Toward Explainable Reasoning and Learning in Robotics (Extended Abstract)

Mohan Sridharan

School of Computer Science, University of Birmingham, UK
m.sridharan@bham.ac.uk
https://www.cs.bham.ac.uk/~sridharm/

Abstract. This paper summarizes our architecture for robots that combines the strengths of knowledge-based reasoning and data-driven learning. The architecture supports non-monotonic logical reasoning and probabilistic reasoning with incomplete commonsense domain knowledge. Reasoning triggers and guides learning of previously unknown domain knowledge when needed using data-driven learning methods. The architecture also enables the robot to provide relational descriptions of its decisions and beliefs during reasoning and learning, and to construct questions that address ambiguities in the human input. The architecture's capabilities are evaluated in simulation and on robots.

Keywords: Non-monotonic logical reasoning · Probabilistic reasoning · Deep learning · Explainability · Disambiguation.

1 Motivation

Consider a *robot assistant* (RA) domain in which a robot has to deliver target objects to particular people or rooms, and estimate and revise the occlusion of objects and stability of object configurations in a particular location. The robot's incomplete domain knowledge includes commonsense knowledge, e.g., statements such as "books are usually in the study" that hold in all but a few exceptional circumstances, e.g., cookbooks are in the kitchen. The robot also extracts information from noisy sensor inputs with quantitative measures of uncertainty, e.g., "I am 90% certain I saw the robotics book in office-1". The robot has some prior knowledge of object attributes such as $size$, $surface$, and $shape$; grounding of some prepositional words such as *above* and *in* representing the spatial relations between objects; and some axioms governing domain dynamics. Examples of these axioms include "placing an object on an irregular surface results in instability" and "an object below another object cannot be picked up". The robot reasons with the knowledge and observations for to perform desired tasks. In any practical domain, it will have to revise this knowledge over time, often accomplished by data-driven (e.g., deep, reinforcement) learning methods that process observations, labeled datasets, and/or human input. Also, enabling the robot to describe its decisions and beliefs at different levels of abstraction will lead to more effective collaboration with humans. We briefly summarize our architecture that seeks to support these capabilities by exploiting the complementary strengths of declarative logic programming, probabilistic reasoning, and data-driven interactive learning; for complete details, please see [8].

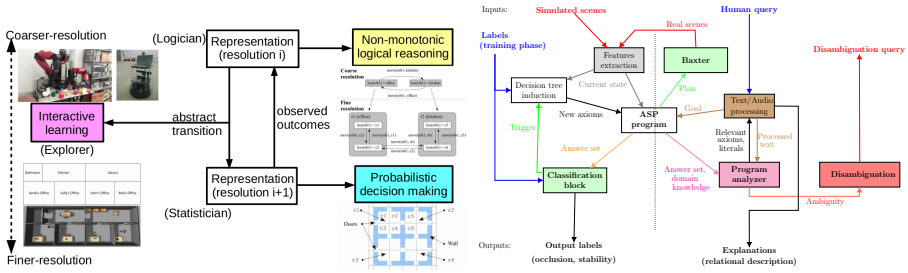

Fig. 1. (left) Architecture combines strengths of declarative programming, probabilistic reasoning, and interactive learning to represent, reason, act, and learn at different resolutions; (right) Non-monotonic logical reasoning triggers and guides (deep, inductive, or reinforcement) learning to complete desired estimation tasks, learn previously unknown domain knowledge, and to provide relational descriptions as explanations.

2 Architecture Overview

Our architecture for knowledge representation, explainable reasoning, and interactive learning, is based on tightly-coupled transition diagrams at different resolutions. It may be viewed as a logician, statistician, and a creative explorer working together–see Figure 1(left). The transition diagrams are described using an action language $\mathcal{AL}_d$ [3], which has a sorted signature and supports three types of statements: causal laws, state constraints, and executability conditions; the fluents can be non-Boolean and axioms can be non-deterministic. Depending on the domain and tasks at hand, the robot chooses to plan and execute actions at two specific resolutions, but can construct and provide explanations at other resolutions; we limit our discussion to two resolutions here.

Knowledge representation and reasoning: The coarse resolution domain description comprises system description $\mathcal{D}_c$ of transition diagram τ_c, a collection of $\mathcal{AL}_d$ statements, and history $\mathcal{H}_c$. $\mathcal{D}_c$ comprises sorted signature Σ_c and axioms. Σ_c includes basic sorts, statics, fluents, and actions. Axioms include causal laws, state constraints, and executability conditions. $\mathcal{H}_c$, which is typically a record of fluents observed to be true or false at a particular time step, and the occurrence of actions at a particular time step, is expanded to include prioritized defaults describing the values of fluents in the initial state.

To reason with the domain description, we construct program $\Pi(\mathcal{D}_c, \mathcal{H}_c)$ in CR-Prolog, a variant of Answer Set Prolog (ASP) that incorporates consistency restoring (CR) rules [2]. ASP is based on stable model semantics, and supports *default negation* and *epistemic disjunction*, e.g., unlike "$\neg a$" that states *a is believed to be false*, "*not a*" only implies *a is not believed to be true*, i.e., each literal can be true, false or "unknown". ASP represents constructs difficult to express in classical logic formalisms and supports non-monotonic logical reasoning. An *answer set* of Π represents the beliefs of the robot associated with Π. Tasks such as computing entailment, planning, and diagnostics can be reduced to computing answer sets of Π; we do so using the SPARC system [1].

For any given goal, coarse-resolution reasoning provides a plan of *abstract actions*. To implement the abstract actions, a fine-resolution system description

$\mathcal{D}_f$ defined formally as a *refinement* of $\mathcal{D}_c$ such that for any given abstract transition between two states $\in \tau_c$, there is a path in τ_f between a refinement of the two states. In the RA domain, the robot would (for example) reason about grid cells in rooms and parts of objects, attributes that were previously abstracted away by the designer. Since the robot interacts with the physical world at the finer resolution, we introduce a *theory of observation* in $\mathcal{D}_f$, specifically *knowledge-producing* actions and fluents to sense the value of domain fluents. Next, $\mathcal{D}_f$ is *randomized* to model non-determinism ($\mathcal{D}_{fr}$). Since reasoning with $\mathcal{D}_{fr}$ becomes computationally unfeasible for complex domains, we enable the robot to automatically *zoom* to $\mathcal{D}_{fr}(T)$, the part of $\mathcal{D}_{fr}$ *relevant* to any given abstract transition T. Reasoning with $\mathcal{D}_{fr}(T)$ provides a sequence of concrete actions that implement T, incorporating any available probabilistic models of uncertainty in perception and actuation. Fine-resolution outcomes with a high probability are committed as statements known with complete certainty. Reasoning with these outcomes provides coarse-resolution outcomes that are added to $\mathcal{H}_c$ for further reasoning; see [9] for details.

Interactive learning: Reasoning with incomplete domain knowledge to achieve desired goals (e.g., fetch objects) or perform desired estimation tasks (e.g., classifying object occlusion or stability) can lead to incorrect or suboptimal outcomes. Many state of the art methods for learning previously unknown actions and axioms, or task-specific object models, are based on deep networks. They often require many labeled examples; it is difficult to provide such examples in complex domains or to interpret the decisions of "end to end" data-driven methods.

Figure 1(right) is an overview of the interactive learning and explainable reasoning components. The main sensor inputs (for these components and architecture) are RGB/D images. These images are processed to extract spatial relations (using learned grounding of prepositions [5]) and other attributes that are encoded as ASP statements. The robot attempts to use ASP-based logical reasoning to complete the desired (e.g., planning, estimation) tasks. If this reasoning does not provide any outcome (e.g., no plan to reach goal), or provides an incorrect outcome (e.g., incorrect label on training image), this is considered to indicate that incomplete or incorrect knowledge, which triggers learning.

Our architecture has two schemes for learning and knowledge revision. The first scheme is used in the context of planning; relational reinforcement learning and decision-tree induction are used to learn actions and axioms from human descriptions of desired behavior, or from observations obtained through active exploration or reactive action execution in response to the unexpected outcomes. Reasoning automatically adapts the search space of learning based to the task(s) and goal(s) at hand [10]. The second scheme is used in the context of estimation tasks on input images. Reasoning with domain knowledge automatically identifies relevant regions of interest (ROIs) from the corresponding images, using information from these ROIs to efficiently train a deep neural network for estimation tasks. This information is also used to incrementally learn decision trees summarizing the robot's experiences, with axioms induced from the trees' branches merged with existing axioms for reasoning [8].

Fig. 2. Example images of simulated scenes and setup for physical robot experiments.

Explainable reasoning: We consider an "explanation" to be a relational description of the robot's decisions or evolution of beliefs in terms of the domain attributes and robot actions. Our architecture's explainable reasoning component is based on a *theory of explanations* comprising: (i) claims about representing, reasoning with, and learning knowledge to support explanations; (ii) a characterization of explanations based on abstraction, specificity, and verbosity; and (iii) a methodology for constructing explanations [11].

The robot first processes human verbal or textual input using existing natural language processing tools and a controlled vocabulary to identify the type of query. This includes simple commands (e.g., goals or action to be achieved), descriptive statements or questions (e.g.,"please describe the plan", "why did you pick up the robotics book from the table?", "why did believe the AI book was in the study at step 3?"), contrastive questions, and counterfactual questions (e.g., "why did not not use the shorter corridor to the library?"). Commands set goals that are passed on to the ASP-based reasoner for planning. To answer other types of questions, our architecture enables the robot to automatically identify relevant beliefs and axioms, and traces the evolution of beliefs through relevant axioms to identify literals relevant to answering the query. These literals are used to construct the answer presented to the human user. The human can interactively obtain the answer at the desired abstraction level [11, 8].

The query posed by the human can be ambiguous in terms of the object, event, or time step being referenced, e.g., "why do you want to pick up the yellow object?" when the computed plan requires the robot to move two different yellow objects at different time steps. Our architecture introduces relevant heuristic measures of ambiguity, human confusion, and the attribute utility, enabling the robot to automatically construct and pose questions likely to provide disambiguation in the fewest number of interactions with a human. Once the human response to the selected disambiguation question resolves the ambiguities, the robot proceeds to answer the original human query; see [7] for details.

3 Execution Traces

Consider the following execution traces of our architecture.

Execution Example 1 *[Explain plans and beliefs]*
In a scene with objects as shown in the fourth image in Figure 2, the robot's goal state has the red cube on the orange cube. The following interaction takes place *after* the robot has executed a plan to achieve the goal.

– **Human:** "Please describe the plan."

Baxter: "I picked up the blue cube. I put it on the table. I picked up the orange cube. I put it on the table. I picked up the red cube. I put it on the orange cube."
- The human may ask the robot to justify a particular action in the plan.
  Human: "Why did you pick up the blue cube at step 0?"
  Baxter: "Because I had to pick up the red cube. It was below blue cube." Image regions that influences the answer are automatically highlighted.
- The human now may ask about actions that were not considered.
  Human: "Why did you not put down the orange cube on the blue cube?"
  Baxter: "Because the blue cube is small." The robot automatically resolved the time reference and used the learned default knowledge that a large object on a small object is typically unstable.
- The human may also ask the robot to justify particular beliefs.
  Human: "Why did you believe that the red cube was below the blue cube in the initial state?"
  Baxter: "Because I observed the red cube below the blue cube in step 0."

For experimental results in simulation and on robots, see [4, 6, 8, 9, 10, 11].

4 Discussion

Some key capabilities of our architecture include:

- Once the designer has provided the domain-specific information (e.g., for refinement), planning, diagnostics, and plan execution can be automated. The formal coupling between the resolutions allows us to introduce more complex theories in the coarse-resolution, and to exploit the complementary strengths of non-monotonic logical reasoning and probabilistic reasoning.
- Second, exploiting the interplay between knowledge-based reasoning and data-driven learning provides a clear separation of concerns, helps focus attention automatically to the relevant knowledge at the appropriate resolution, thus improving the reliability and efficiency of reasoning and learning.
- Third, it is easier to understand and modify the observed behavior than with architectures that consider all the available knowledge or only support data-driven learning. The robot is able to provide relational descriptions of its decisions and the evolution of its beliefs, automatically resolving any ambiguities in the human query by constructing suitable clarification questions.
- Fourth, there is smooth transfer of control and relevant knowledge between components of the architecture, and confidence in the correctness of the robot's behavior. Also, the underlying methodology can be used with different robots and in different application domains.

Acknowledgements

The research threads summarized here were pursued in collaboration with Ben Meadows, Tiago Mota, Heather Riley, Rocio Gomez, Michael Gelfond, Shiqi Zhang, and Jeremy Wyatt. This work was supported in part by the U.S. ONR Awards N00014-13-1-0766, N00014-17-1-2434 and N00014-20-1-2390, AOARD award FA2386-16-1-4071, and U.K. EPSRC award EP/S032487/1.

Bibliography

[1] Balai, E., Gelfond, M., Zhang, Y.: Towards Answer Set Programming with Sorts. In: International Conference on Logic Programming and Nonmonotonic Reasoning. Corunna, Spain (September 15-19, 2013)

[2] Gebser, M., Kaminski, R., Kaufmann, B., Schaub, T.: Answer Set Solving in Practice, Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan Claypool Publishers (2012)

[3] Gelfond, M., Inclezan, D.: Some Properties of System Descriptions of AL_d. Journal of Applied Non-Classical Logics, Special Issue on Equilibrium Logic and Answer Set Programming **23**(1-2), 105–120 (2013)

[4] Gomez, R., Sridharan, M., Riley, H.: What do you really want to do? Towards a Theory of Intentions for Human-Robot Collaboration. Annals of Mathematics and Artificial Intelligence, special issue on commonsense reasoning **89**, 179–208 (Februray 2021)

[5] Mota, T., Sridharan, M.: Incrementally Grounding Expressions for Spatial Relations between Objects. In: International Joint Conference on Artificial Intelligence. Stockholm, Sweden (July 2018)

[6] Mota, T., Sridharan, M.: Commonsense Reasoning and Knowledge Acquisition to Guide Deep Learning on Robots. In: Robotics Science and Systems. Freiburg, Germany (June 22-26, 2019)

[7] Mota, T., Sridharan, M.: Answer me this: Constructing Disambiguation Queries for Explanation Generation in Robotics. In: IEEE International Conference on Development and Learning (ICDL) (August 23-26, 2021)

[8] Mota, T., Sridharan, M., Leonardis, A.: Integrated Commonsense Reasoning and Deep Learning for Transparent Decision Making in Robotics. Springer Nature Computer Science **2**(242), 1–18 (2021)

[9] Sridharan, M., Gelfond, M., Zhang, S., Wyatt, J.: REBA: A Refinement-Based Architecture for Knowledge Representation and Reasoning in Robotics. Journal of Artificial Intelligence Research **65**, 87–180 (May 2019)

[10] Sridharan, M., Meadows, B.: Knowledge Representation and Interactive Learning of Domain Knowledge for Human-Robot Collaboration. Advances in Cognitive Systems **7**, 77–96 (December 2018)

[11] Sridharan, M., Meadows, B.: Towards a Theory of Explanations for Human-Robot Collaboration. Kunstliche Intelligenz **33**(4), 331–342 (December 2019)