

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.

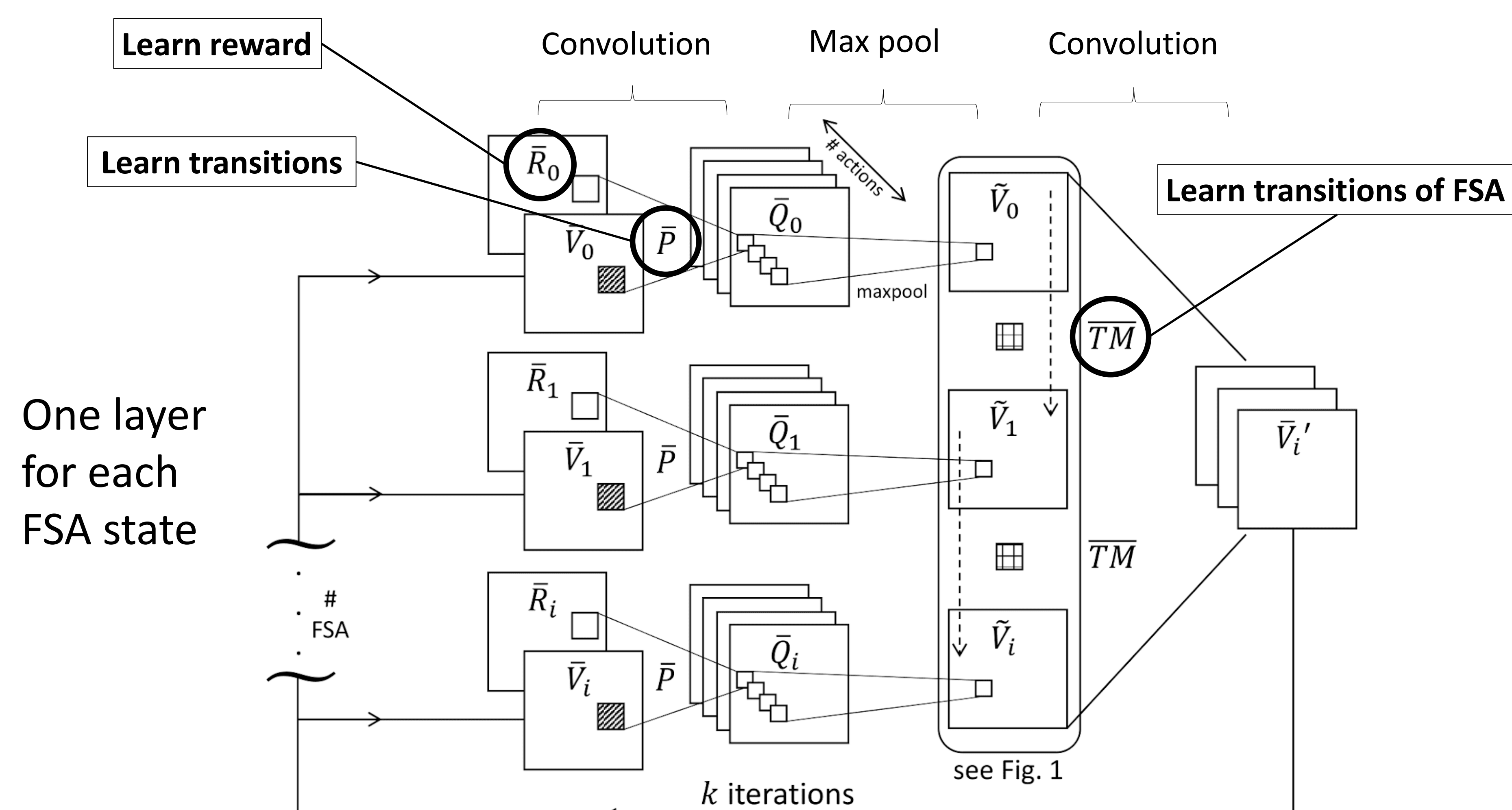
Manipulable: A human operator can easily modify the learned policy to perform similar but different policies.

Assumptions

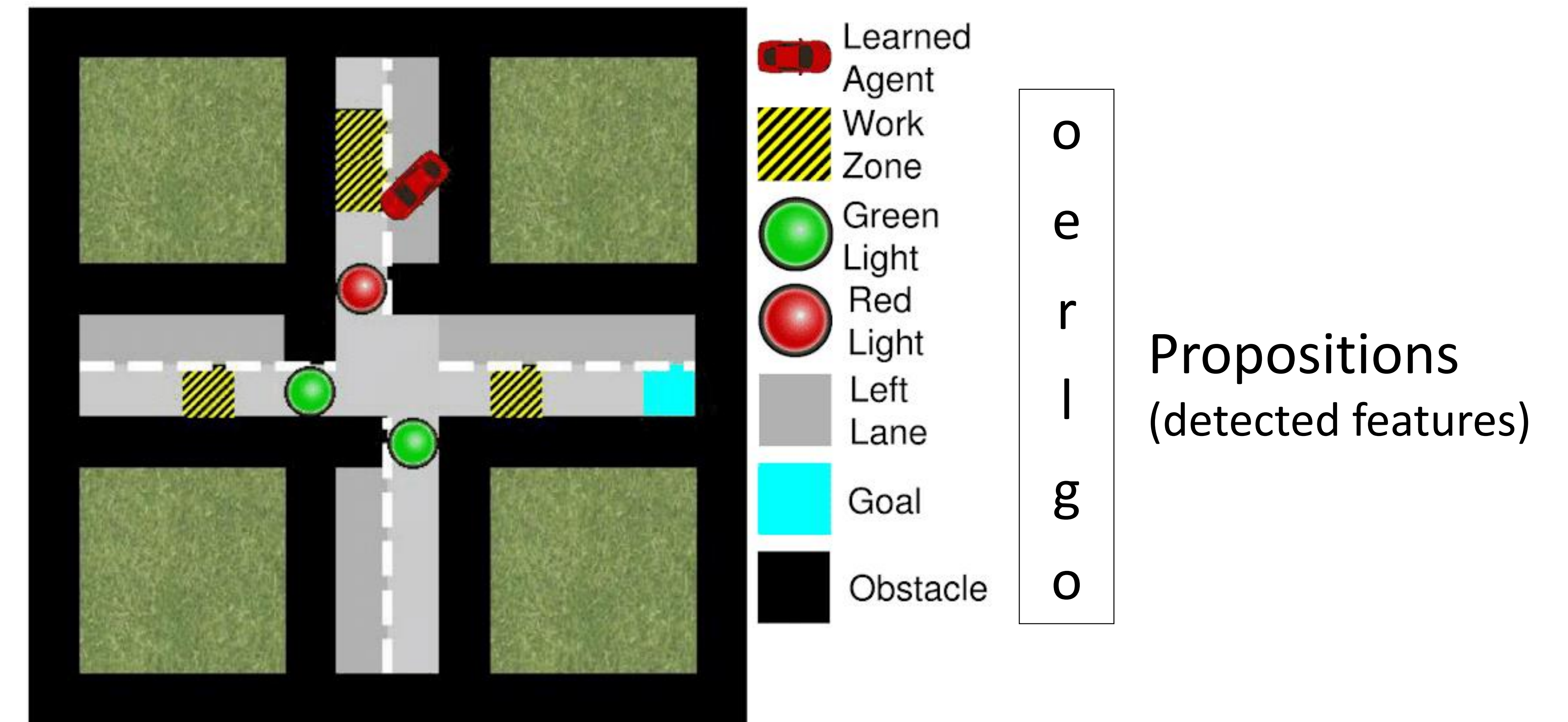
- Rules can be encoded as a Finite State Automaton (FSA)
- Relevant features of the environment can be detected
- Number of FSA states is known
- Environment outputs current FSA state

Algorithm

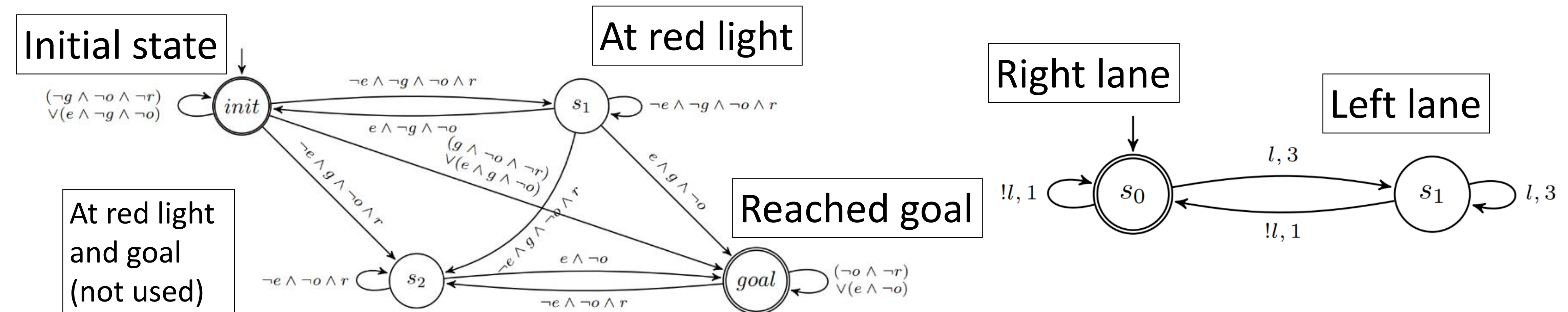
Logic-Based Value Iteration Networks



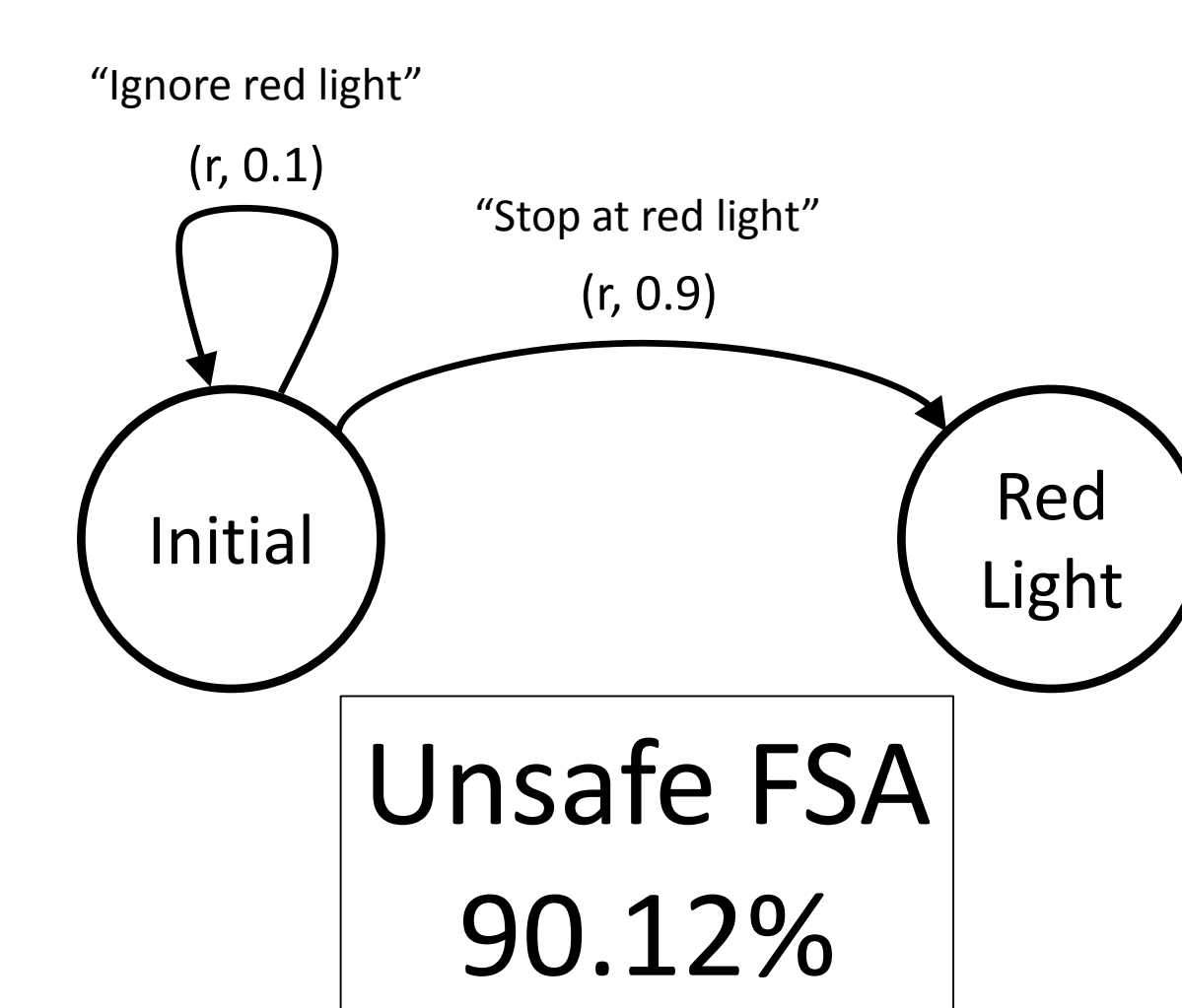
Case Study: Driving



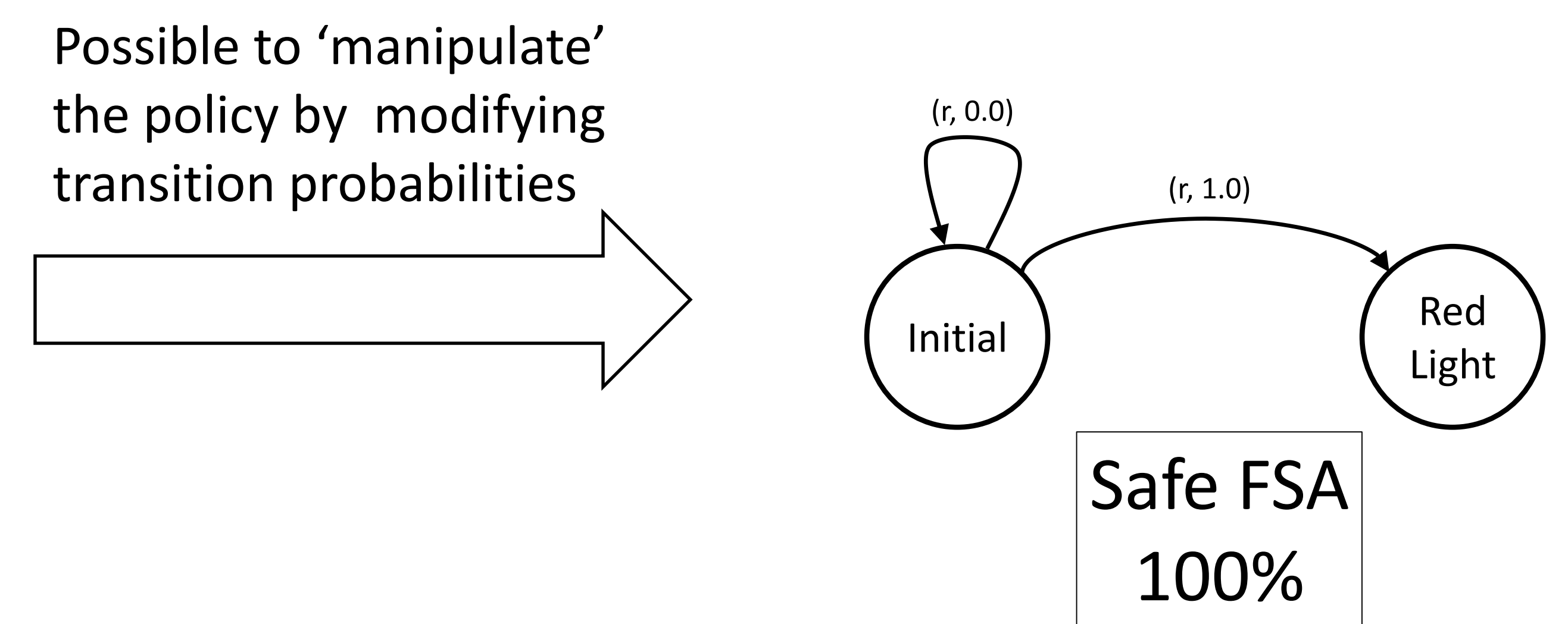
Ground-truth FSA



Unsafe Fragment of Learned FSA



Safe Modified Fragment of Learned FSA



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