

# A Cognitive Architecture for Integrated Robot Systems

Mohan Sridharan<sup>1</sup>

<sup>1</sup>Intelligent Robotics Lab, School of Computer Science, University of Birmingham, UK

## Abstract

This paper describes an integrated architecture for robots that combines knowledge-based and data-driven methods for transparent reasoning, control, and learning. Specifically, the architecture builds on the principle of step-wise iterative refinement to support non-monotonic logical reasoning and probabilistic reasoning with tightly-coupled transition diagrams of the domain at different resolutions. Reasoning with prior domain knowledge and heuristic methods guide the interactive learning and revision of knowledge in the form of axioms governing change, predictive models controlling the robot's movement, and predictive models of the behavior of other agents. Furthermore, the interplay between these components is used to embed the principles of explainable agency, enabling a robot to provide on-demand relational descriptions of its decisions and beliefs in response to different types of questions.

## 1. Motivation

Consider a robot delivering objects or stacking objects in desired configurations. Such robots have to reason with different descriptions of prior domain knowledge and uncertainty. These descriptions include commonsense knowledge, e.g., relations between some domain objects and default statements such as “textbooks are usually in the library” that hold true in all but a few exceptional circumstances. Also, information extracted from noisy sensor inputs is often associated with quantitative measures of uncertainty, e.g., “I am 90% certain the robotics book is in the office”. In addition, the robot will have to revise its theory of actions and change over time, often using data-driven methods and noisy observations. Furthermore, for effective collaboration with other *agents* (e.g., humans, robots), the robot will need to reason with incrementally-revised models of the behavior of these agents, and provide on-demand descriptions of its decisions such that they make contact with human-level concepts such as goals and beliefs. In state of the art architectures that combine knowledge-based reasoning (e.g., for planning) and data-driven learning (e.g., for object recognition) for such *integrated robot systems*, the desired behavior thus poses open problems in knowledge representation, reasoning, control, and learning. This paper summarizes the capabilities of an architecture designed to address these problems.

## 2. Architecture and Insights

Figure 1(left) is an overview of the architecture that encodes the principle of *stepwise iterative refinement*. It is based on tightly-coupled transition diagrams at different resolutions, and may be viewed as a logician, statistician, and an explorer working together. Statements in an action


---


Cognitive AI 2023, 13th-15th November, 2023, Bari, Italy.

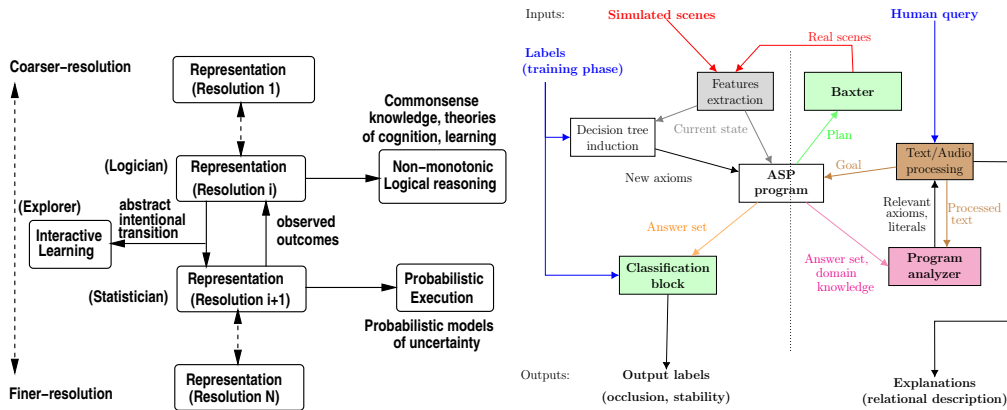
✉ [m.sridharan@bham.ac.uk](mailto:m.sridharan@bham.ac.uk) (M. Sridharan)

🌐 <https://www.cs.bham.ac.uk/~sridharm/> (M. Sridharan)

🆔 0000-0001-9922-8969 (M. Sridharan)

 © 2023 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)



**Figure 1:** Architecture represents and reasons with transition diagrams at different resolutions, combining strengths of declarative programming, probabilistic reasoning, and interactive learning.

language are used to describe these diagrams in the form of a sorted signature with statics, fluents, and actions; and three types of (deterministic, non-deterministic) axioms governing domain dynamics: causal laws, state constraints, and executability conditions. The domain's history includes the robot's observations, action executions, and prioritized defaults in the initial state. For any given task, the robot plans and executes actions at two resolutions, but is able to construct on-demand relational descriptions of decisions at other resolutions.

**Knowledge representation and reasoning:** The prior domain knowledge that the robot represents (as relational statements) and reasons with in the coarse resolution includes cognitive theories. For example, in addition to reasoning about the attributes and default room location of objects, a robot in an office building also considers an *adaptive theory of intentions* encoding principles of *non-procrastination* and *persistence* to respond quickly to unexpected successes and failures. The fine-resolution transition diagram is defined as a *refinement* of the coarse-resolution diagram, with a *theory of observations* modeling the robot's ability to sense the values of domain fluents. A robot in an office building now considers grid cells in rooms and object parts, attributes that were previously abstracted away, and reasons about *knowledge fluents* whose values are changed by observation actions. The definition of refinement guarantees that for any given coarse-resolution transition, there exists a path in the fine-resolution diagram between states that are refinements of the coarse-resolution states. Also, the refined diagram is *randomized* to model non-determinism. For any given goal, a plan of *intentional abstract actions* is obtained at the coarse-resolution through non-monotonic logical reasoning by translating the action language description to a *Answer Set Programming* program and solving it [1]. The robot implements each abstract transition as a sequence of concrete actions by automatically *zooming* to and reasoning with the *relevant* part of the fine-resolution diagram. The fine-resolution reasoning and execution uses probabilistic models of uncertainty (e.g., in perception and actuation) and relevant methods, adding outcomes to coarse-resolution history for subsequent reasoning [2, 3].

**Interactive learning, control, and transparency:** It is often difficult to use state of the art machine learning methods (e.g., based on deep networks) to revise the robot's knowledge over time. These methods require many training examples and considerable computational resources that are not available in many robot domains. Our architecture supports three strategies for

incremental, efficient acquisition of previously unknown action capabilities and axioms: (i) verbal descriptions of observed behavior; (ii) active exploration of new transitions; and (iii) reactive exploration of unexpected transitions. These strategies are formulated as interactive (e.g., inductive, reinforcement) learning problems. *Reasoning and learning guide each other*, enabling the robot to automatically identify and use the relevant information to construct mathematical models for these formulations [4]. For example, to estimate the stability of objects in a scene, the robot first attempts to reason with domain knowledge and spatial relations extracted from input images. Relevant regions of interest are automatically extracted from images for which reasoning is unable to make a decision (or makes an incorrect decision), and used to train a data-driven model (e.g., a deep network) for stability estimation. Information from these regions also induces axioms used for subsequent reasoning—Figure 1(right) provides an overview of this architecture. This approach substantially improves reliability and efficiency in comparison with data-driven models [5, 6, 7]. Our architecture supports a similar approach to address the discontinuous interaction dynamics experienced by a robot making and breaking contacts with objects and surfaces (e.g., while cleaning a table). The robot learns from a few trials to predict contact regions and end-effector measurements, using the error between prediction and measurements to adapt control laws in order to ensure smooth motion [8, 9].

Our architecture supports *explainable agency*, i.e., transparent reasoning and learning that makes contact with human concepts such as goals and beliefs. It encodes a *theory of explanations* comprising: (i) claims about representing, reasoning with, and learning knowledge to support relational descriptions of decisions; (ii) a characterization of explanations based on representational abstraction, and explanation specificity and verbosity; and (iii) a methodology for constructing such explanations. This theory is implemented in conjunction with the components summarized above—see Figure 1(right). The robot then provides on-demand relational descriptions of decisions and beliefs in response to different types of questions (e.g., descriptive, contrastive, counterfactual) posed by a human. The human is able to interactively obtain descriptions at the desired abstraction, specificity, and verbosity, with the robot automatically constructing and posing disambiguation questions to the human as needed [6, 10].

**Ad hoc teamwork:** The final component of our architecture enables collaboration without prior coordination, known as *ad hoc teamwork* (AHT), with the ad hoc robot (agent) selecting and executing actions to collaborate with teammates it has not worked with before. This robot performs non-monotonic logical reasoning with prior commonsense domain knowledge and predictive models of the behavior of the other agents (i.e., teammates and opponents). Our architecture encodes the principle of *ecological rationality*, which builds on the principle of bounded rationality and focuses on using heuristic methods for adaptive satisficing in decision making [11]. For example, the models predicting the behavior of other agents in benchmark multiagent collaboration domains are learned and revised rapidly using an ensemble of *fast and frugal trees*, with the performance of the team being better than (or comparable with) that provided by state of the art deep network methods that require orders of magnitude more training examples and computational resources [12, 13].

### 3. Execution Traces and Results

The following execution traces demonstrate some capabilities of our architecture.

**Execution Example 1.** *[Planning and learning]*

The robot in the *study* is asked to bring a cup to the *study*, i.e., the goal state contains:  $loc(C, study)$ ,  $not\ in\_hand(rob_1, C)$ , where  $C$  is a *cup*.

- The computed plan of abstract actions is:

$$\begin{aligned} &move(rob_1, kitchen),\ pickup(rob_1, C), \\ &move(rob_1, study),\ putdown(rob_1, C) \end{aligned}$$

which uses the default knowledge that cups are usually in the *kitchen* next to the *study*.

- To implement each abstract transition, the robot zooms to the relevant fine-resolution knowledge, e.g., only cells in the *study* and *kitchen* are relevant to the first *move* action.
- The zoomed description is used to obtain a probabilistic policy that is invoked repeatedly to execute a sequence of concrete actions that implements the abstract action, e.g., robot is in a cell in the *kitchen* after first *move*. Other actions are executed in a similar manner.
- The robot's attempt to pick up a cup in the kitchen fails. The robot observes that the cup is *heavy* while its arm is *light*. It then learns the executability condition:

$$\begin{aligned} \text{impossible } pickup(rob_1, C) \text{ if } arm(rob_1, light), \\ \qquad \qquad \qquad obj\_weight(C, heavy) \end{aligned}$$

Any such learned axiom is merged with the existing knowledge.

- The robot also provides on-demand explanations at a suitable level of abstraction.

**Human:** "Please describe the executed plan in detail."

**Robot:** "I moved to cell  $c_2$  in the *kitchen*. I picked the large cup by its handle from the counter [...] I moved to cell  $c_4$  of the *study*. I put the cup down on the red table."

**Execution Example 2.** *[Explain plans and beliefs]*

In the scene in the first two images of Figure 2, the robot's goal is to achieve a state in which the red cube is on top of the orange cube. The following interaction takes place *after* the robot executes a plan to achieve the goal.

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

**Baxter:** "I picked up the blue cube. I put the blue cube on the table. I picked up the orange cube. I put the orange cube on the table. I picked up the red cube. I put the red cube on the orange cube."

- The robot may have to justify a particular action.

**Human:** "Why did you pick up the blue cube at step 0?"

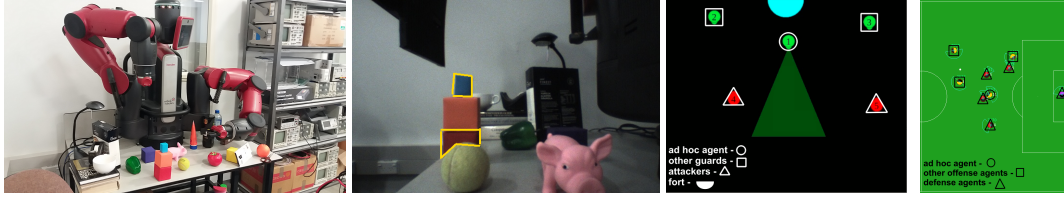
**Baxter:** "Because I had to pick up the red cube, and it was below the blue cube."

This answer is also used to automatically highlight the relevant image regions that influenced this answer, providing additional transparency—see Figure 2(right).

- The robot may have to answer a contrastive question.

**Human:** "Why did you not put down the orange cube on the blue cube?"

**Baxter:** "Because the blue cube is small." In the absence of any reference to a particular time step, the robot considers the single instance (in the executed plan) of putting the orange cube on another cube or surface. The answer is based on learned default knowledge that any structure with a large object on a small object is unstable.



**Figure 2:** Scenario for some robot experiments and screenshots of simulation environments.

- The human may ask the robot to justify beliefs.  
**Human:** "Why did you believe 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."
- The robot can run *mental simulations* to answer counterfactual questions.  
**Human:** "What would happen if the ball is rolled?"  
**Baxter:** "The structure of blocks would be unstable".

For more extensive evaluation of our architecture’s capabilities, we also used complex simulation environments. For example, for evaluating the AHT capability, we explored the *Fort Attack* (FA) domain [14] and *Half Field Offense* (HFO) domain [15]—last two images of Figure 2—benchmarks for multiagent collaboration. In FA, guards (in green, including one ad hoc agent) had to protect a fort from attackers (in red). Any episode ended when all members of a team were killed, an attacker reached the fort, or guards protected the fort for a sufficient time period. Each agent could move in a particular direction or shoot an opponent within a range. In HFO, members of the offense team (including one ad hoc agent) had to score a goal against a team of defenders and one goalkeeper; the game ended when the offense team scored a goal, a defender gained possession of the ball, the ball went out of bounds, or a maximum time limit was exceeded. Each agent could dribble the ball, pass to another agent, or kick the ball toward the goal. We were able to experimentally demonstrate that our architecture enables the ad hoc agent to: (i) adapt to different teammate and opponent types, and to changes in team composition; (ii) incrementally learn and revise other agents’ behavioral models from limited examples; (iii) improve team performance in comparison with a state of the art data-driven method that involved deep reinforcement learning in graph neural networks; and (iv) generate relational descriptions as explanations of its decisions and beliefs in response to different types of questions.

Complete details and experimental results of evaluating our architecture in simulation and on physical robots are described in relevant papers [2, 3, 4, 6, 8, 10, 12, 13].

## Acknowledgments

The architecture described in this paper is the result of research threads pursued in collaboration with Hasra Dodamegama, Michael Gelfond, Rocio Gomez, Ben Meadows, Tiago Mota, Heather Riley, Saif Sidhik, Jeremy Wyatt, and Shiqi Zhang. This work was supported in part by the U.S. Office of Naval Research Awards N00014-13-1-0766, N00014-17-1-2434 and N00014-20-1-2390, the Asian Office of Aerospace Research and Development award FA2386-16-1-4071, and the U.K. Engineering and Physical Sciences Research Council award EP/S032487/1. All conclusions reported in this paper are those of the author alone.

## References

- [1] M. Gebser, R. Kaminski, B. Kaufmann, T. Schaub, *Answer Set Solving in Practice*, Synthesis Lectures on Artificial Intelligence and Machine Learning, Morgan Claypool Publishers, 2012.
- [2] R. Gomez, M. Sridharan, H. Riley, 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 (2021) 179–208.
- [3] M. Sridharan, M. Gelfond, S. Zhang, J. Wyatt, REBA: A Refinement-Based Architecture for Knowledge Representation and Reasoning in Robotics, *Journal of Artificial Intelligence Research* 65 (2019) 87–180.
- [4] M. Sridharan, B. Meadows, Knowledge Representation and Interactive Learning of Domain Knowledge for Human-Robot Collaboration, *Advances in Cognitive Systems* 7 (2018) 77–96.
- [5] M. Sridharan, T. Mota, Towards Combining Commonsense Reasoning and Knowledge Acquisition to Guide Deep Learning, *Autonomous Agents and Multi-Agent Systems* 37 (2023).
- [6] T. Mota, M. Sridharan, A. Leonardis, Integrated Commonsense Reasoning and Deep Learning for Transparent Decision Making in Robotics, *Springer Nature CS* 2 (2021) 1–18.
- [7] H. Riley, M. Sridharan, Integrating Non-monotonic Logical Reasoning and Inductive Learning With Deep Learning for Explainable Visual Question Answering, *Frontiers in Robotics and AI*, special issue on Combining Symbolic Reasoning and Data-Driven Learning for Decision-Making 6 (2019) 20.
- [8] S. Sidhik, M. Sridharan, D. Ruiken, Towards a Framework for Changing-Contact Manipulation Tasks, in: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021.
- [9] M. Mathew, S. Sidhik, M. Sridharan, M. Azad, A. Hayashi, J. Wyatt, Online Learning of Feed-Forward Models for Task-Space Variable Impedance Control, in: *IEEE-RAS International Conference on Humanoid Robotics*, 2019.
- [10] M. Sridharan, B. Meadows, Towards a Theory of Explanations for Human-Robot Collaboration, *Kunstliche Intelligenz* 33 (2019) 331–342.
- [11] G. Gigerenzer, What is Bounded Rationality?, in: *Routledge Handbook of Bounded Rationality*, Routledge, 2020.
- [12] H. Dodampegama, M. Sridharan, Knowledge-based Reasoning and Learning under Partial Observability in Ad Hoc Teamwork, *Theory and Practice of Logic Programming* 23 (2023) 696–714.
- [13] H. Dodampegama, M. Sridharan, Back to the Future: Toward a Hybrid Architecture for Ad Hoc Teamwork, in: *AAAI Conference on Artificial Intelligence*, Washington DC, USA, 2023.
- [14] A. Deka, K. Sycara, Natural Emergence of Heterogeneous Strategies in Artificially Intelligent Competitive Teams, Technical Report, <https://arxiv.org/abs/2007.03102>, 2020.
- [15] M. Hausknecht, P. Mupparaju, S. Subramanian, S. Kalyanakrishnan, P. Stone, Half field offense: An environment for multiagent learning and ad hoc teamwork, in: *AAMAS Adaptive Learning Agents Workshop*, 2016.