Markov Decision Process II

Sanjiban Choudhury

Cornell Bowers CIS
Computer Science

Learning

Robot

Decision

Making

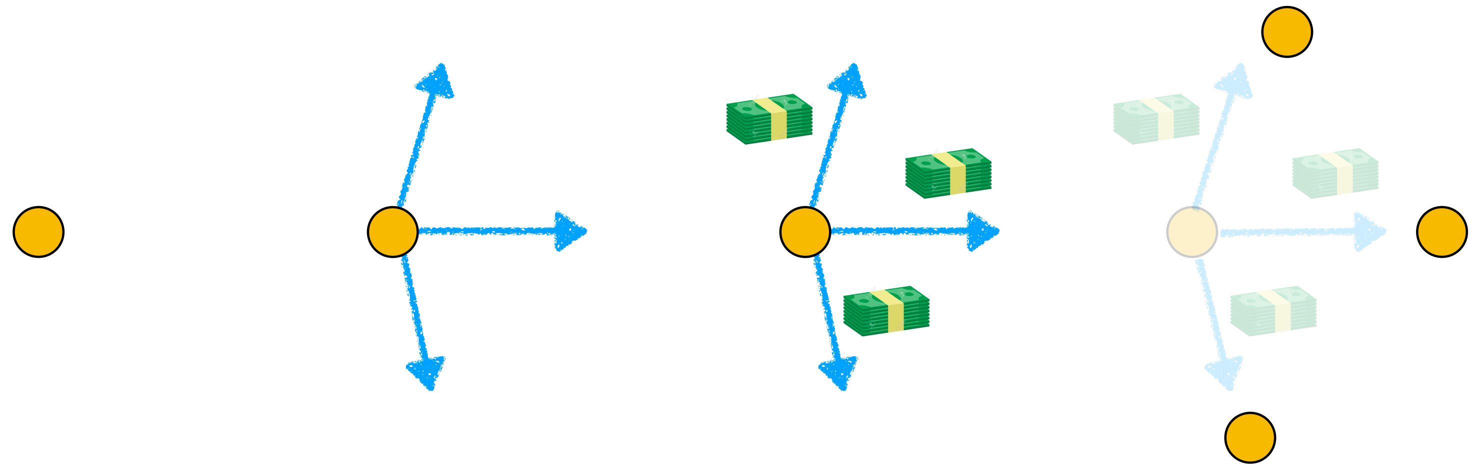
Today!

Recap

Markov Decision Process

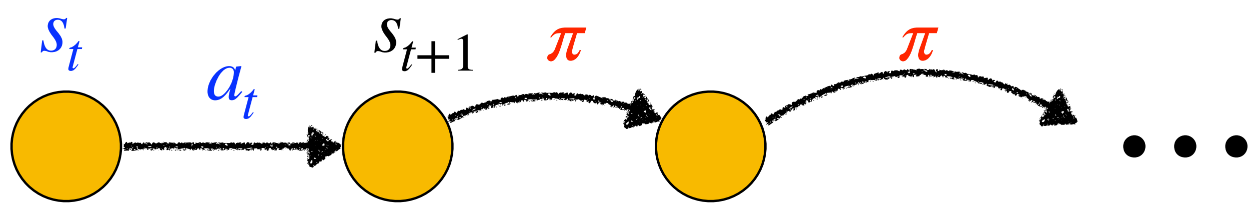
A mathematical framework for modeling sequential decision making

$$< S \; , A \; , C \; , \mathcal{T} \; >$$



x

Value of a state-action



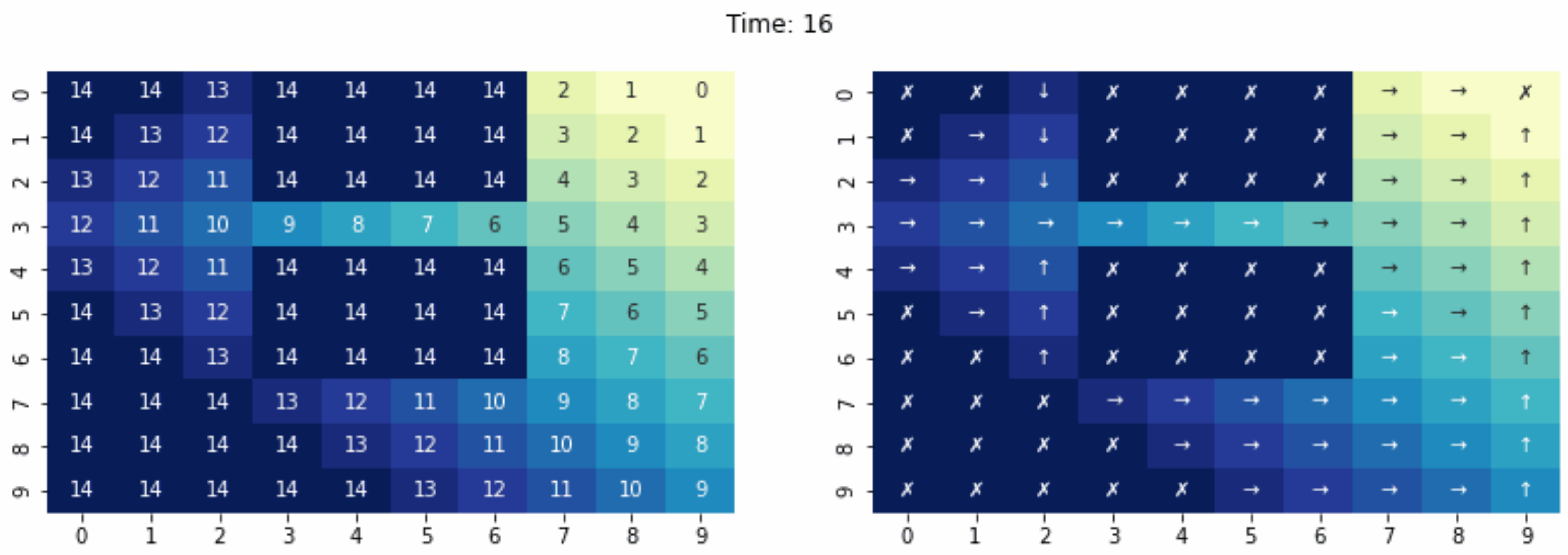
$$Q^{\pi}(s_t, a_t) \; = c_t \; + \; \gamma c_{t+1} \; + \; \gamma^2 c_{t+2} \; + \cdots$$

Expected discounted sum of cost from starting at a state, executing action and following a policy from then on

$$Q^{\pi}(s_t, a_t) = c(s_t, a_t) + \gamma \mathbb{E}_{s_{t+1} \sim \mathcal{T}(s_t, a_t)} V^{\pi}(s_{t+1})$$

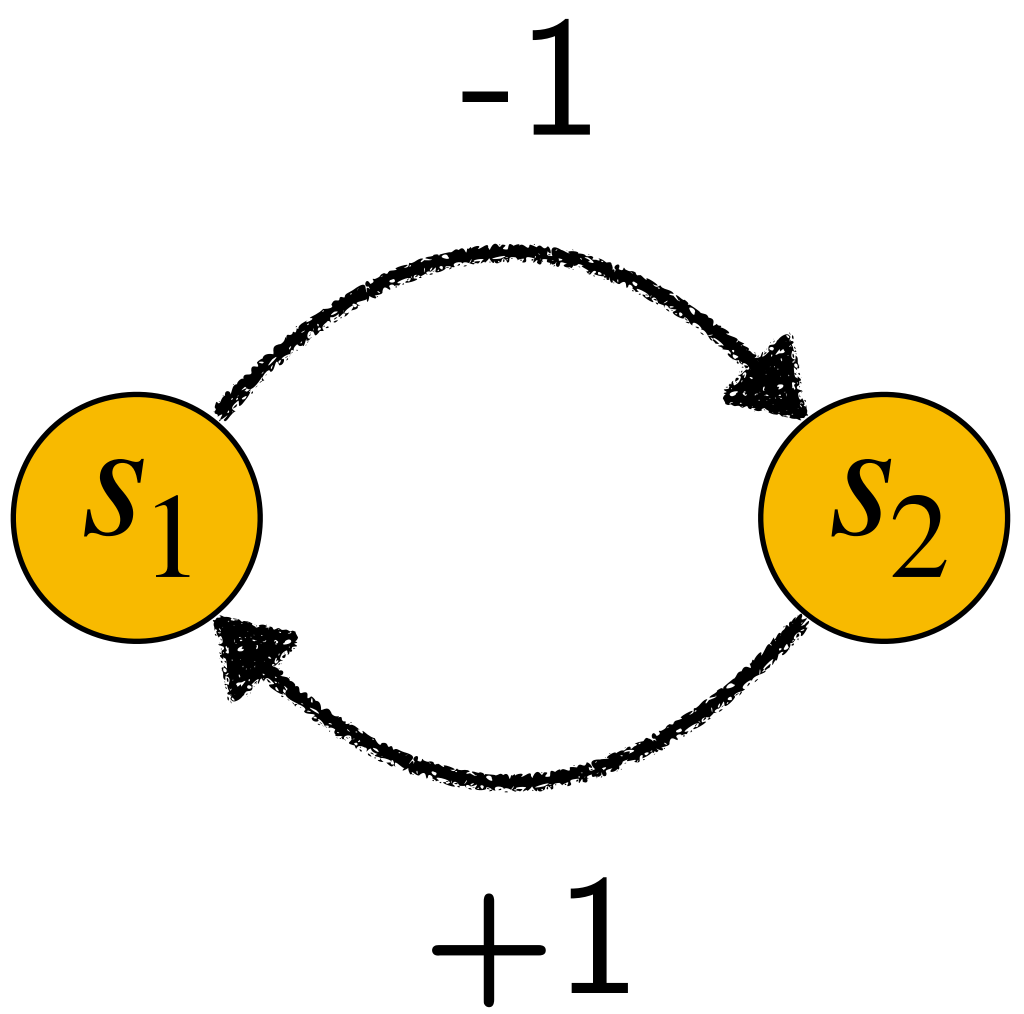
x

Dynamic Programming all the way!



$$V^*(s_t) = \min_a [c(s_t, a) + V^*(s_{t+1})] \qquad \pi^*(s_t) = \arg\min_a [c(s_t), a) + V^*(s_{t+1})]$$

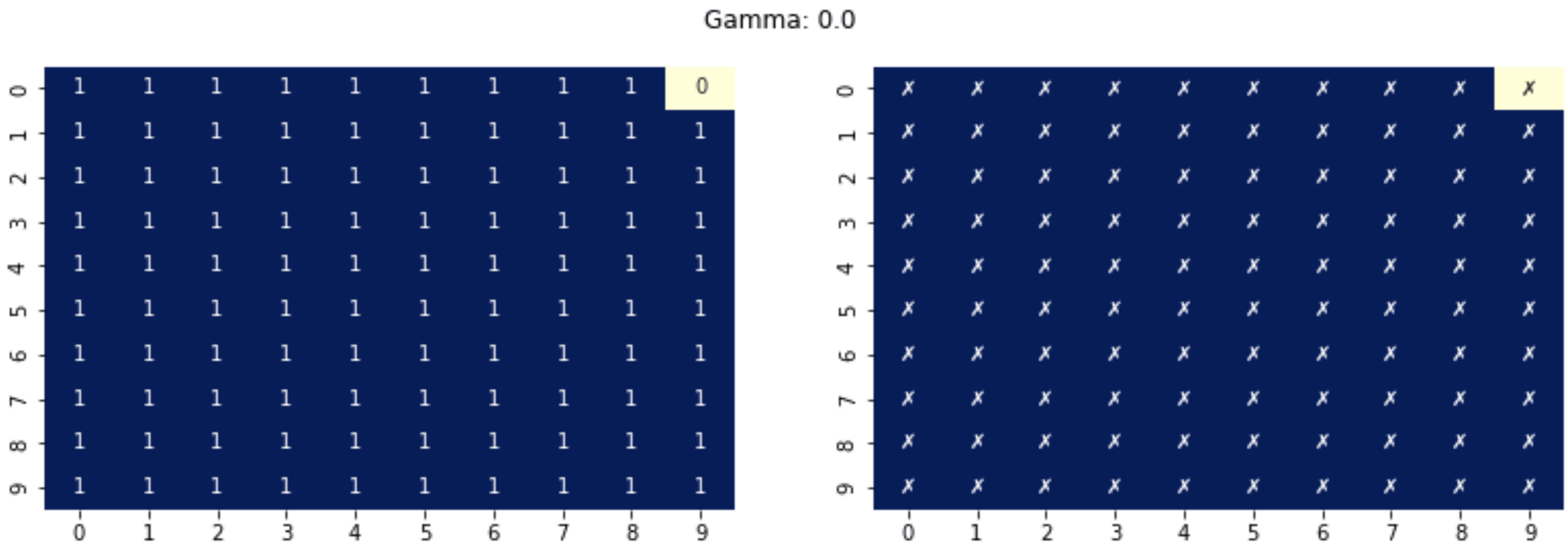
Does value iteration converge?



$\gamma = 1$

What is $V^*(s_1)$? What is $V^*(s_2)$?

What is the effect of discount factor?



Activity!

Think-Pair-Share

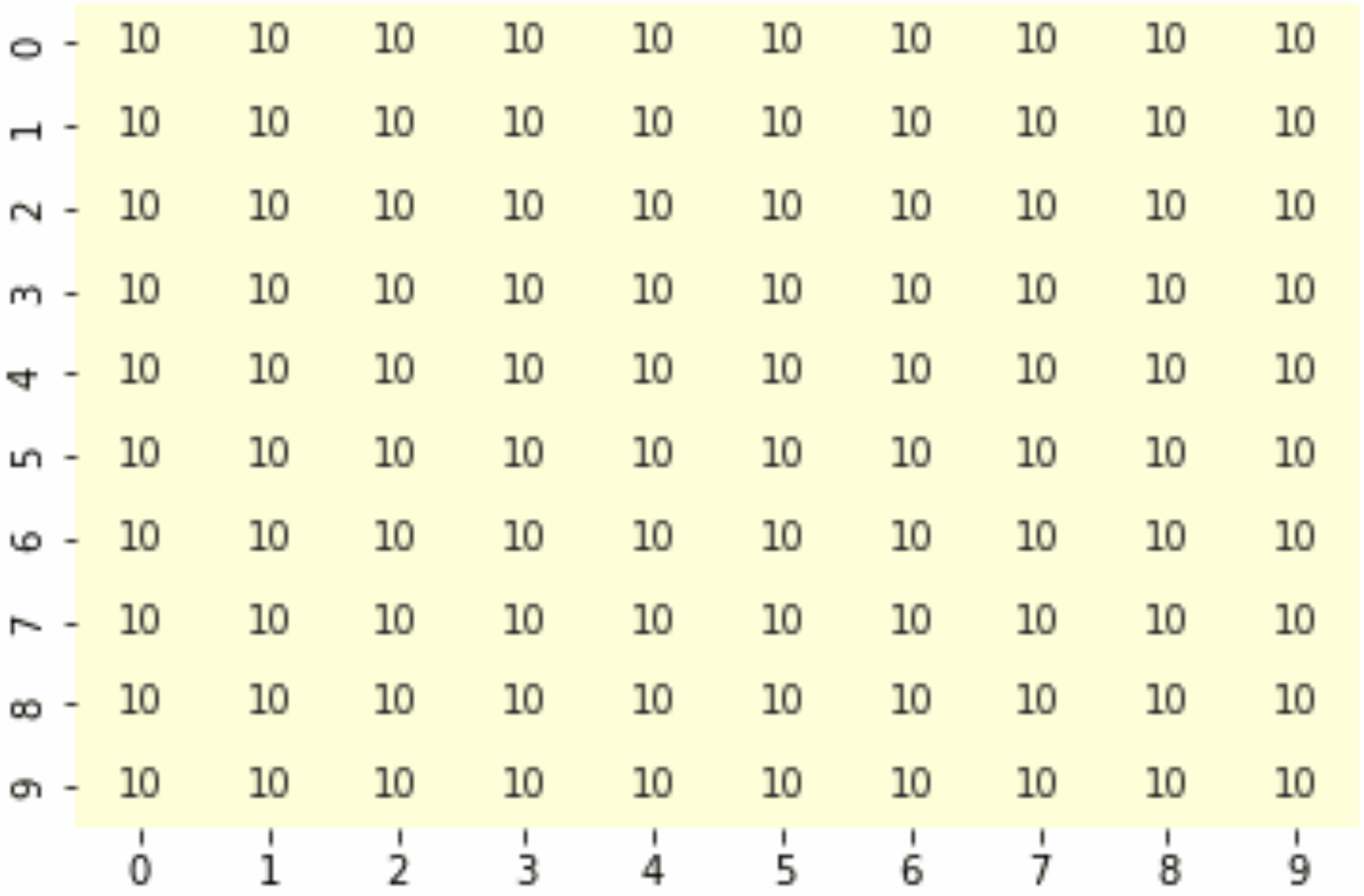
Think (30 sec): What are some attributes of a hard MDP?

Pair: Find a partner

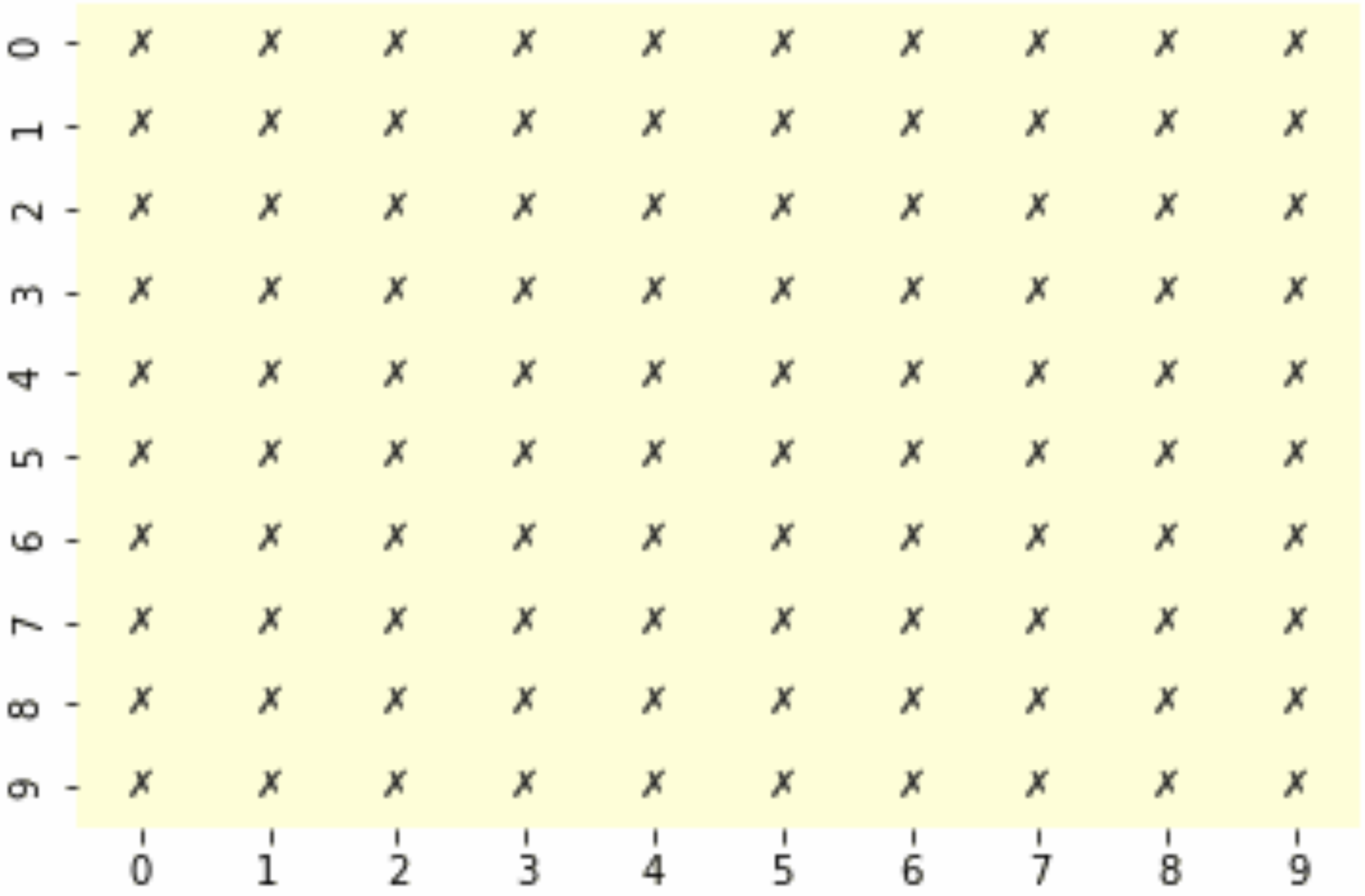
Share (45 sec): Partners exchange ideas

Policy Iteration

How frequently does the best action change?



Values



Policy

Policy converges faster than the value

Can we iterate over policies?

Policy Iteration

Init with some policy π

Repeat forever

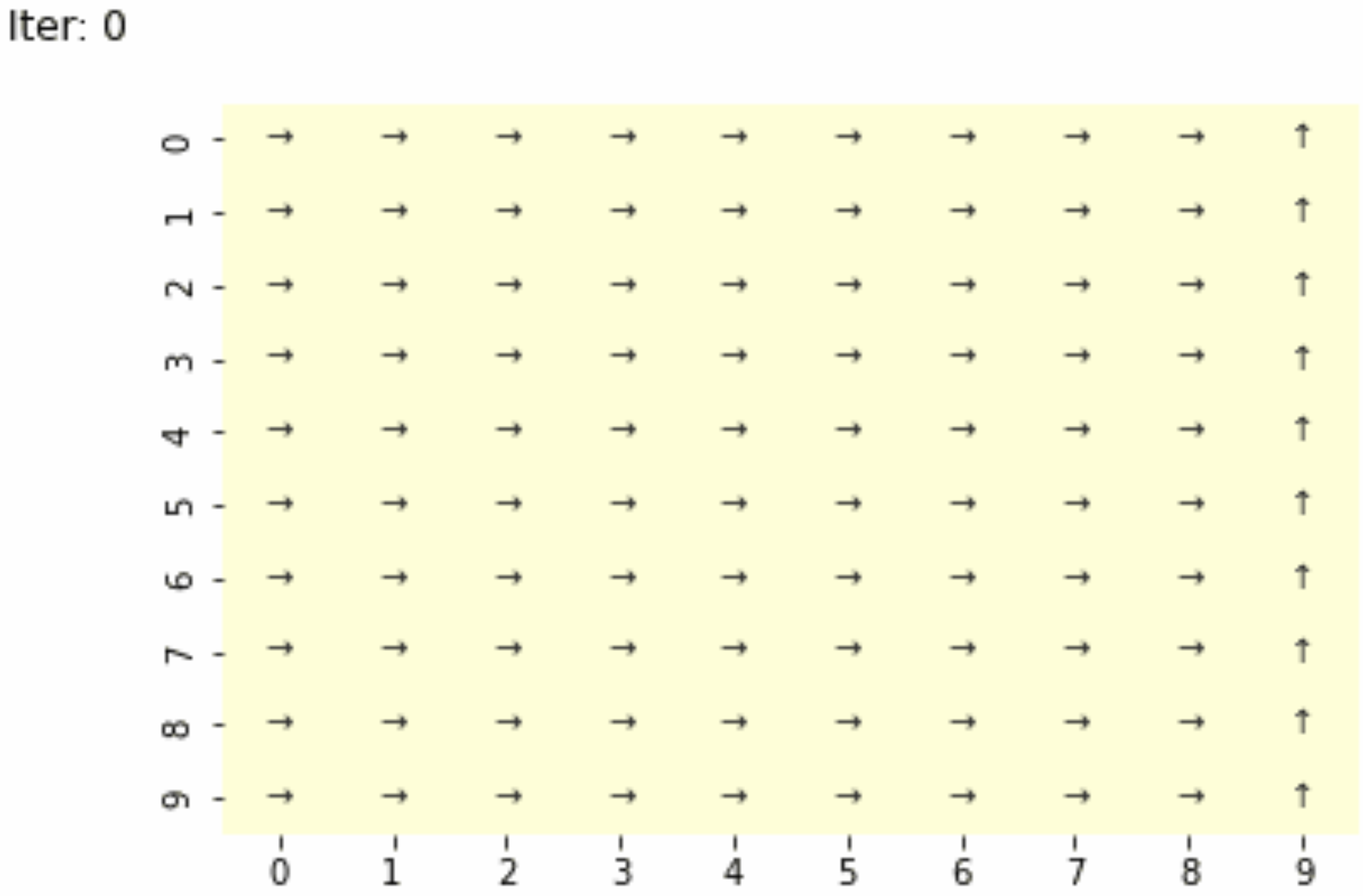
Evaluate policy

$$V^{\pi}(s) = c(s, \pi(s)) + \gamma \mathbb{E}_{s' \sim \mathcal{T}(s,a)} V^{\pi}(s')]$$

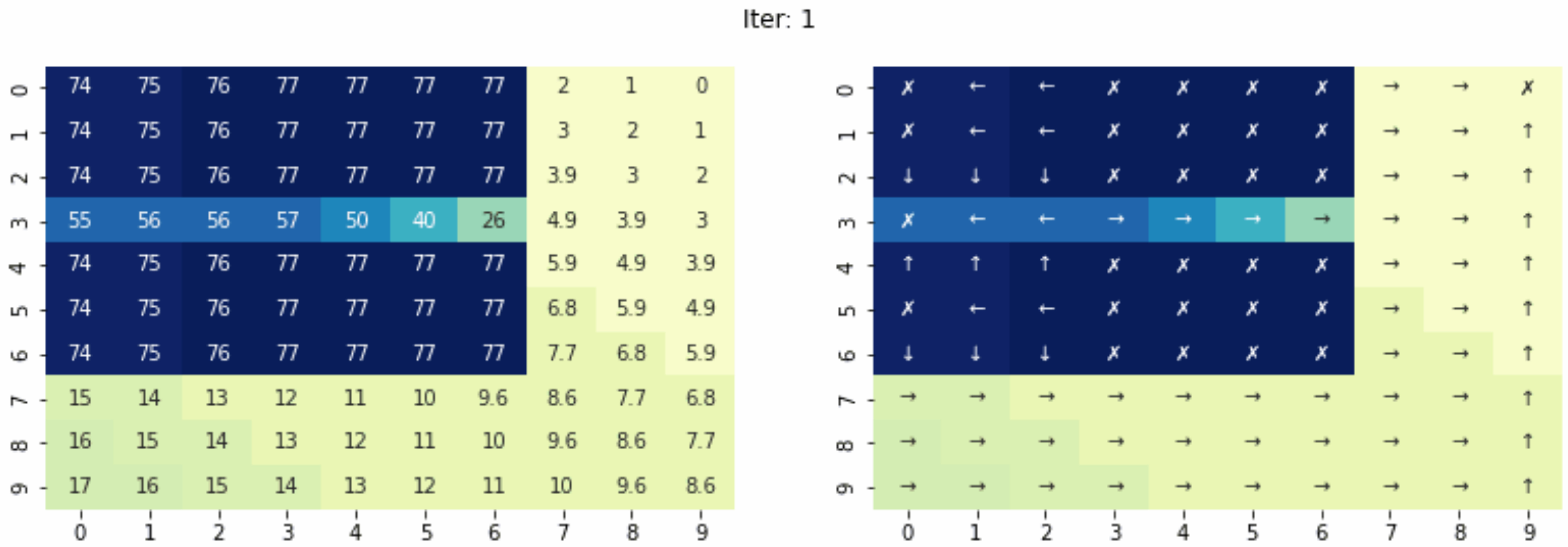
Improve policy

$$\pi^{+}(s) = \arg\min_{a} c(s, a) + \gamma \mathbb{E}_{s' \sim \mathcal{T}(s,a)} V^{\pi}(s')]$$

Init with some policy π



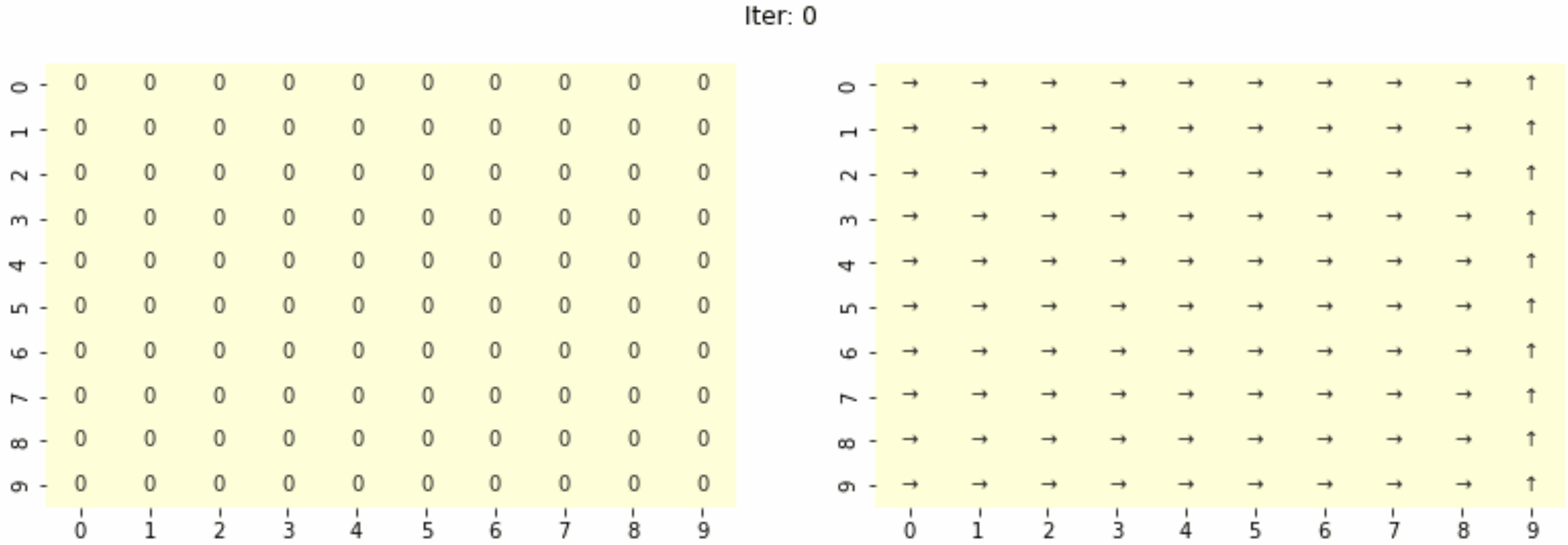
Iteration 1



$V^{\pi}(s) = c(s, \pi(s)) + \gamma \mathbb{E}_{s' \sim \mathcal{T}(s,a)} V^{\pi}(s')]$

$\pi^{+}(s) = \arg\min_{a} c(s, a) + \gamma \mathbb{E}_{s' \sim \mathcal{T}(s,a)} V^{\pi}(s')]$

Policy Iteration



$$V^{\pi}(s) = c(s, \pi(s)) + \gamma \mathbb{E}_{s' \sim \mathcal{T}(s,a)} V^{\pi}(s')] \qquad \pi^{+}(s) = \arg\min_{a} c(s, a) + \gamma \mathbb{E}_{s' \sim \mathcal{T}(s,a)} V^{\pi}(s')]$$

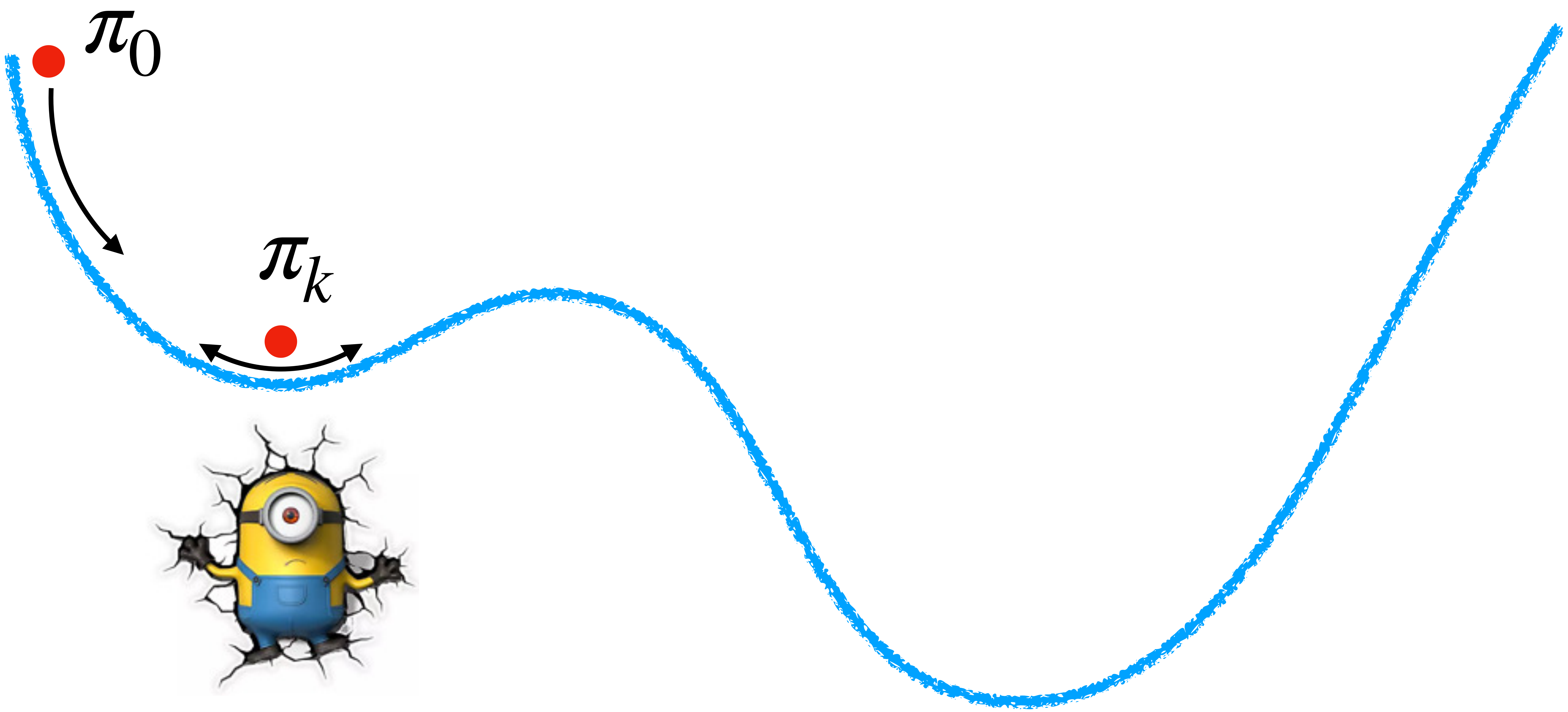
So many questions ....

Q1. Does policy iteration converge to the optimal policy?

Q2. Does it converge faster than value iteration?

Q3. Can we bootstrap policy evaluation?

Q1. Does policy iteration converge to optimal policy?



Stuck at local minima??

π^*

Q1. Does policy iteration converge to optimal policy?

Proof has 2 step argument

Step 1: Policy iteration *monotonically* improves

Step 2: Once it reaches the optimal policy, it does not change

Performance Difference Lemma

Q1. Does policy iteration converge to optimal policy?

2 step argument

✓ Step 1: Policy iteration *monotonically* improves

All advantages <=0 implies monotonic performance improvement

✓ Step 2: Once it reaches the optimal policy, it does not change

Advantage >=0 for the optimal policy

PDL answers *all* ...

In Imitation Learning

In Model Free Reinforcement Learning

In Model Based Reinforcement Learning

How it partners with online learning

So many questions ....

Q1. Does policy iteration converge to the optimal policy?

Q2. Does it converge faster than value iteration?

Q3. Can we bootstrap policy evaluation?

So many questions ....

Q1. Does policy iteration converge to the optimal policy?

¯_(ツ)_/¯

Q2. Does it converge faster than value iteration?

Empirically yes, but no rigorous theory about when ..

Q3. Can we bootstrap policy evaluation?

So many questions ....

Q1. Does policy iteration converge to the optimal policy?

¯_(ツ)_/¯

Q2. Does it converge faster than value iteration?

Empirically yes, but no rigorous theory about when ..

Q3. Can we bootstrap policy evaluation?

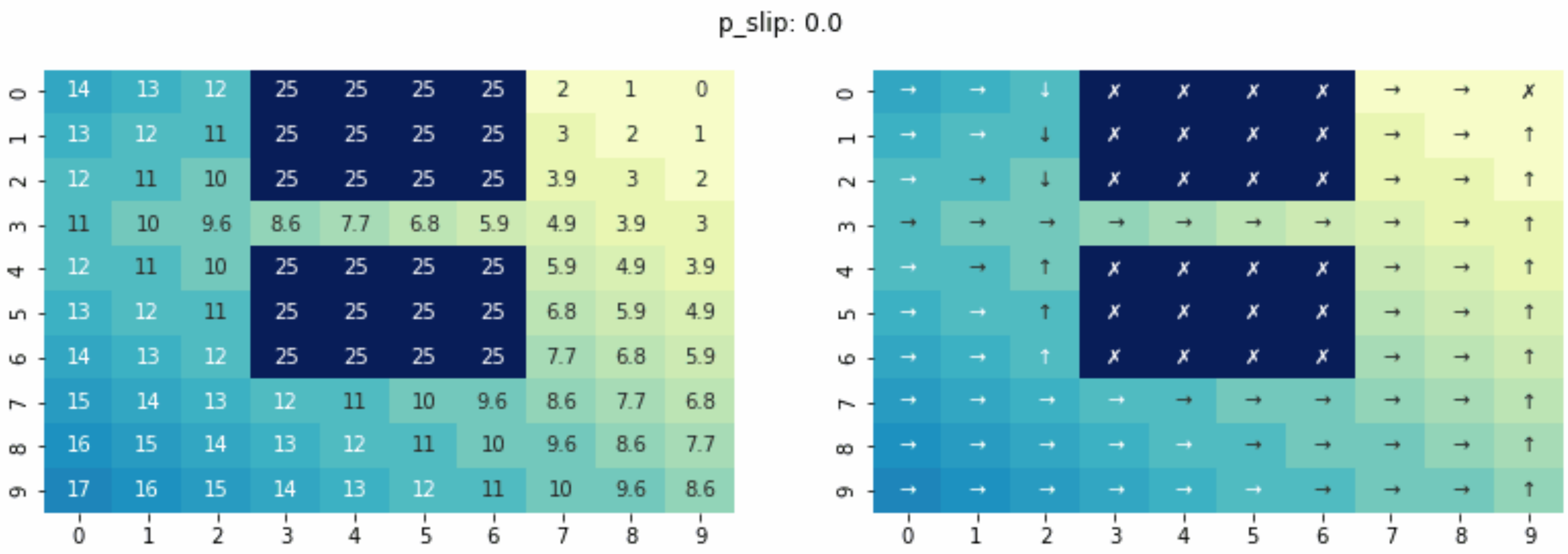
Yes => Modified policy iteration

Messing

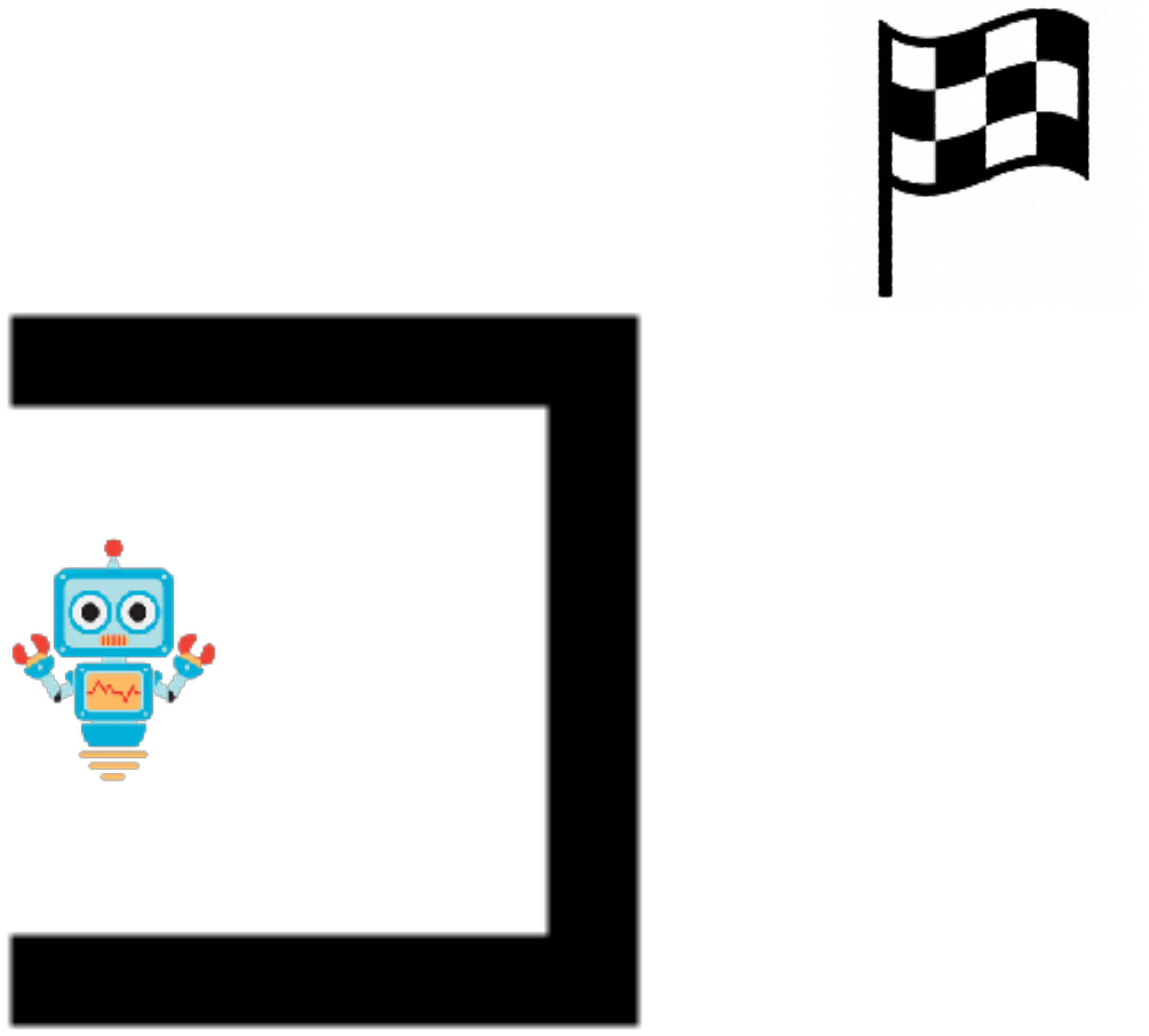
with

MDPs

What happens as you increase the slipperiness of bridge?



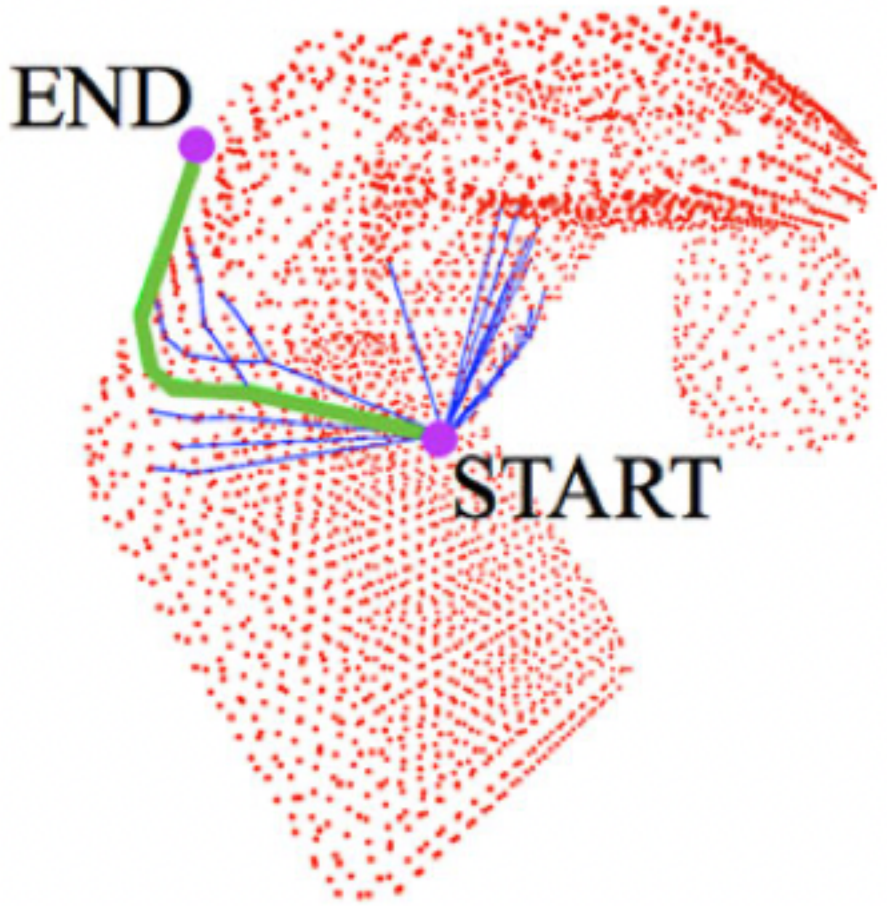
What happens if you change the cost function?

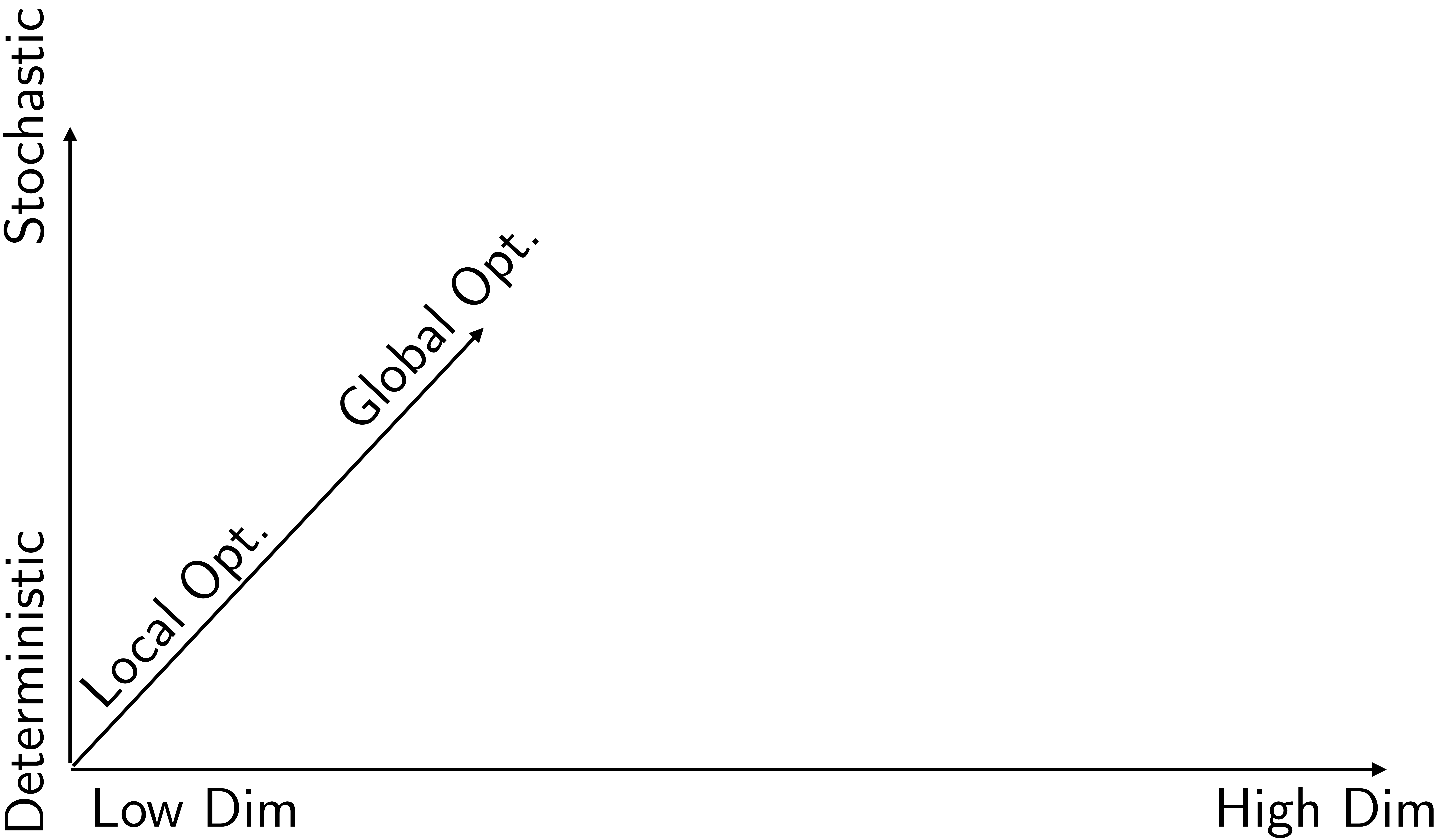


$$C(s) = ||s - s_{goal}||$$

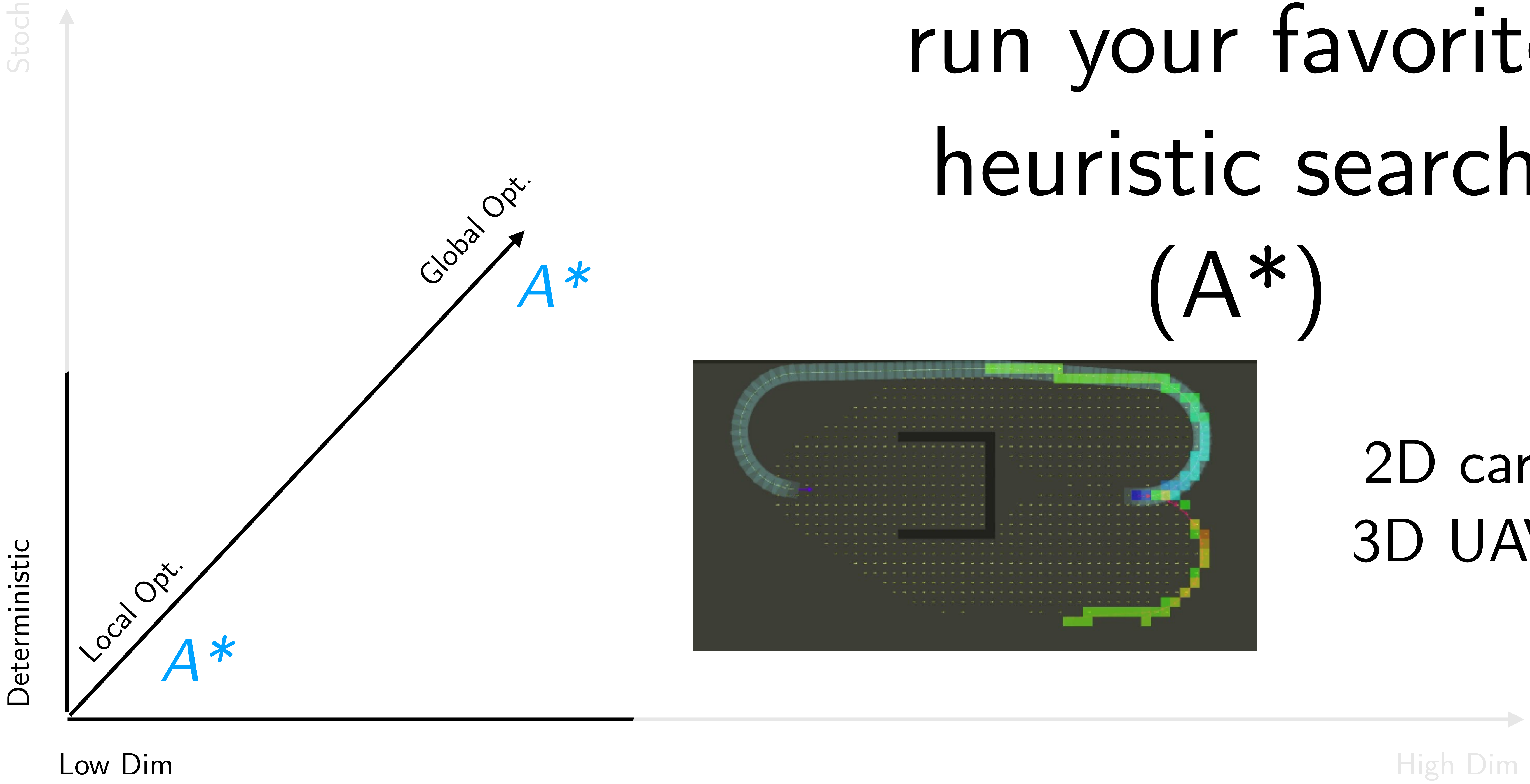
How do you convert a HARD MDP into an EASY one?

Solving continuous MDPs



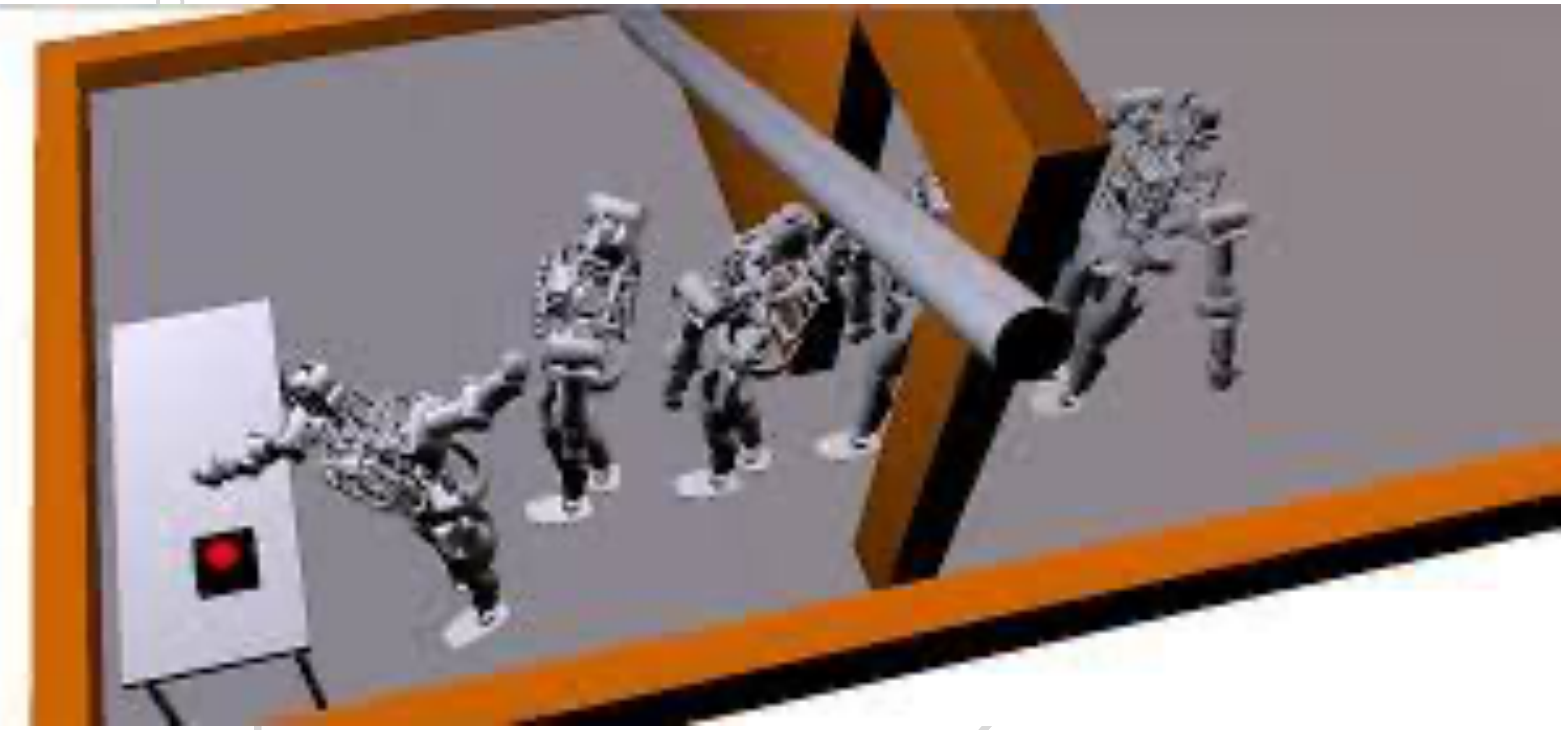

Stochastic
Deterministic
Local Opt.
Global Opt.
Low Dim
High Dim

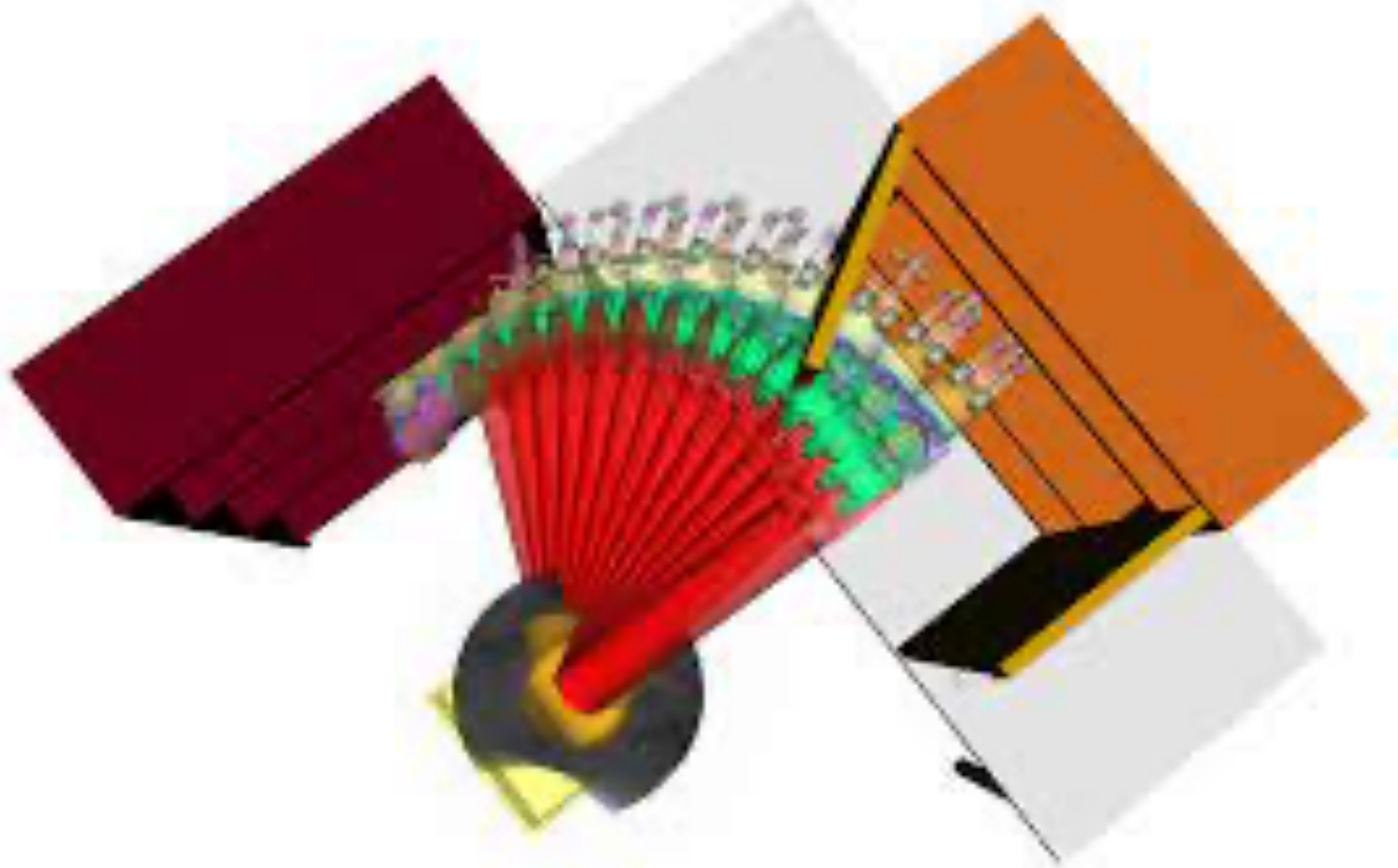
Just discretize and run your favorite heuristic search (A*)



2D car,
3D UAV

Run some sequential convex trajectory optim.





Deterministic

Local Opt.

A*

CHOMP,
TrajOPT

Low Dim

High Dim

Call a (slow) probabilistic planner that “almost surely” gives you the optimum
(Don’t hold your breath...)

Stochastic

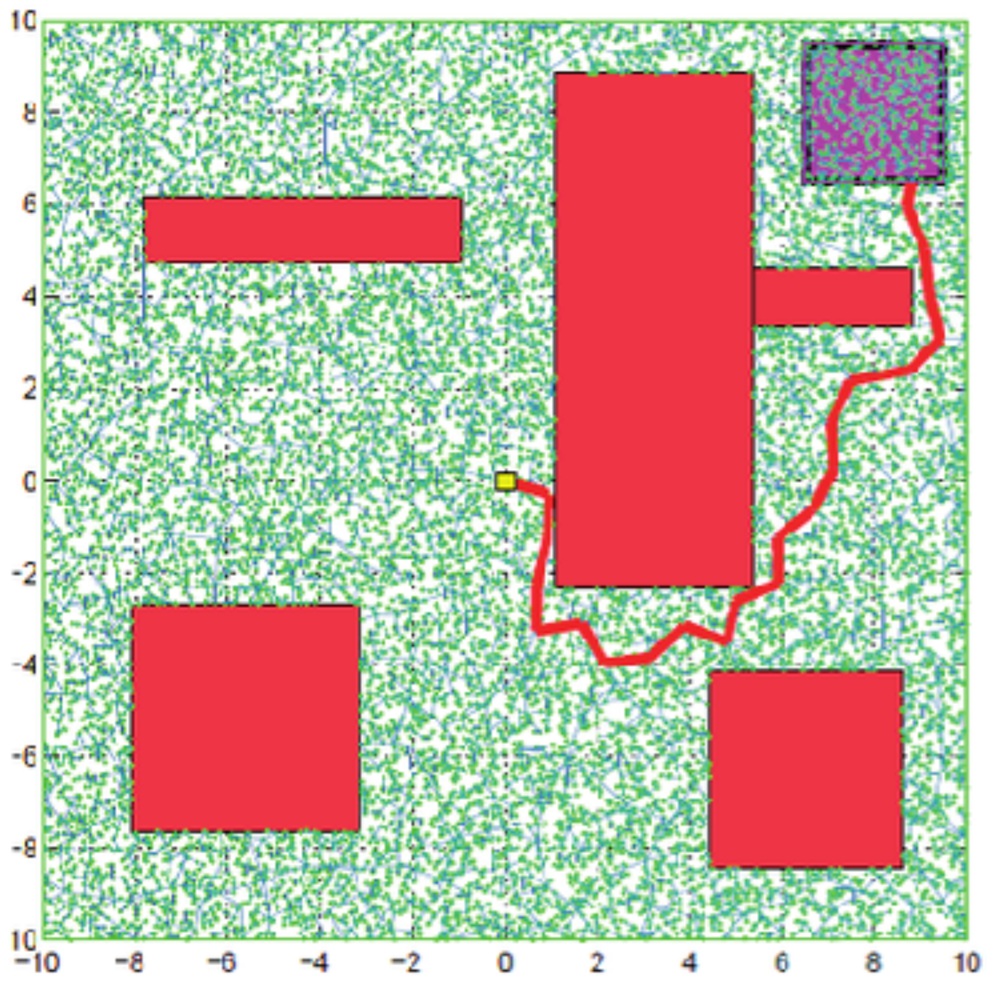
Deterministic

Global Opt.

Local Opt.

A*

A*



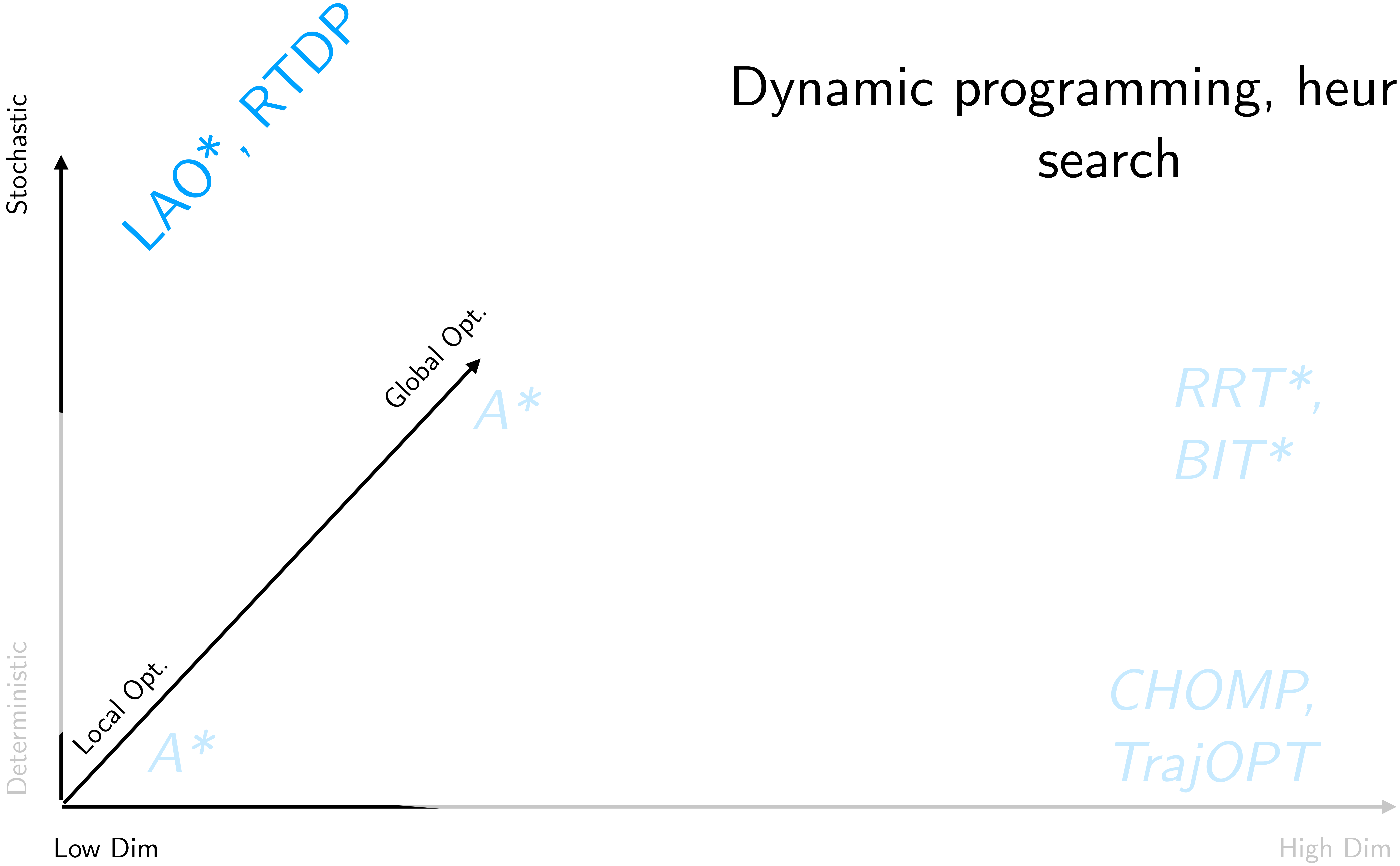
RRT,*
*BIT**

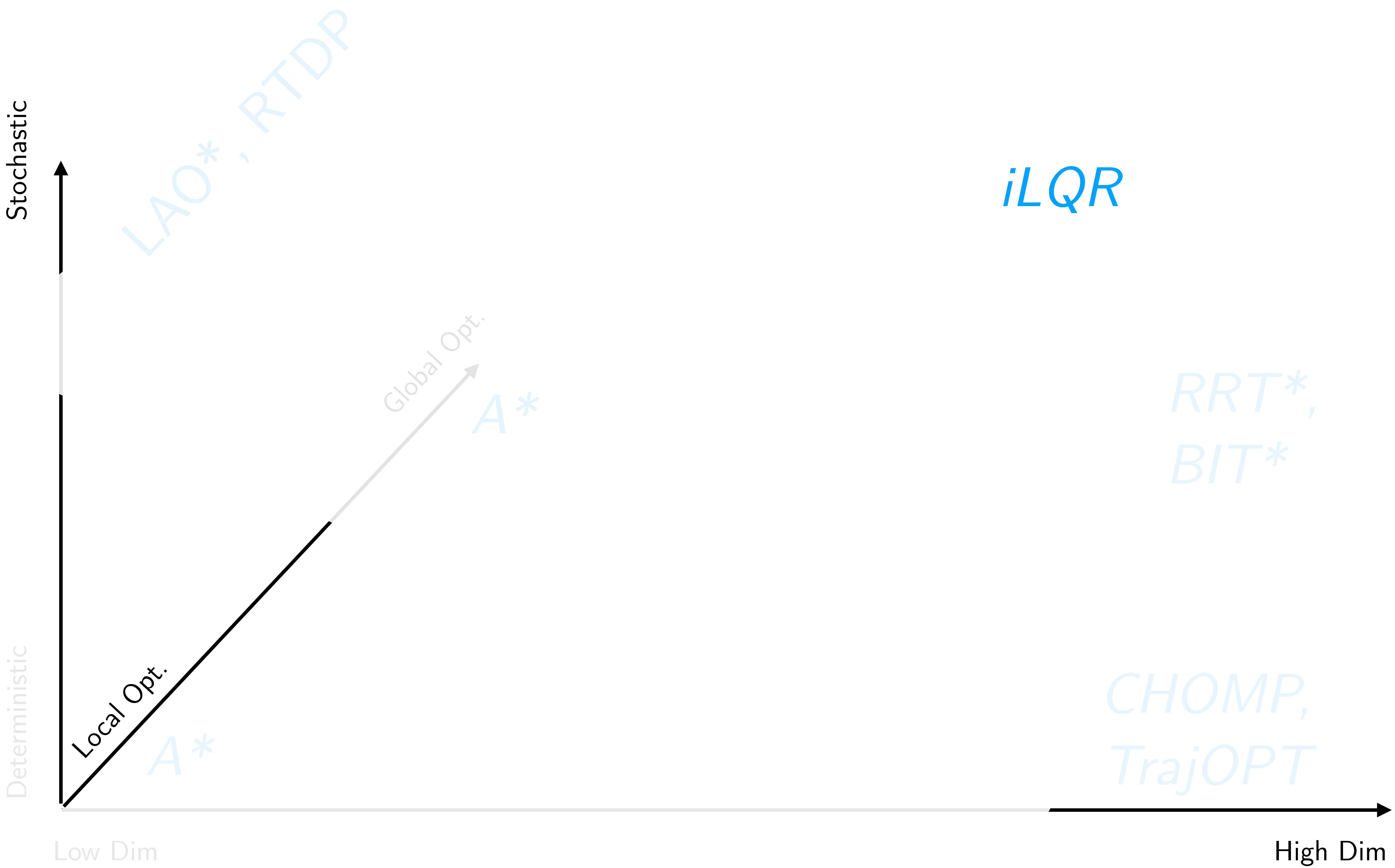
CHOMP,
TrajOPT

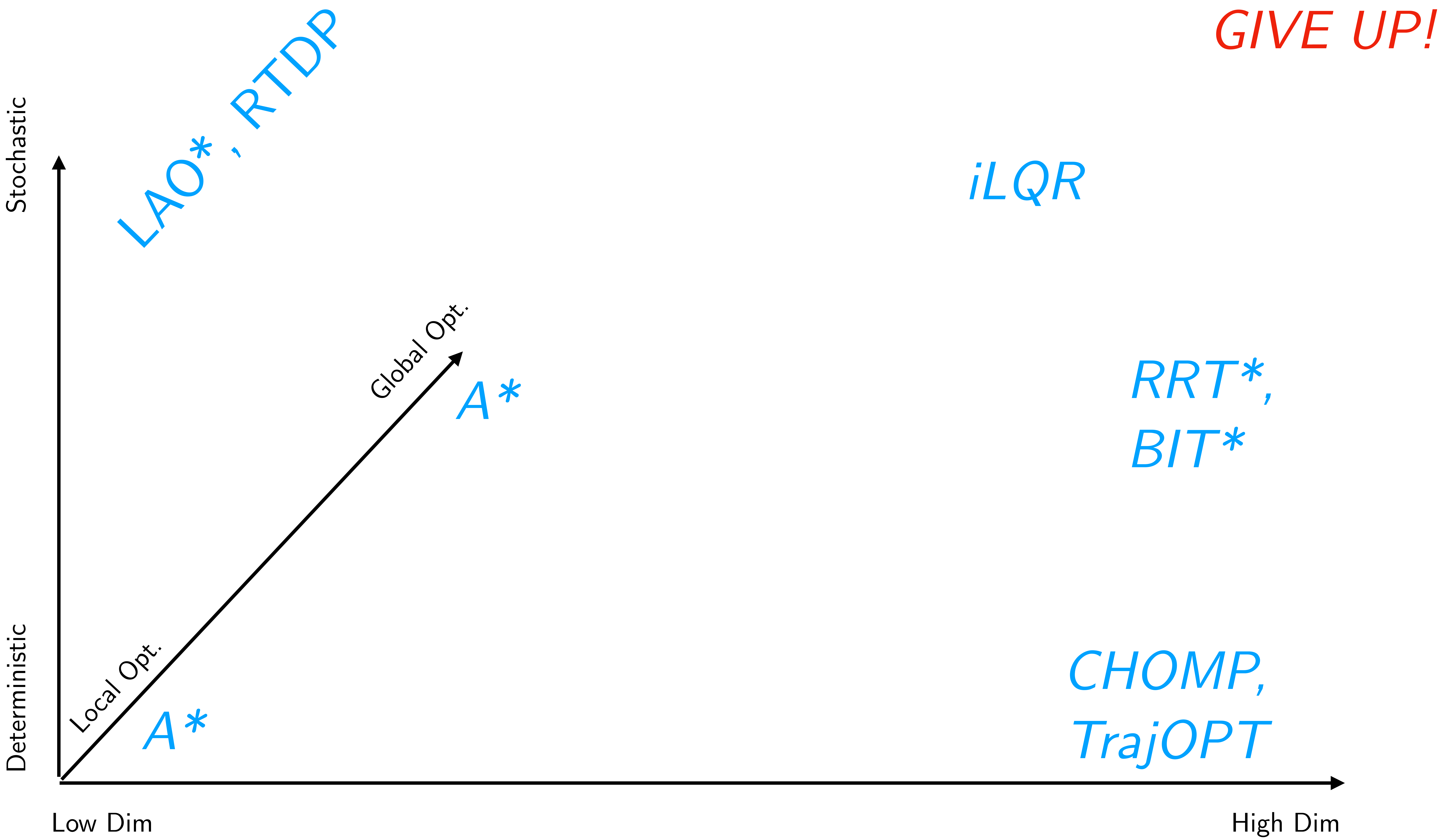
Low Dim

High Dim

Dynamic programming, heuristic search




Stochastic
LAO*, RTDP
iLQR
Global Opt.
A*
RRT*,
BIT*
Deterministic
Local Opt.
A*
CHOMP,
TrajOPT
Low Dim
High Dim


GIVE UP!
LAO*, RTDP
Stochastic
iLQR
Global Opt.
A*
RRT*,
BIT*
Deterministic
Local Opt.
A*
CHOMP,
TrajOPT
Low Dim
High Dim