NEURAL INFORMATION PROCESSING SYSTEMS

Generalizable Imitation Learning from Observation via Inferring Goal Proximity

Youngwoon Lee*, Andrew Szot*, Shao-Hua Sun, Joseph J. Lim




Assemble table

https://www.youtube.com/watch?v=_valw8rmZIM


Assemble table
Make progress towards goal
https://www.youtube.com/watch?v=_valw8rmZIM

https://www.youtube.com/watch?v=4620X342yGo

Assemble new table

Make progress to an assembled table

https://www.youtube.com/watch?v=4620X342yGo

Simple but **generalizable** imitation of demonstrations:

(1) understand underlying task structure, **task progress**

(2) learn to make progress

https://www.youtube.com/watch?v=4620X342yGo

Goal Proximity — Task Progress

"How close a state is to the goal?"

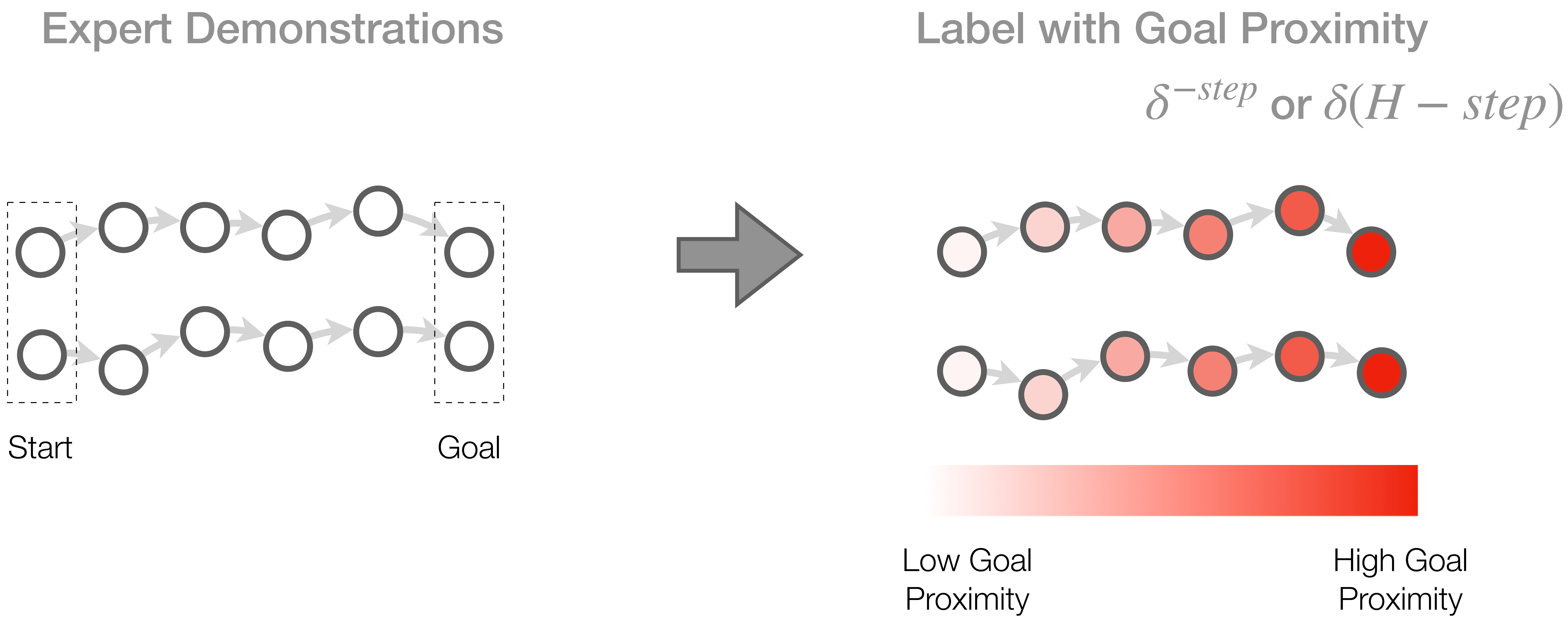
Number of actions required to complete the task

4 steps → 3 steps → 2 steps → 1 step → 0 step

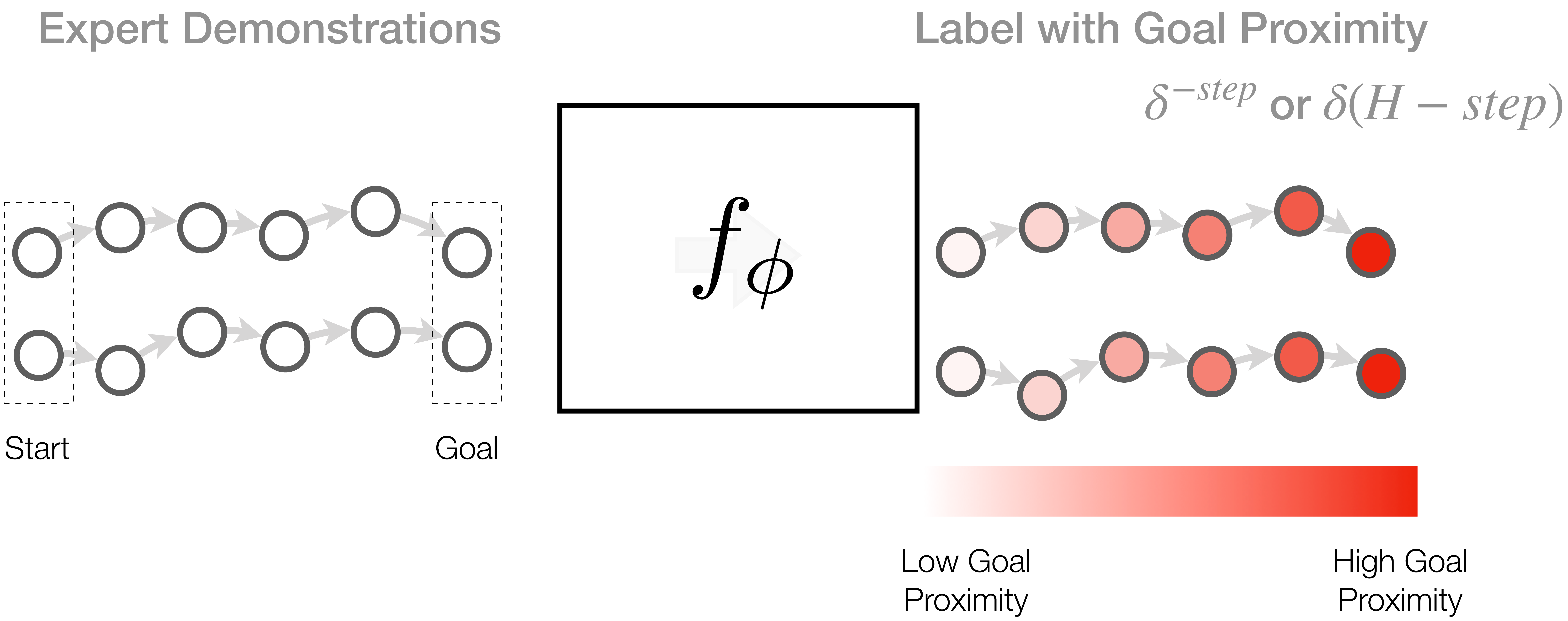
4 steps: Low goal proximity

0 step: High goal proximity

Learning Goal Proximity Function

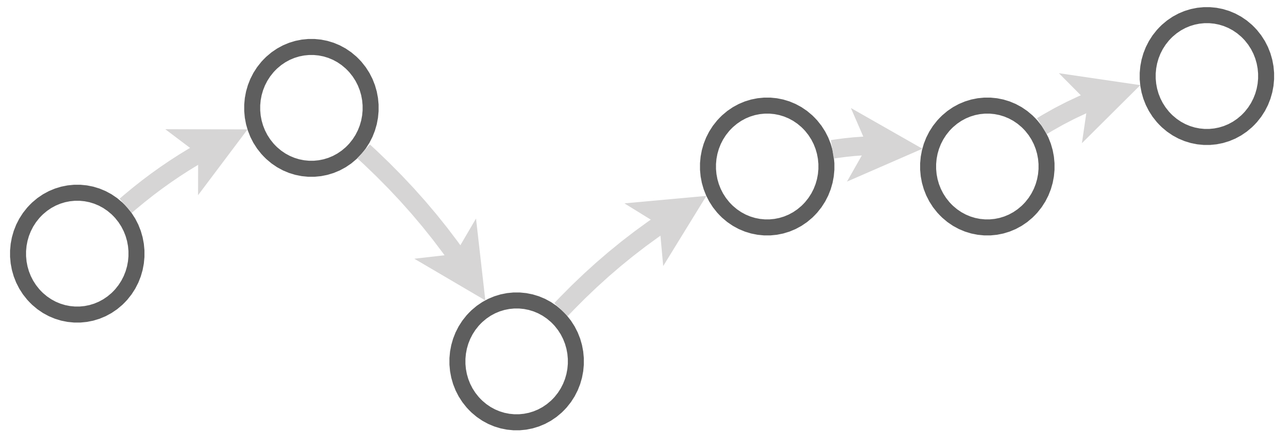


Learning Goal Proximity Function



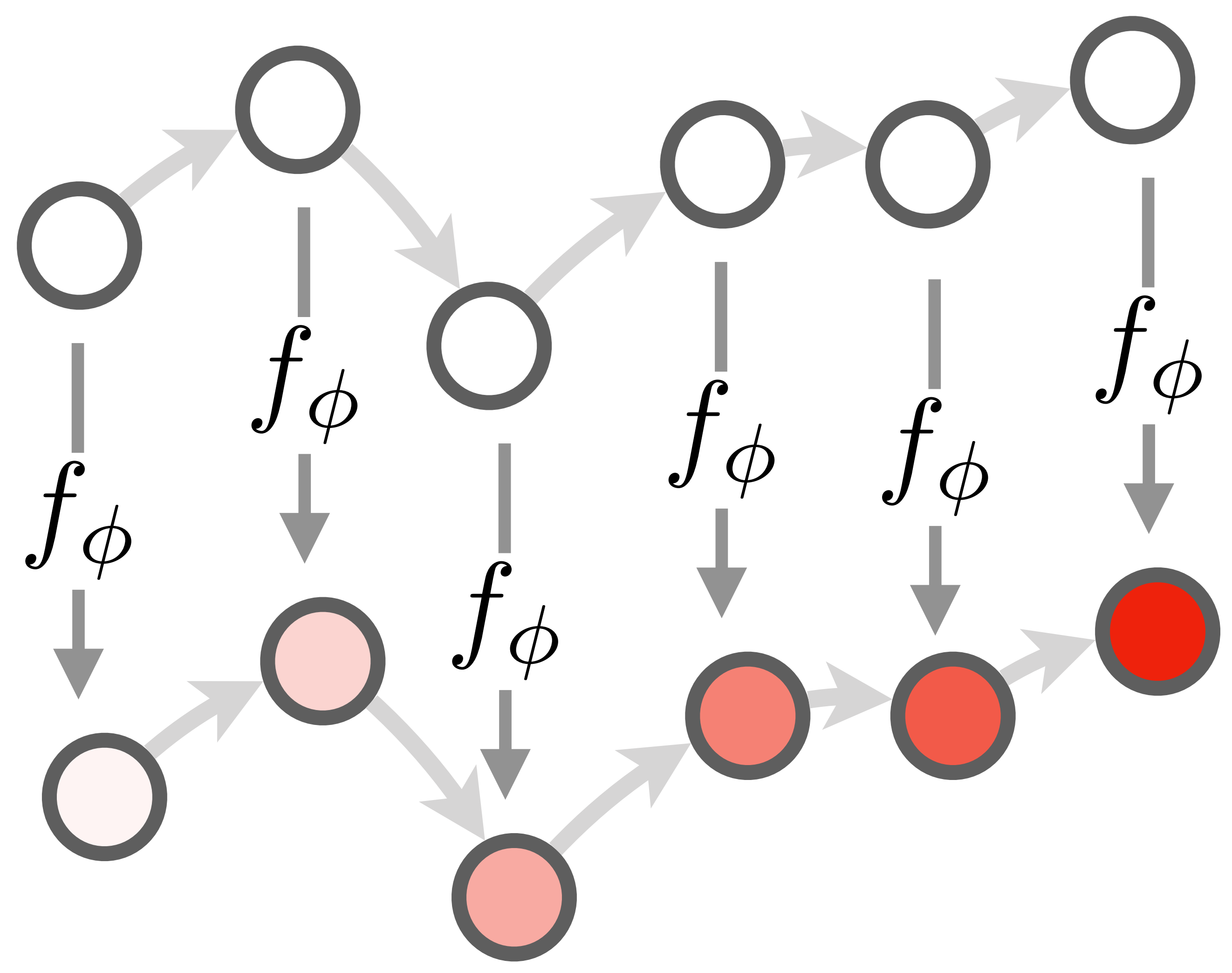
Learning Policy

Unseen Trajectories



Learning Policy

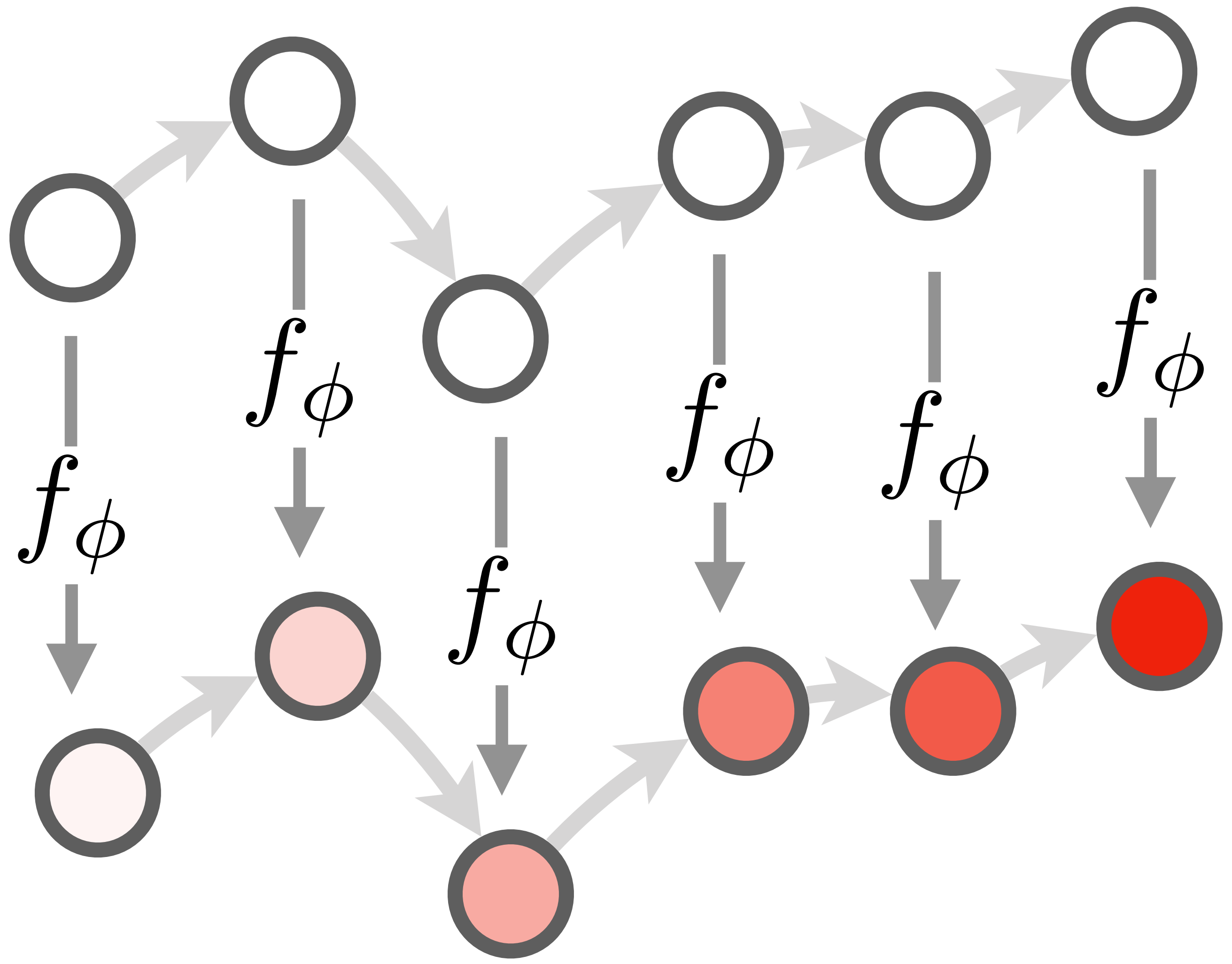
Unseen Trajectories



Label with Proximity Function

Learning Policy

Unseen Trajectories



Label with Proximity Function

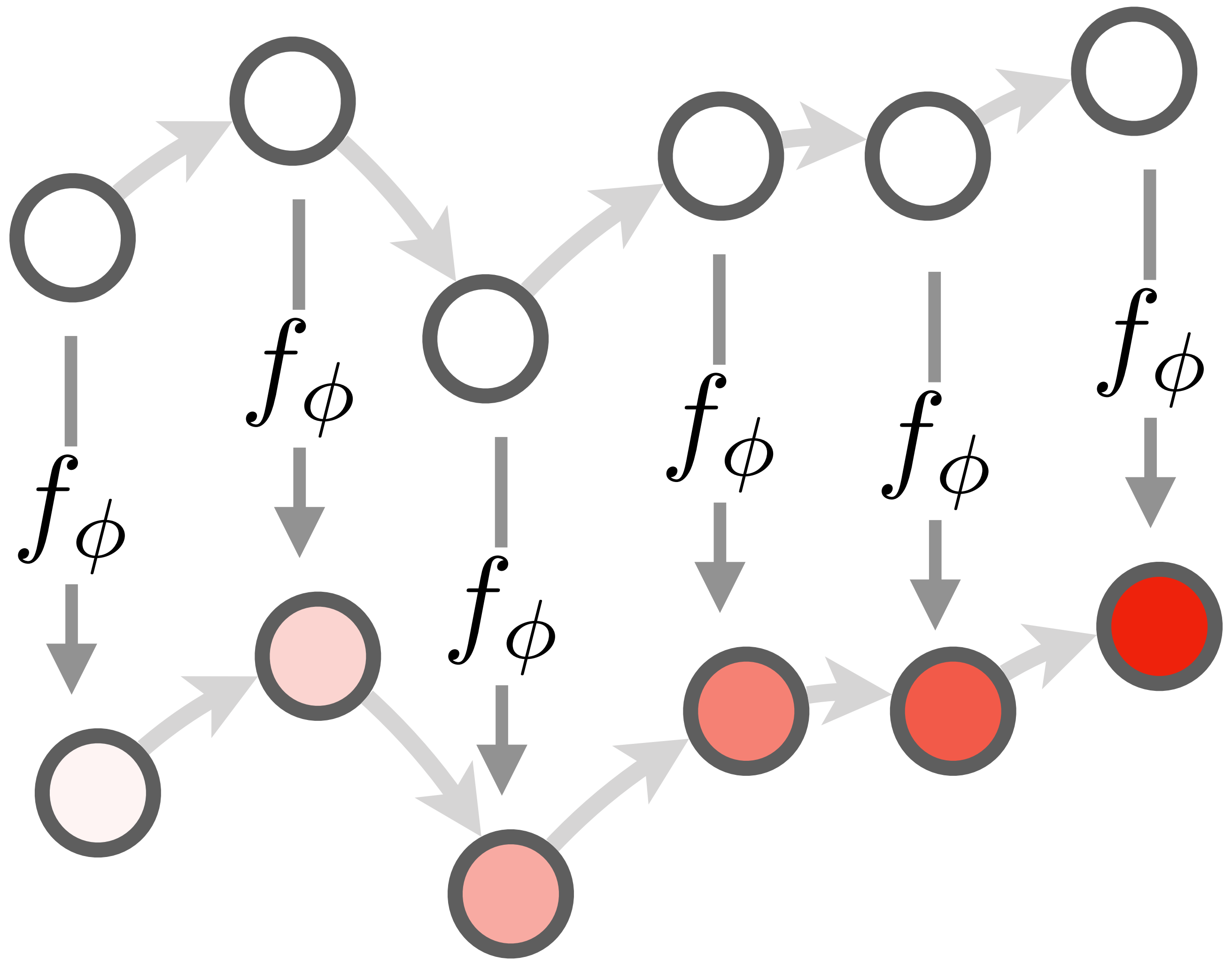
Train Agent

Proximity Reward

$$f_\phi(s_{t+1}) - f_\phi(s_t)$$

Learning Policy

Unseen Trajectories



Label with Proximity Function

Train Agent

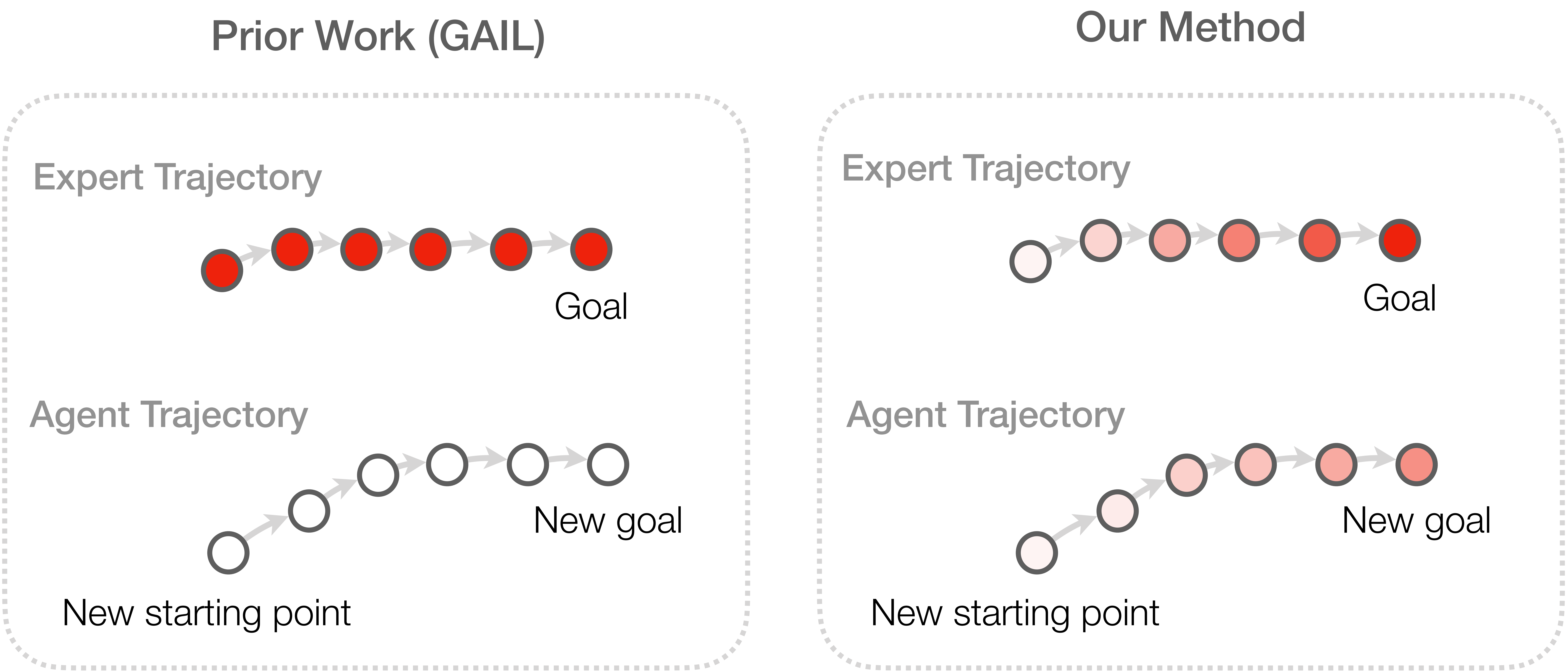
Proximity Reward

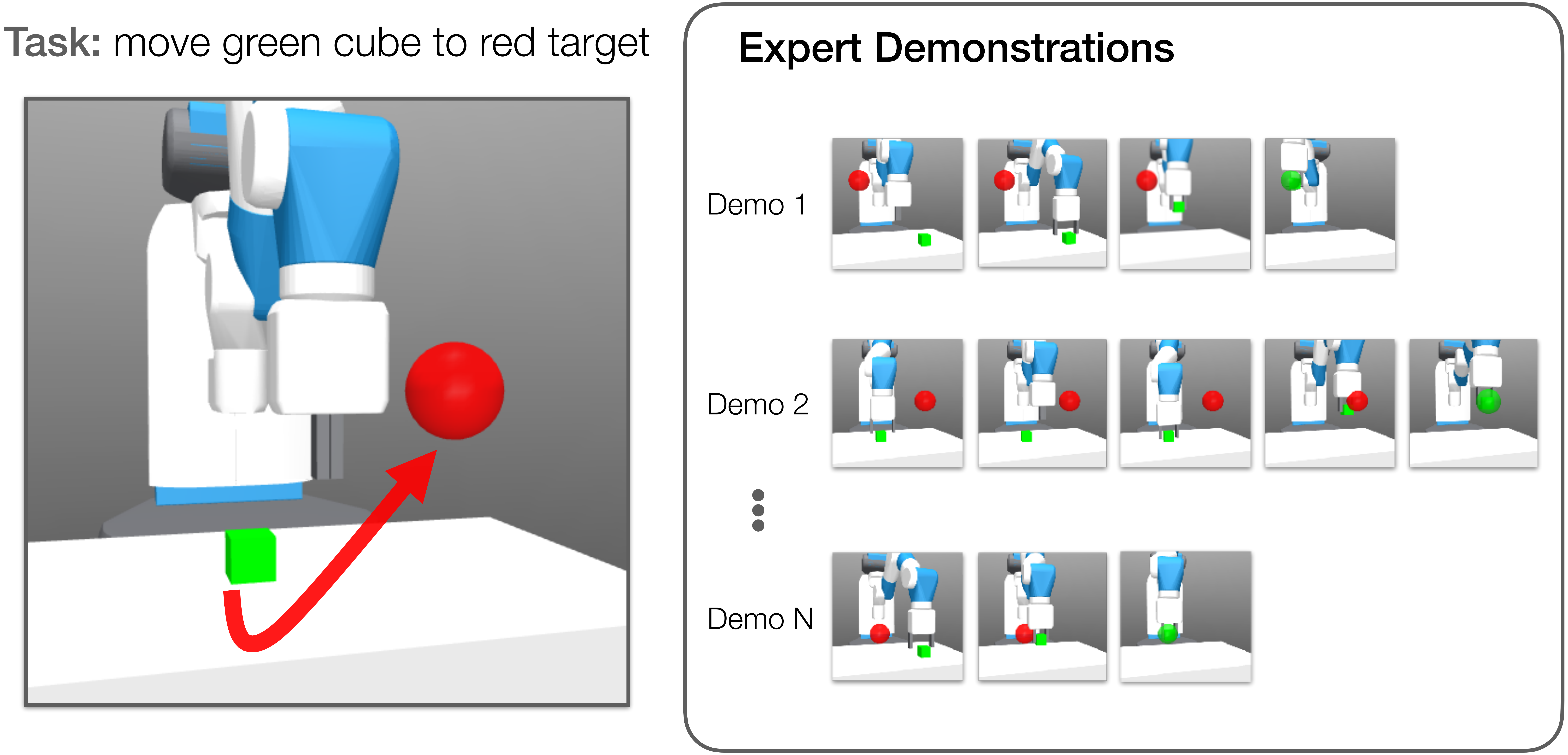
$$f_\phi(s_{t+1}) - f_\phi(s_t)$$

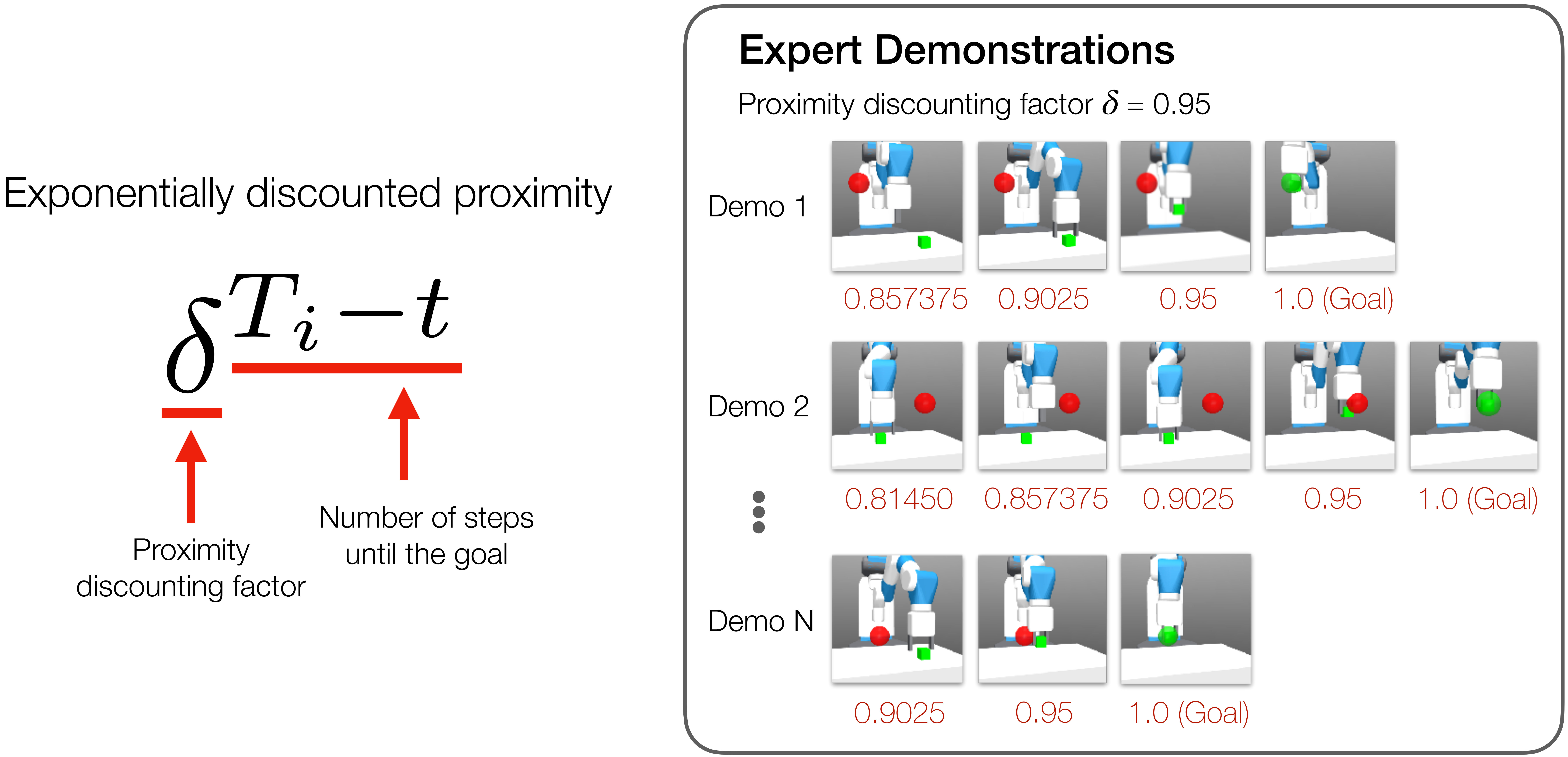
+ Move closer to the goal

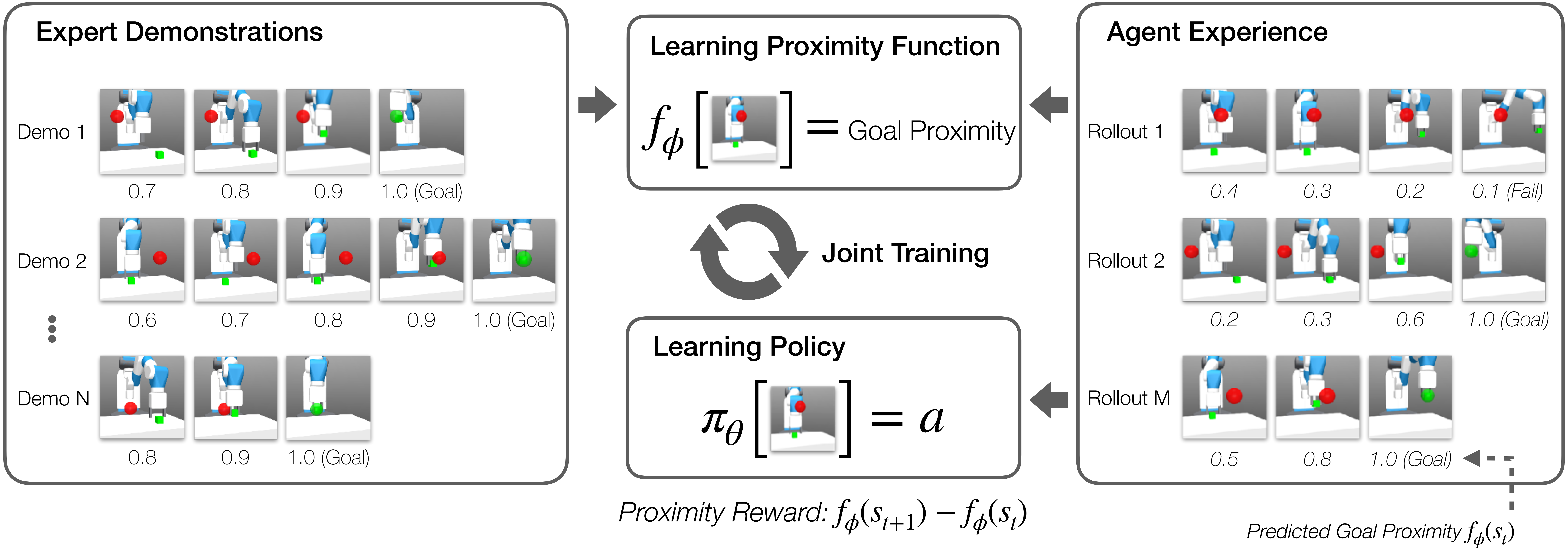
− Move away from the goal

Related Work

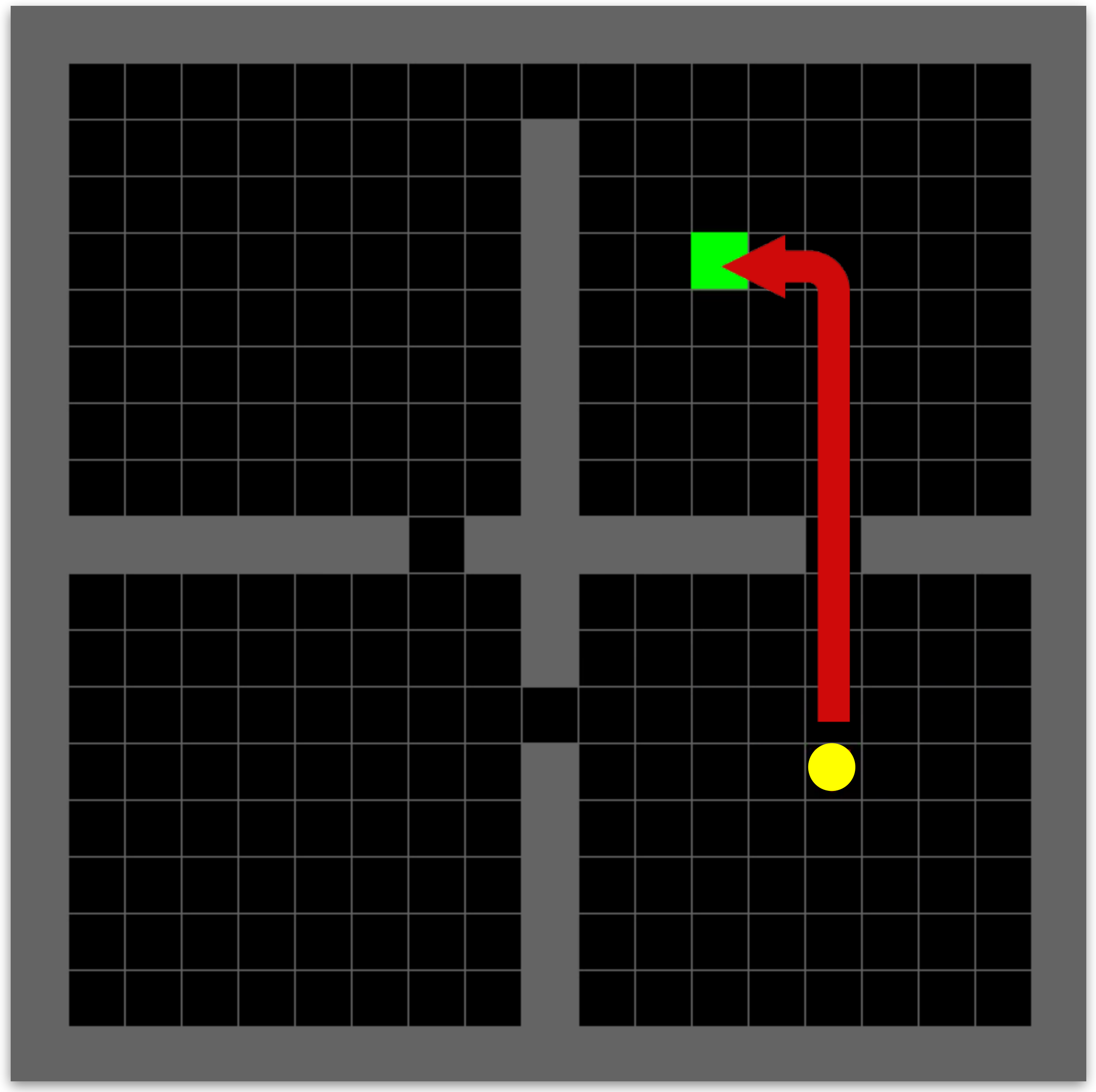


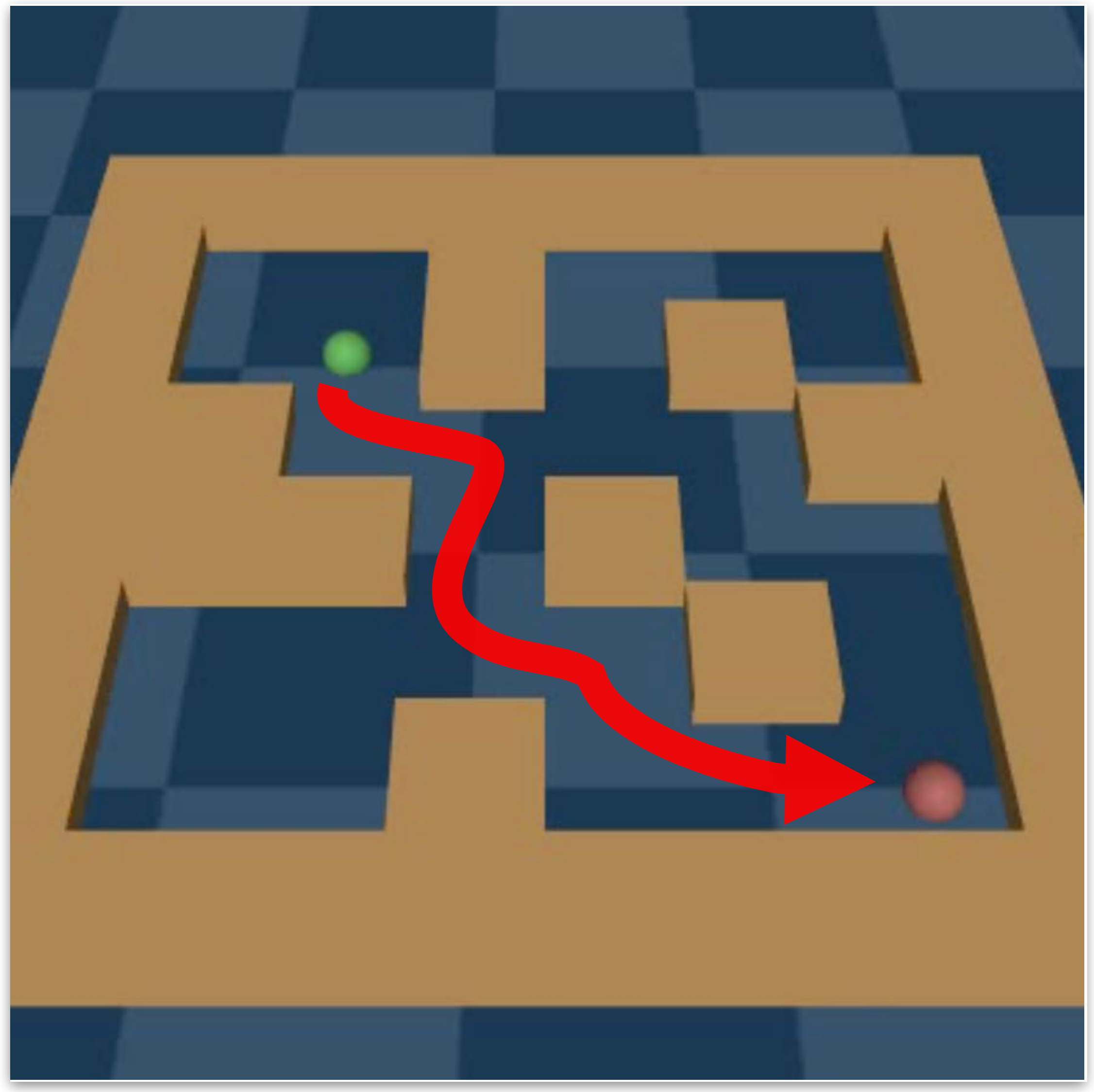

Task: move green cube to red target
Expert Demonstrations
Demo 1
Demo 2
Demo N

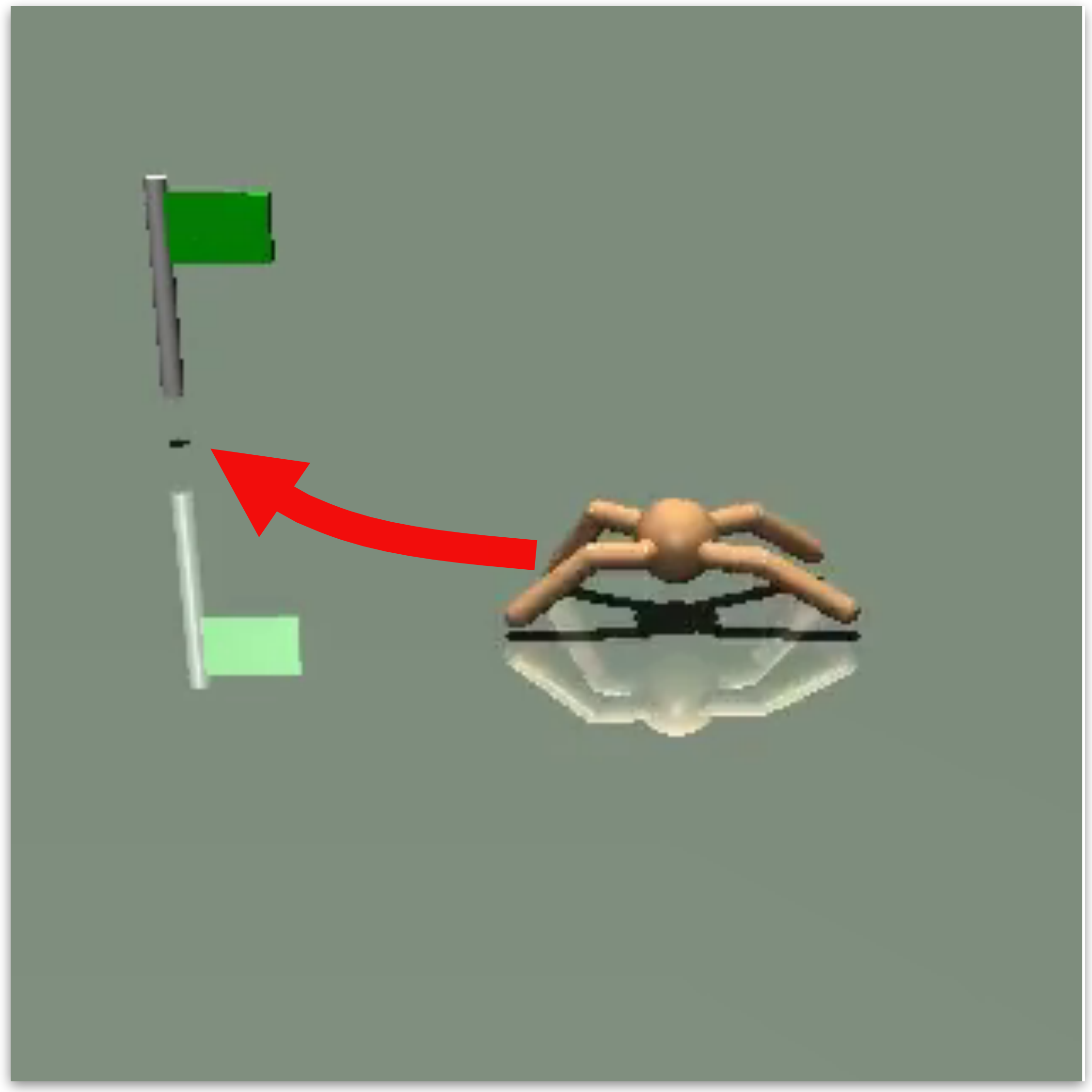

Exponentially discounted proximity
$\delta^{T_i - t}$
Proximity discounting factor
Number of steps until the goal
Expert Demonstrations
Proximity discounting factor δ = 0.95
Demo 1
0.857375
0.9025
0.95
1.0 (Goal)
Demo 2
0.81450
0.857375
0.9025
0.95
1.0 (Goal)
Demo N
0.9025
0.95
1.0 (Goal)

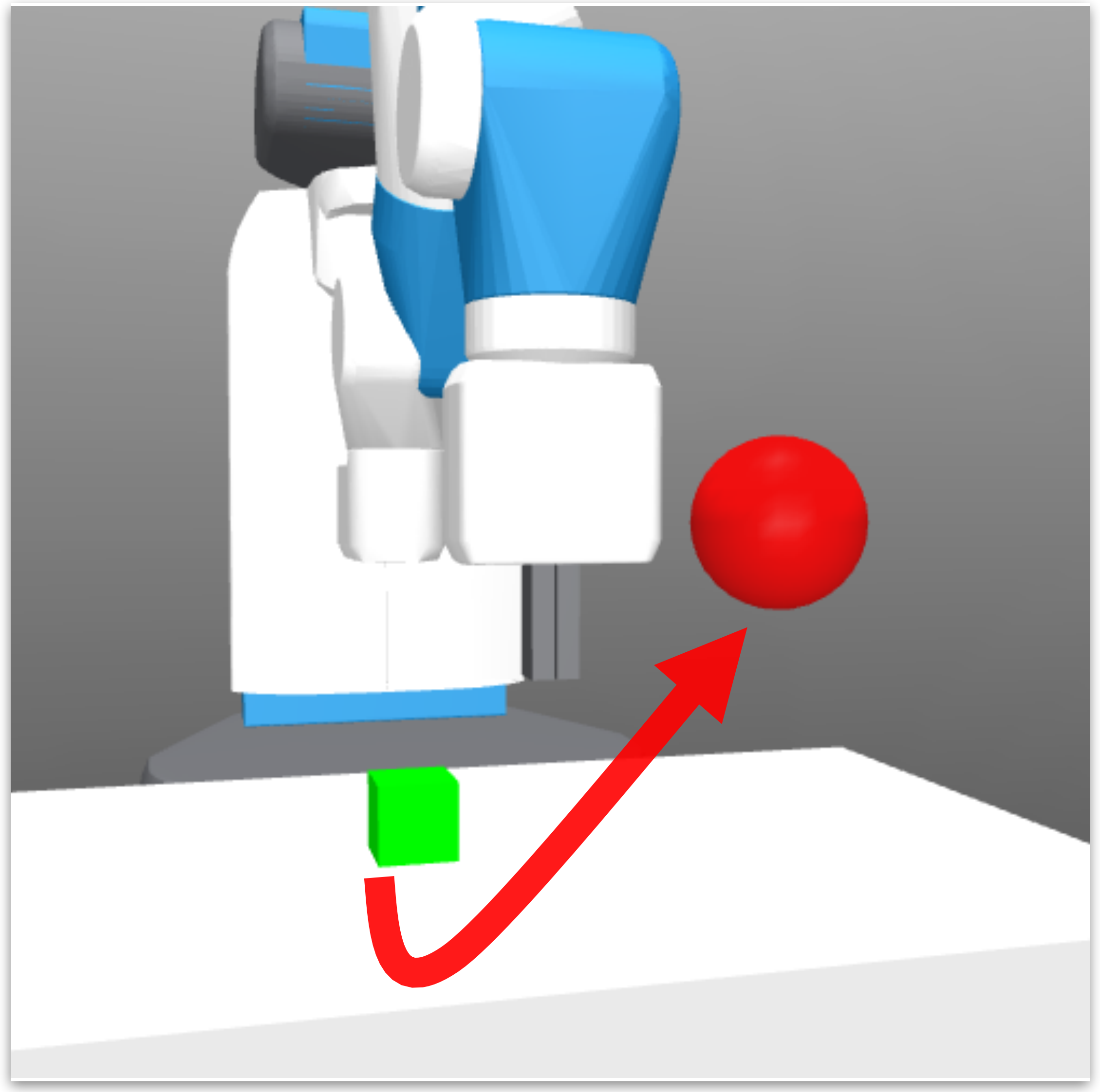


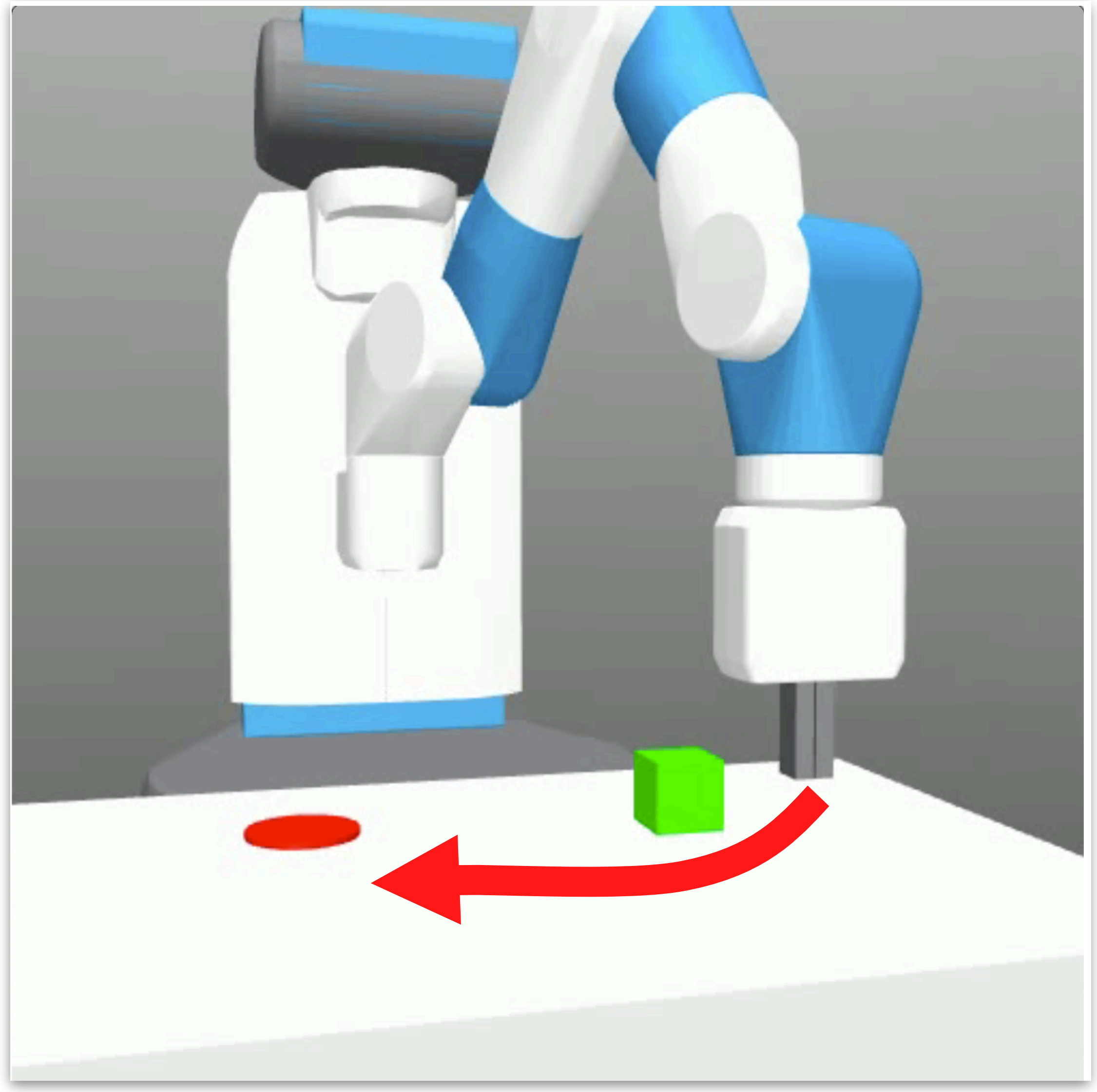
Expert Demonstrations
Demo 1
0.7
0.8
0.9
1.0 (Goal)
Demo 2
0.6
0.7
0.8
0.9
1.0 (Goal)
Demo N
0.8
0.9
1.0 (Goal)
Learning Proximity Function
f_ϕ [ ] = Goal Proximity
Joint Training
Learning Policy
π_θ [ ] = a
Proximity Reward: $f_\phi(s_{t+1}) - f_\phi(s_t)$
Agent Experience
Rollout 1
0.4
0.3
0.2
0.1 (Fail)
Rollout 2
0.2
0.3
0.6
1.0 (Goal)
Rollout M
0.5
0.8
1.0 (Goal)
Predicted Goal Proximity $f_\phi(s_t)$

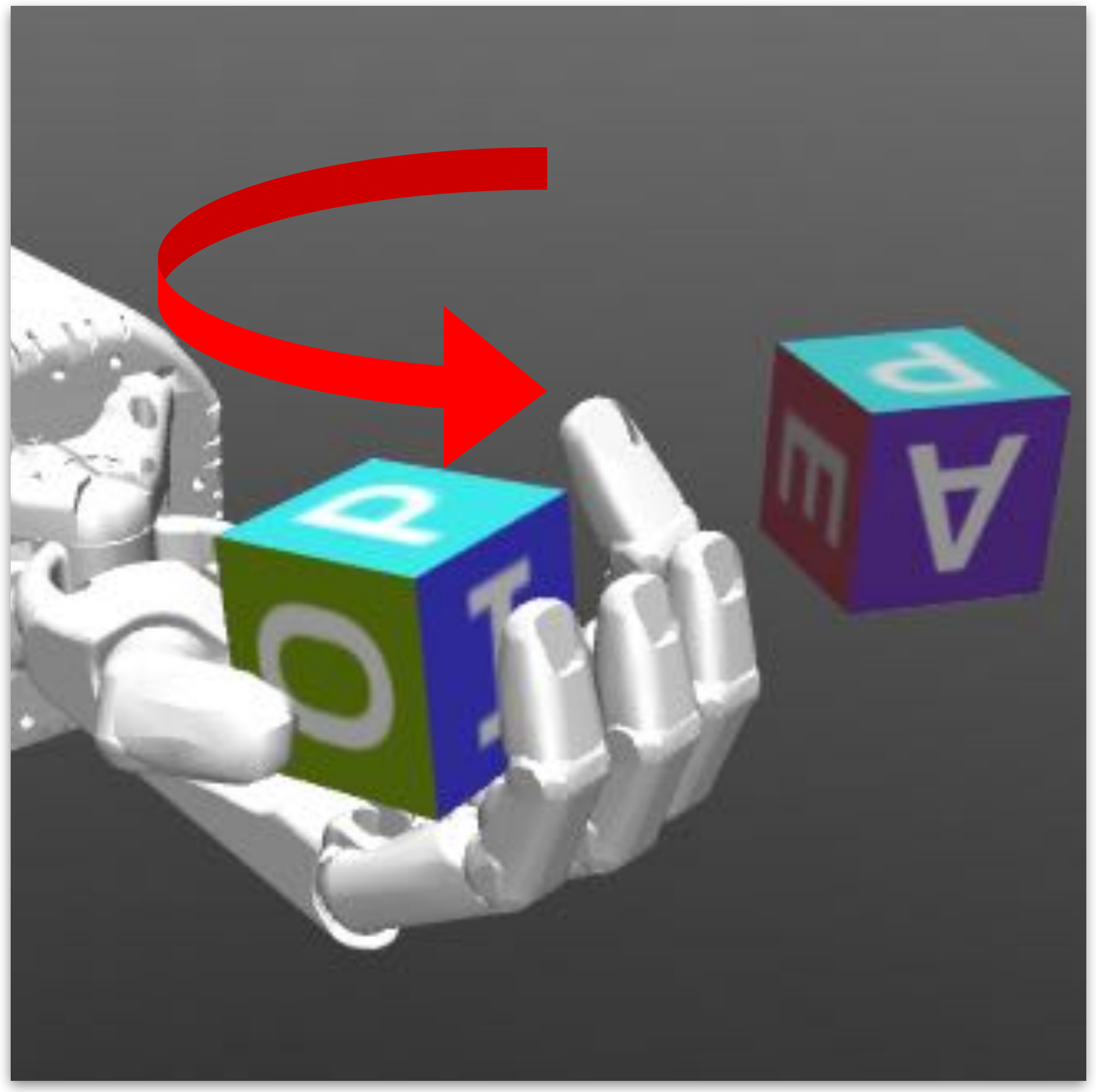

Navigation


Maze2D

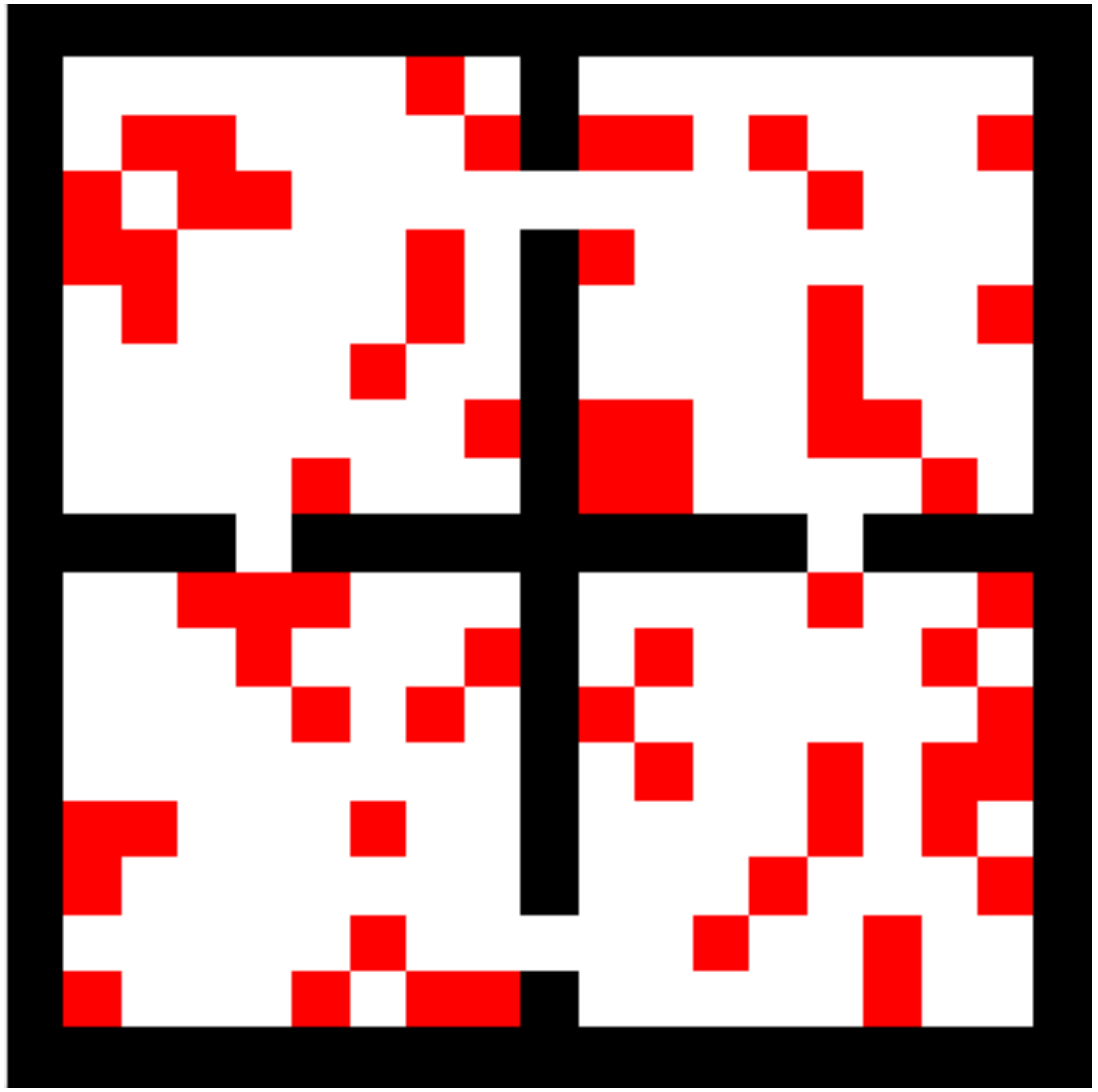

Ant Reach


Fetch Pick

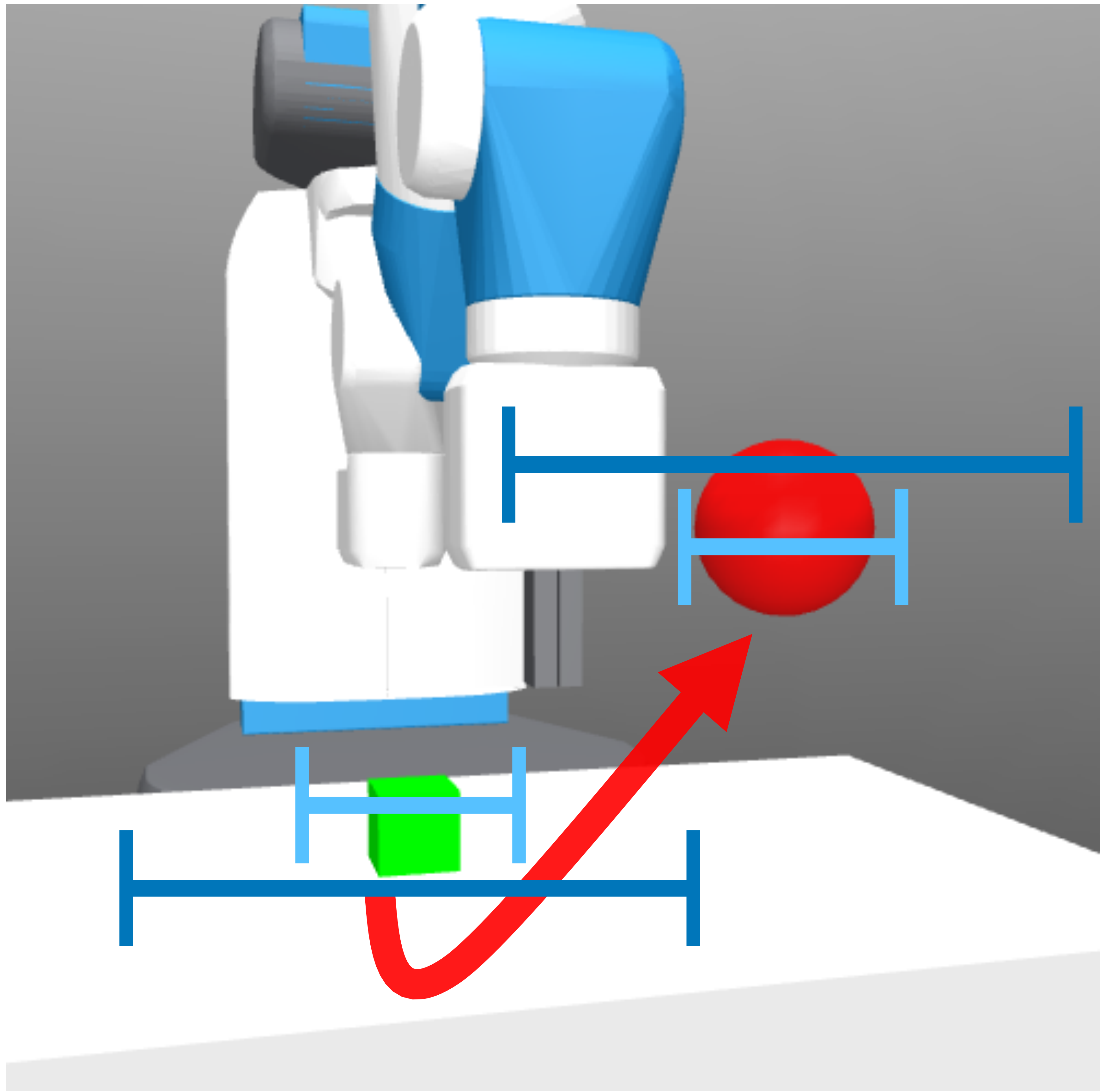

Fetch Push


Hand Rotate

Generalization Experiments

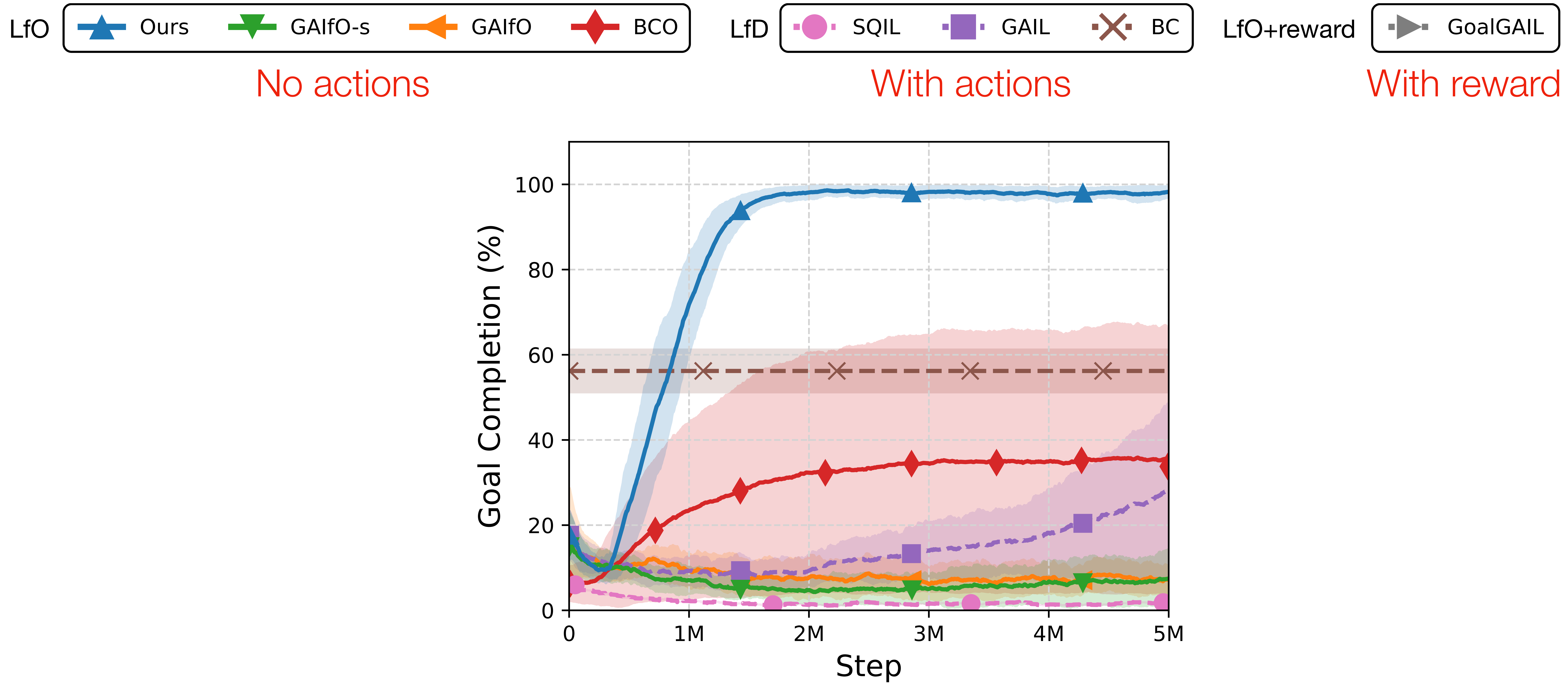


Case 1: Demonstrations cover only part of the state space



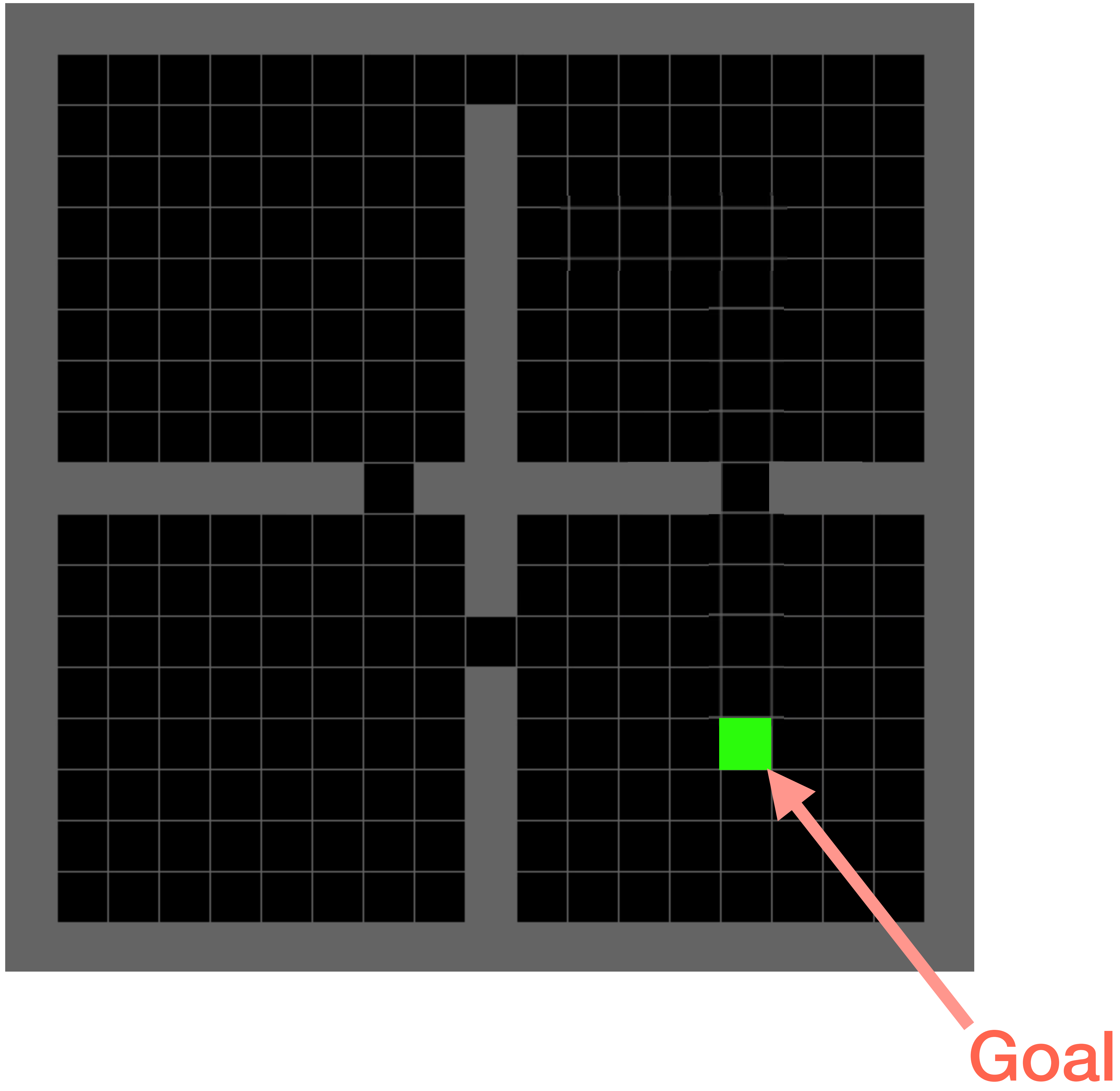
Case 2: Small expert sampling noise vs. Large agent sampling noise

Navigation — 25% Coverage

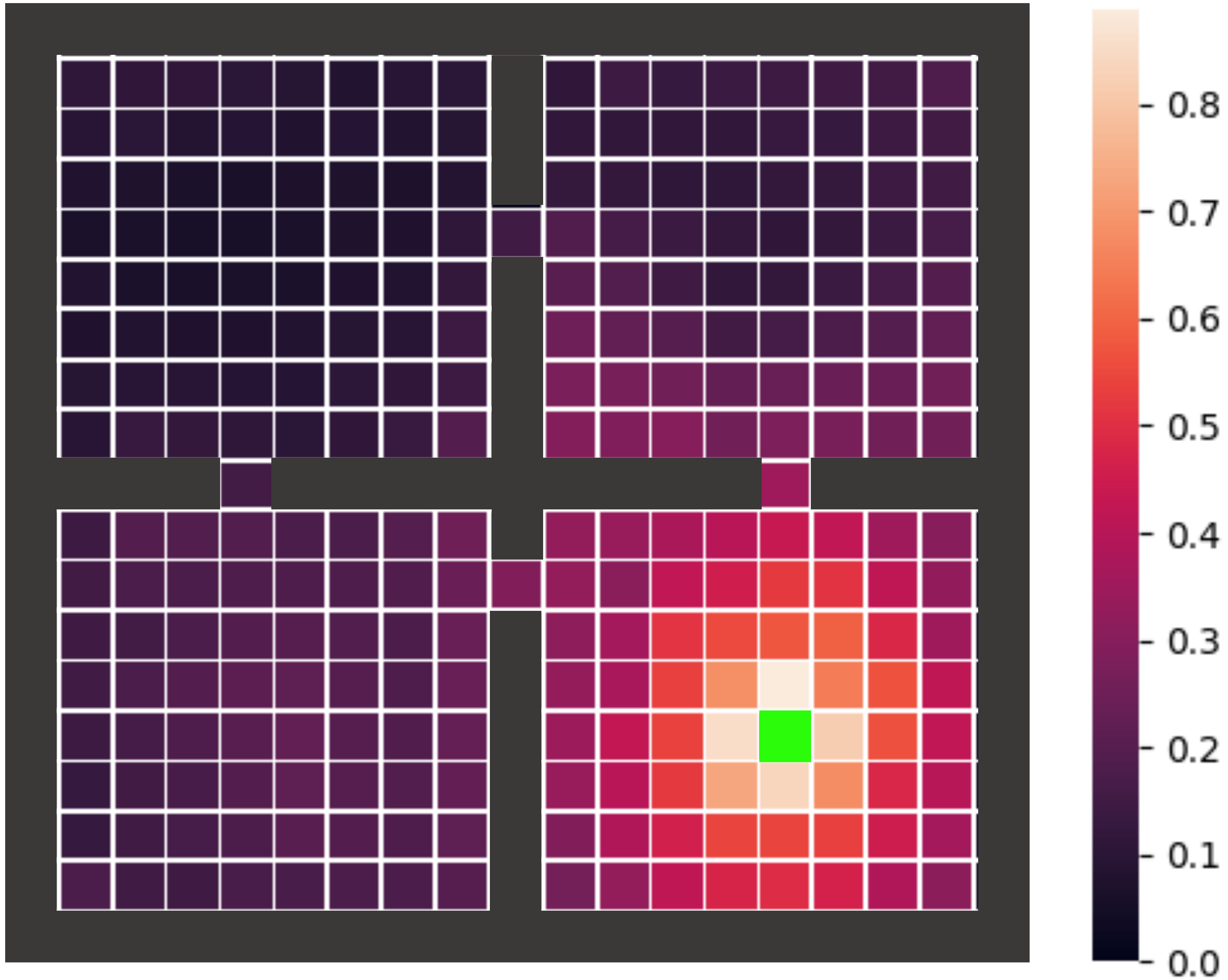


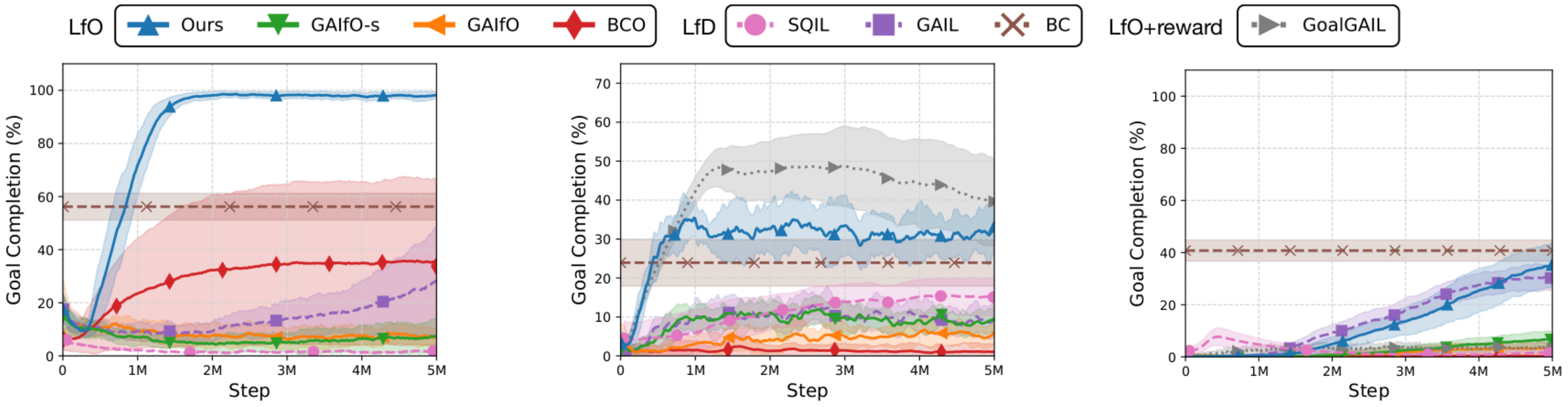
Navigation — Different Coverages

Navigation Task



Learned Proximity Function

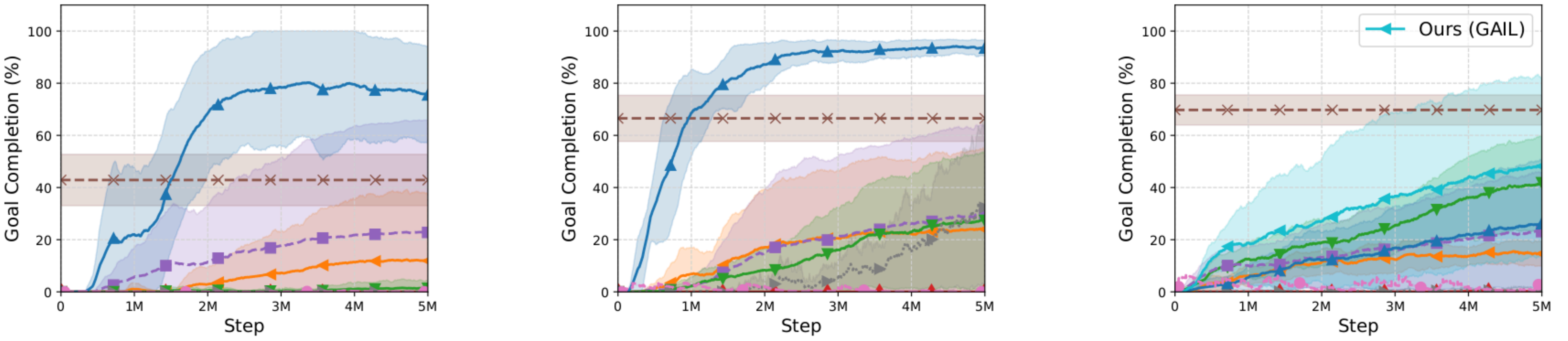




(a) NAVIGATION 25%

(b) MAZE2D 50%

(c) ANT REACH 0.05 noise



(d) FETCH PICK 1.75x noise

(e) FETCH PUSH 1.75x noise

(f) HAND ROTATE 0.35 noise

Takeaways

- **Goal proximity** is **generalizable**, **freely available** task information, and effectively guides an agent to imitate demonstrations
- Our approach jointly learns **goal proximity function** and **policy**
- Our method outperforms LfO baselines and is comparable to LfD baselines in multiple tasks: navigation, locomotion, and manipulation

Generalizable Imitation Learning from Observation via Inferring Goal Proximity

Youngwoon Lee*, Andrew Szot*, Shao-Hua Sun, Joseph J. Lim

For more details: clvrai.com/GPIL