Explainable Agency in Integrated Cognitive Systems

Advanced Course at ESSAI 2024

Mohan Sridharan
Chair in Robot Systems
School of Informatics, University of Edinburgh (UK)
m.sridharan@ed.ac.uk
https://homepages.inf.ed.ac.uk/msridhar/

July 15-19, 2024

Objectives

- **Advanced course** at the intersection of multiple topics.
  - Knowledge-based reasoning.
  - Data-driven learning.
  - Integrated cognitive systems.
  - Explainable agency.

- **Scope** of this course:
  - Non-monotonic logic, probability theory.
  - Machine learning, reinforcement learning, deep learning.
  - Robots that sense, reason, act, learn.
  - Relational descriptions of decisions; theory of mind.

- **Format:** interactive, discussions, examples.

Tentative Outline

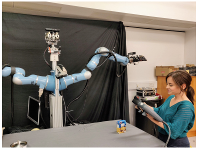
1. (L1) Knowledge representation and reasoning (KRR) I.
2. (L2) KRR II and learning.
3. (L3) KRR, learning, and control.
4. (L4) KRR, teamwork, and learning.
5. (L5) Explanations, integrated systems, closing the loop.

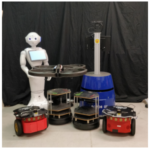
Prerequisites and Assumptions

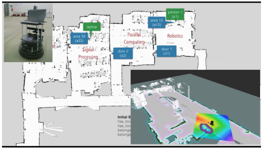
- Basic proficiency in logic and probability theory.
- Basic knowledge of reasoning, learning.
- Interest in integrated cognitive systems.
- Interest in interdisciplinary topics.

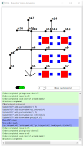
Illustrative Domain: Robot Assistants

Robot assistant finding and manipulating objects.









Integrated Cognitive Robot Systems: Desiderata

- Enable robots to represent, reason, and act with different descriptions of domain knowledge and uncertainty.
  "Books are usually in the library"
  "I am 90% certain the robotics book is in the library"

- Enable robots to learn interactively and cumulatively from sensor inputs and limited human feedback.
  Learn actions, action capabilities, domain dynamics
  "Robot with weak arm cannot lift heavy box"

- Enable designers to understand the robot's behavior and establish that it satisfies desirable properties.
  Explainable agency, intentions, goals, measures
  "What would happen if I dropped the spoon on the table?"

Inspiration and Core Ideas

- Cognitive systems inspired by human cognition, control.
- Represent, reason, act, learn jointly at different abstractions with different schemes.
- Logician, statistician, creative explorer; formal coupling not unified representation.
- Combine knowledge-based and data-driven reasoning and learning; predictive, cumulative, interactive, relevant.
- Explanations: relational descriptions of decisions, beliefs; Questions: descriptive, causal, contrastive, counterfactual.

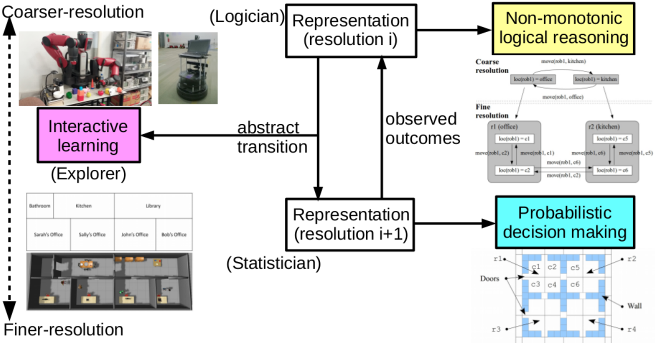
Shiqi Zhang and Mohan Sridharan. **A Survey of Knowledge-based Sequential Decision Making under Uncertainty**. Artificial Intelligence Magazine, 43(2):249-266, 2022.

Claims: Representation + Reasoning + Learning

1. Distributed representation of knowledge (commonsense, probabilistic) at different coupled abstractions.
2. Separation of concerns (domain-specific/independent knowledge, observations); common methodology.
3. Knowledge elements support non-monotonic revision; revise previously held conclusions.
4. "Here and there" reasoning; satisfiability, stochastic policies. Often focus on rationality and not on optimality!

Illustrative domains: visual planning, scene understanding and manipulation problems in robotics.

Refinement-Based Architecture: Overview



Exploit complementary strengths of non-monotonic logical reasoning, probabilistic reasoning, and interactive learning.

Mohan Sridharan. **REBA-KRL: Refinement-Based Architecture for Knowledge Representation, Explainable Reasoning, and Interactive Learning in Robotics**. In the European Conference on Artificial Intelligence, 2020.

Mohan Sridharan, Michael Gelfond, Shiqi Zhang and Jeremy Wyatt. **REBA: Refinement-based Architecture for Knowledge Representation and Reasoning in Robotics**. In Journal of Artificial Intelligence Research, 65:87-180, May 2019.

Tentative Outline

1. (L1) Knowledge representation and reasoning (KRR) I.
2. (L2) KRR II and learning.
3. (L3) KRR, learning, and control.
4. (L4) KRR, teamwork, and learning.
5. (L5) Explanations, integrated systems, closing the loop.

Robot Waiter Example: Reasoning (Video)

Example

Action Language + Logician's System Description

- AL_d: formal description of transition diagrams.
- System description $\mathcal{D}_C$: sorted signature Σ_C and axioms as statements in AL_d.
- Statics: $next\_to(place, place)$.
- Fluents: $loc : thing \rightarrow place$, $in\_hand : robot \times object \rightarrow boolean$.
- Actions: $move(robot, place)$, $grasp(robot, object)$, $exo\_move(object, place)$, $exo\_lock(place)$.

Logician's System Description: Axioms

- Causal law, state constraint, executability condition.
- Causal laws:

$$move(rob1, Pl) \ \textbf{causes} \ loc(rob1) = Pl$$
$$grasp(rob1, Ob) \ \textbf{causes} \ in\_hand(rob1, Ob)$$
$$putdown(rob1, Ob) \ \textbf{causes} \ \neg in\_hand(rob1, Ob)$$

- State constraints:

$$loc(Ob) = Pl \ \textbf{if} \ loc(rob1) = Pl, \ \ in\_hand(rob1, Ob)$$
$$loc(Th) \neq Pl_1 \ \textbf{if} \ loc(Th) = Pl_2, \ Pl_1 \neq Pl_2$$

- Executability conditions:

$$\textbf{impossible} \ grasp(rob1, Ob) \ \textbf{if} \ loc(rob1) \neq loc(Ob)$$
$$\textbf{impossible} \ grasp(rob1, Ob) \ \textbf{if} \ in\_hand(rob1, Ob)$$
$$\textbf{impossible} \ putdown(rob1, Ob) \ \textbf{if} \ not \ in\_hand(rob1, Ob)$$

Histories with Defaults

- History contains records of observations and actions:

$$obs(fluent, boolean, step)$$
$$hpd(action, step)$$

- Expand to include initial state defaults:

$$\textbf{initial default } loc(X) = library \;\; \textbf{if} \;\; textbook(X)$$
$$\textbf{initial default } loc(X) = office \;\; \textbf{if} \;\; textbook(X), \; loc(X) \neq library$$

- Consistency-restoring rules for recovery and diagnostics.

$$loc(X) \neq library \stackrel{+}{\leftarrow} textbook(X)$$

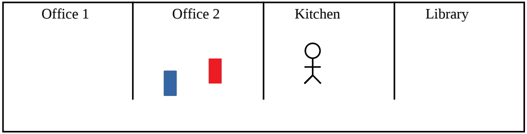
Modeling Intentions: What are they?

- Many different "definitions" proposed; also survey papers.
- Inferred from sensor inputs (gaze, gestures), intermediate features (tracked body pose and movement), or "meta" concepts.
- Intention as joint high-level concept defined over robot's beliefs and actions.

Tom Carlson and Yiannis Demiris. **Human-Wheelchair Collaboration Through Prediction of Intention and Adaptive Assistance**. International Conference on Robotics and Automation, 2008.
Adam Norton, Henny Admoni, Jacob Crandall, Tesca Fitzgerald, Alvika Gautam, Michael Goodrich, Amy Saretsky, Matthias Scheutz, Reid Simmons, Aaron Steinfeld, and Holly Yanco. **Metrics for Robot Proficiency Self-Assessment and Communication of Proficiency in Human-Robot Teams**. Transactions on Human-Robot Interaction, 11(3),2022.

Modeling Intentions I

- Unexpected success and failure.



- Persistence, non-procrastination, relevance.
- Expand to $\Pi(\mathcal{D}'_C$ and $\mathcal{H}'_C)$; activities; mental fluents and mental actions.

Rocio Gomez, Mohan Sridharan, and Heather Riley. **What do you really want to do? Towards a Theory of Intentions for Human-Robot Collaboration**. In Annals of Mathematics and Artificial Intelligence, special issue on Commonsense Reasoning, 89(1): 179-208, February 2021.

Modeling Intentions II

- Expand Σ_H:
  - Activity: goal, plan, name.
  - Mental fluents and actions.
- Expand axioms to represent action effects, start/stop activity, generate intentional actions.
- Expand $\mathcal{H}$, e.g., to model attempted actions:

$$obs(fluent, boolean, step),\ \ hpd(action, step)$$
$$attempt(action, step), \neg hpd(action, step)$$

Modeling Affordances: What are they?

- Multiple interpretations and surveys, surveys of surveys?
- Attribute of object, agent, environment?
- Behavior determined by agent's cognitive process and environment.
- Affordance as joint attribute of agent and object in the context of specific actions.

Keith S. Jones. **What is an Affordance?** Ecological Psychology, 15(2):104-114, 2003.
L. Jamone, E. Ugur, A. Cangelosi, L. Fadiga, A. Bernardino, J. Piater, and J. Santos-Victor. **Affordances in Psychology, Neuroscience and Robotics: A Survey**. IEEE Transactions on Cognitive and Developmental Systems, 2016.
P. Zech, S. Haller, S. R. Lakani, B. Ridge, E. Ugur, and J. Piater. **Computational Models of Affordance in Robotics: A Taxonomy and Systematic Classification**. Adaptive Behavior,25(5): 235-271, 2017.
V. Sarathy and M. Scheutz. **A Logic-based Computational Framework for Inferring Cognitive Affordances**. IEEE Transactions on Cognitive and Developmental Systems, 10(1):26-43, 2018.

Modeling Affordances

- Affordance as combination of attributes of object(s) and agent(s) with reference to an action.

- Action can have one or more enabling or forbidding affordances.

$$
\begin{aligned}
\textbf{impossible} \;\; pickup(R, O) \;\; &\textbf{if} \;\; obj\_weight(O, heavy), \\
&\quad not \;\; aff\_enables(id_1, pickup(R, O)) \\
aff\_enables(id_1, pickup(R, O)) \;\; &\textbf{if} \;\; strength(R, strong) \\
\textbf{impossible} \;\; A \;\; &\textbf{if} \;\; aff\_forbids(ID, A) \\
aff\_forbids(id_j, A) \;\; &\textbf{if} \;\; \ldots
\end{aligned}
$$

- Distributed representation supports information reuse.

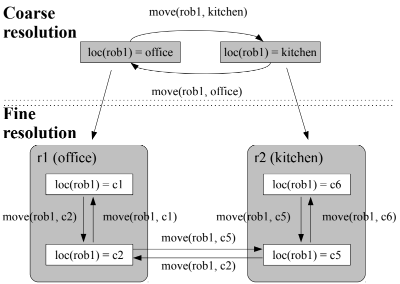
Pat Langley, Mohan Sridharan, and Ben Meadows. **Representation, Use, and Acquisition of Affordances in Cognitive Systems**. AAAI Spring Symposium on Integrating Representation, Reasoning, Learning, and Execution for Goal Directed Autonomy, Stanford, USA, March 26-28, 2018.
Mohan Sridharan and Ben Meadows. **Knowledge Representation and Interactive Learning of Domain Knowledge for Human-Robot Collaboration**. Advances in Cognitive Systems Journal, 7:77-96, 2018.

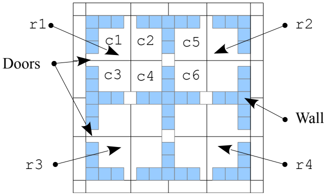
Logician's Reasoning

- Logician's description:
  - **Input**: (a) $\mathcal{D}_C$ and history $\mathcal{H}_C$; (b) Goal.
  - **Output**: plan and next transition $T = \langle \sigma_1, a^C, \sigma_2 \rangle$ to execute.
  - Can translate to different formalisms for reasoning.
- Answer Set Prolog program $\Pi(\mathcal{D}_C, \mathcal{H}_C)$. Reason by computing answer sets. Supports non-monotonic logical reasoning.
- Default negation and epistemic disjunction.

$\neg$ l	l is believed to be false
not l	it is not believed that l is true
p $\vee$ $\neg$ p	is a tautology
p or $\neg$ p	is not tautological

Refinement: Overview





- Refinement: describe ($\mathcal{D}_C$) at finer resolution ($\mathcal{D}_F$).
- Formal relationships; add knowledge fluents and actions.

Mohan Sridharan, Michael Gelfond, Shiqi Zhang and Jeremy Wyatt. **REBA: Refinement-based Architecture for Knowledge Representation and Reasoning in Robotics**. In Journal of Artificial Intelligence Research, 65:87-180, May 2019.

Weak Refinement ($\mathcal{D}_{F,nobs}$) I

Refine signature Σ_F of τ_F:

- Inherit basic sorts and define s^* counterparts.

$$place = \{r_1, \ldots, r_n\}, \quad place^* = \{c_1, \ldots, c_m\}$$
$$cup = \{cup_1\}, \quad cup^* = \{cup\_base_1, cup\_handle_1\}$$

- Add new statics, fluents, and actions; define component relationships.

$$next\_to^*(place^*, place^*)$$
$$loc^* : thing \rightarrow place^*, \quad cup \notin thing, \quad loc^* : cup^* \rightarrow place^*$$
$$move^*(robot, place^*), \quad grasp^*(robot, cup^*)$$
$$component(place^*, place), \quad component(cup^*, cup)$$

Weak Refinement ($\mathcal{D}_{F,nobs}$) II

- Causal laws:

$$move^*(R, C) \ \textbf{causes} \ loc^*(R) = C$$
$$grasp(R, O) \ \textbf{causes} \ in\_hand(R, O), O \neq cup_1$$
$$putdown^*(R, O) \ \textbf{causes} \ \neg in\_hand^*(R, O), \ O \in cup^*$$

- State constraints (including bridge axioms):

$$loc^*(O) = C \ \textbf{if} \ loc^*(R) = C, \ in\_hand(R, O)$$
$$next\_to^*(C_2, C_1) \ \textbf{if} \ next\_to^*(C_1, C_2)$$
$$loc(Th) = P \ \textbf{if} \ component(C, P), \ loc^*(Th) = C$$
$$loc^*(O) = C \ \textbf{if} \ loc^*(OPart) = C, \ component(OPart, O)$$

- Executability conditions:

$$\textbf{impossible} \ move^*(R, C_2) \ \textbf{if} \ loc^*(R) = C_1, \ \neg next\_to^*(C_1, C_2)$$
$$\textbf{impossible} \ grasp(R, O) \ \textbf{if} \ loc^*(R) \neq loc^*(O)$$
$$\textbf{impossible} \ putdown(R, O) \ \textbf{if} \ not \ in\_hand(R, O)$$

Strong Refinement ($\mathcal{D}_F$)

- Introduce theory of observations: knowledge fluents, knowledge-producing actions.

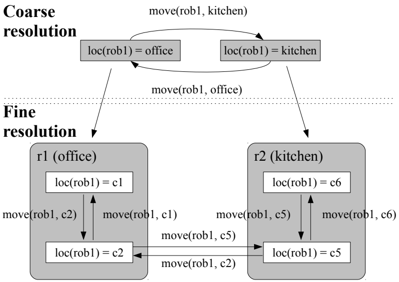
- Introduce new fluents, actions, and axioms to observe the environment.

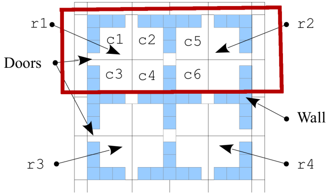
$$observed_f : robot \times dom(f) \times range(f) \rightarrow \{true, false, undet\}$$
$$test_f : robot \times dom(f) \times range(f) \rightarrow boolean$$

- Inherit axioms of $\mathcal{D}_C$; expand as appropriate.

$$test_{f*}(R, \bar{X}, Y) \;\; \textbf{causes} \;\; observed_{f*}(R, \bar{X}, Y) \;\; \textbf{if} \;\; f^*(\bar{X}) = Y$$
$$test_{f*}(R, \bar{X}, Y) \;\; \textbf{causes} \;\; \neg observed_{f*}(R, \bar{X}, Y) \;\; \textbf{if} \;\; f^*(\bar{X}) \neq Y$$
$$\textbf{impossible} \;\; test_{f*}(R, \bar{X}, Y) \;\; \textbf{if} \;\; \neg can\_be\_observed_{f*}(R, \bar{X}, Y)$$

Randomize and Zoom to $\mathcal{D}_{FR}(T)$



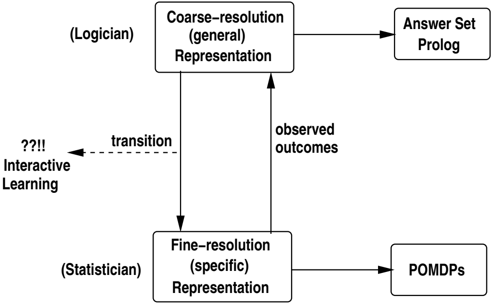


- Randomization to capture non-determinism ($\mathcal{D}_{FR}$).

$$move^*(R, C_2) \ \textbf{causes} \ loc^*(R) = \{C : range(loc^*(R), C)\}$$

- Collect statistics to compute probabilities.
- Automatically zoom to $\mathcal{D}_{FR}(T)$ for $T = \langle \sigma_1, a^C, \sigma_2 \rangle$.

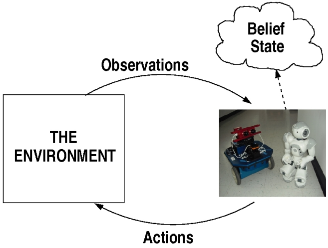
Statistician's task



- $\mathcal{D}_{FR}(T)$ and statistics to construct and solve Partially Observable Markov Decision Process (POMDP).
- Compute policy mapping belief states to actions. Invoke to execute sequence of actions.
- Add observed outcomes to $\mathcal{H}_C$ to be used by logician.

POMDP: Overview

- Tuple: $\langle S, A, Z, T, O, R \rangle$
  Object in domain with four rooms:
  $$B_t = [0.2, 0.1, 0.05, 0.65]$$
  Policy $\pi : B_t \mapsto a_{t+1}$



- **Challenges:**
  - Model parameters may not be known and may change.
  - State space and computational complexity.
- **Observation:**
  - Only a subset of scenes relevant to task.
  - *Visual processing can be organized hierarchically*.

POMDP Construction

- POMDP tuple: $\langle S^F, A^F, Z^F, T^F, O^F, R^F \rangle$.
- Belief states: probability distributions over physical states (p-state): $B_t = [0.1, 0.8, 0.05, 0.05]$.
- Transition function $T(s, a, s') \rightarrow [0, 1]$ is probability of state transitions.
- Observation function $O(s, a, f = v) \rightarrow [0, 1]$ is probability of observing $f = v$ by executing a in s.

POMDP Construction (contd.)

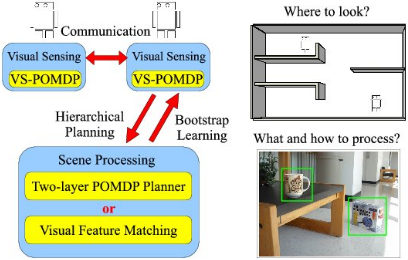
- Reward function $R(s, a, s') \rightarrow \Re$ assigns higher utility to transitions to goal state, and assigns costs to other actions.
- POMDP solved to obtain policy $\pi : B_t \rightarrow a_t$.
- Use policy to repeatedly choose actions, obtain observations and update belief state, until goal achieved with high probability:

$$b_{t+1}(s_{t+1}) \propto O(s_{t+1}, a_t, o_{t+1}) \sum_s T(s, a_t, s_{t+1}) \cdot b_t(s)$$

- Communicate observations and action outcomes to logician ($\mathcal{H}_C$) to revise knowledge base.

Hierarchical POMDPs

- Where to look? What to process? How to process?



- Policy kernels and adaptive observation functions.
- Automatic belief propagation and model generation at all levels for reliable and efficient operation.

Shiqi Zhang, Mohan Sridharan and Jeremy Wyatt. **Mixed Logical Inference and Probabilistic Planning for Robots in Unreliable Worlds**. In the IEEE Transactions on Robotics, 31(3):699-713, June 2015.
Shiqi Zhang, Mohan Sridharan and Christian Washington. **Active Visual Planning for Mobile Robot Teams using Hierarchical POMDPs**. In the IEEE Transactions on Robotics, 29(4): 975-985, 2013.

Detour: Bayesian Filtering Basics

- Inputs:
  - Stream of observations z and actions u: $\{u_1, z_1, \ldots, u_t, z_t\}$
  - Sensor model: $p(z|x)$
  - Action model: $p(x'|u, x)$
  - Prior probability of system state: $p(x)$
- Outputs:
  - Estimate the state $\boldsymbol{x}$ of a *dynamical system*.
  - Posterior of state, called the belief:

$$bel(x_t) = p(x_t|u_1, z_1, \ldots, u_t, z_t)$$

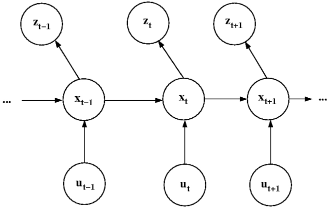
Markov Assumption

- First-order Markov (conditional independence) assumption:

$$p(x_t|x_0,\ldots,x_{t-1}) = p(x_t|x_{t-1})$$

- Bayesian filtering:

$$p(z_t|x_{0:t}, z_{1:t}, u_{1:t}) = p(z_t|x_t)$$
$$p(x_t|x_{1:t-1}, z_{1:t}, u_{1:t}) = p(x_t|x_{t-1}, u_t)$$



Bayes Inference

- Bayes prediction and correction:

$$\forall x_t : \; bel(x_t) = \eta \; p(z_t|x_t) \int p(x_t|u_t, x_{t-1}) \, bel(x_{t-1}) \, dx_{t-1}$$

$$\forall k : \; p_{k,t} = \eta \; p(z_t|X_t = x_k) \sum_i p(X_t = x_k|u_t, X_{t-1} = x_i) \, p_{i,t-1}$$

- Bayes filter:

$$\forall x_t : \overline{bel}(x_t) = \int p(x_t|u_t, x_{t-1}) \, bel(x_{t-1}) \, dx_{t-1}$$

$$bel(x_t) = \eta \; p(z_t|x_t) \, \overline{bel}(x_t)$$

- Discrete Bayes filter:

$$\forall k : \overline{p}_{k,j} = \sum_i p(X_t = x_k|u_t, X_{t-1} = x_i) \, p_{i,t-1}$$

$$p_{k,j} = \eta \; p(z_t|X_t = x_k) \, \overline{p}_{k,j}$$

Bayesian Filters in Practice: Kalman Filter

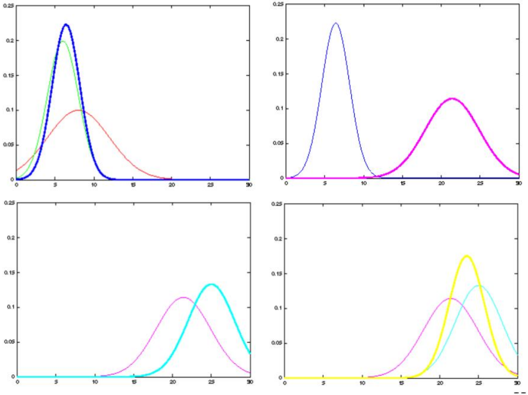


Image from *Probabilistic Robotics* book by Thrun, Burgard, and Fox.

- Bayesian Networks, POMDP, Hidden Markov Model.

Bayesian Filters in Practice: Particle Filter I

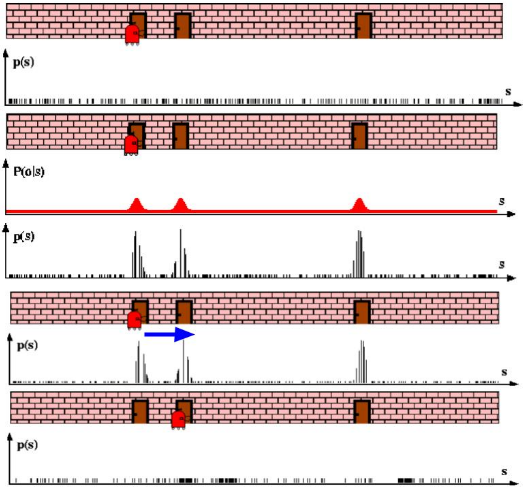


Image from *Probabilistic Robotics* book by Thrun, Burgard, and Fox.

Bayesian Filters in Practice: Particle Filter II

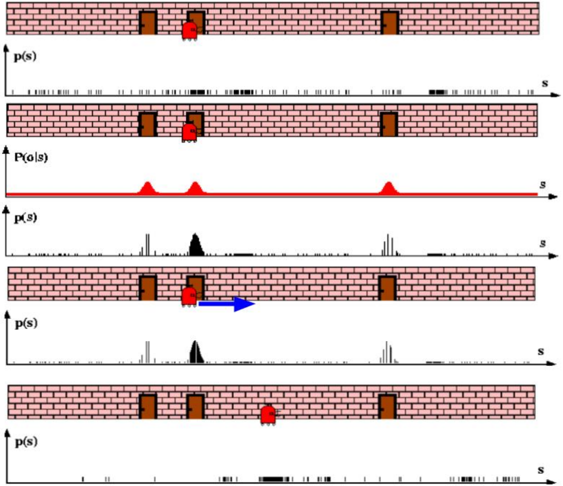


Image from *Probabilistic Robotics* book by Thrun, Burgard, and Fox.

Execution Trace: Reasoning

- **Goal:** some cup C has to be in the office: $loc(C) = office, \;\; \neg in\_hand(rob_1, C)$.

- **Initial knowledge** (subset): $loc(rob_1, office)$, $obj\_weight(cup_1, heavy)$, $arm\_type(rob_1, electromagnetic)$.

- Based on **default**: $loc(cup_1) = kitchen$.

- One possible plan from ASP-based inference:

$$move(rob_1, kitchen), \;\; grasp(rob_1, cup_1)$$
$$move(rob_1, office), \;\; putdown(rob_1, cup_1)$$

- Assume rob_1 is in $kitchen$. Has to locate and grasp cup_1.

Execution Trace: Reasoning

- Some **relevant** literals: $loc(rob_1) = c_i$, $loc(cup_1) = c_j$, where $c_i, c_j \in kitchen$.
- Possible action sequence (executed probabilistically):

$$\begin{array}{ll} move(rob_1, c_3) & \\ test(rob_1, loc(cup_1), c_3) & \%\ cup_1 \text{ not observed} \\ move(rob_1, c_5) & \\ test(rob_1, loc(cup_1), c_5) & \%\ cup_1 \text{ observed} \\ grasp(rob_1, cup_1) & \end{array}$$

- Proceed if grasping succeeds; what to do when it fails?

Robot Waiter Revisited: Reasoning (Video)

Example

Robot Waiter Video 2

Example

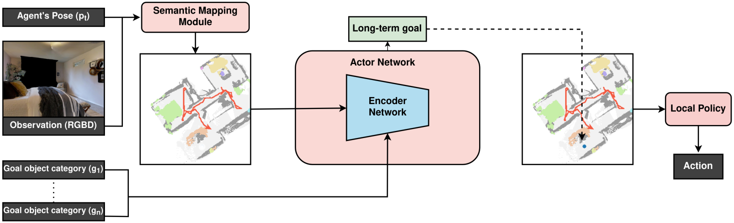
Advantages

- Step-wise refinement simplifies design and implementation.
- Increases confidence in behavior, promotes scalability.
- Separation of concerns: domain-independent/specific knowledge.
- Designer follows pre-defined steps; otherwise automated.
- Non-monotonic logical reasoning and probabilistic reasoning inform and guide each other.

Tentative Outline

1. (L1) Knowledge representation and reasoning (KRR) I.
2. (L2) KRR II and learning.
3. (L3) KRR, learning, and control.
4. (L4) KRR, teamwork, and learning.
5. (L5) Explanations, integrated systems, closing the loop.

Using Learned Knowledge I: Overview



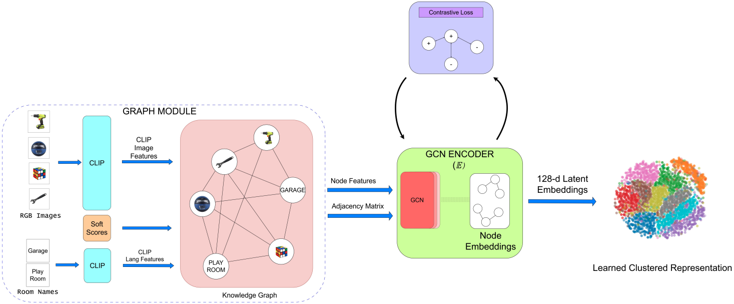
N. Gireesh, A. Agrawal, A. Datta, S. Banerjee, M. Sridharan, B. Bhowmick, and M. Krishna. **Sequence-Agnostic Multi-Object Navigation**. IEEE International Conference on Robotics and Automation (ICRA), May 2023.

Using Learned Knowledge I: Video

SAM Example

N. Gireesh, A. Agrawal, A. Datta, S. Banerjee, M. Sridharan, B. Bhowmick, and M. Krishna. **Sequence-Agnostic Multi-Object Navigation**. IEEE International Conference on Robotics and Automation (ICRA), May 2023.

Using Learned Knowledge II: Overview



A. Agrawal, R. Arora, A. Datta, S. Banerjee, B. Bhowmick, K.M. Jatavallabhula, M. Sridharan, and M. Krishna. **CLIPGraphs: Multimodal Graph Networks to Infer Object-Room Affinities**. In the IEEE International Conference on Robot and Human Interactive Communication (RO-MAN), August 2023.

Using Learned Knowledge II: Video

CLIP Example

A. Agrawal, R. Arora, A. Datta, S. Banerjee, B. Bhowmick, K.M. Jatavallabhula, M. Sridharan, and M. Krishna. **CLIPGraphs: Multimodal Graph Networks to Infer Object-Room Affinities**. In the IEEE International Conference on Robot and Human Interactive Communication (RO-MAN), August 2023.

Anticipate and Act: Video

LLM-PDDL Example

R. Arora, S. Singh, K. Swaminathan, S. Banerjee, B. Bhowmick, K. M. Jatavallabhula, M. Sridharan, and M. Krishna. **Anticipate & Act: Integrating LLMs and Classical Planning for Efficient Task Execution in Household Environments**. In the IEEE International Conference on Robotics and Automation (ICRA), May 2024.

Coming Up...

- Learning to augment and revise existing knowledge.
- Reasoning and learning informing and guiding each other.
- Robot control, teamwork, and learning.
- Explanations, integrated cognitive systems.

That's all folks!