mechanisms from both, human and robot perspective, which allows us to reason about the agent's capacities with (2) a symbolic knowledge representation, which allows us to reason about the agents' knowledge about the state of the world.

VII. Conclusions

We have presented a set of mechanisms to ease interaction between humans and robots while communicating the robot's internal knowledge about the world. More precisely, we have introduced a model for knowledge representation along with a geometric reasoning engine that provides symbolic descriptions of geometric relations, as well as agent's abilities. The overall system is completely platform independent and has been integrated in two different platforms.

References

[1] S. Lemaignan, R. Ros, L. Mösenlechner, R. Alami, and M. Beetz, "Oro, a knowledge management module for cognitive architectures in robotics," 2010, submitted to IROS.

[2] T. Siméon, J.-P. Laumond, and F. Lamiraux, "Move3D: A generic platform for path planning," in *IEEE International Symposium on Assembly and Task Planning, ISATP*, 2001, pp. 25–30.

[3] L. F. Marin-Urias, E. A. Sisbot, and R. Alami, "Geometric tools for perspective taking for human-robot interaction," in *7th International Conference on Artificial Intelligence*, 2008.

[4] A. K. Pandey and R. Alami, "Mightability maps: A perceptual level decisional framework for co-operative and competitive human-robot interaction," 2010, submitted to IROS.

[5] B. Tversky, P. Lee, and S. Mainwaring, "Why do speakers mix perspectives?" *Spatial Cognition and Computation*, vol. 1, 1999.

[6] R. Ros, S. Lemaignan, E. A. Sisbot, R. Alami, J. Steinwender, K. Hamann, and F. Warneken, "Which One? Grounding the Referent Based on Efficient Human-Robot Interaction," 2010, submitted to RO-MAN.

[7] R. Ros, E. A. Sisbot, R. Alami, J. Steinwender, K. Hamann, and F. Warneken, "Solving ambiguities with perspective taking," in *In Proceedings of International Conference on Human-Robot Interaction*, 2010, pp. 181–182.

[8] B. Tversky and B. M. Hard, "Embodied and disembodied cognition: Spatial perspective-taking," *Cognition*, vol. 110, pp. 124–129, 2009.

[9] S. Wu and B. Keysar, "The effect of culture on perspective taking," *Psychological Science*, vol. 18, no. 7, pp. 600–606, 2007.

[10] J. G. Trafton, N. L. Cassimatis, M. D. Bugajska, D. P. Brock, F. E. Mintz, and A. C. Schultz, "Enabling effective human-robot interaction using perspective-taking in robots," *IEEE Transactions on systems, man and cybernetics - Part A: Systems and Humans*, vol. 35, no. 4, pp. 460–470, 2005.

[11] J. G. Trafton, A. C. Schultz, , D. Perznowski, M. D. Bugajska, W. Adams, N. L. Cassimatis, and D. P. Brock, "Children and robots learning to play hide and seek," in *Proceedings of the 1st Conference on Human-Robot Interaction*. ACM, 2006, pp. 242–249.

[12] M. Berlin, J. Gray, A. L. Thomaz, and C. Breazeal, "Perspective taking: An organizing principle for learning in human-robot interaction," in *In Proceedings of AAAI*, 2006.

[13] M. Johnson and Y. Demiris, "Perceptual perspective taking and action recognition," *International Journal of Advanced Robotic Systems*, vol. 4, no. 4, pp. 301–308, 2005.

[14] F. Zacharias, C. Borst, and G. Hirzinger, "Capturing robot workspace structure: Representing robot capabilities," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2007, pp. 3229–3236.