

Solving Markov Decision Processes

Sanjiban Choudhury

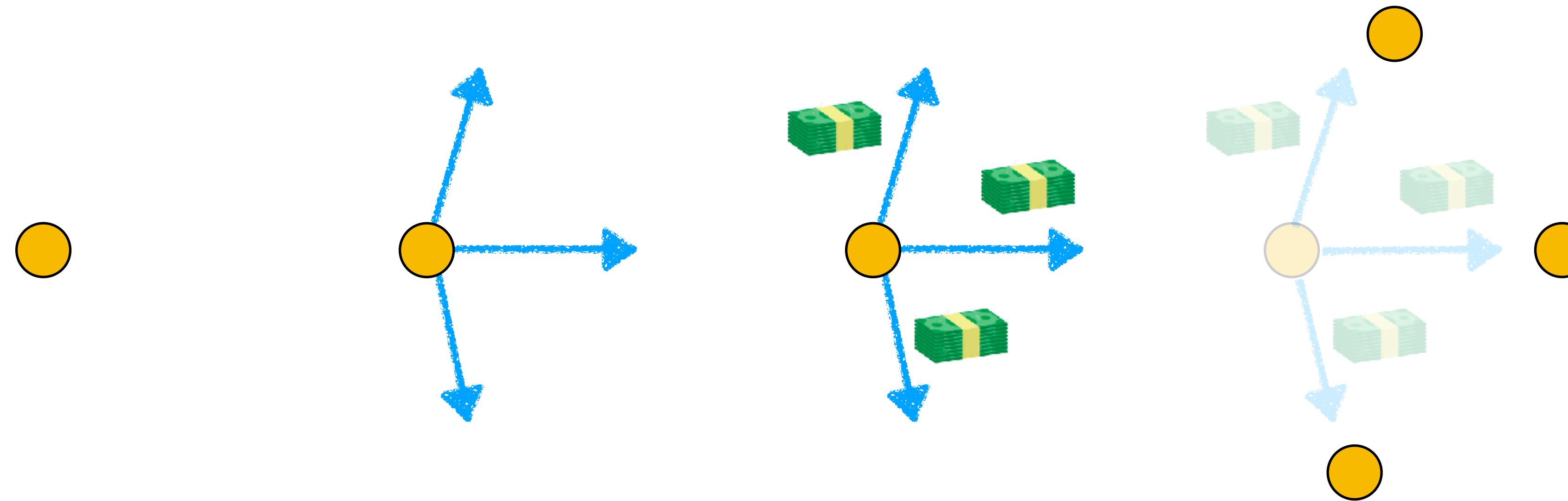


Cornell Bowers CIS
Computer Science

Markov Decision Process

A mathematical framework for modeling sequential decision making

$\langle S, A, C, \mathcal{T} \rangle$



Today's class

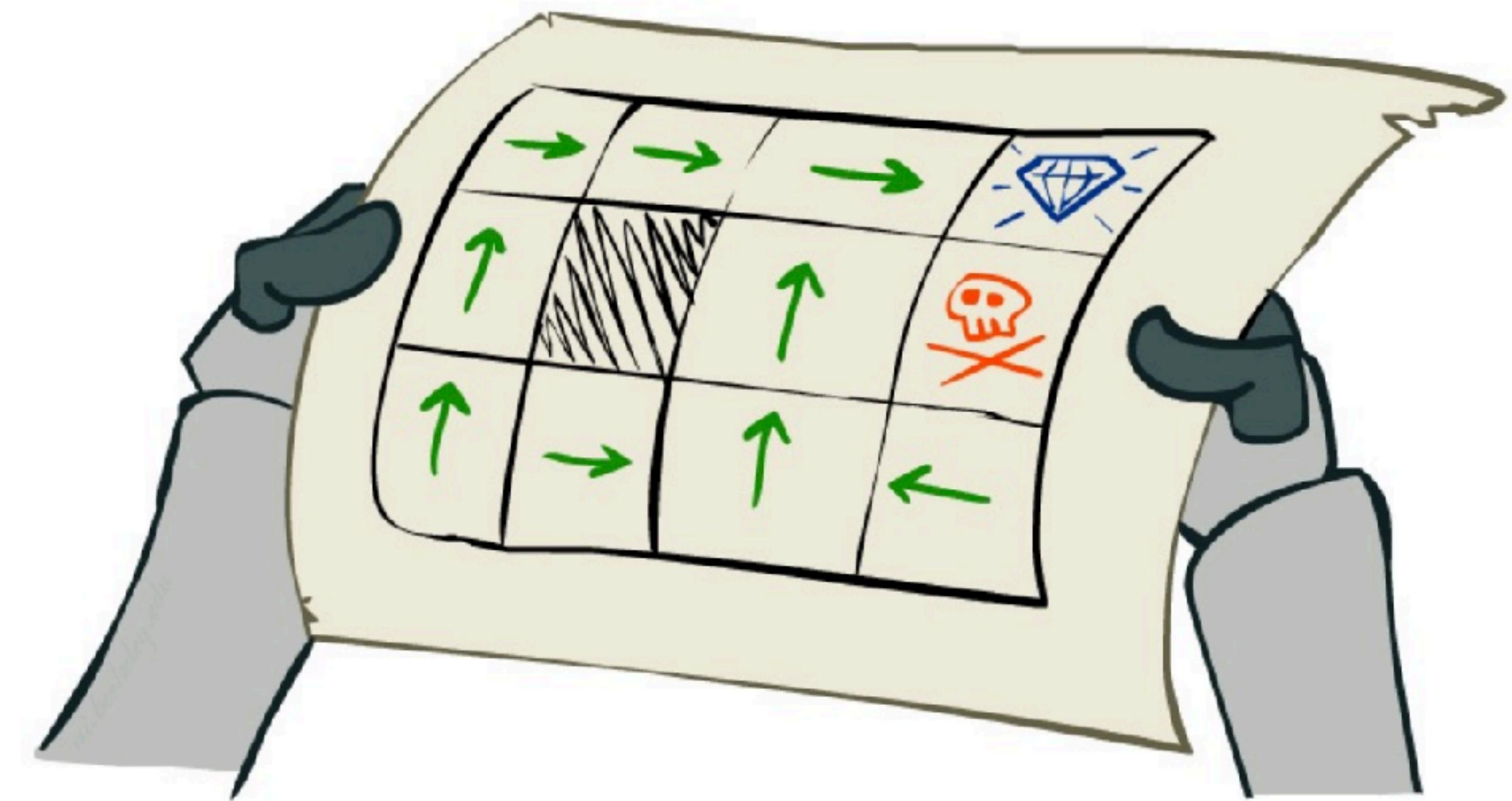
- What does it mean to solve a MDP?
- Bellman Equation
- Value Iteration

What does it mean to solve
a MDP?

Solving an MDP means finding a **Policy**

$$\pi : s_t \rightarrow a_t$$

A function that maps state (and time) to action

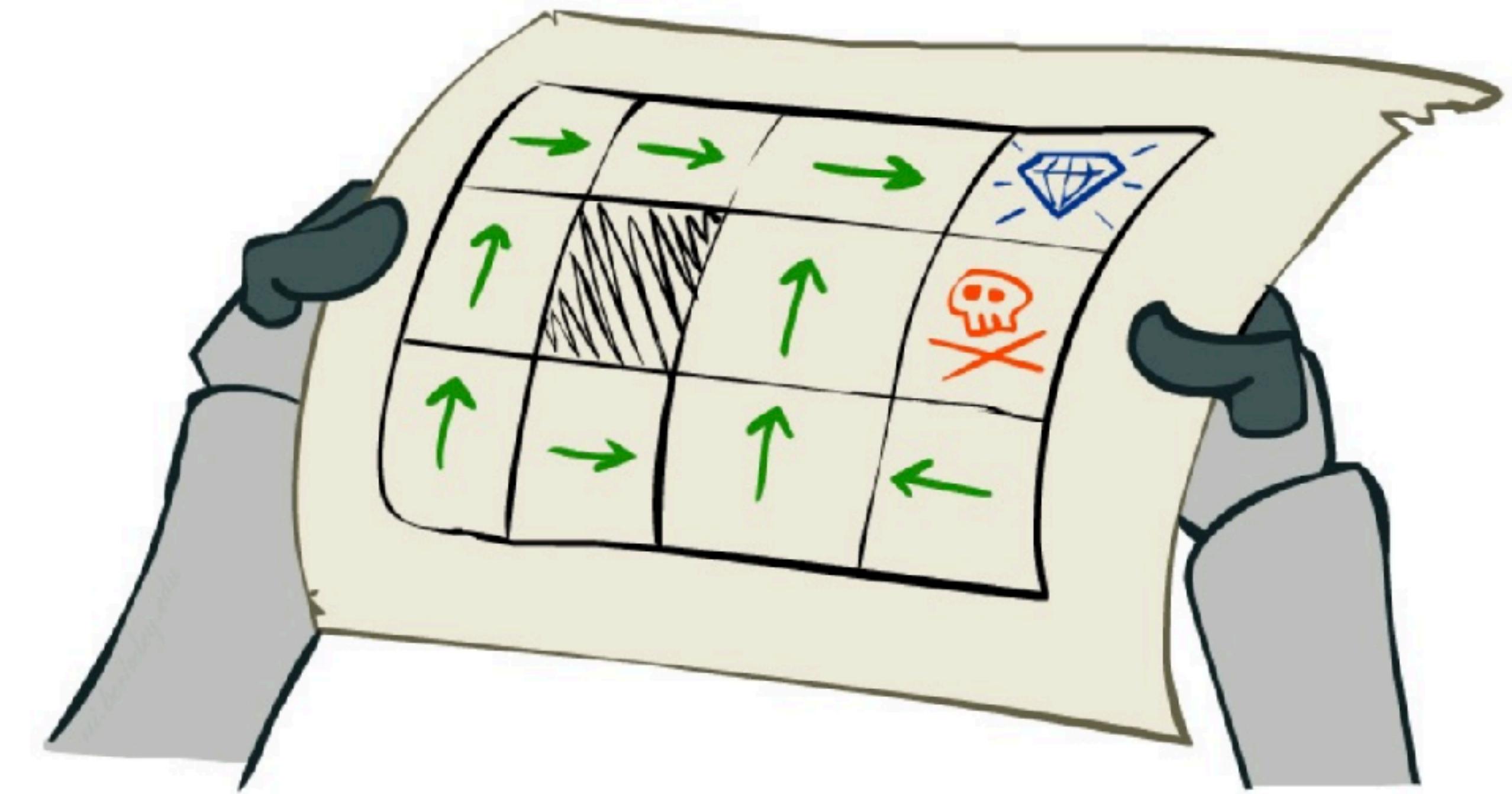


Policy: What action should I choose at any state?

Solving an MDP means finding a **Policy**

$$\pi : s_t \rightarrow a_t$$

A function that maps state (and time) to action

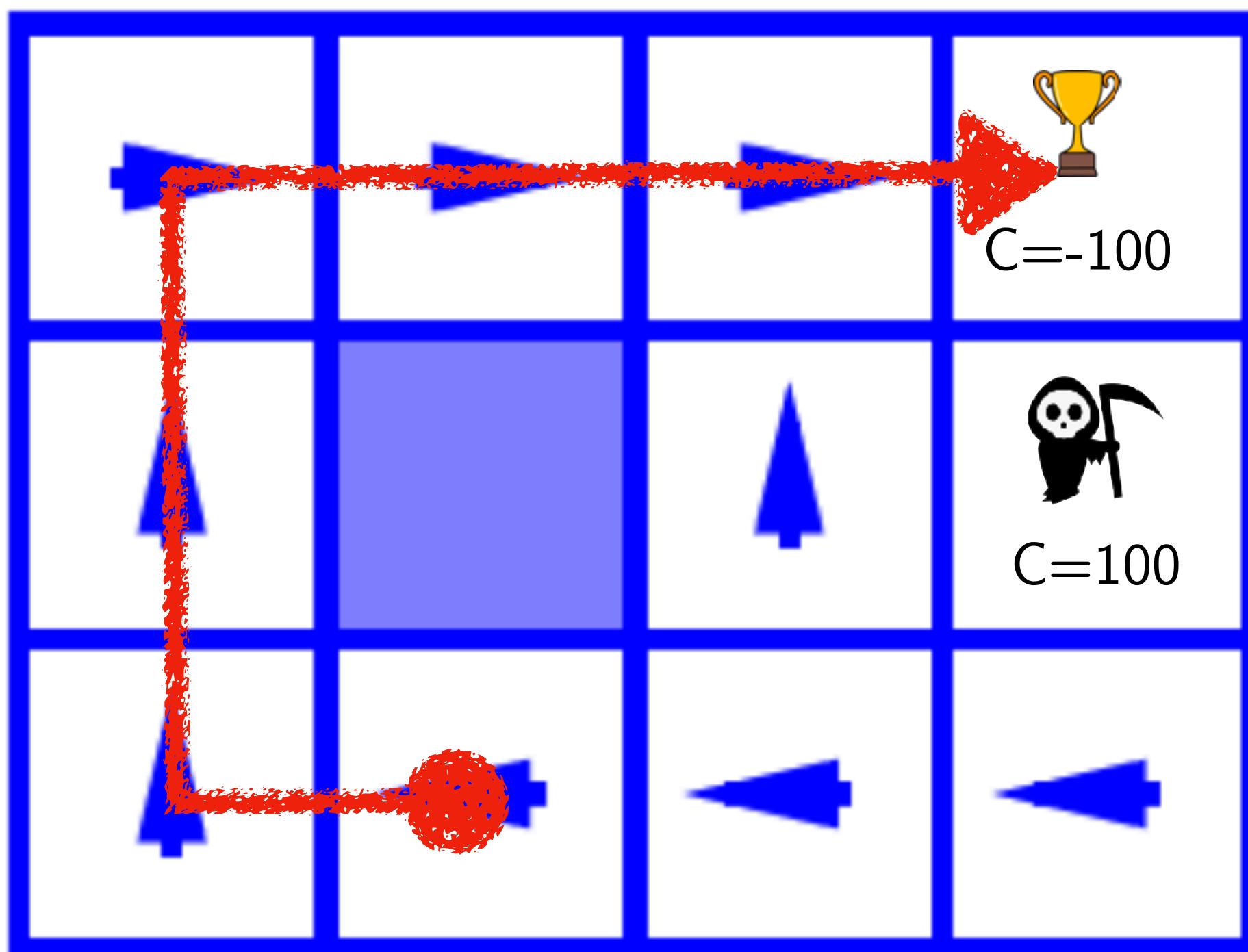


Policy: What action should I choose at any state?

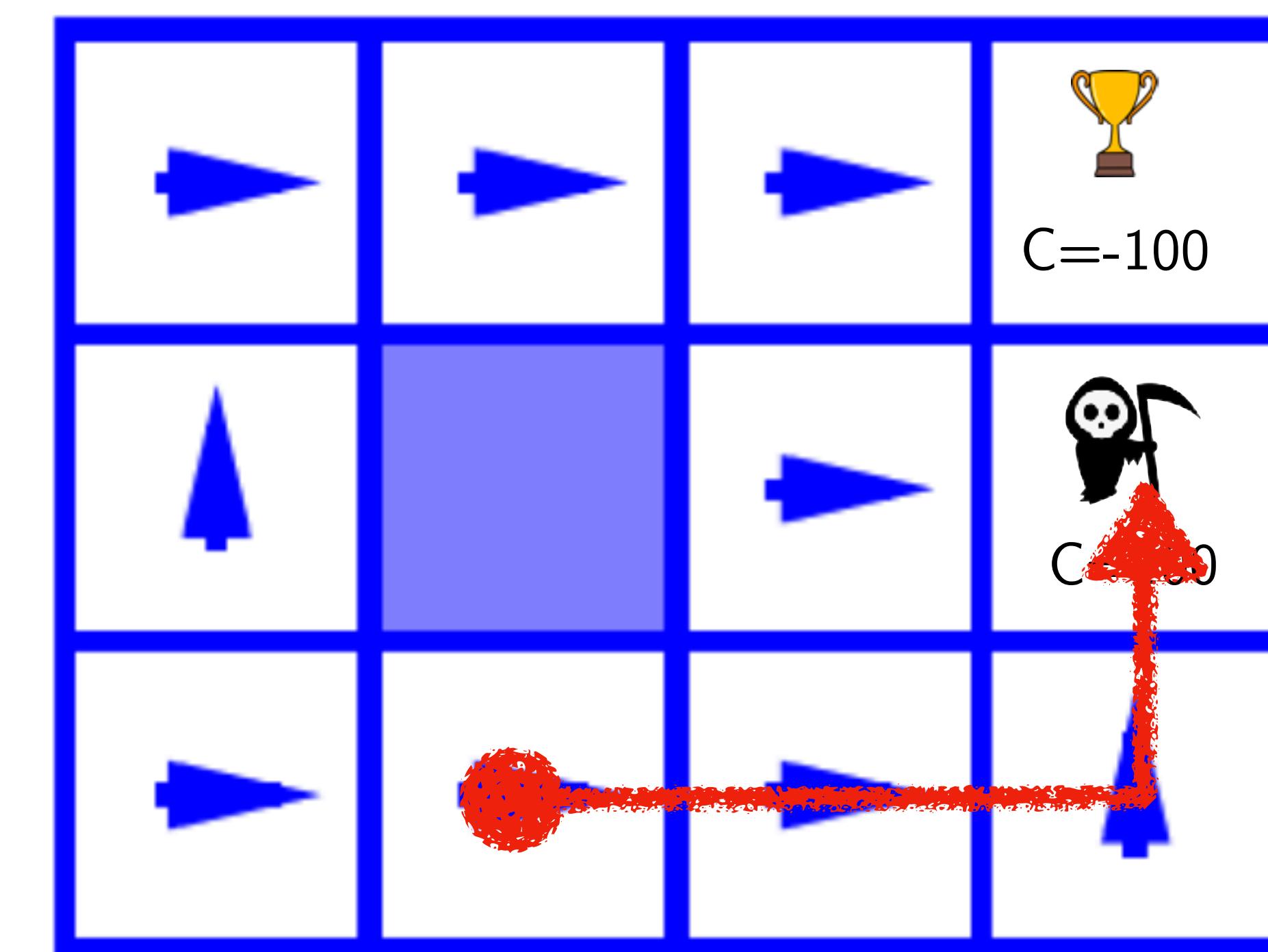
Can be deterministic or stochastic

What makes a policy *optimal*?

Which policy is better?



Policy π_1



Policy π_2

What makes a policy *optimal*?

$$\min_{\pi} \mathbb{E}_{\substack{s_0 \sim P(s_0) \\ a_t \sim \pi(s_t)}} \left[\sum_{t=0}^{T-1} c(s_t, a_t) \right]$$

(Search over Policies)

(Sum over all costs)

$$s_{t+1} \sim \mathcal{T}(s_t, a_t)$$

(Sample a start state,
then follow π till end
of episode)

One last piece ...

Which of the two outcomes do you prefer?

\$50 today

\$1 million
a 1000 days later

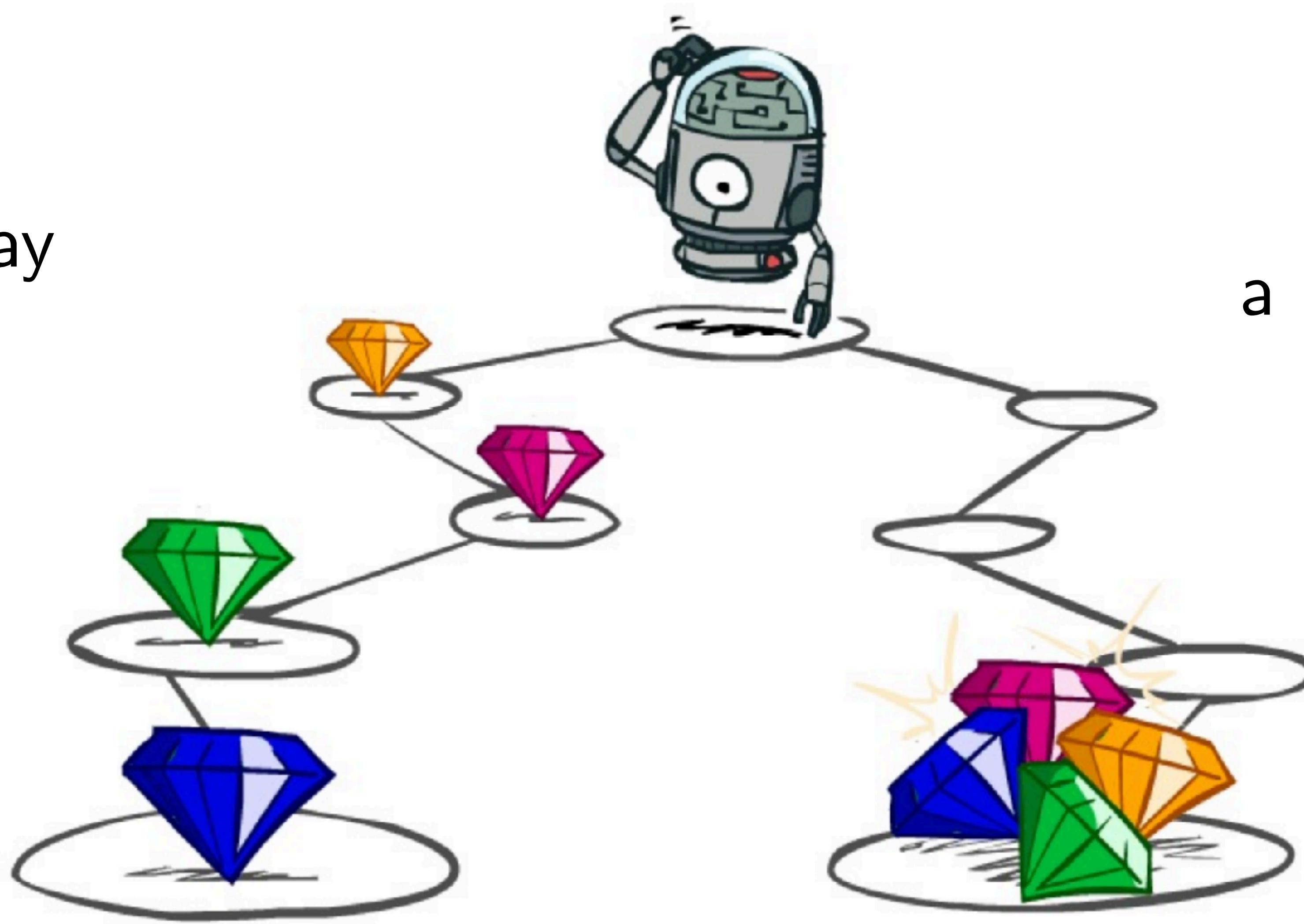


Image courtesy Dan Klein

Discount: Future rewards / costs matter less



1

Worth Now



γ

Worth Next Step



γ^2

Worth In Two Steps

At what discount value does it make sense to take
\$50 today than \$1million in 1000 days?

Image courtesy Dan Klein

What makes a policy *optimal*?

$$\min_{\pi} \mathbb{E}_{\begin{array}{l} s_0 \sim P(s_0) \\ a_t \sim \pi(s_t) \\ s_{t+1} \sim \mathcal{T}(s_t, a_t) \end{array}} \left[\sum_{t=0}^{T-1} \gamma^t c(s_t, a_t) \right]$$

(Search over Policies)

(Sample a start state, then follow π till end of episode)

(Discounted sum of costs)

How do we solve a MDP?

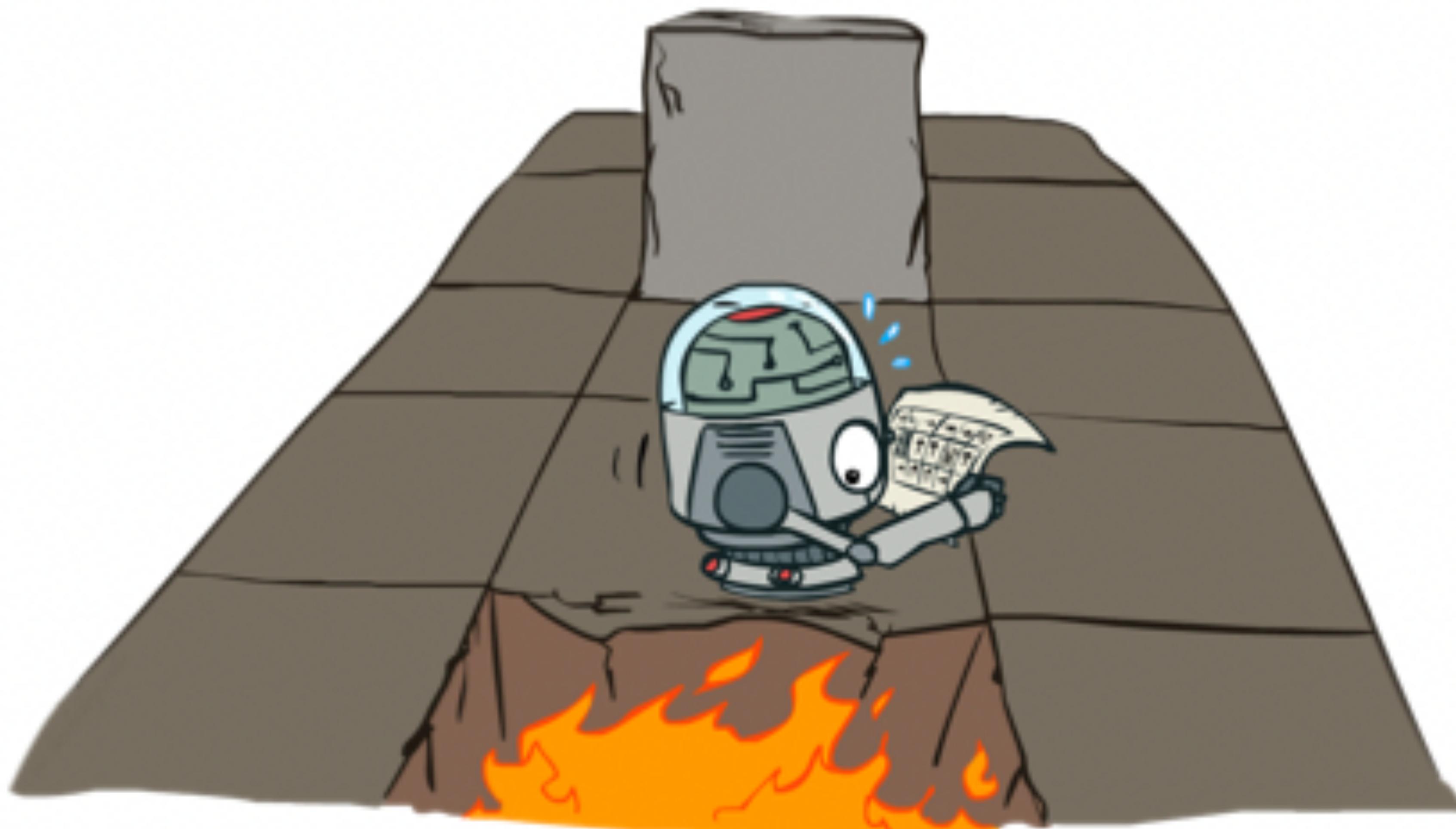


Image courtesy Dan Klein

Let's start with how NOT
to solve MDPs

What would brute force do?

$$\min_{\pi} \mathbb{E}_{\substack{s_0 \sim P(s_0) \\ a_t \sim \pi(s_t)}} \left[\sum_{t=0}^{T-1} \gamma^t c(s_t, a_t) \right]$$
$$s_{t+1} \sim \mathcal{T}(s_t, a_t)$$

How much work would brute force have to do?

What would brute force do?

$$\min_{\pi} \mathbb{E}_{\substack{s_0 \sim P(s_0) \\ a_t \sim \pi(s_t) \\ s_{t+1} \sim \mathcal{T}(s_t, a_t)}} \left[\sum_{t=0}^{T-1} \gamma^t c(s_t, a_t) \right]$$

1. Iterate over all possible policies
2. For every policy, evaluate the cost
3. Pick the best one

There are at most $(A^S)^T$ deterministic policies!!!!

Today's class

- What does it mean to solve a MDP?
- Bellman Equation
- Value Iteration

MDPs have a very special
structure

Introducing the “Value” Function

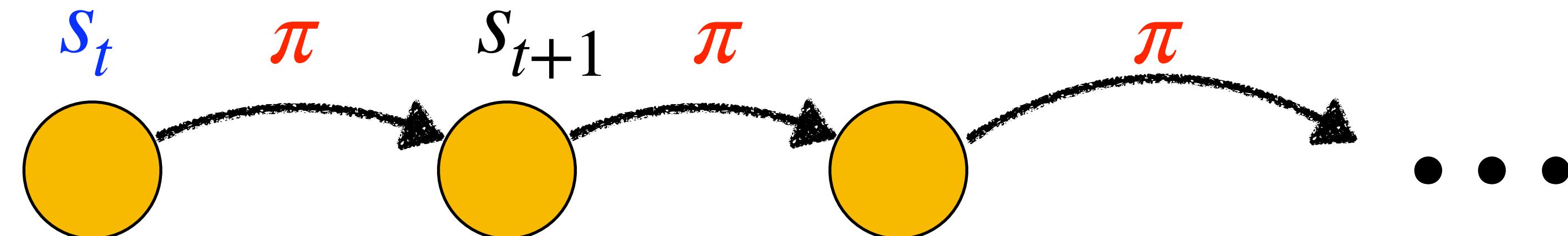
$$V^{\pi}(s_t)$$

Read this as: Value of a **policy** at a given **state and time**

Introducing the “Value” Function

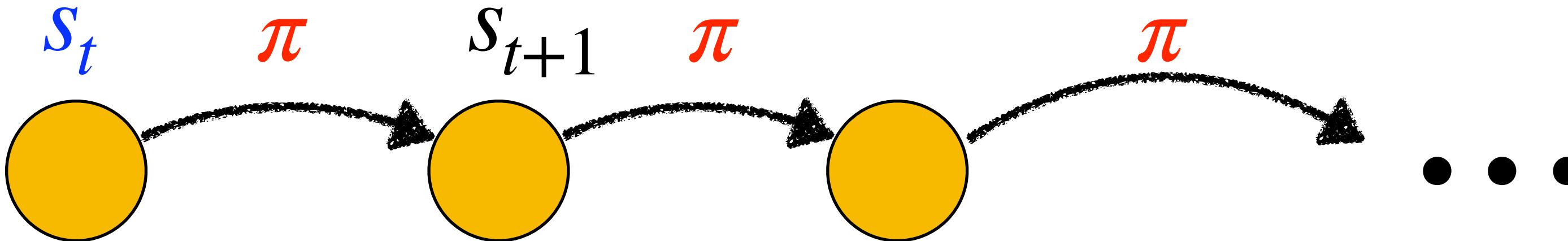
$$V^{\pi}(s_t)$$

Read this as: Value of a **policy** at a given **state and time**



$$V^{\pi}(s_t) = c_t + \gamma c_{t+1} + \gamma^2 c_{t+2} + \dots$$

Introducing the “Value” Function



$$V^{\pi}(s_t) = \mathbb{E}_{\pi} \left[\sum_{k=0}^{T-t-1} \gamma^k c(s_{t+k}, a_{t+k}) \mid s_t \right]$$

The Bellman Equation

$$V^{\pi}(s_t) = \mathbb{E}_{a_t \sim \pi(\cdot | s_t)} \left[c(s_t, \pi(s_t)) + \gamma \mathbb{E}_{s_{t+1} \sim \mathcal{T}(\cdot | s_t, a_t)} V^{\pi}(s_{t+1}) \right]$$

*Value of
current state*

Cost

*Value of
future state*

Exercise: Why is this true?

The Bellman Equation (for deterministic policies)

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

*Value of
current state*

Cost

*Value of
future state*

Optimal policy

$$\pi^* = \arg \min_{\pi} \mathbb{E}_{s_0} V^\pi(s_0)$$

Bellman Equation for the Optimal Policy

$$V^{\pi^*}(s_t) = \min_{a_t} [c(s_t, a_t) + \gamma \mathbb{E}_{s_{t+1}} V^{\pi^*}(s_{t+1})]$$

*Optimal
Value*

Cost

*Optimal
Value of
Next State*

Why is this true?

We use V^* to denote optimal value

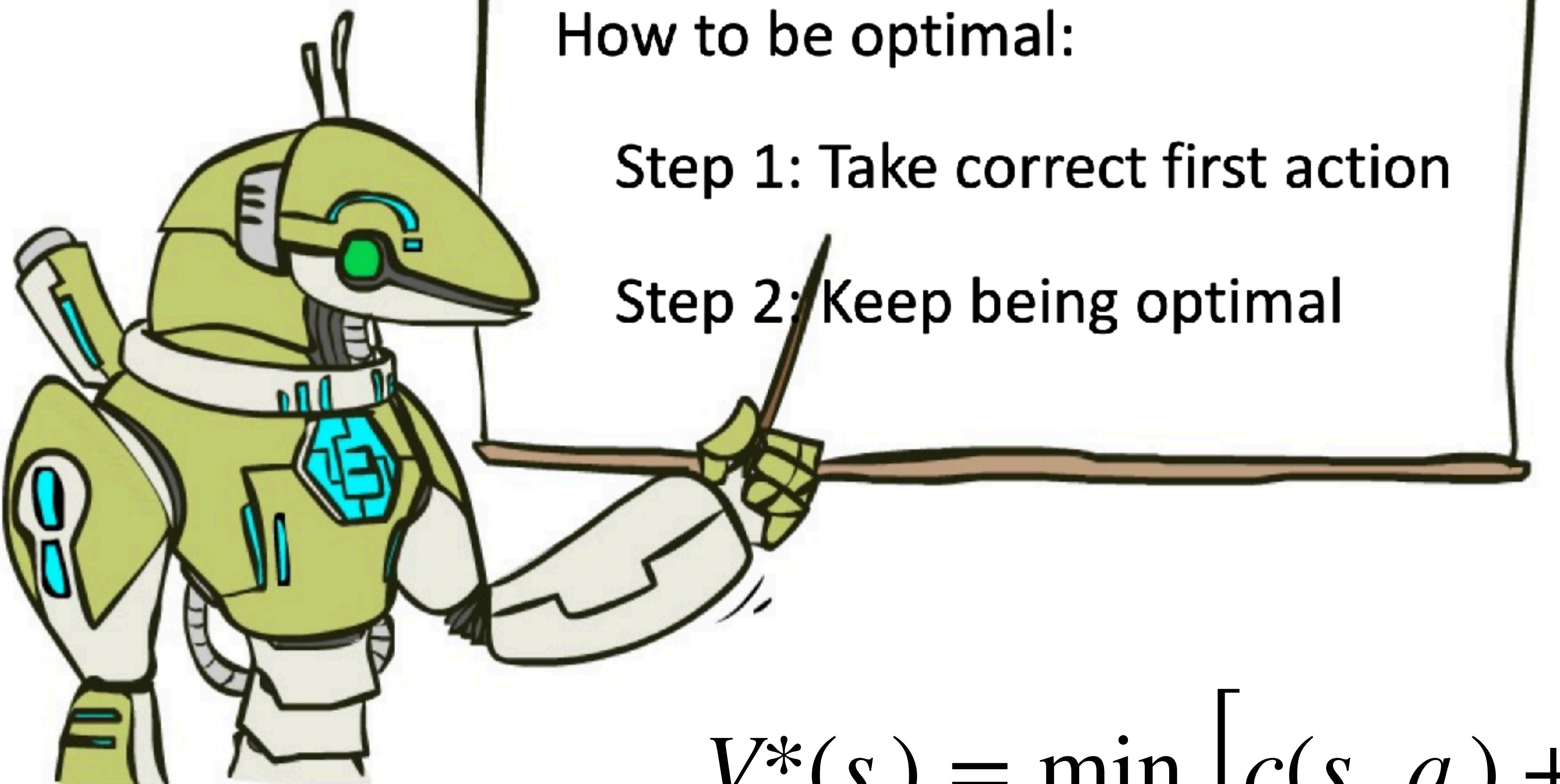
$$V^*(s_t) = \min_{a_t} \left[c(s_t, a_t) + \gamma \mathbb{E}_{s_{t+1}} V^*(s_{t+1}) \right]$$

*Optimal
Value*

Cost

*Optimal
Value of
Next State*

The Bellman Equation



Activity!



Today's class

- What does it mean to solve a MDP?
- Bellman Equation
- Value Iteration

Value Iteration

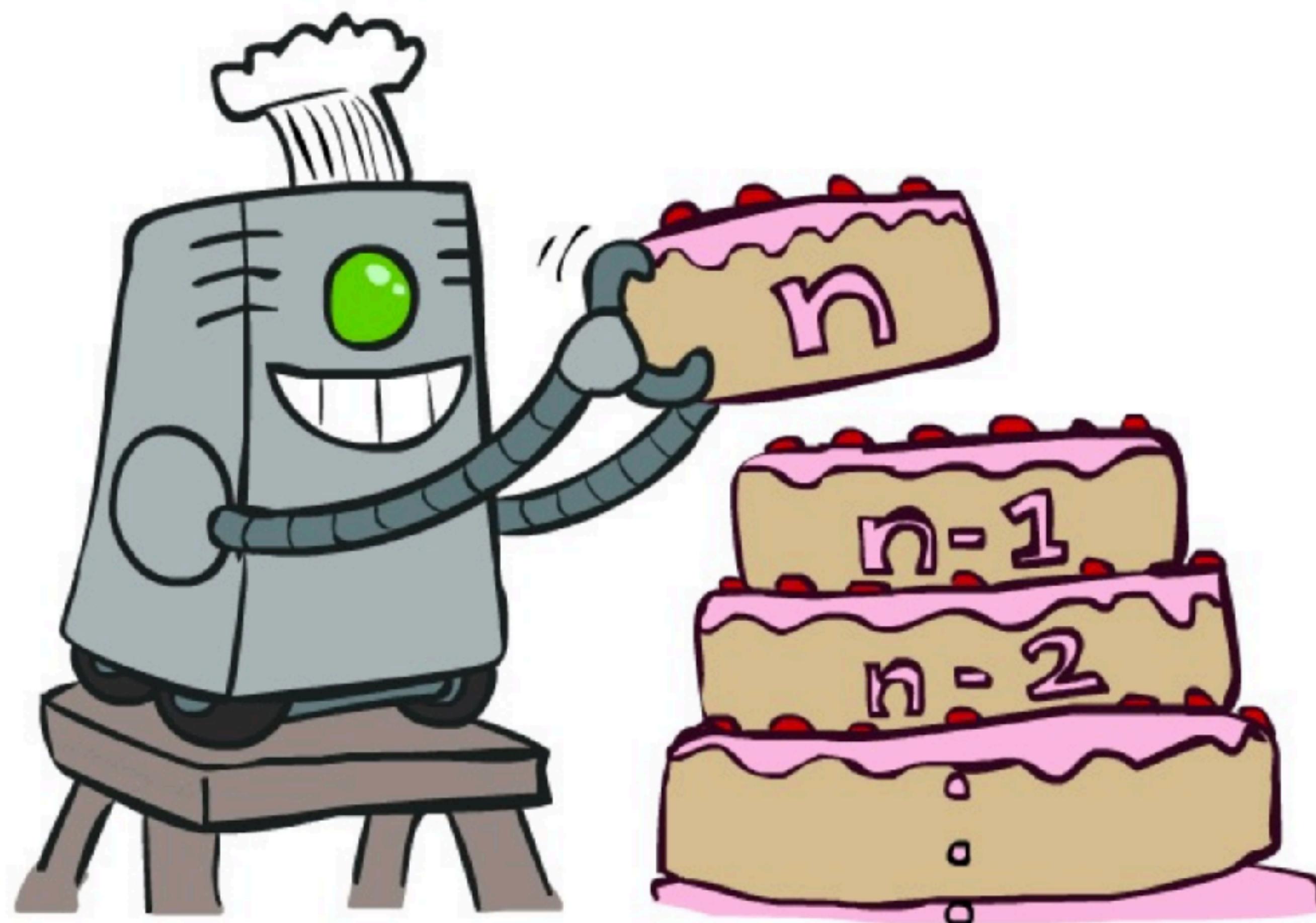
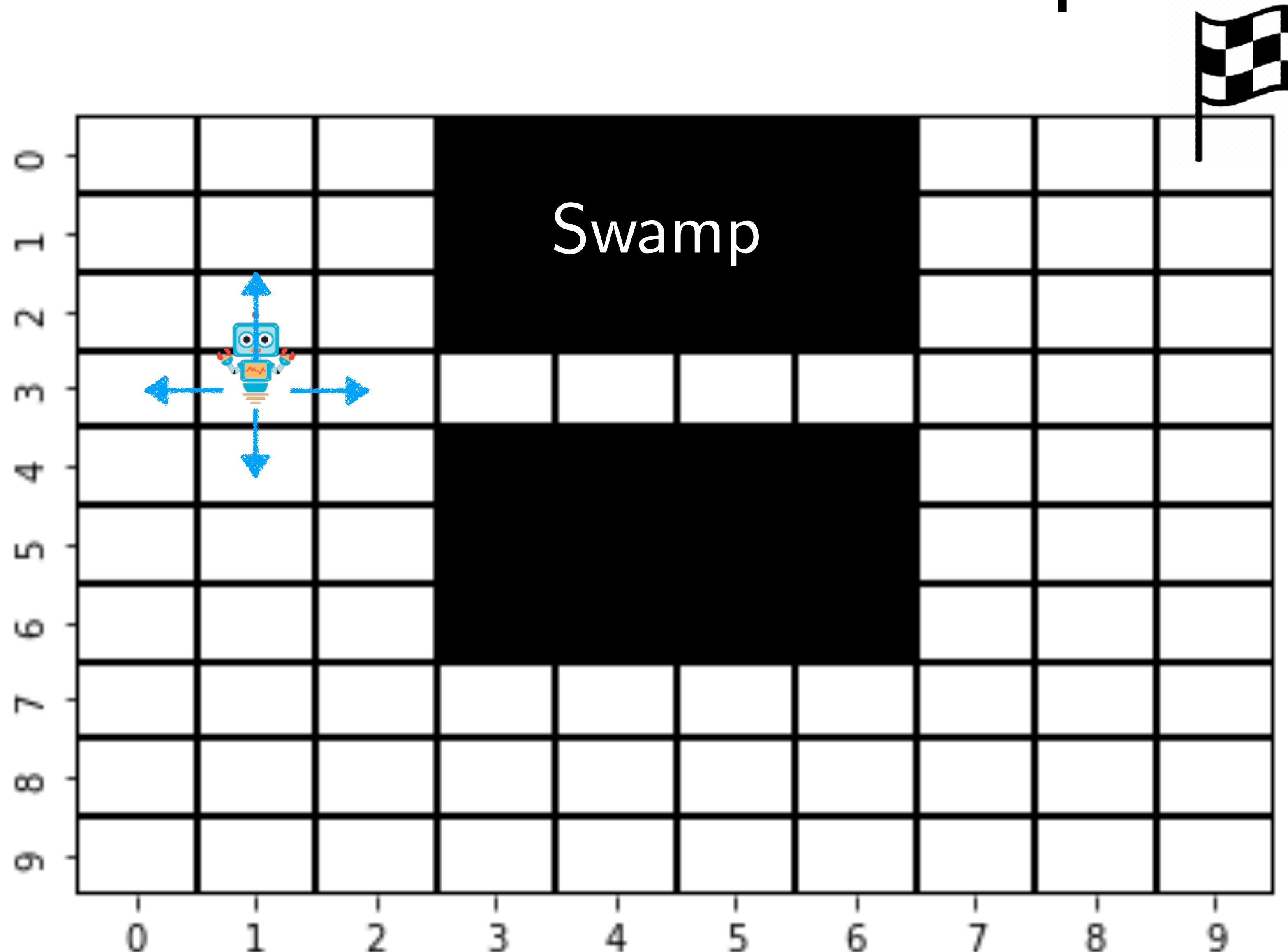


Image courtesy Dan Klein

Setup



$\langle S, A, C, \mathcal{T} \rangle$

- Two absorbing states:
Goal and Swamp
(can never leave)
- $c(s) = 0$ at the goal,
 $c(s) = 1$ everywhere else
- Transitions deterministic
- Time horizon $T = 30$
- Discount $\gamma = 1$

What is the optimal value at T-1?

Time: 29

0	1	1	1	1	1	1	1	1	1	0
1	1	1	1	1	1	1	1	1	1	1
2	1	1	1	1	1	1	1	1	1	1
3	1	1	1	1	1	1	1	1	1	1
4	1	1	1	1	1	1	1	1	1	1
5	1	1	1	1	1	1	1	1	1	1
6	1	1	1	1	1	1	1	1	1	1
7	1	1	1	1	1	1	1	1	1	1
8	1	1	1	1	1	1	1	1	1	1
9	1	1	1	1	1	1	1	1	1	1

0	x	x	x	x	x	x	x	x	x	0
1	x	x	x	x	x	x	x	x	x	1
2	x	x	x	x	x	x	x	x	x	2
3	x	x	x	x	x	x	x	x	x	3
4	x	x	x	x	x	x	x	x	x	4
5	x	x	x	x	x	x	x	x	x	5
6	x	x	x	x	x	x	x	x	x	6
7	x	x	x	x	x	x	x	x	x	7
8	x	x	x	x	x	x	x	x	x	8
9	x	x	x	x	x	x	x	x	x	9

$$V^*(s_{T-1}) = \min_a c(s_{T-1}, a)$$

$$\pi^*(s_{T-1}) = \arg \min_a c(s_{T-1}, a)$$

What is the optimal value at T-2?

Time: 28

0	2	2	2	2	2	2	2	2	1	0
1	2	2	2	2	2	2	2	2	2	1
2	2	2	2	2	2	2	2	2	2	2
3	2	2	2	2	2	2	2	2	2	2
4	2	2	2	2	2	2	2	2	2	2
5	2	2	2	2	2	2	2	2	2	2
6	2	2	2	2	2	2	2	2	2	2
7	2	2	2	2	2	2	2	2	2	2
8	2	2	2	2	2	2	2	2	2	2
9	2	2	2	2	2	2	2	2	2	2
	0	1	2	3	4	5	6	7	8	9

0	x	x	x	x	x	x	x	x	→	x
1	x	x	x	x	x	x	x	x	↑	
2	x	x	x	x	x	x	x	x	x	x
3	x	x	x	x	x	x	x	x	x	x
4	x	x	x	x	x	x	x	x	x	x
5	x	x	x	x	x	x	x	x	x	x
6	x	x	x	x	x	x	x	x	x	x
7	x	x	x	x	x	x	x	x	x	x
8	x	x	x	x	x	x	x	x	x	x
9	x	x	x	x	x	x	x	x	x	x
	0	1	2	3	4	5	6	7	8	9

$$V^*(s_{T-2}) = \min_a [c(s_{T-2}, a) + V^*(s_{T-1})]$$

$$\pi^*(s_{T-2}) = \arg \min_a [c(s_{T-2}, a) + V^*(s_{T-1})]$$

Dynamic Programming all the way!

Time: 16

0	14	14	13	14	14	14	14	2	1	0
1	14	13	12	14	14	14	14	3	2	1
2	13	12	11	14	14	14	14	4	3	2
3	12	11	10	9	8	7	6	5	4	3
4	13	12	11	14	14	14	14	6	5	4
5	14	13	12	14	14	14	14	7	6	5
6	14	14	13	14	14	14	14	8	7	6
7	14	14	14	13	12	11	10	9	8	7
8	14	14	14	14	13	12	11	10	9	8
9	14	14	14	14	14	13	12	11	10	9

0	x	x	↓	x	x	x	x	→	→	x
1	x	→	↓	x	x	x	x	→	→	↑
2	→	→	↓	x	x	x	x	→	→	↑
3	→	→	→	→	→	→	→	→	→	↑
4	→	→	↑	x	x	x	x	→	→	↑
5	x	→	↑	x	x	x	x	→	→	↑
6	x	x	↑	x	x	x	x	→	→	↑
7	x	x	x	→	→	→	→	→	→	↑
8	x	x	x	x	→	→	→	→	→	↑
9	x	x	x	x	→	→	→	→	→	↑

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

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

Value Iteration

Initialize value function at last time-step

$$V^*(s, T - 1) = \min_a c(s, a)$$

for $t = T - 2, \dots, 0$

Compute value function at time-step t

$$V^*(s, t) = \min_a \left[c(s, a) + \gamma \sum_{s'} \mathcal{T}(s' | s, a) V^*(s', t + 1) \right]$$

Quiz!



Computational complexity of value iteration

Initialize value function at last time-step

$$V^*(s, T - 1) = \min_a c(s, a)$$

for $t = T - 2, \dots, 0$

Compute value function at time-step t

$$V^*(s, t) = \min_a \left[c(s, a) + \gamma \sum_{s'} \mathcal{T}(s' | s, a) V^*(s', t + 1) \right]$$

When poll is active respond at PollEv.com/sc2582



Why is the optimal policy a function of time?

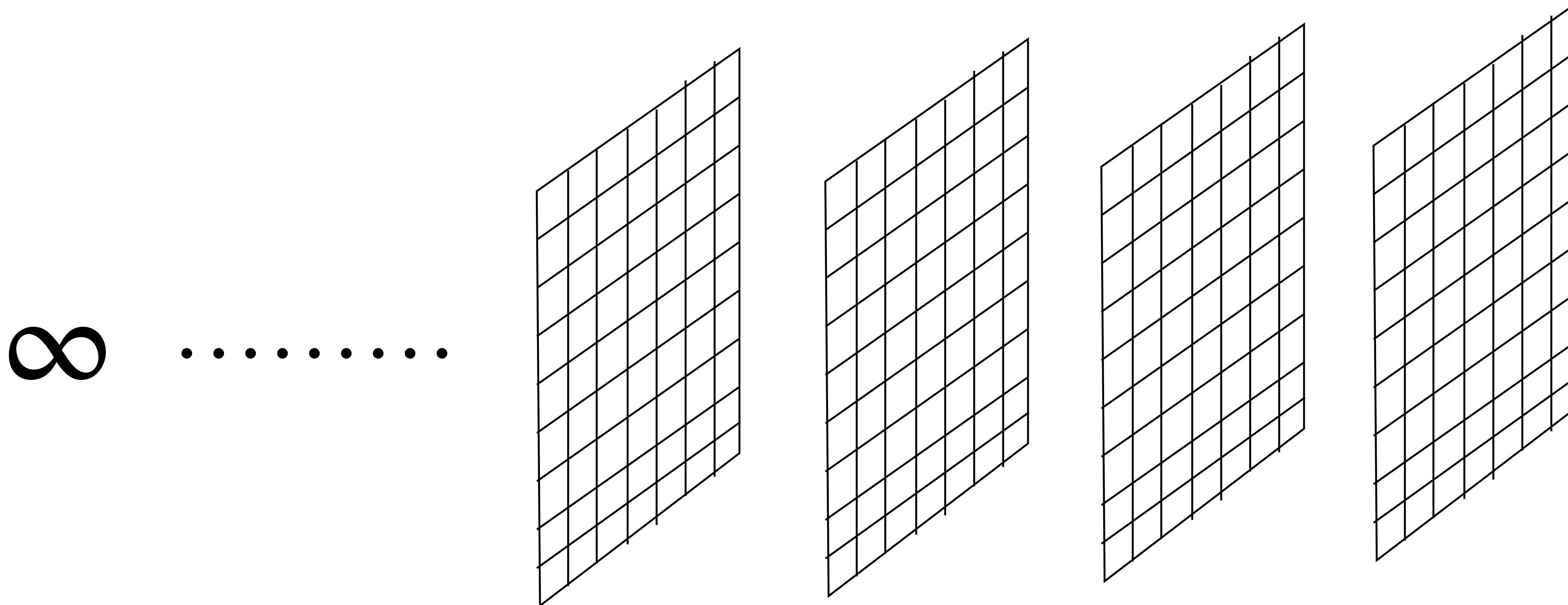


Pulling the goalie
when you
are losing and have
seconds left ..

What happens when horizon is infinity?

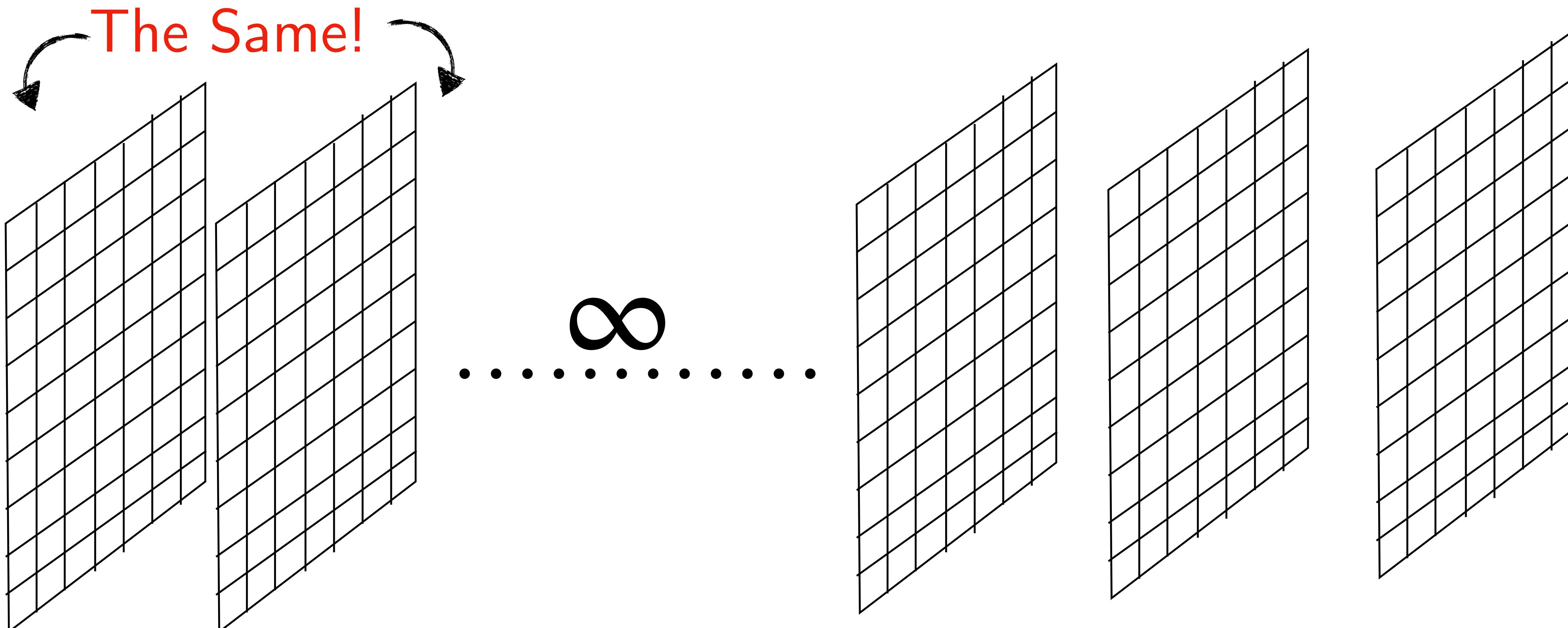


What happens when horizon is infinity?



$$V^{\pi^*}(s_t) = \min_{a_t} [c(s_t, a_t) + \gamma \mathbb{E}_{s_{t+1}} V^{\pi^*}(s_{t+1})]$$

Value Function Converges! (For $\gamma < 1$)



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

Infinite Horizon Value Iteration

Initialize with some value function $V^*(s)$

Repeat forever

Update values

$$V^*(s) = \min_a \left[c(s, a) + \gamma \sum_{s'} \mathcal{T}(s' | s, a) V^*(s') \right]$$

Today's class

- What does it mean to solve a MDP?
- Bellman Equation
- Value Iteration