Commonsense Reasoning and Deep Learning for Transparent Decision Making in Robotics (Extended Abstract)

Tiago Mota[1], Mohan Sridharan[2], and Aleš Leonardis[2]

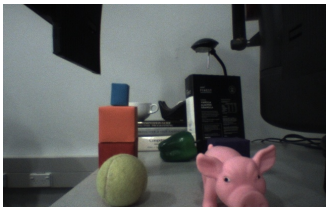
[1] Electrical and Computer Engineering, The University of Auckland, New Zealand
tmot987@aucklanduni.ac.nz
[2] School of Computer Science, University of Birmingham, United Kingdom
m.sridharan@bham.ac.uk, a.leonardis@bham.ac.uk

1 Motivation

Imagine the robot in Figure 1a arranging objects in desired configurations on a table, and estimating the occlusion of objects and stability of object configurations. An object is occluded if the view of any fraction of its frontal face is hidden by another object, and a configuration (i.e., a stack of objects) is unstable if any object in it is unstable. To perform these tasks, the robot extracts information from on-board camera images, reasons with this information and incomplete domain knowledge, and executes actions to achieve desired outcomes. The robot also incrementally learns previously unknown constraints governing domain states, and responds to questions about its plans, decisions, and underlying beliefs. For instance, the plan to achieve the goal of having the pig on top of the orange block in Figure 1b may be to move the blue block on to the table and place the pig on the orange block. When asked to justify a plan step, e.g., "why do you want to pick up the blue block first?", the robot answers "I have to put the pig on the orange block. The blue block is on the orange block"; when asked to explain an action choice, e.g., "why didn't you pick up the pig first?", the robot responds "Because the blue block is on the orange block".

Our work seeks to enable such on-demand *explanations* of a robot's decisions and beliefs, and hypothetical situations, in the form of descriptions of relations between relevant objects, actions, and domain attributes. This "explainability" can help improve the underlying algorithms and establish accountability, but it is challenging to achieve

(a) Test scenario.



(b) Image from robot's camera.

Fig. 1: (a) Motivating scenario of a manipulator arranging objects in desired configurations on a tabletop; (b) Image from the camera on the Baxter robot's left gripper.

with integrated robot systems that include knowledge-based reasoning methods (e.g., for planning) and data-driven (deep) learning algorithms (e.g., for pattern recognition). Drawing on research in cognitive systems, which indicates the advantages of reasoning with different representations [11], our architecture couples the complementary strengths of knowledge-based and data-driven algorithms, and provides transparent decision making. It builds on our prior work that combined non-monotonic logical reasoning and deep learning for scene understanding in simulated images [14]. Here, we summarize the ability of the architecture to:

- Incrementally merge newly acquired information with existing knowledge, exploiting the interplay between representation, reasoning, and learning.
- Automatically extract relevant information and construct relational descriptions of the robot's decisions and beliefs, including under hypothetical situations.

These capabilities are evaluated in the context of planning and scene understanding tasks in simulated scenes and on a physical robot manipulating tabletop objects. The robot (i) computes and executes actions to arrange objects in desired configurations; and (ii) estimates occlusion of scene objects and stability of object configurations. This paper is an extended abstract of our recent conference paper [15].

2 Related Work

Early work on explanation generation drew on research in psychology and linguistics to characterize explanations [6], and developed computational methods for explaining unexpected observations [8]. Recent work can be broadly categorized into two groups. Methods in one group modify or transform learned models to make their decisions more interpretable [9], or bias a reasoning system towards making decisions easier for humans to understand [18]. Methods in the other group seek to make decisions more transparent, e.g., by combining classical first order logical reasoning with interface design [3], or defining *proof trees* that describe the trace of a computation [5]. There has also been work on describing why a particular solution was obtained for a given problem using non-monotonic logical reasoning [4]. These methods are agnostic to how an explanation is structured or assume comprehensive domain knowledge. Methods are also being developed to make deep network models more interpretable, e.g., by constructing *heat maps* of relevant features [2].

Our work focuses on integrated robot systems that use a combination of knowledge-based and data-driven algorithms to represent, reason with, and learn from incomplete commonsense domain knowledge and noisy observations. We enable such robots to provide relational descriptions of decisions, beliefs, and hypothetical situations, capabilities that are not supported by existing systems [1,12]. We build on existing work on making decisions more transparent, and on work in our group on explainable agency [10], a theory of explanations [17], and on combining non-monotonic logical reasoning and deep learning for scene understanding [14].

3 Architecture

Figure 2 shows the architecture. Components to the left of the dashed vertical line combine non-monotonic logical reasoning and deep learning for classification in simulated

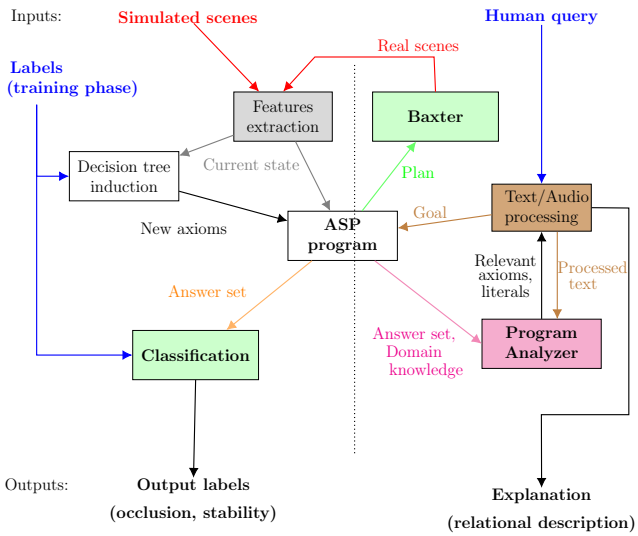


Fig. 2: Architecture combines strengths of deep learning, non-monotonic logical reasoning with incomplete knowledge, and inductive learning. New components to the right of the dotted line support desired explainability.

images [14]. Components to the right of the dashed line expand reasoning capabilities and answer questions about decisions, beliefs, and hypothetical situations. We summarize these components here; see [15] for more details.

The primary sensor inputs to the architecture are RGB images of simulated scenes, or noisy views of any given scene from the robot's cameras. From each image, the *feature extraction* component uses a probabilistic algorithm to extract objects and attributes. Also, the spatial relations between objects is computed using a learned histogram-based *grounding* for prepositional words such as "above", "far", and "in" [13].

To represent and reason with incomplete domain knowledge, we use CR-Prolog, an extension to Answer Set Prolog (ASP)[3]. ASP is a declarative language that encodes *default negation* and *epistemic disjunction*, and supports non-monotonic logical reasoning [7]. A domain's description comprises a *system description* $\mathcal{D}$ and a *history* $\mathcal{H}$. $\mathcal{D}$ comprises a *sorted signature* Σ with basic sorts, actions, and domain attributes (statics and fluents); and axioms that encode the domain dynamics. $\mathcal{H}$ comprises records of observations (of fluents) and of the execution of an action at a particular time step. Planning, diagnostics, and inference are reduced to computing *answer sets* of $\Pi(\mathcal{D},\mathcal{H})$ [7] and extracting relevant relations. Other work in our group combined coarse-resolution non-monotonic logical reasoning with probabilistic reasoning over the relevant part of a finer resolution representation [16]. In this work, we limit ourselves to logical reasoning at one resolution to focus on the interplay between reasoning and learning.

For any given image, the robot first tries to estimate occlusion of objects and stability of object configurations using ASP-based reasoning. If an answer is not found, or an incorrect answer is found (during training), the robot automatically reasons with

[3] We use the terms "CR-Prolog" and "ASP" interchangeably.

(a) Execution Example 1. (b) Execution Example 2. (c) Additional example.

Fig. 3: (a) Relation between blue cube and red cube is important for the explanation in Execution Example 1; (b) The rubber duck is the focus of attention in Execution Example 2; and (c) Example of another trial (not described in this paper).

knowledge of the task to identify and ground the relevant axioms and relations in the image to determine the relevant regions of interest (ROIs). For instance, to explore the stability of object configurations in Figure 3a, the robot would automatically identify the stack with the blue, orange, and red blocks. Parameters of existing Convolutional Neural Network (CNN) architectures are tuned to map information from each ROI to the classification labels [14].

Image features and spatial relations extracted from ROIs and used to train a CNN, along with the labels for occlusion and stability, are also used to incrementally construct a decision tree (during training) that summarizes the corresponding state transitions. Branches of the tree that satisfy certain thresholds are used to construct candidate axioms that are validated and added to the ASP program for subsequent reasoning.

Human verbal/text input is parsed using existing software and a controlled vocabulary, labeled using a part-of-speech (POS) tagger, and normalized with the lemma list and WordNet. This text helps identify the type of request, which may be a desired goal or a question about decisions, beliefs, or hypothetical events. In the former case, the goal is sent to the ASP program for planning. The robot executes the plan, replanning when unexpected action outcomes cannot be explained, until the goal is achieved. In the latter case, the "Program Analyzer" reasons with the domain knowledge and current beliefs, and traces the evolution of beliefs by inferring the application of relevant axioms. The relevant literals are then inserted into generic response templates based on the controlled vocabulary, resulting in human-understandable (textual) descriptions that are converted to synthetic speech if needed. For more details, please see [15].

4 Execution Traces

The following execution traces illustrate our architecture's ability to construct relational descriptions explaining the decisions, beliefs, and the outcomes of hypothetical actions.

Execution Example 1 *[Plans, actions, and beliefs]*
Consider a robot that starts with objects as shown in Figure 3a. The robot is assigned the goal of achieving a state in which the red block is on top of the orange block, i.e., *holds(relation(on, red_block, orange_block), I)*. The robot answers the following questions *after executing* a plan and successfully achieving the assigned goal:

- **Human:** "Please describe the plan."
  Baxter: "I picked up the blue block. I put the blue block on the table. I picked up the orange block. I put the orange block on the table. I picked up the red block. I put the red block on the orange block."
- The human may ask the robot to justify a particular action in the executed plan.
  Human: "Why did you pick up the blue block at step 0?"
  Baxter: "Because I had to pick up the red block, and it was below the blue block."
  The constructed answer is also used to automatically highlight the relevant image regions that influenced this answer, as shown in Figure 3a.
- The human now may ask about particular actions that were not considered.
  Human: "Why did you not put down the orange block on the blue block?"
  Baxter: "Because the blue block is small." In the absence of a reference to a particular time step, the robot answers the question based on the single instance (in the executed plan) of putting the orange block on another block or surface. In addition, the answer is based on learned axiom, i.e., that objects with a small base are typically unstable.
- The human may also ask about particular beliefs.
  Human: "Why did you believe that the red block was below the blue block in the initial state?"
  Baxter: "Because I observed the red block below the blue block in step zero."

Execution Example 2 *[Learning and explanation]*
In some situations, the robot may be unable to respond to the human request or question because it is not possible to achieve the desired object configuration or belief. Even in such cases, our architecture enables the robot to answer explanatory questions. For instance, consider the simulated scene in Figure 3b, with the following interaction:

- **Human:** "Please put the pitcher on the duck."
  This action is not executed because of a constraint learned during a previous trial that any object configuration that has an object on another object with an irregular surface will be unstable.
- If asked, the robot can justify its decision of not executing the action.
  Human: "Why did you not put the pitcher on the duck?".
  Robot: "Because the duck has an irregular surface."
  The image region relevant to the construction of the robot's answer to the question posed by the human is automatically highlighted in the corresponding image, as indicated in Figure 3b above.

This example illustrates how integrating reasoning and learning helps justify the decision to not execute a requested action that will have an unfavorable outcome.

Summary: Overall, our architecture automatically reasons with just the relevant information; incrementally revises axioms; identifies image regions, attributes, and actions contributing to particular decisions and beliefs; and provides a partial understanding of the behavior of the learned CNNs. Experimental results indicate the ability to (i) incrementally reduce uncertainty about the scene by learning previously unknown axioms;

and (ii) reliably and efficiently construct explanations in the form of relational descriptions by automatically identifying and reasoning with the relevant knowledge despite noisy sensing and actuation [15]. In the future, we will integrate these capabilities with the architecture that tightly couples coarse-resolution non-monotonic logical reasoning and fine-resolution probabilistic reasoning [16].

References

1. Anjomshoae, S., Najjar, A., Calvaresi, D., Framling, K.: Explainable agents and robots: Results from a systematic literature review. In: International Conference on Autonomous Agents and Multiagent Systems. Montreal, Canada (2019)
2. Assaf, R., Schumann, A.: Explainable Deep Neural Networks for Multivariate Time Series Predictions. In: International Joint Conference on Artificial Intelligence. pp. 6488–6490. Macao, China (July 2019)
3. Bercher, P., Biundo, S., Geier, T., Hoernle, T., Nothdurft, F., Richter, F., Schattenberg, B.: Plan, repair, execute, explain - how planning helps to assemble your home theater. In: Twenty-Fourth International Conference on Automated Planning and Scheduling (2014)
4. Fandinno, J., Schulz, C.: Answering the "Why" in Answer Set Programming: A Survey of Explanation Approaches. Theory and Practice of Logic Programming **19**(2), 114–203 (2019)
5. Ferrand, G., Lessaint, W., Tessier, A.: Explanations and Proof Trees. Computing and Informatics **25**, 1001–1021 (2006)
6. Friedman, M.: Explanation and scientific understanding. Philosophy **71**(1), 5–19 (1974)
7. Gelfond, M., Kahl, Y.: Knowledge Representation, Reasoning and the Design of Intelligent Agents. Cambridge University Press (2014)
8. Genesereth, M.: The Use of Design Descriptions in Automated Diagnosis. Artificial Intelligence **24**, 411–436 (1984)
9. Koh, P.W., Liang, P.: Understanding Black-box Predictions via Influence Functions. In: International Conference on Machine Learning. pp. 1885–1894 (2017)
10. Langley, P., Meadows, B., Sridharan, M., Choi, D.: Explainable Agency for Intelligent Autonomous Systems. In: Innovative Applications of Artificial Intelligence (2017)
11. Langley, P.: Progress and Challenges in Research on Cognitive Architectures. In: AAAI Conference on Artificial Intelligence. San Francisco, USA (February 4-9, 2017)
12. Miller, T.: Explanations in Artificial Intelligence: Insights from the Social Sciences. Artificial Intelligence **267**, 1–38 (2019)
13. Mota, T., Sridharan, M.: Incrementally Grounding Expressions for Spatial Relations between Objects. In: International Joint Conference on Artificial Intelligence. pp. 1928–1934 (2018)
14. Mota, T., Sridharan, M.: Commonsense Reasoning and Knowledge Acquisition to Guide Deep Learning on Robots. In: Robotics Science and Systems (2019)
15. Mota, T., Sridharan, M.: Commonsense Reasoning and Deep Learning for Transparent Decision Making in Robotics. In: European Conference on Multiagent Systems (2020)
16. 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)
17. Sridharan, M., Meadows, B.: Towards a Theory of Explanations for Human-Robot Collaboration. Kunstliche Intelligenz **33**(4), 331–342 (2019)
18. Zhang, Y., Sreedharan, S., Kulkarni, A., Chakraborti, T., Zhuo, H.H., Kambhampati, S.: Plan explicability and predictability for robot task planning. In: International Conference on Robotics and Automation. pp. 1313–1320 (2017)