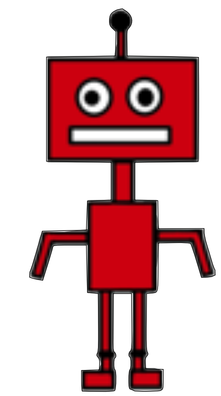
Toward Affordance-Aware Planning



David Abel & Gabriel Barth-Maron
James MacGlashan, Stefanie Tellex

Dept. of Computer Science, Brown University

Outline

Problem Statement

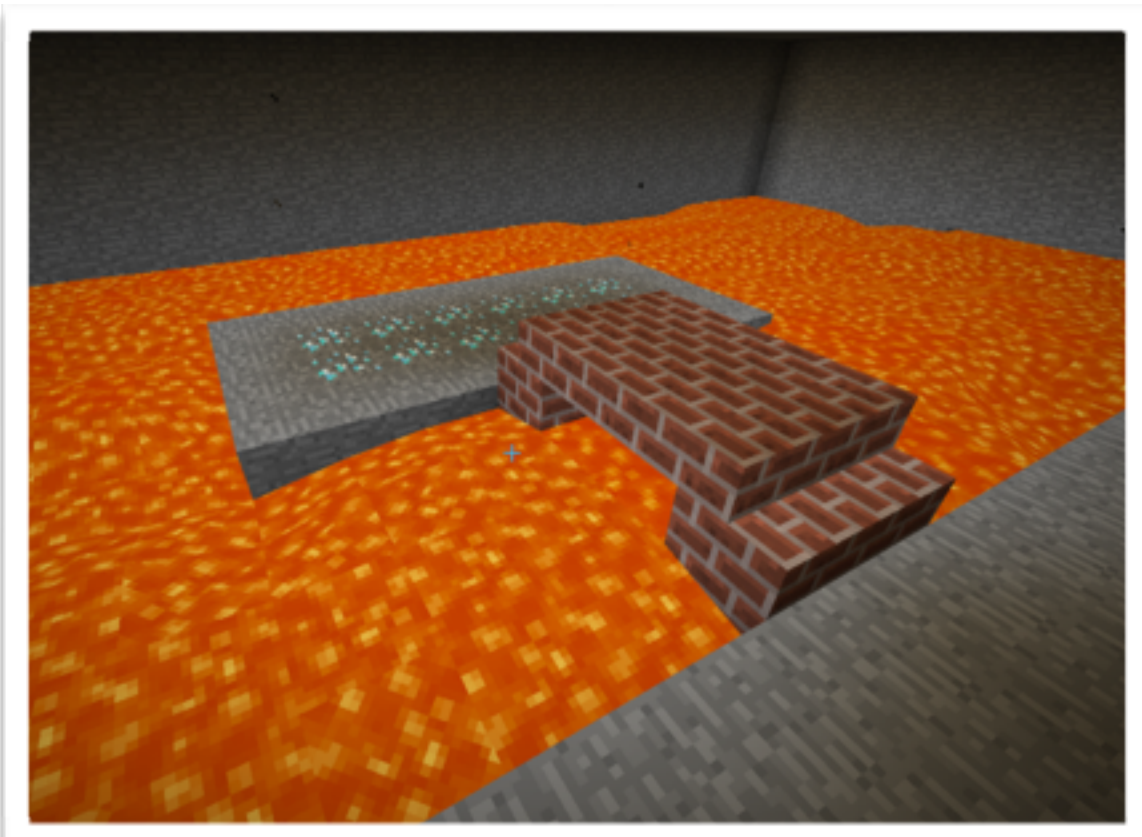
Enable Autonomous Agents to learn to plan effectively in massive stochastic state-spaces.

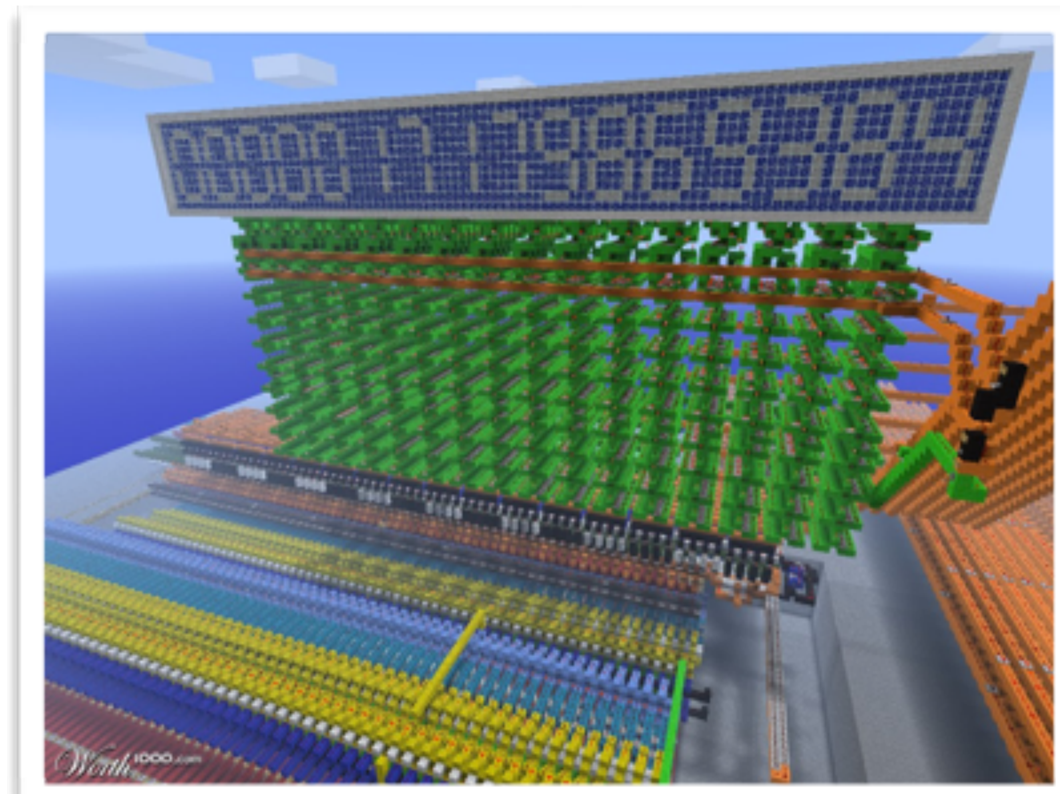
Minecraft

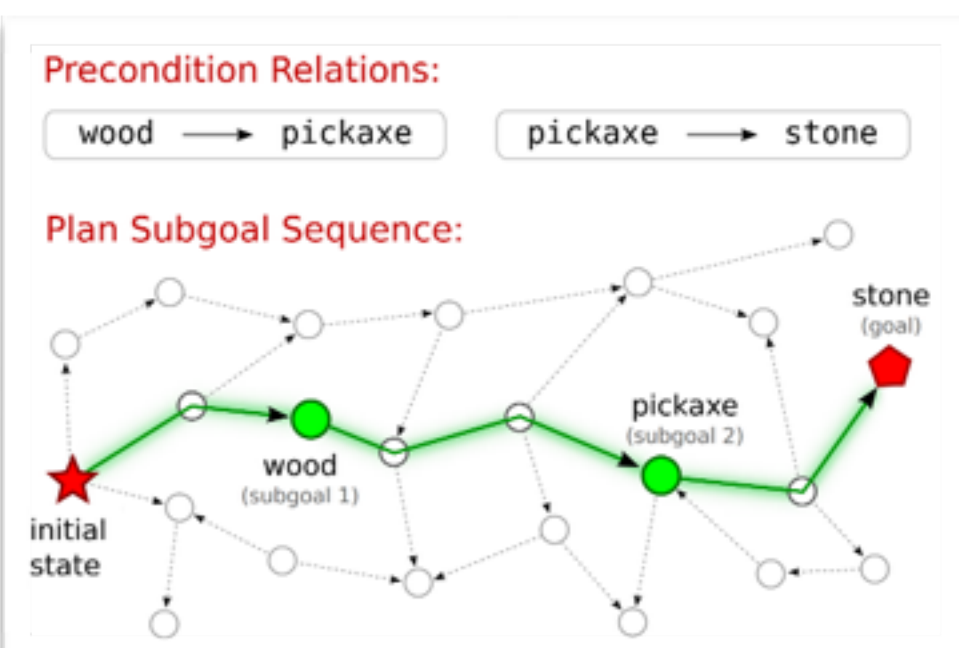
Insert video (Have it already, didn't push b/c file size)

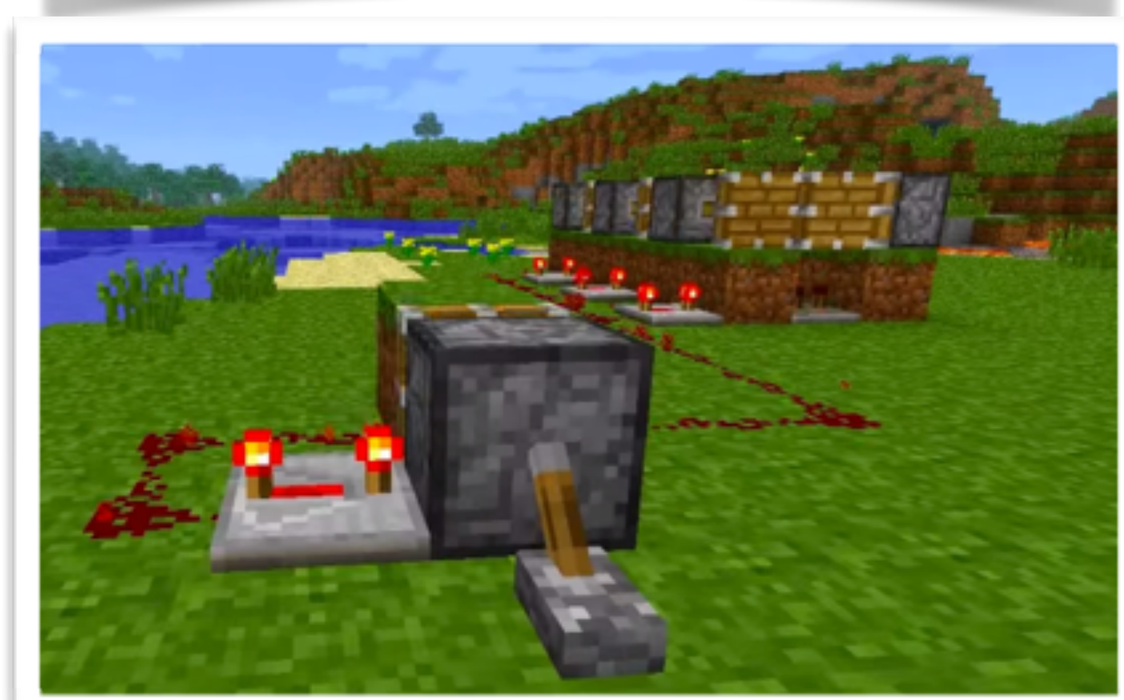
≈Turing Complete LEGO

Minecraft & Robotics

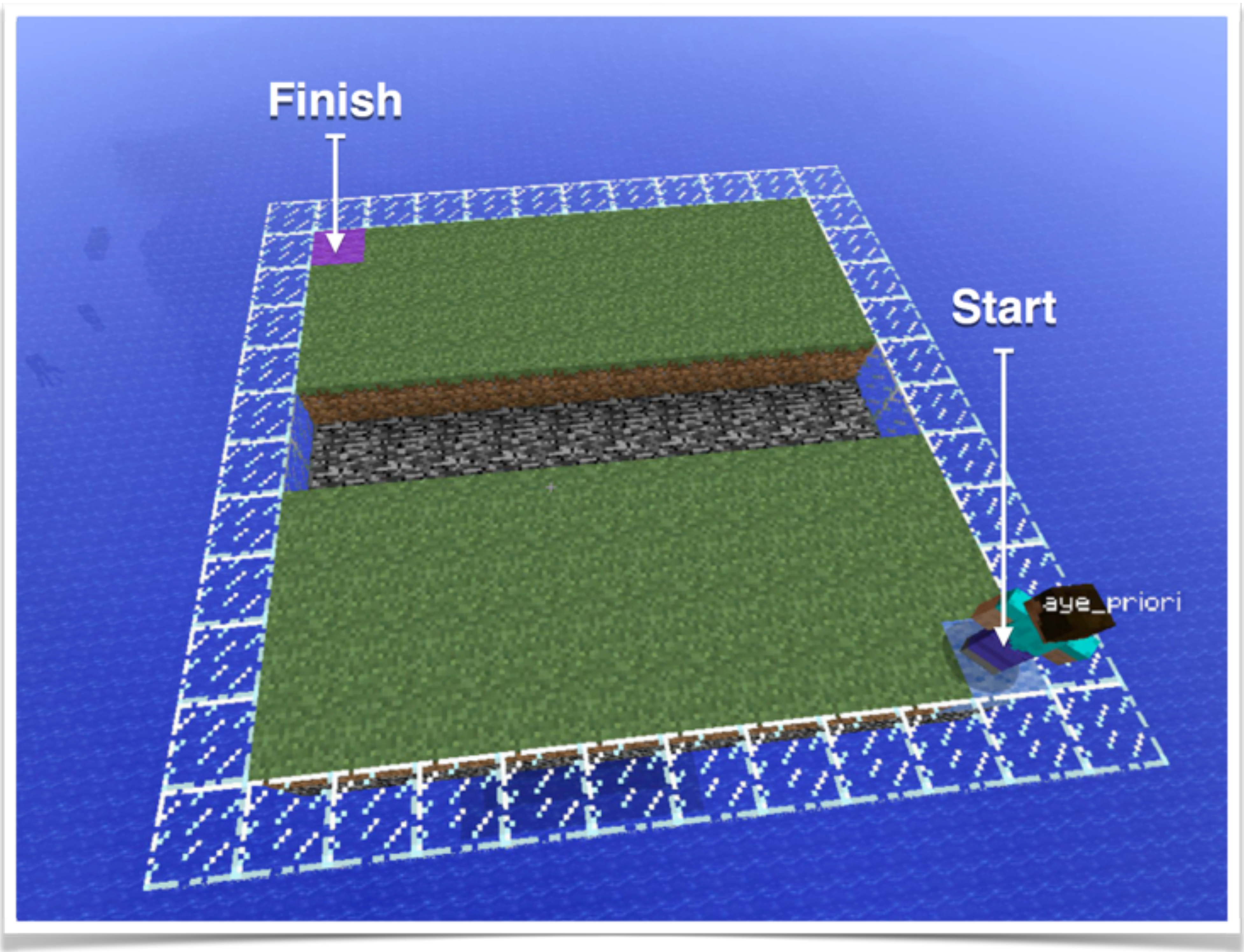








Minecraft: The Problem



ACTIONS

- Move
- Place
- Destroy
- Use
- Jump
- Rotate
- Look
- Craft
- …

Affordance Formalism

We formalise an affordance as:

$$\Delta = \langle p, g \rangle \longmapsto \mathcal{A}'$$

Where:

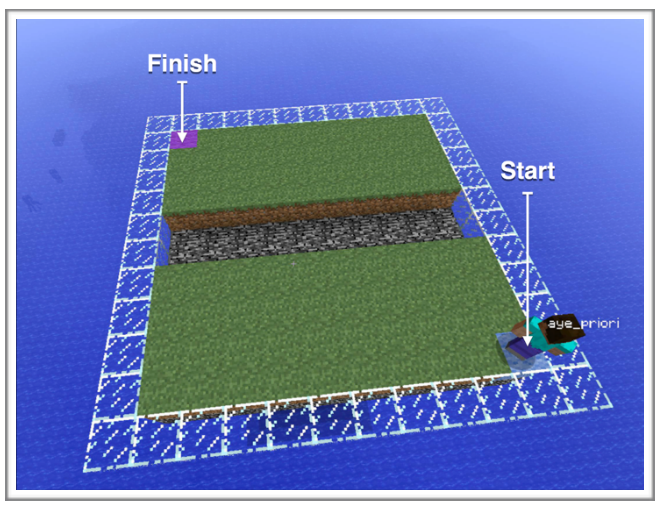
$p =$ a predicate on states

$g =$ a lifted goal description

$\mathcal{A}' =$ a set of relevant actions

Affordance Formalism Intuition

Idea: Affordances focus the agent on *relevant action possibilities* by pruning irrelevant actions on a state by state basis

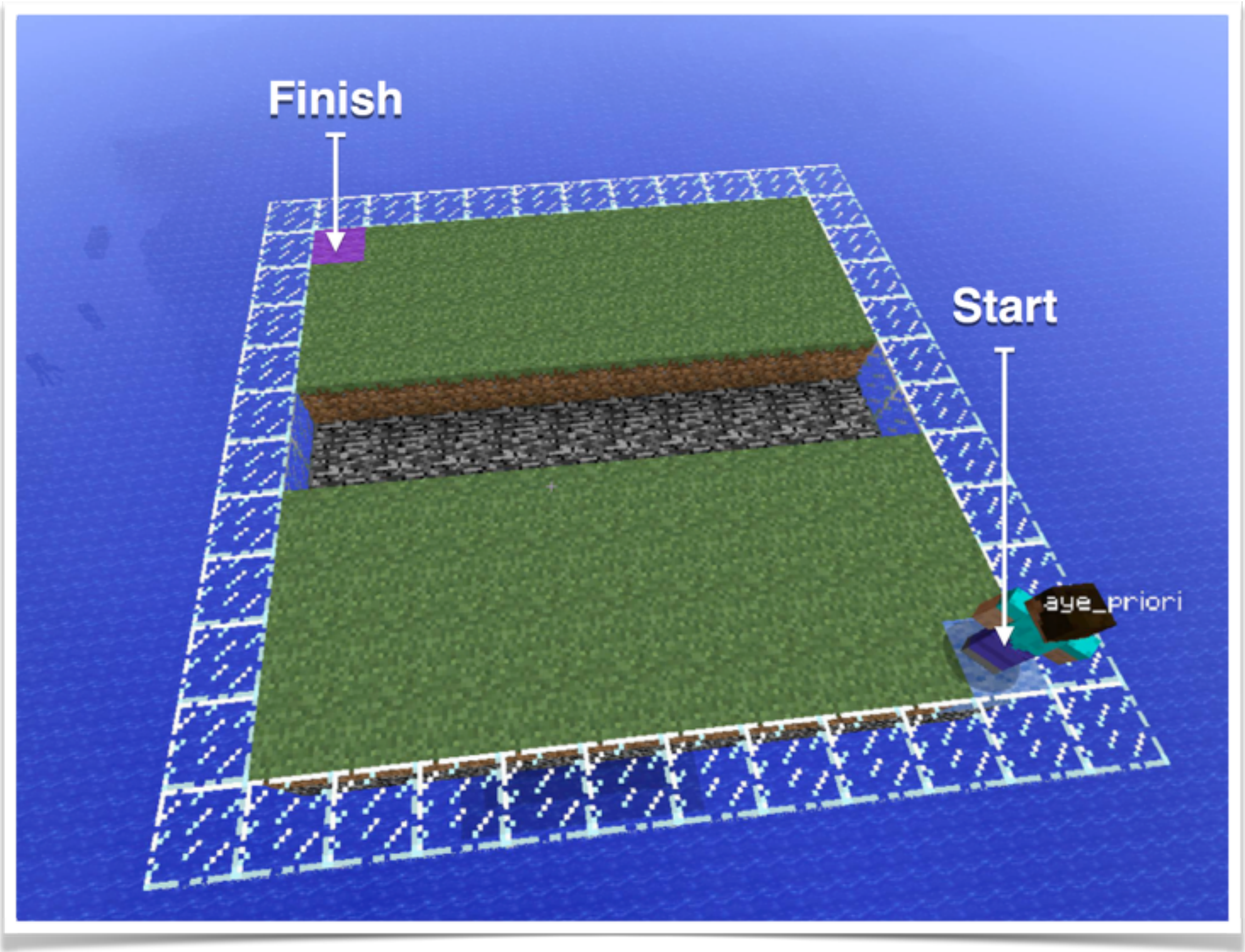


(OO)-MDP $\longrightarrow$ Policy

$$\Delta_1 = \langle nearTrench, AtGoal \rangle \longmapsto \{place, jump\}$$

$$\Delta_2 = \langle nearPlane, AtGoal \rangle \longmapsto \{move, rotate\}$$

Affordances In Planning



Demo

Insert video (Have it already, didn't push b/c file size)

Learning Framework

Goal: For each reachable state in the MDP, *s*, infer:

$$\Pr(\mathcal{A}^* \mid s, \Delta_1 \ldots \Delta_K)$$

≈ Probability that an action set contains the optimal action for that state

Where:

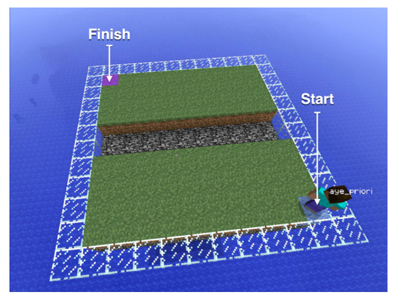
$$\mathcal{A}^* = \bigcup_{i=1}^{K} \mathcal{A}'_i$$

Full Learning Process

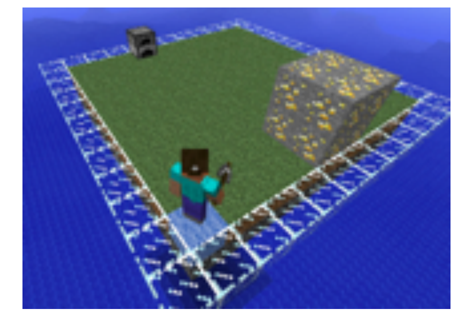
AtGoal

MakeGold

GENERATE SMALL WORLDS:



. . . . . .



SOLVE EACH MDP

π_1 . . . . . . π_N

COUNT:

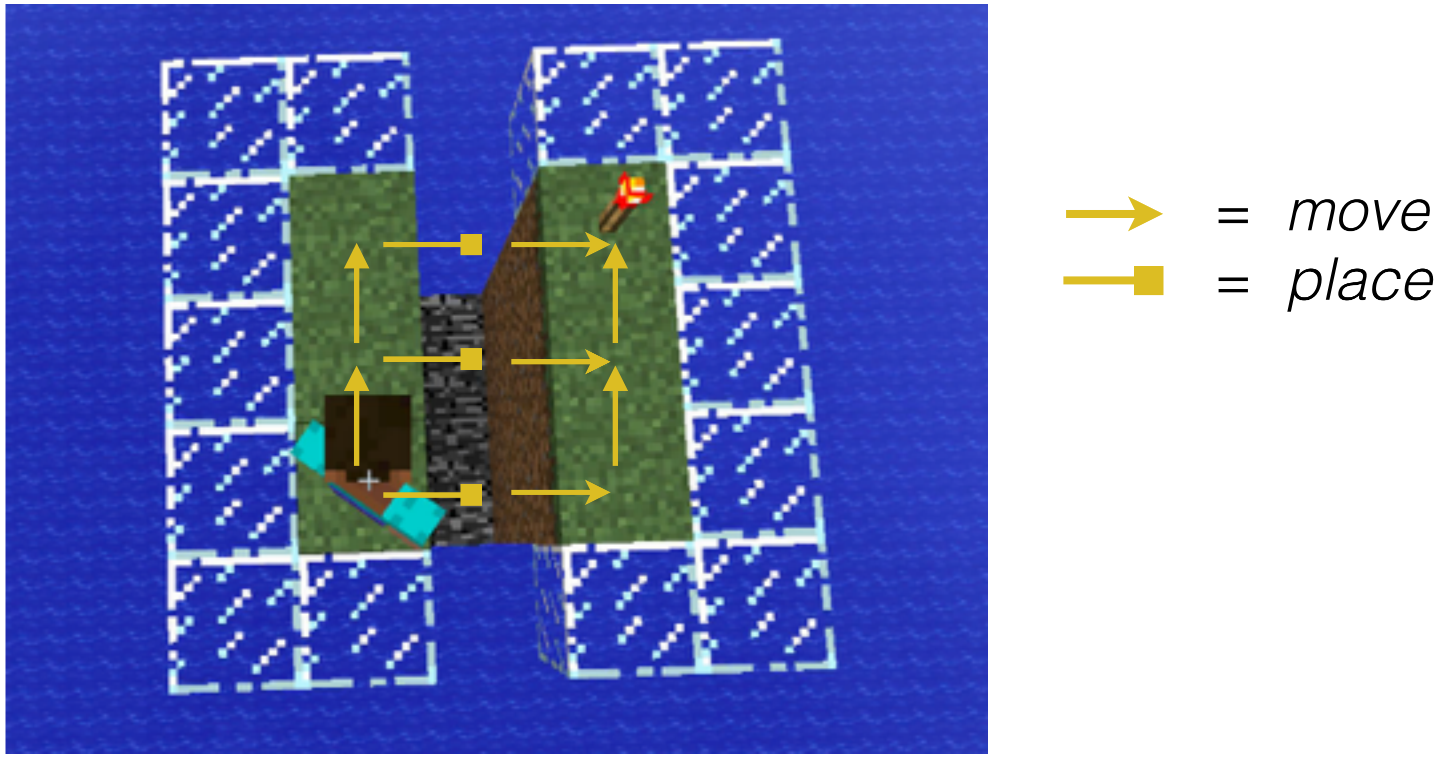
$$\Delta_i.updateCounts(\pi_1, \ldots, \pi_K)$$

Counts

$$\Delta_i.\alpha = \text{ number of worlds in which each action was used}$$

$$\Delta_i.\beta = \text{ number of unique actions used in each world}$$

$\Delta_i.updateCounts(\pi_1, \ldots, \pi_K)$



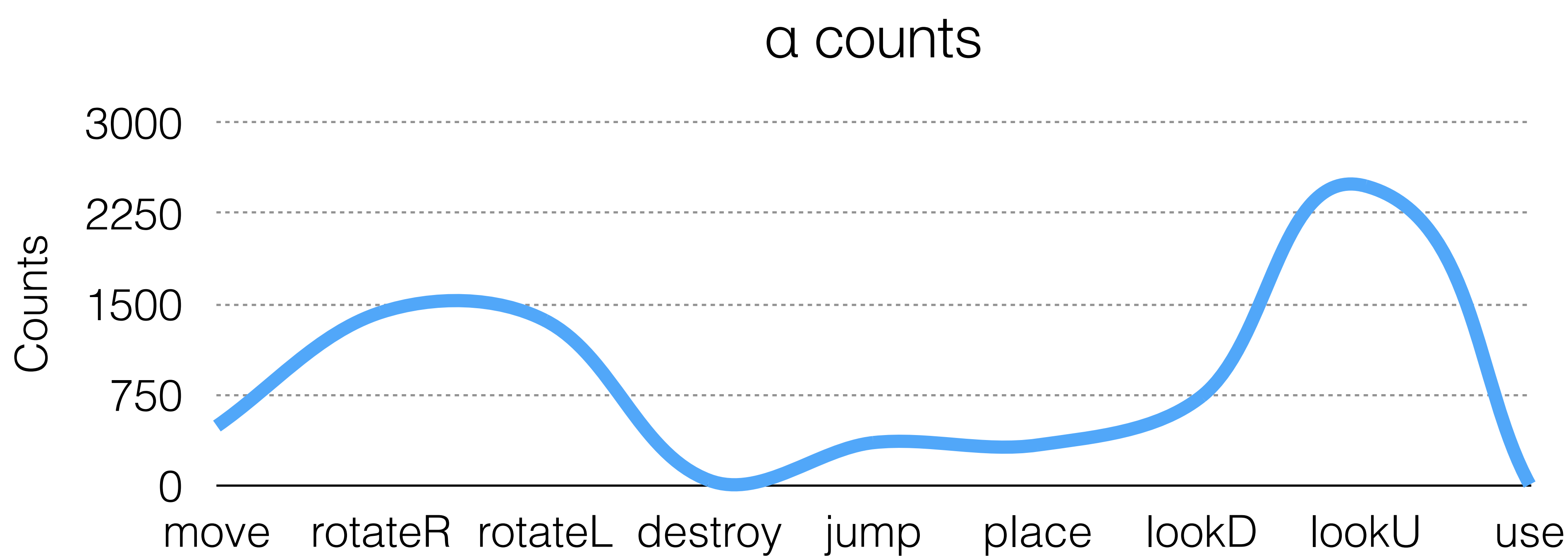
$$\Delta_1 = \langle \checkmark\ nearTrench, \checkmark\ atGoal \rangle$$

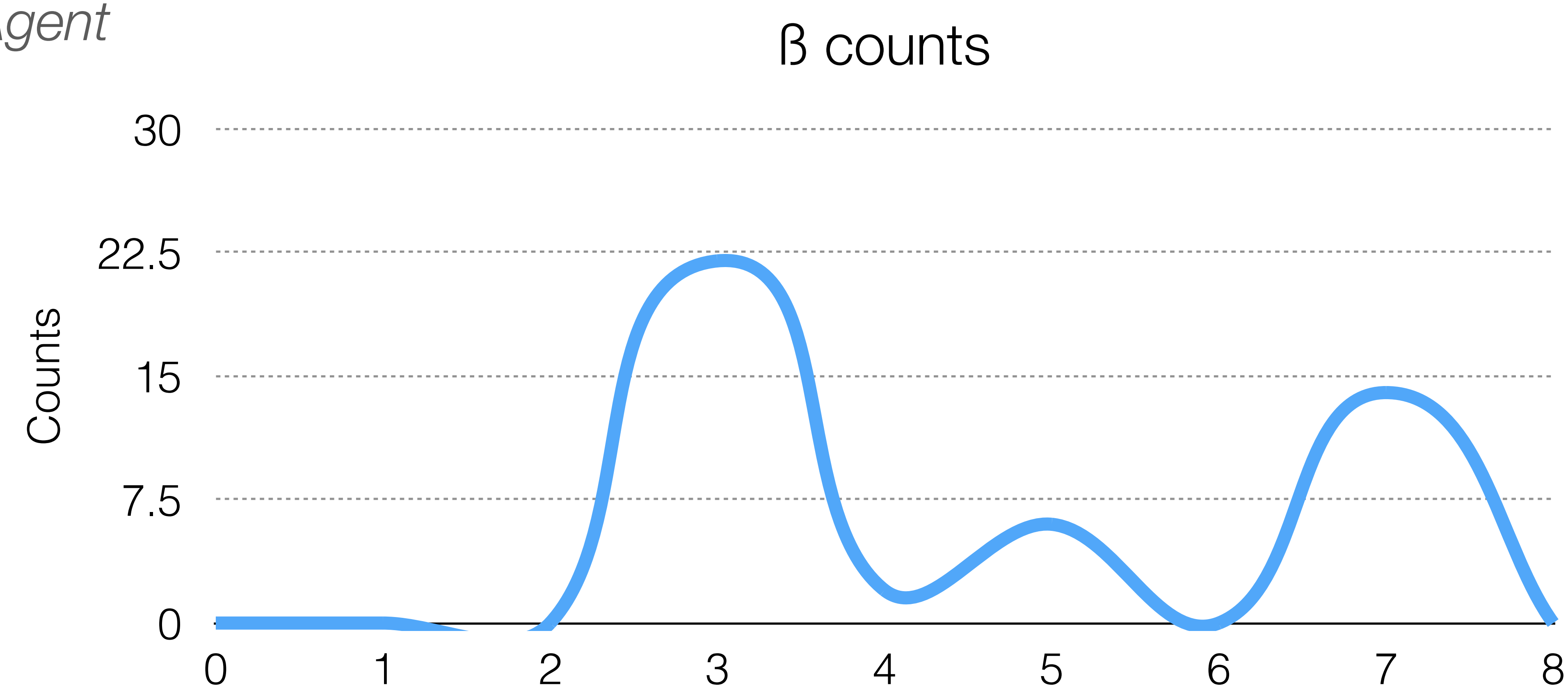
$\Delta_1.\alpha.moveRight$++, $\Delta_1.\alpha.moveForward$++, $\Delta_1.\alpha.placeRight$++

$\Delta_1.\beta.3$++

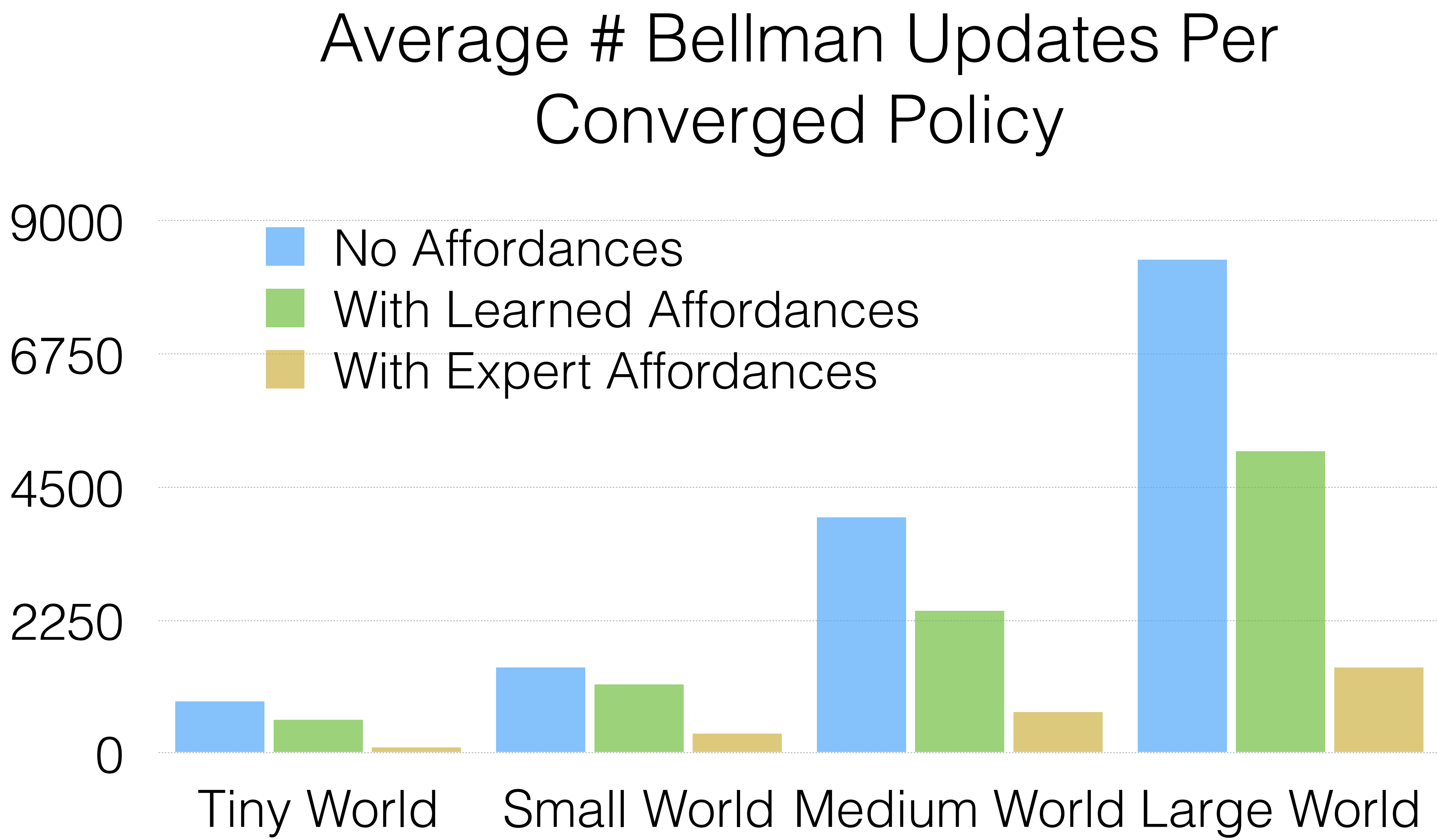
Learning Example

p = *trenchFrontOfAgent*
g = *AtGoal*





Learning Results



Current Work

- Extending predicates to logical expressions
- Clustering problem types
- Incorporating perception
- Pragmatic natural language extensions (e.g. “bridge”)
- Subgoal Planning
- Deploy on robots, other domains (cooking, javascript, Atari)

In Summary

- Proposed **Affordance-Aware Planning**
- Proposed complete learning framework for Affordances
- Demonstrated speedups in planning in large stochastic state spaces

References