Learning to Plan with Logical Automata

Brandon Araki\textsuperscript{1,*}, Kiran Vodrahalli\textsuperscript{2,*}, Thomas Leech\textsuperscript{1,3}, Cristian-Ioan Vasile\textsuperscript{1}, Mark Donahue\textsuperscript{3}, Daniela Rus\textsuperscript{1}

\textsuperscript{1}MIT; \textsuperscript{2}Columbia University; \textsuperscript{3}MIT Lincoln Laboratory; *Equal contributors

Overview

**GOAL:** Learn from demonstrations not just a low-level policy but also a high-level policy that is *interpretable* and *manipulable*.

*Interpretable:* The structure and weights of the learned policy are grounded directly in a formal language.

*Manipulable:* A human operator can easily modify the learned policy to perform similar but different policies.

Representation of Rules

**Rules:**

- Pack sandwich or burger; Then pack banana

**Formal logic (LTL):**

- \( (\Diamond \text{Packed}) \land (\Diamond \text{Sandwich}) \land (\Diamond \text{Burger}) \land (\Diamond \text{Banana}) \)

**Finite State Automaton (FSA):**

- States: S0, S1, S2, S3, G
- Transitions:
  - From S0: Pack sandwich or burger; then pack banana
  - From S1: Pack banana; then pack only burger

**Transition Matrix (TM):**

- Transition probabilities

Logic-based Value Iteration Networks (LVIN)

Case Study: Lunchbox Packing

- **Learned FSA:**
  - Pack sandwich or burger; Then pack banana

- **Modified FSA:**
  - Pack banana; Then pack only burger

Case Study: Driving

- **Ground-truth FSA:**
  - Initial state
  - At red light
  - Reached goal

- **Unsafe Fragment of Learned FSA:**
  - "Ignore red light" (r, 0.1)

- **Safe Modified Fragment of Learned FSA:**
  - "Stop at red light" (r, 1.0)

What Makes LVIN Different?

- Interpret the high level of a hierarchical model as a FSA/logical specification
- *Interpretable*
- Incorporate the FSA into value iteration so that changes to the FSA result in changes to the policy
- *Manipulable*
- Interpretable and manipulable policies enable the crafting of safe policies