A Representation for Spatial Reasoning in Robotic Planning

Masoumeh Mansouri[1] and Federico Pecora[1]

Abstract— **In order to close the sense-plan-act loop, a robot requires several capabilities: it must match perceived context with general knowledge about the environment, instantiate plans into the metric space of the real world, and detect and react to contingencies. All of these capabilities include some form of spatial reasoning — however, at different levels of abstraction. Perception generates metric spatial knowledge, while general knowledge about the environment is often qualitative in nature. Similarly, plans may call for the achievement of qualitative spatial relations, but actions must be precisely instantiated in metric space. This paper focuses on integrating qualitative and metric spatial reasoning for closing the loop around perception and actuation. We propose a knowledge representation and reasoning technique, grounded on well-established spatial calculi, for combining qualitative and metric knowledge and obtaining solutions expressed in actionable metric terms.**

I. Introduction

When we plan to achieve activities we rely on several, different abstractions of the world around us. These abstractions, many of which are learned through experience, are often qualitative: we know that knives should be put to the right of dishes and forks to the left when setting a table. When performing the actions to set the table, this qualitative knowledge is used to instantiate precise placing actions in metric space.

Whereas bridging this gap is natural for humans, it is not at all evident how to endow robots with this capability. A robot's world is entirely metric: it perceives events and carries out actions in metric time, it can localize, displace itself and perceive objects in a reference frame. Yet specifying sophisticated robot behavior in purely metric terms is difficult, as the specification would have to be long and overly specialized to the particular setting in which the robot operates. Conversely, qualitative representations facilitate modeling by humans — although they often fail to capture the details that are necessary for proper execution. Integrating temporal, spatial, and other reasoning capabilities in the sense-plan-act loop is an important step towards building general purpose robots (see Related Work).

In this paper we focus on spatial knowledge. We address the issue of bridging the gap between the metric and qualitative dimensions of a robot's knowledge (whether it is perceived or modeled). We attach metric semantics to qualitative spatial relations and state formal properties of the obtained calculus. These properties enable reasoning in three important phases of the sense-plan-act loop, namely (1) matching perceived context with general knowledge about the environment, (2) instantiating plans into the metric space of the real world, and (3) detecting and reacting to contingencies. In all three processes, different levels of abstraction are used: perception generates metric spatial knowledge, while general knowledge about the environment is often qualitative in nature; similarly, plans may call for the achievement of qualitative spatial relations, but actions must be precisely instantiated in metric space.

Our knowledge representation formalism is grounded on well-established spatial calculi, and allows to uniformly account for metric and qualitative spatial knowledge in processes (1), (2) and (3). The calculus affords efficient algorithms for use on-line, and is expressive enough to avoid strong assumptions when modeling the perceived world. The representation is based on spatial relations which allow to model both topology and direction. Along the course of this paper, we will explicitly refer to processes (1–3).

II. Related Work

Bridging the gap between the robot's metric world and its symbolic knowledge is an issue that has been studied in automated planning [1]. Planning domains provide a causal abstraction of the real world, based on which a robot can derive which actions should be performed to achieve a given goal. AI techniques (predominantly constraint-based) have been explored for the purpose of bridging the gap between symbolic planning problems and real-world execution with metric time [2] and resources [3]. However, time and resources remain virtually the only metric aspects of a robot's environment that are considered at a qualitative level in the sense-plan-act loop.

Work on combining qualitative spatial knowledge (e.g., knives should be placed to the right of forks) with perception, planning and actuation is sparse. This problem has been addressed in the context of perceptual anchoring [4], proposing qualitative spatial relations for scene interpretation (partially addressing point (1) above). The work is an example of integrating metric and qualitative spatial relations, in that qualitative relations are inferred from observed metric relations. Work in Cognitive Vision has addressed this issue [5] by focusing on scene understanding. Work on structural pattern recognition [6] has also provided techniques for matching qualitative spatial knowledge (representing a specified structure) to perceived context. In all of the above, qualitative relations do not belong to a well-defined calculus, which would facilitate logical reasoning, rather they are tailored to capture specific features (e.g., distance, orientation, shape) which are useful for pattern specification and recognition in the particular application.

[1]Center for Applied Autonomous Sensor Systems, Örebro University, SE-70182 Sweden. {mmi, fpa}@aass.oru.se

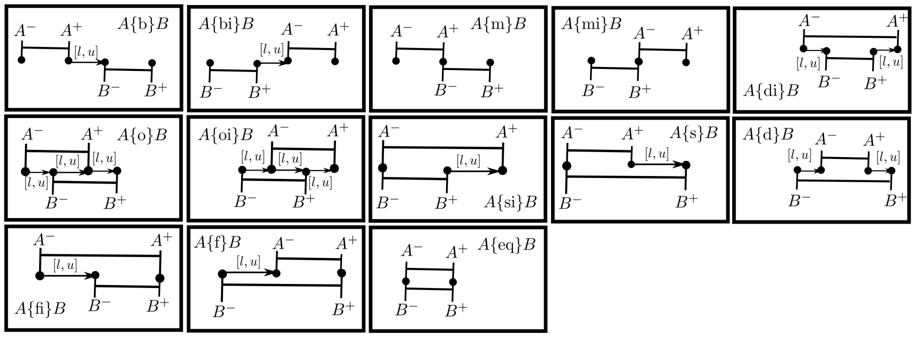


Fig. 1. The thirteen relations in Allen's Interval Algebra with specified bounds.

Significant work has been done in endowing robots with metric (as opposed to qualitative) spatial reasoning capabilities. Many focus on geometric reasoning, some employing metric constraints in combination with planning [7], thus partially addressing point (2), others proposing ad-hoc metric spatial reasoning for analyzing perceived context [8].

Robotics has leveraged the richness of qualitative spatial calculi predominantly for representation rather than reasoning [9]. Examples include robot navigation and self-localization [10], motion planning [11] and task planning [12]. Research has started to study how to combine qualitative and metric spatial reasoning in robotics, specifically by modeling qualitative knowledge through domain-specific predicates and performing metric spatial reasoning though procedural attachment (point (2) above) [13]. To the best of our knowledge, no work has employed well-founded qualitative spatial calculi in conjunction with metric spatial reasoning.

III. Combining Qualitative and Metric Relations

Spatial expressions in natural language assert properties like distance, size, shape, topology, and direction. Although often completely qualitative, these expressions subsume numerous specific metric relations. The spatial calculus that is chosen for representing a robot's knowledge should have a similar level of abstraction, as it is often humans who specify this knowledge. There exist several well-known and well-studied qualitative calculi, most of which represent spatial relations as *constraints*. Each of these constraint-based calculi focuses on one category of spatial concepts — e.g., Region Connection Calculus (RCC) [14] allows to represent topologies, while Cardinal Direction Calculus (CDC) [15] is appropriate for orientation. Rectangle Algebra (RA) [16] can be regarded as an approach to combining topological and cardinal relations. RA is an extension of Allen's Interval Algebra (IA) [17] to two dimensions. It considers as a spatial entity a bounding box (rectangle) whose sides are parallel to the axes of some orthogonal basis in a two-dimensional Euclidean space.

The set of atomic relations in RA is defined as $B_{\mathrm{RA}} = \{\langle r_1, r_2\rangle : r_1, r_2 \in B_{\mathrm{IA}}\}$ where B_{IA} is the set of Allen's Interval relations, namely the thirteen possible relations between intervals (see Figure 1): "before" (b), "meets" (m), "overlaps" (o), "during" (d), "starts" (s), "finishes" (f), their inverses (e.g., bi), and "equals" (eq). The set of RA relations is the power set of B_{RA}. Each RA relation is a disjunction of atomic relations that model the possible mutual placement of two spatial entities represented as axis-parallel rectangles. For instance (see Figure 2), if object B is in the relation

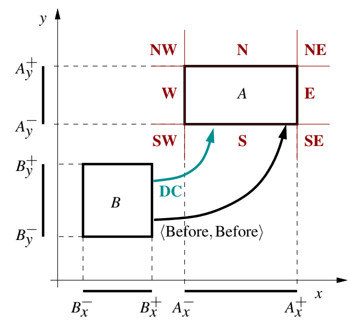


Fig. 2. Relation $B\,\langle \mathrm{b},\mathrm{b}\rangle\,A$ in RA.

$\langle \mathrm{b},\mathrm{b}\rangle$ with object A, then A is Northeast of B (relation in CDC); also, these two objects are disjoint from each other (relation in RCC). Relations between bounding boxes can be represented as binary constraints in a constraint network:

Definition 1: A *rectangle constraint network* is a pair $N = (V, C)$, where

- $V = \{V_1, \ldots, V_n\}$ is a set of *variables* representing axis-parallel rectangles;
- $C : V \times V \to 2^{B_{\mathrm{RA}}}$ is a *mapping* which defines the binary constraints over the variables.

In the following paragraphs we outline two incremental additions to RA which facilitate the important processes in the sense-plan-act loop (1–3).

A. Attaching Bounds to RA Relations

Given a set $V = \{A_1, \ldots, A_n\}$ of intervals, we denote $A_1^-, A_1^+, \ldots, A_n^-, A_n^+$ the extreme points of the intervals in V (see Figure 2). Given a rectangle A, A_x is the interval corresponding to the projection of A onto the first axis (resp. A_y for the second axis).

Qualitative relations subsume metric relations between rectangles. For instance, $B\,\langle \mathrm{b},\mathrm{b}\rangle\,A$ represents all possible placements in which $A_x^- > B_x^+$ and $A_y^- > B_y^+$. We wish to attach bounds to the metric semantics of RA, and we choose to do so through minimum and maximum distances between the axis intervals. For instance, $B \langle \mathrm{b}[5,13], \mathrm{b}[0,+\infty) \rangle A$ restricts placements to those in which $A_x^- > B_x^+ + 5$ and $A_x^- < B_x^+ + 13$. Constraining the distance between points is a well known concept in temporal reasoning [18], where metric knowledge is represented as *simple distance constraints* in the form $l \leq v_2 - v_1 \leq u$, where v_1 and v_2 are variables representing points and $[l,u]$ are bounds on their distance. We employ simple distance constraints to represent the metric semantics of qualitative relations, as well as the metric bounds of these relations. We call the algebra so obtained Augmented Rectangle Algebra (ARA). The relations in ARA are qualitative RA relations, that are augmented with metric bounds on each component IA relation (see Figure 1).

The question which remains to be answered is how to reason upon constraints in ARA. Reasoning about qualitative and metric relations using a single constraint network which encodes both aspects has been studied for temporal problems [19]. This idea was extended to fuzzy spatial constraints [20]. The only approach we are aware of which tackles the problem of combining crisp qualitative and metric spatial constraints is proposed by Condotta [21]. However, the approach falls short of providing a usable, hybrid qualitative and metric language that is provably tractable.

For the purposes of spatial reasoning for robots, we do not require the full expressiveness of ARA. In particular, we focus on *convex* ARA relations, which are relations whose qualitative components are convex RA relations. Convex RA relations are composed exclusively of convex IA relations, which are defined by [22]. For example, {b,m,o} is a convex IA relation, whereas {b,o} is not. Convex ARA relations impose convex disjunctions of IA relations on each axis. For example, $A \{\langle \mathrm{b},\mathrm{o} \rangle, \langle \mathrm{m},\mathrm{o} \rangle\} B$, is convex because the IA relation in the x dimension $A_x \{\mathrm{b},\mathrm{m}\} B_x$ is convex, as is the elementary relation $A_y \{\mathrm{o}\} B_y$. Convex IA relations can be translated to a set of simple distance constraints. Specifically, the metric translation of convex IA relations is the union of the translation into simple distance constraints of each atomic IA relation in the disjunction. The simple distance constraint translation of a convex ARA relation is thus two sets of simple distance constraints: one deriving from the translation of IA relations in the x axis, one deriving from the IA relations in the y axis. Henceforth, we denote with $\mathrm{metric}(r) = \mathrm{metric}_x(r) \cup \mathrm{metric}_y(r)$ the translation into simple distance constraints of a convex ARA relation r.

Note that specifying bounds on an otherwise convex qualitative relation may lead to the impossibility to translate the relation to simple distance constraints. For instance, $A_x^+ \{\mathrm{b},\mathrm{m}\} B_x^-$ is convex and translates to $A_x^+ \leq B_x^-$, whereas $A_x^+ \{\mathrm{b}[5,\infty),\mathrm{m}\} B_x^-$ is not convex, and requires the disjunctive metric translation $A_x^+ + 5 < B_x^- \vee A_x^+ = B_x^-$. We thus disallow to specify bounds on ARA relations composed of non-atomic RA relations. This allows to employ simple distance constraints (which do not admit disjunction) for modeling the metric semantics of ARA relations. Sets of simple distance constraints constitute a Simple Temporal Problem [18], which is tractable.

Convex ARA relations are a powerful representational tool. For instance, we can model the requirement that object A is "to the left of" object B by at least 5cm while allowing the disjunction of all other relations in the y axis. This is expressed by the constraint $A \{\langle \mathrm{b}[5,\infty), \mathrm{convexify}(\mathrm{b},\mathrm{a}) \rangle\} B$, where $\mathrm{convexify}(\mathrm{b},\mathrm{a})$ represents the disjunction of all (qualitative) IA relations. Indeed, given the IA relation $\{r_1, r_r\}$, it is possible to compute the disjuncts that must be added in order to render the relation convex [23]. The resulting $\mathrm{convexify}(\cdot)$ function runs in $O(|C|)$.

IV. Consistency Checking

It is possible to show that the consistency of convex ARA constraint networks is tractable:

Theorem 1: The consistency of a rectangle constraint network $M = (V,C)$, where C is a set of convex ARA relations, is decided by the consistency of the set of simple distance constraints $\bigcup_{r \in C} \mathrm{metric}(r)$.

The proof, omitted here for lack of space, involves a method introduced by van Beek [24] for translating qualitative IA relations to metric ones, as well as results by Condotta ([21], Theorem 2). This formal property allows to process both qualitative constraints and their metric bounds uniformly at the metric level, i.e., reasoning at the metric level computes the consequence of both qualitative and metric relations among rectangles. Consistency of a set of simple distance constraints can be proved by low-order polynomial constraint propagation algorithms [25], [26]. The result of constraint propagation is a set of admissible bounds on the placement of rectangles, a metric solution which is directly understandable by the robot. Notice that placement does not include orientation of the object.

Although expressive, ARA alone is not sufficiently versatile for our needs. Specifically, it lacks unary relations for modeling the size and placement of objects. Note that size and placement are essential for modeling perceived context. We thus introduce the unary relation $Size[l_x,u_x][l_y,u_y]$, which bounds the distances between two points of the same rectangle along one axis, i.e., two constraints imposing minimum and maximum x and y dimensions l_x, u_x, l_y and u_y. ARA constraints together with *Size* constraints express a robot's knowledge about orientation and topology of spatial entities. We also introduce the relation $At[l_x^1,u_x^1][l_y^1,u_y^1][l_x^2,u_x^2][l_y^2,u_y^2]$, which bounds the absolute placement of bounding boxes. The bounds $[l_x^1,u_x^1][l_y^1,u_y^1]$ provide 2D bounds for the position of the lower left corner (x^1,y^1), while $[l_x^2,u_x^2][l_y^2,u_y^2]$ provide 2D bounds for the position of the upper right corner (x^2,y^2).

We denote the algebra obtained by enriching ARA with *Size* and *At* constraints ARA$^+$. It is straightforward to prove that ARA$^+$ with convex relations remains tractable:

Theorem 2: The consistency of a rectangle constraint network $M = (V,C)$, where C is a set of convex ARA$^+$ relations, is decided by the consistency of $\bigcup_{r \in C} \mathrm{metric}(r)$.

An ARA$^+$ network can be used to represent uniformly both a desired spatial layout of objects and the observed spatial layout. We reduce the problem of matching observed spatial relations and perceived context (1) to consistency checking as follows: a set of rectangles V_o is created to model the observed objects, and each $v_o \in V_o$ is constrained with an *At* constraint reflecting its position; all knowledge is encoded as further rectangles V and ARA$^+$ relations; a constraint $v_o\{\text{eq},\text{eq}\}v$ is added to unify each observed object $v_o \in V_o$ with its counterpart $v \in V$. If this network is consistent, then the observed state of the world adheres to the the robot's spatial knowledge.

Maintaining a network representing both qualitative and metric relations enables to do more than matching perceived context to knowledge. More in general, the constraint network forms a query of our desire. We can use constraint networks to answer queries for instantiating planned actions. For instance, we can construct a constraint network containing *At* relations for all perceived objects, as well as variables for object(s) that are not in the scene and need to be placed. A solution to this spatial constraint satisfaction problem (i.e., a substitution of coordinates to points that is consistent) represents a placement of the objects in the scene that is consistent with qualitative and metric knowledge. In order to support this capability (2), we require a way to extract and discern between different solutions in a way that is appropriate for the robotic domain, as shown below.

V. Solution extraction

Enforcing the consistency of the network of simple distance constraints $\bigcup_{r\in C} \text{metric}(r)$ updates the bounds of all rectangles. The assignment of all lower bounds or of all upper bounds after consistency enforcement are both valid solutions [18]. Other possible solutions can be obtained through incremental propagation.

In a robotic context, assignments other than the lower and upper bound solutions are preferable. Specifically, we are interested in obtaining the solution that has maximum distance from these two solutions, as the region that is given to a robot to place an object should tolerate the inaccuracy of manipulation. In other words, if the robot does not place an object exactly within the region, the spatial layout should still be consistent. For this reason, we prefer assignments that are close to the center of the solution space. Obtaining the exactly centered solution is an optimization problem that is too computationally demanding to solve on-line, therefore we sacrifice the optimality of the solution in favor of efficiency.

Given $M = (V,C)$, we compute an approximation of the most centered solution for each rectangle $A \in V$ by leveraging the concept of 2D representation of an interval [27]. The interval A_x (similarly for A_y) is represented as a window in the space of start and end position (see Figure 3). The window is characterized by four numbers, namely, minimum and maximum positions, and minimum and maximum lengths. All possible placements of A_x after consistency is enforced are within a convex polygon in the 2D space. We choose as "most centered" placement for A_x the center of mass of this polygon, thus obtaining an assignment of A_x^+ and A_x^-.

As opposed to lower and upper bounds, which together constitute a consistent assignment for all points, the collection of all assignments extracted as described above does not constitute a consistent assignment [18]. In order to obtain one, the extracted coordinates of each rectangle must be encoded as additional *At* constraints and incrementally propagated. The procedure for determining a solution thus consists of (a) extracting the center of mass for the x and y intervals of one rectangle, (b) adding one *At* constraint reflecting this choice, and (c) applying incremental constraint propagation to update the bounds of the points of other rectangles. This quadratic[1] procedure is repeated $|V|$ times, yielding a complexity of $\Theta(|V|^3)$.

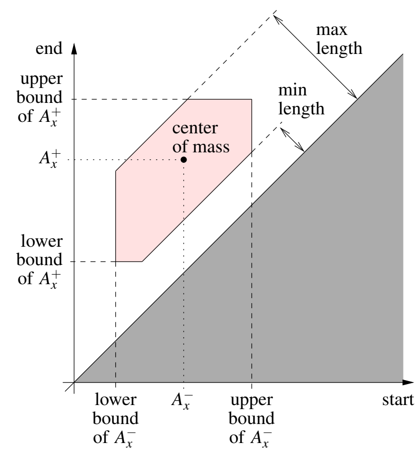


Fig. 3. Two-dimensional representation of an interval.

The query network for the task of action instantiation is formulated as follows: as for the scene/knowledge matching task, an initial network is constructed with observed and known relations; objects to be placed are represented by variables in V, and a consistent assignment computed as above for the objects in V can be used to determine the coordinates for placing actions (2).

VI. Culprit Detection, Recovery Recommendation

We now turn our attention to spatial reasoning for detecting and reacting to contingencies (3). Suppose a robot has to place a dish on a table in which several other dishware have already been placed, and the robot is given a set of spatial relations which describe a well-set table (e.g., that a dish is to be placed between fork and knife, possibly within certain bounds). As described above, the robot instantiates the placing action by computing the solution to the query constraint network. However, the network turns out to be inconsistent, indicating that it cannot achieve the goal of placing a dish.

It may be possible to recover from the failure by finding the source(s) of inconsistency in the ARA$^+$ network. The source(s) of inconsistency are constraints, specifically *At* constraints deriving from observation (e.g., the fork and

[1]We employ an incremental all-pairs-shortest-path algorithm to enforce consistency of the simple distance constraints [25], which requires $\Theta(|V|^2)$ time.

knife are so close that a dish cannot fit between them). Eliminating the inconsistency thus means to relax one or more *At* constraints. The power set of *At* constraints in the network contains all possible *culprit sets*[2]. If there exists at least one culprit set such that its removal from the network makes the network consistent, then a solution of this network contains new positions for the objects related to constraints in the culprit set. These new positions can be used to instantiate new actions that bring about a consistent situation — e.g., moving the knife to the right, moving the fork to the left, or both. Which of the alternative courses of action (culprit sets) is more convenient?

The process of finding a desirable culprit set is a search process which employs two heuristics. The process is shown in Algorithm `RecommendRecovery`. The input is an

Function `RecommendRecovery` (V,C) : $V_{\text{replace}} \subseteq V$

1 $C_{At} \leftarrow \{v\, At : v\, At \in C\}$
2 **for** $i = 1$ **to** $|C_{At}|$ **do**
3 culprits $\leftarrow \{\{v_1\, At_1, \ldots, v_i\, At_i\} \in 2^{C_{At}}\}$
4 culprit $\leftarrow \arg\min_{C' \in \text{culprits}} \{Rig(V, C \setminus C')\}$
5 **if** $(V, C \setminus \text{culprit})$ *is consistent* **then**
6 **return** $\{v_i : v_i\, At \in culprit\}$
7 **return** $\emptyset$

inconsistent ARA$^+$ network, and the output is a set of rectangles representing objects whose replacement leads to consistency. The first heuristic favors small sets, the rationale being that moving fewer objects is less prone to failure than moving many (line 2). The second heuristic favors moves which least affect the *spatial rigidity* of the network (line 4). To define this heuristic, we employ a measure known as the root mean square (RMS) rigidity of a network of simple distance constraints [28]. The measure builds on the notion of relative rigidity of pairs of points (A_x^+, B_x^+), (A_x^+, B_x^-), and so on (similarly for the y axis), namely

$$Rig(A_x^{(\cdot)}, B_x^{(\cdot)}) = \frac{1}{1 + d_{\max}(A_x^{(\cdot)}, B_x^{(\cdot)}) - d_{\min}(A_x^{(\cdot)}, B_x^{(\cdot)})},$$

where $d_{\max}(\cdot)$ and $d_{\min}(\cdot)$ are the maximum and minimum distances between the points. The measure has a maximum value of 1, since $d_{\max}(A_x^{(\cdot)}, B_x^{(\cdot)}) - d_{\min}(A_x^{(\cdot)}, B_x^{(\cdot)}) \geq 0$, which occurs when the points are both fixed. The RMS rigidity of the network $S_x = \bigcup_{r \in C} \text{metric}_x(r)$ is defined as

$$Rig(S_x) = \sqrt{\frac{2}{2V\,(2V-1)} \sum_{(A,B) \in V} \left(Rig(A_x^{(\cdot)}, B_x^{(\cdot)})\right)^2}$$

The rigidity of a RA constraint network $M = (V, C)$ is the average of $Rig(S_x)$ and $Rig(S_y)$. This measure also ranges in $[0,1]$: low rigidity entails that the admissible bounds of objects are such that there is significant slack in determining new placements, whereas high rigidity means that the constraints in the network afford placement options which are close to failure; therefore manipulation must be more precise.

[2]We exclude *At* constraints on immovable objects such as the table.

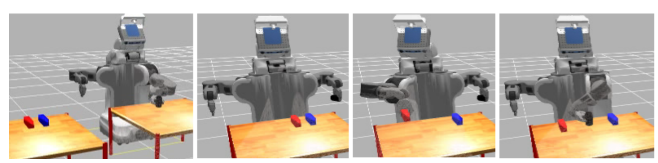
The complexity of culprit detection is exponential in the number of *At* constraints, however, this number is often small as it corresponds to the number of objects in the scene.

VII. Experiment

Several experiments were performed with a real robotic platform (Scitos G5 mobile platform with a Kinova Jaco arm) and with a simulated PR2 using ROS/Gazebo. Here we describe the latter. The robot's *general knowledge* states that well-set tables are such that forks are to the left and knives to the right of cups, that all objects should be at least 5cm from the edge of the table, and that all objects should be at most 20cm far from the side of the table from which the robot is operating (see Figure 4), due to reachability constraints:

$$\begin{array}{rl}
\texttt{Fork} & \langle \text{d}[5,+\infty)[5,+\infty), \text{d}[5,20][5,+\infty)\rangle\ \texttt{Table} \\
\texttt{Knife} & \langle \text{d}[5,+\infty)[5,+\infty), \text{d}[5,20][5,+\infty)\rangle\ \texttt{Table} \\
\texttt{Cup} & \langle \text{d}[5,+\infty)[5,+\infty), \text{d}[5,20][5,+\infty)\rangle\ \texttt{Table} \\
\texttt{Fork} & \langle \text{b}[10,15], \text{d}\rangle\ \texttt{Cup} \\
\texttt{Knife} & \langle \text{bi}[10,15], \text{d}\rangle\ \texttt{Cup} \\
\texttt{Fork} & Size[4,8][18,24] \\
\texttt{Knife} & Size[4,8][18,24] \\
\texttt{Cup} & Size[4,7][4,7]
\end{array}$$

The fork, the knife and the cup are modeled as bounding boxes whose sides are parallel to the axes of the reference frame of their support plane (table). Cups were used instead of dishes and cuboids instead of forks and knives to facilitate manipulation. The planner used was a simple implementation of a Hierarchical Task Network planner. There are two tables in the environment (table-1 and table-2). The initial condition (see Figure 4(a)) consisted in the robot being in front of table-1, on top of which a cup was placed; a fork and a knife were placed on table-2 (see Figure 4(b)), which was located to the right of table-1; the goal consisted in placing the cup on table-2 such that the table was well set. The planner generated a high-level plan prescribing the robot to pick the cup from table-1, move to and approach table-2, and place the cup on it. Upon arriving at table-2 (see Figure 4(b)), four



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

Fig. 4. Salient moments of a test run with a simulated PR2.

variables representing rectangles are generated: `table2`, `cup`, `fork`, `knife`. The robot observes the placement of the fork, knife and table, and appropriate *At* constraints (in the table reference frame) are imposed on the rectangles, reflecting their *observed placement*:

$$\begin{array}{r}
\texttt{fork}\ At[31,31][11,11][37,37][30,30] \\
\texttt{knife}\ At[40,40][10,10][46,46][29,29] \\
\texttt{table2}\ At[0,0][0,0][100,100][100,100]
\end{array}$$

Observed and general knowledge are combined by imposing constraints that unify the observed objects with the objects in the general knowledge:

$$\texttt{fork}\ \langle \text{eq}, \text{eq} \rangle\ \texttt{Fork}$$
$$\texttt{table2}\ \langle \text{eq}, \text{eq} \rangle\ \texttt{Table}$$
$$\texttt{knife}\ \langle \text{eq}, \text{eq} \rangle\ \texttt{Knife}$$

All relations and variables combined constitute an ARA$^+$ constraint network $M = (V, C)$. Through consistency checking, M is found to be inconsistent, as the fork and knife are too close to each other (see Figure 4(a)). At this point, two possible culprit sets with cardinality one are identified (line 3), namely $c_1 = \{\texttt{fork}\ At[\ldots]\}$ and $c_2 = \{\texttt{knife}\ At[\ldots]\}$. Their heuristic values are respectively $Rig((V, C \setminus c_1)) = 0.2720$ and $Rig((V, C \setminus c_2)) = 0.2716$. Consequently, the plan is modified to achieve the further goal of picking and replacing the knife with the free arm.

When execution of the modified plan resumes (see Figure 4(c)), the placing actions in the plan are dispatched to the PR2's executive. The executive requires a bounding box for each placement action, within which possible points for placing an object are assessed for reachability by a kinematic solver and a 3D motion planner. The bounding box provided to the executive is obtained through the procedure described in Section V.

In all experiments, both real and simulated, the same planning and spatial reasoning algorithm was used, and the general knowledge (encoded as ARA$^+$ relations) was never changed. Videos of the experiments are available at `http://aass.oru.se/~mmi/IROS-2013-WS-video/`.

VIII. Conclusions and Future Work

We have presented ARA$^+$, a knowledge representation formalism for integrated metric and qualitative spatial reasoning. By uniformly representing metric bounds alongside qualitative relations, the calculus can be used both to specify spatial knowledge in human-accessible terms and to represent perceived context. As we have shown through proof-of-concept experiments, ARA$^+$ can be leveraged to realize the three important reasoning tasks in the sense-plan-act loop. Future work will focus on developing a planner which can take into account spatial relations in ARA$^+$ in the causal reasoning process. This will allow the planner to automatically infer the plan modifications necessary in response to recovery recommendation as part of its search.

Acknowledgements. This work is supported by the EC Seventh Framework Program under project RACE ("Robustness by Autonomous Competence Enhancement", see `project-race.eu`), grant agreement no. 287752.

References

[1] M. Ghallab, D. Nau, and P. Traverso, *Automated Planning: Theory and Practice*. Morgan Kaufmann, 2004.

[2] W. Cushing, S. Kambhampati, Mausam, and D. Weld, "When is temporal planning really temporal?" in *Proc. of the 20th Int'l Joint Conf. on Artif. Intell.*, 2007.

[3] S. Fratini, F. Pecora, and A. Cesta, "Unifying Planning and Scheduling as Timelines in a Component-Based Perspective," *Archives of Control Sciences*, vol. 18, no. 2, pp. 231–271, 2008.

[4] A. Loutfi, S. Coradeschi, M. Daoutis, and J. Melchert, "Using knowledge representation for perceptual anchoring in a robotic system," *International Journal on Artificial Intelligence Tools*, vol. 17, no. 5, pp. 925–944, 2008.

[5] W. Kennedy, M. Bugajska, M. Marge, W. Adams, B. Fransen, D. Perzanowski, A. Schultz, and J. Trafton, "Spatial representation and reasoning for human-robot collaboration," in *Proc. of the 22nd Nat. Conf. on Artif. Intell.*, 2007.

[6] O. Colliot, O. Camara, and I. Bloch, "Integration of fuzzy spatial relations in deformable models-application to brain mri segmentation," *Pattern Recogn.*, vol. 39, no. 8, pp. 1401–1414, Aug. 2006.

[7] J. Guitton and J.-L. Farges, "Taking into account geometric constraints for task-oriented motion planning," in *Workshop on Bridging the Gap between Task and Motion Planning (BTAMP)*, 2009.

[8] H.-Y. Jang, H. Moradi, S. Hong, S. Lee, and J. Han, "Spatial reasoning for real-time robotic manipulation," in *Proc. of IEEE/RSJ Int'l Conf. on Intelligent Robots and Systems*, 2006.

[9] A. Cohn, J. Renz, and M. Sridhar, "Thinking inside the box: A comprehensive spatial representation for video analysis," in *Proc. of the 13th Int'l Conf. on Principles of Knowledge Representation and Reasoning*, 2012.

[10] D. Wolter and J. Wallgrün, *Qualitative spatial reasoning for applications: New challenges and the SparQ toolbox*. IGI Global, 2010.

[11] M. Westphal, C. Dornhege, S. Wölfl, M. Gissler, and B. Nebel, "Guiding the generation of manipulation plans by qualitative spatial reasoning," *Spatial Cognition and Computation: An Interdisciplinary Journal*, vol. 11, no. 1, pp. 75–102, 2011.

[12] L. Belouaer, M. Bouzid, and A. Mouaddib, "Ontology based spatial planning for human-robot interaction," in *Proc of the 17th Int'l Symp. on Temporal Representation and Reasoning*, 2010.

[13] F. Lagriffoul, D. Dimitrov, A. Saffiotti, and L. Karlsson, "Constraint propagation on interval bounds for dealing with geometric backtracking," in *Proc. of IEEE/RSJ Int'l Conf. on Intelligent Robots and Systems*, 2012.

[14] D. Randell, Z. Cui, and A. Cohn, "A spatial logic based on regions and connection," in *Proc. of the Int'l Conf. on Principles of Knowledge Representation and Reasoning*, 1992.

[15] S. Skiadopoulos and M. Koubarakis, "Composing cardinal direction relations," *Artif. Intell.*, vol. 152, no. 2, pp. 143–171, 2004.

[16] P. Balbiani, J.-F. Condotta, and L. F. Del Cerro, "A new tractable subclass of the rectangle algebra," in *Proc. of the 16th Int'l Joint Conf. on Artif. Intell.*, vol. 1, 1999, pp. 442–447.

[17] J. Allen, "Towards a general theory of action and time," *Artif. Intell.*, vol. 23, no. 2, pp. 123–154, 1984.

[18] R. Dechter, I. Meiri, and J. Pearl, "Temporal constraint networks." *Artif. Intell.*, vol. 49, no. 1-3, pp. 61–95, 1991.

[19] I. Meiri, "Combining qualitative and quantitative constraints in temporal reasoning," in *Artif. Intell.*, 1996, pp. 260–267.

[20] M. Falda, "Spatial reasoning with integrated qualitative-metric fuzzy constraint networks," *Universal Computer Science*, vol. 16, no. 11, pp. 1390–1409, 2010.

[21] J.-F. Condotta, "The augmented interval and rectangle networks," in *Proc. of 7th Int'l Conf. on Principles of Knowledge Representation and Reasoning*, 2000.

[22] G. Ligozat, "A new proof of tractability for ORD-Horn relations," in *AAAI Workshop on Spatial and Temporal Reasoning*, 1996.

[23] D. Mitra and F. Launay, "Explanation generation over temporal interval algebra," in *Qualitative Spatio-Temporal Representation and Reasoning: Trends and Future Directions*, S. Hazarika, Ed. Information Science Publishing, 2010.

[24] P. van Beek, "Exact and approximate reasoning about qualitative temporal relations," Ph.D. dissertation, University of Waterloo, Waterloo, Ont., Canada, Canada, 1990, uMI Order No. GAXNN-61098.

[25] R. Floyd, "Algorithm 97: Shortest path," *Commun. ACM*, vol. 5, p. 345, June 1962.

[26] L. Xu and B. Choueiry, "A new efficient algorithm for solving the simple temporal problem," in *Proc. of the 4th Int'l Conf. on Temporal Logic*, 2003.

[27] J.-F. Rit, "Propagating temporal constraints for scheduling," in *Proc. of the 2nd Nat'l Conf. on Artif. Intell.*, 1986.

[28] L. Hunsberger, "Algorithms for a temporal decoupling problem in multi-agent planning," in *Proc. of the 18th Nat'l Conf. on Artif. Intell.*, 2002.