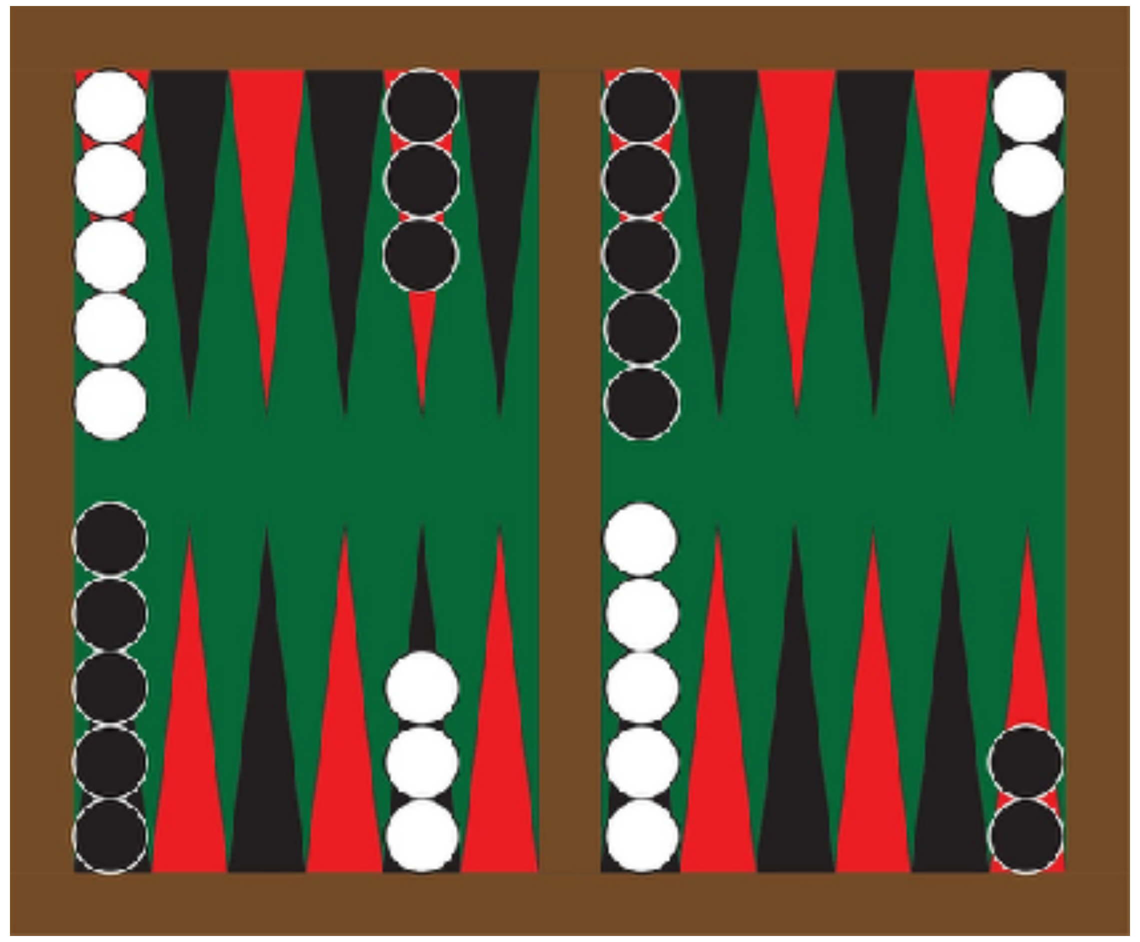
Statistical Foundations of Reinforcement Learning: I

COLT 2021

Akshay Krishnamurthy (MSR, akshaykr@microsoft.com)

Wen Sun (Cornell, ws455@cornell.edu)

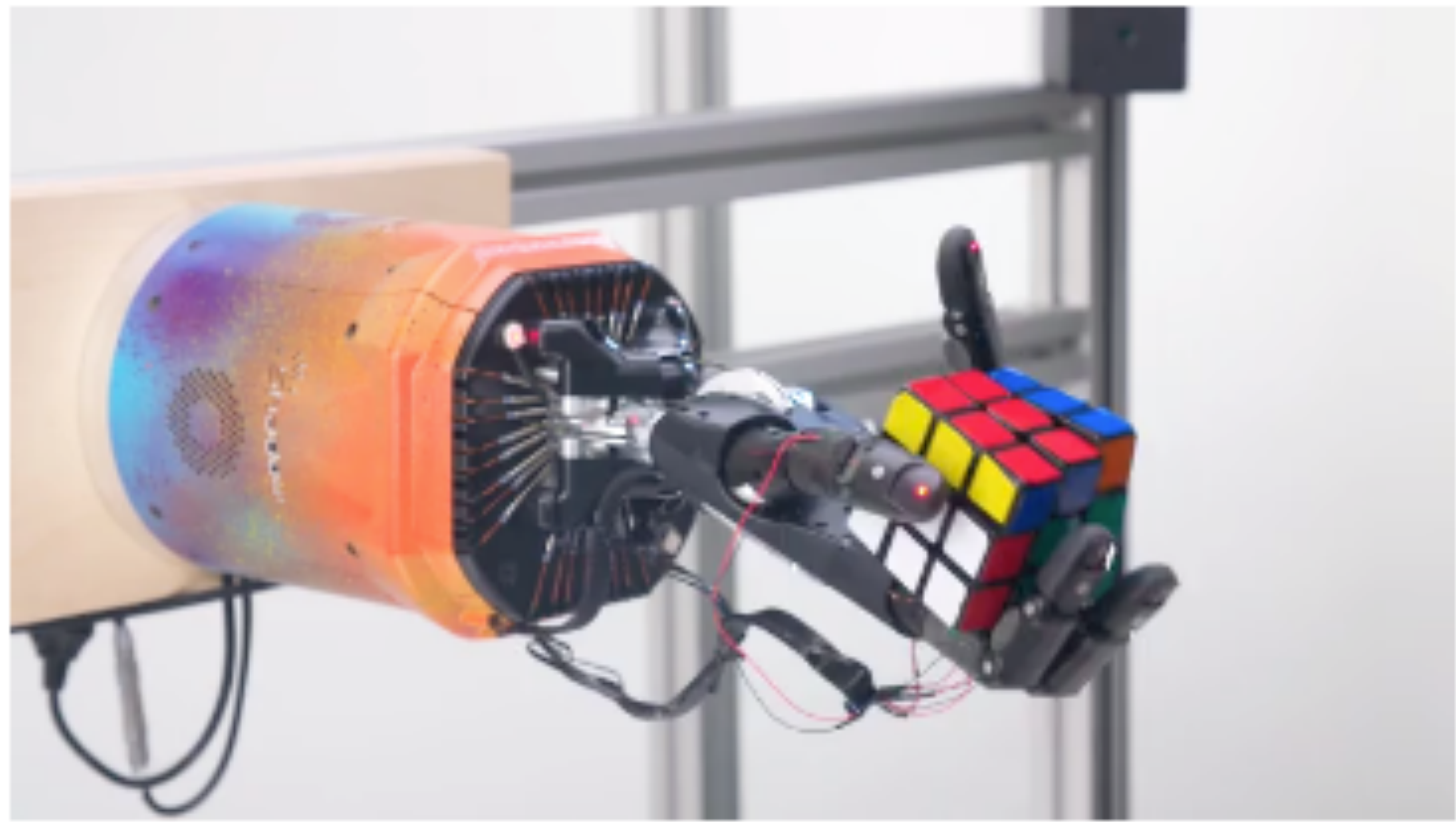
Reinforcement Learning: Motivation and empirical progress



TD Gammon [Tesauro]

DeepMind Starcraft [Vinyals et.al]

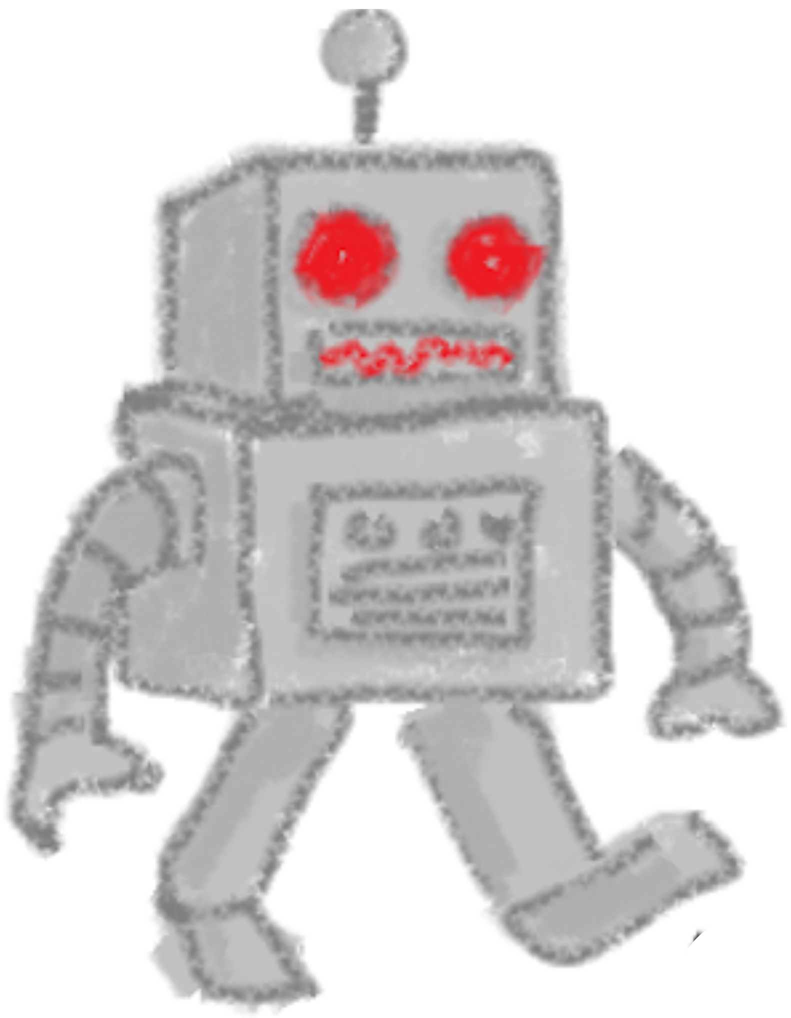
Stratospheric balloons [Bellemare et.al]



OpenAI Dexterous manipulation [Akkaya et.al]

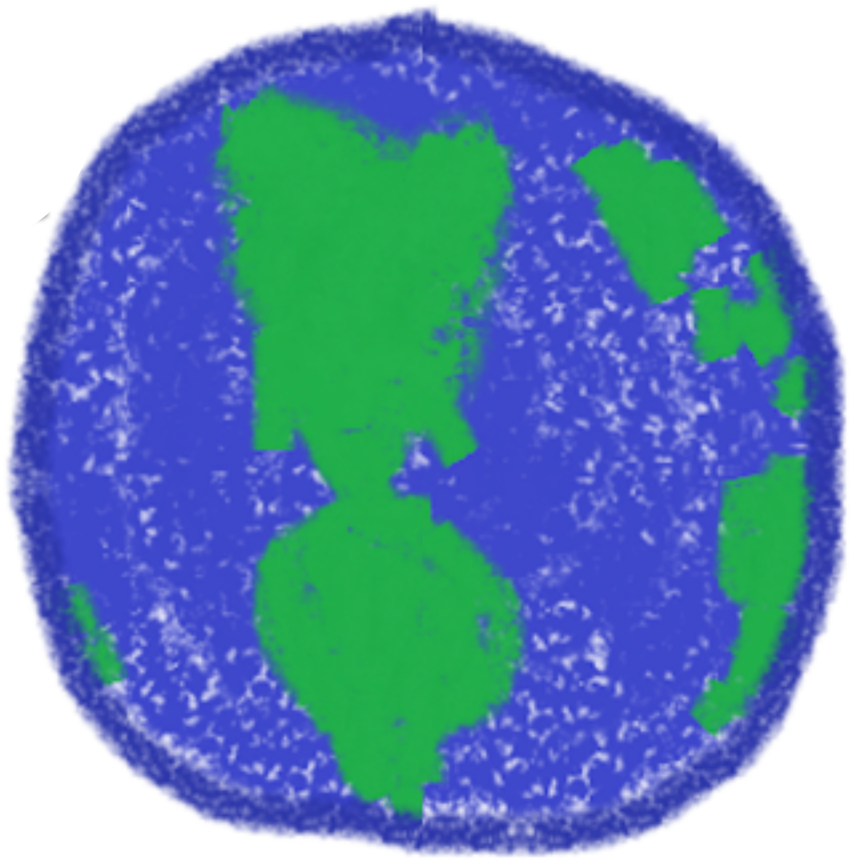
What is reinforcement learning?

Learning Agent

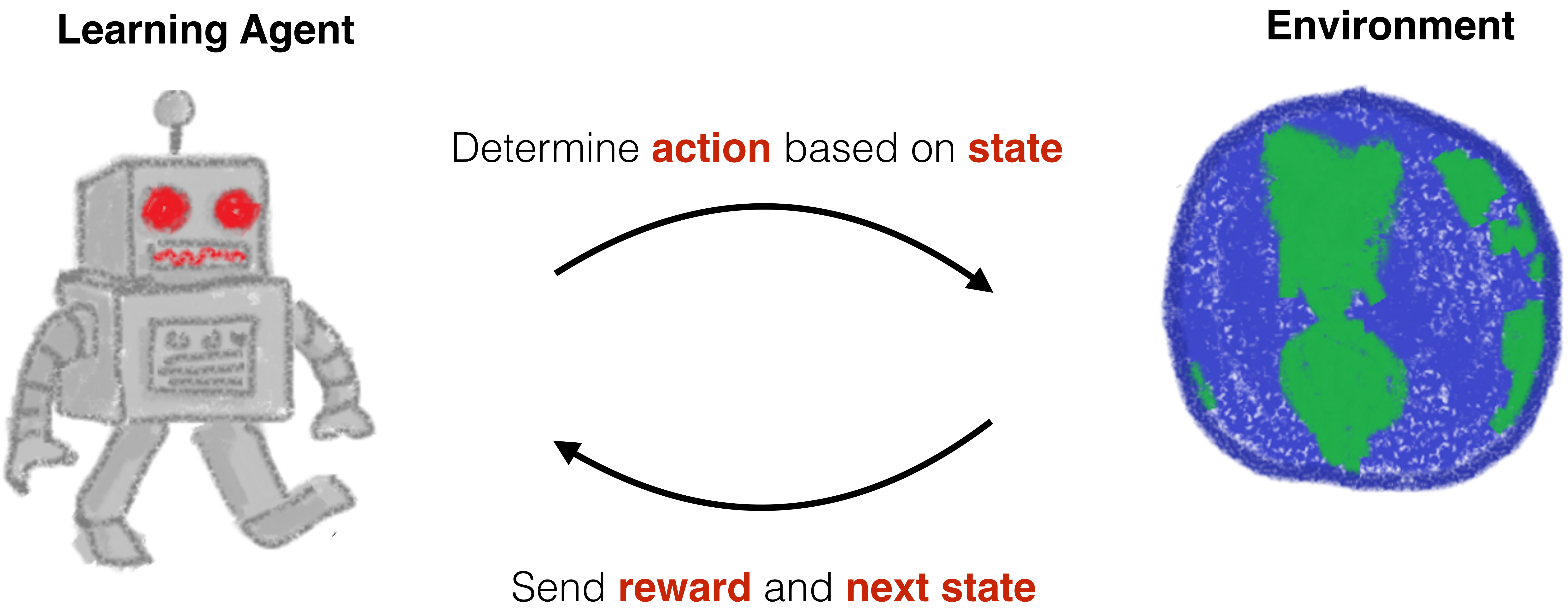


Determine **action** based on **state**

Environment

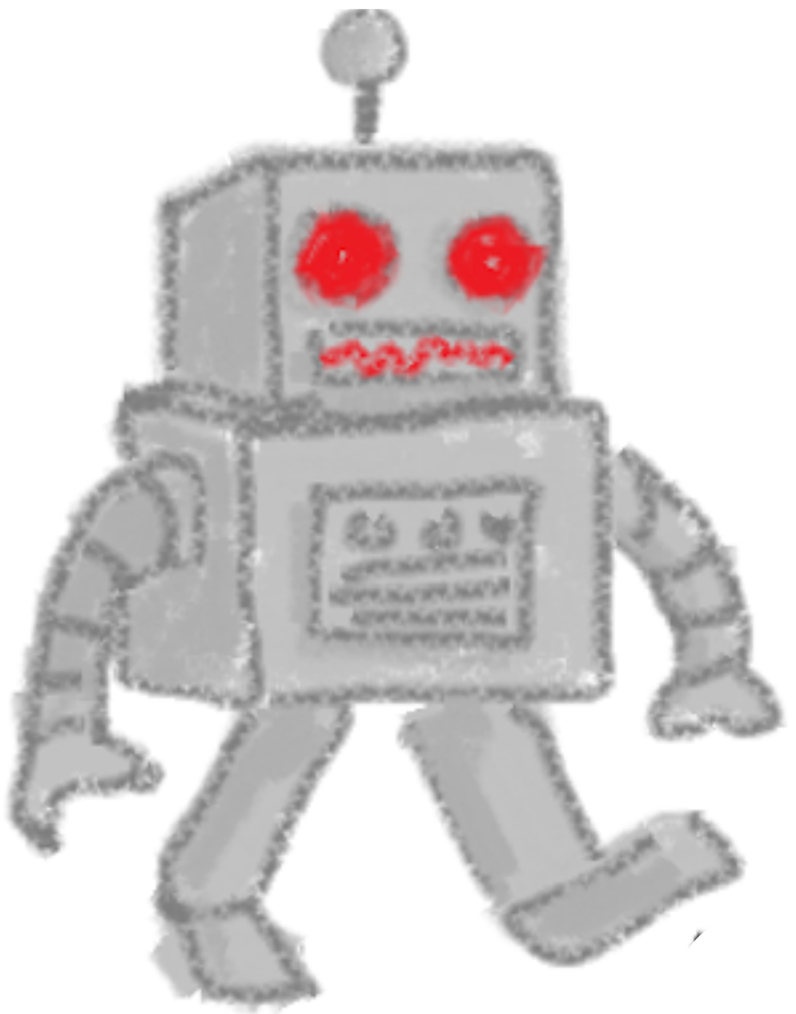


What is reinforcement learning?

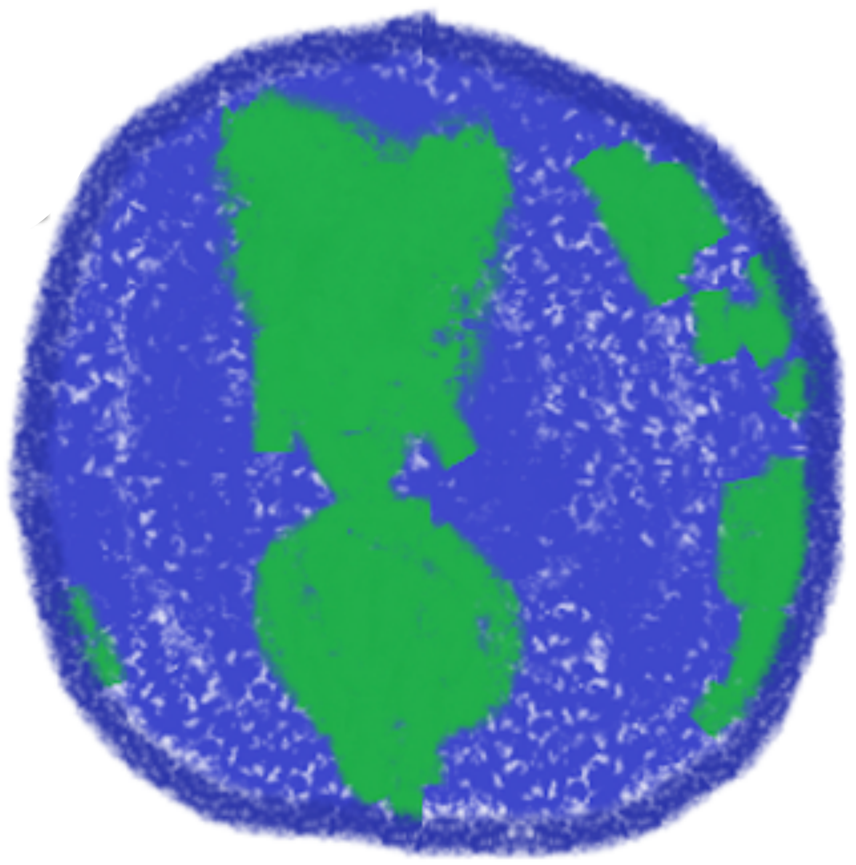


What is reinforcement learning?

Learning Agent



Environment



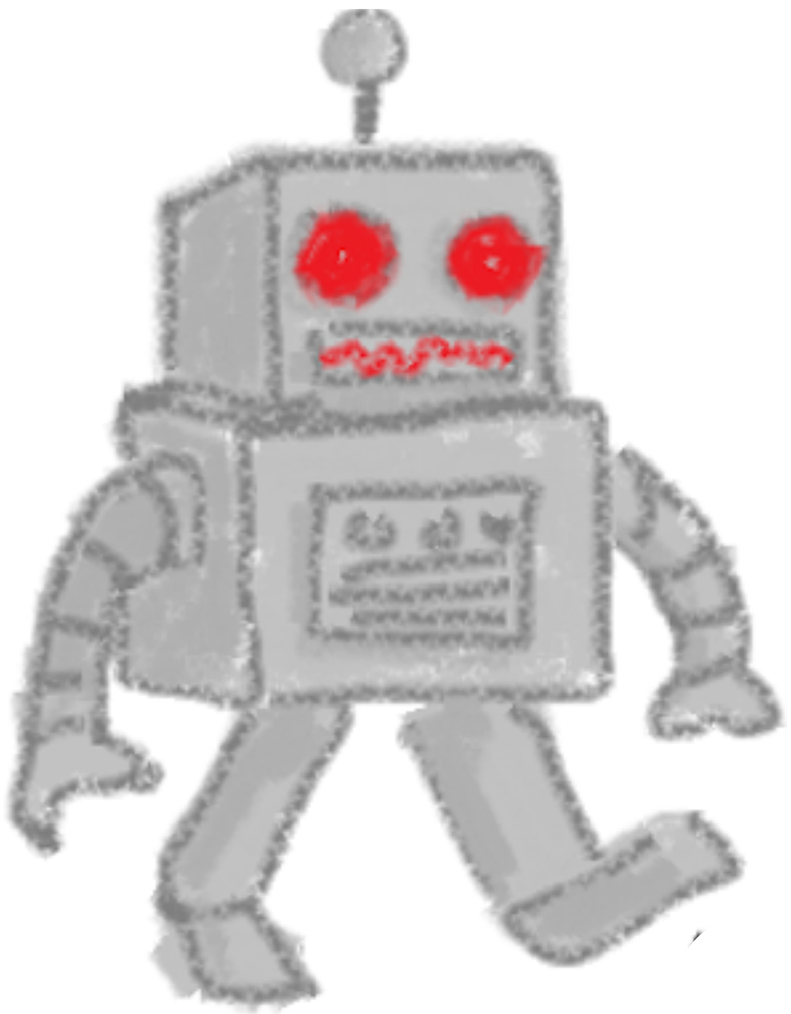
Determine **action** based on **state**

Multiple Steps

Send **reward** and **next state**

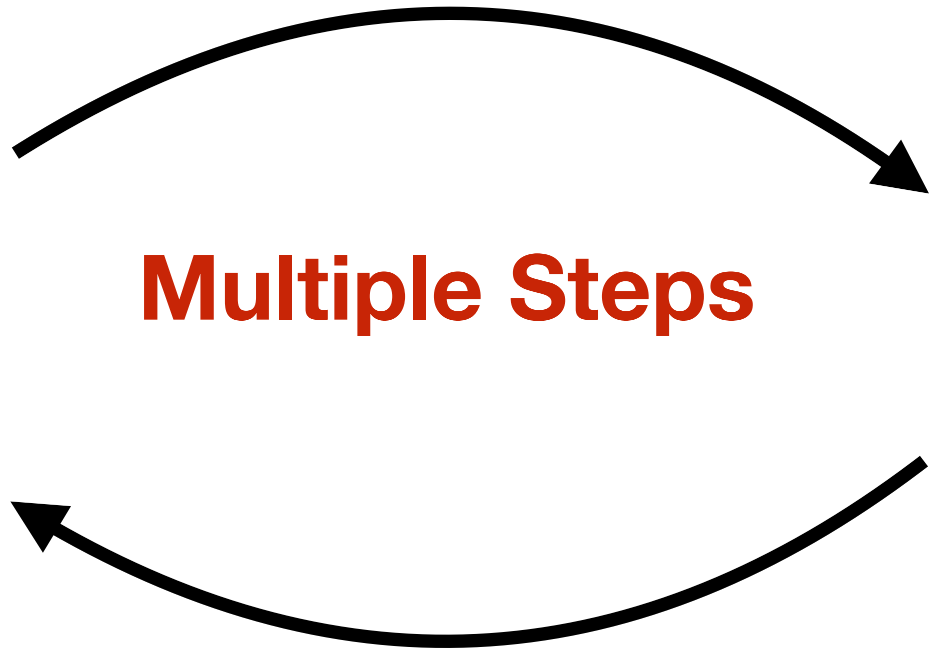
What is reinforcement learning?

Learning Agent



Environment

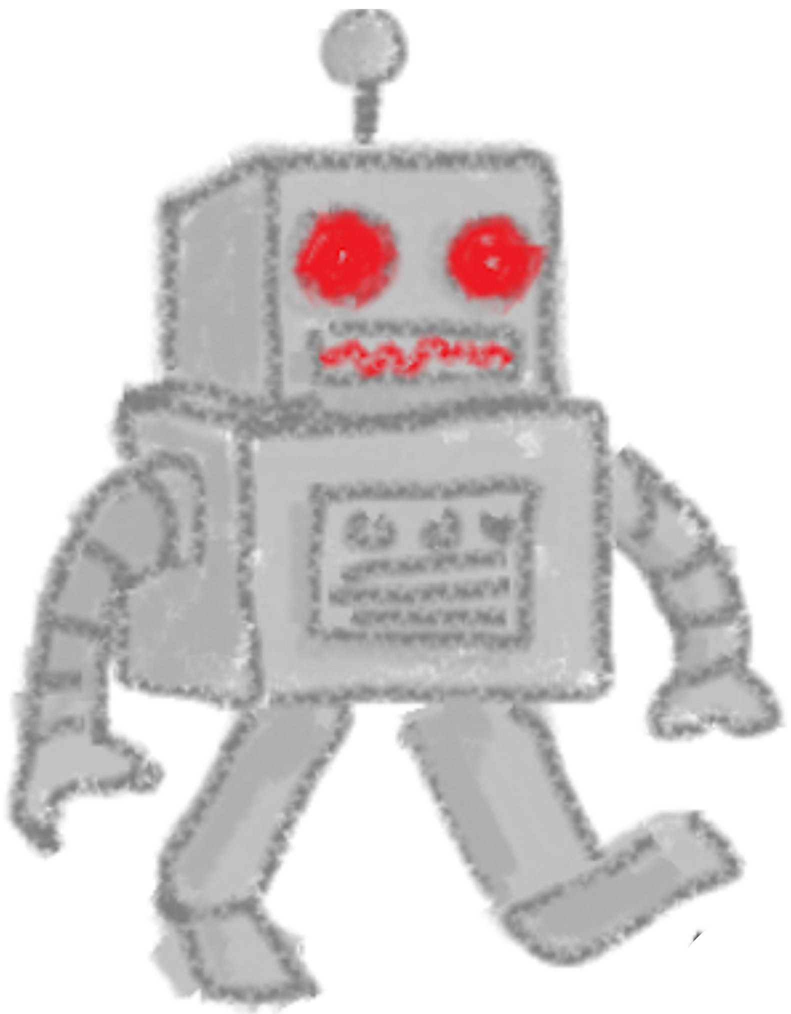
Determine **action** based on **state**



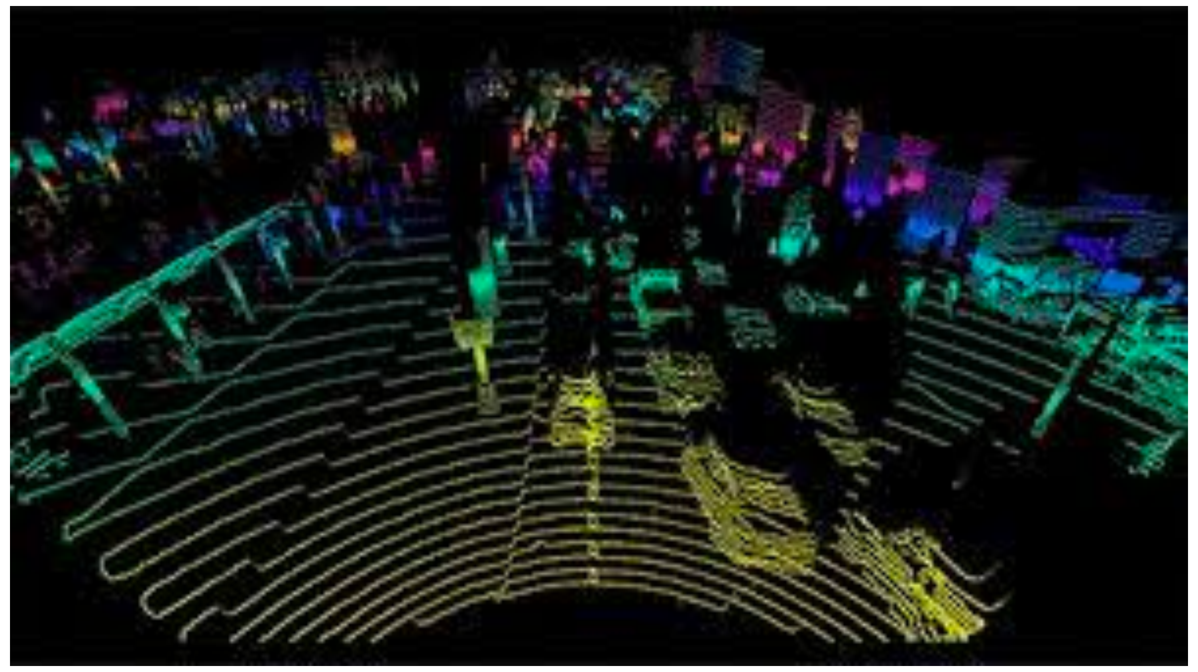
Send **reward** and **next state**

What is reinforcement learning?

Learning Agent



Environment



Determine **action** based on **state**

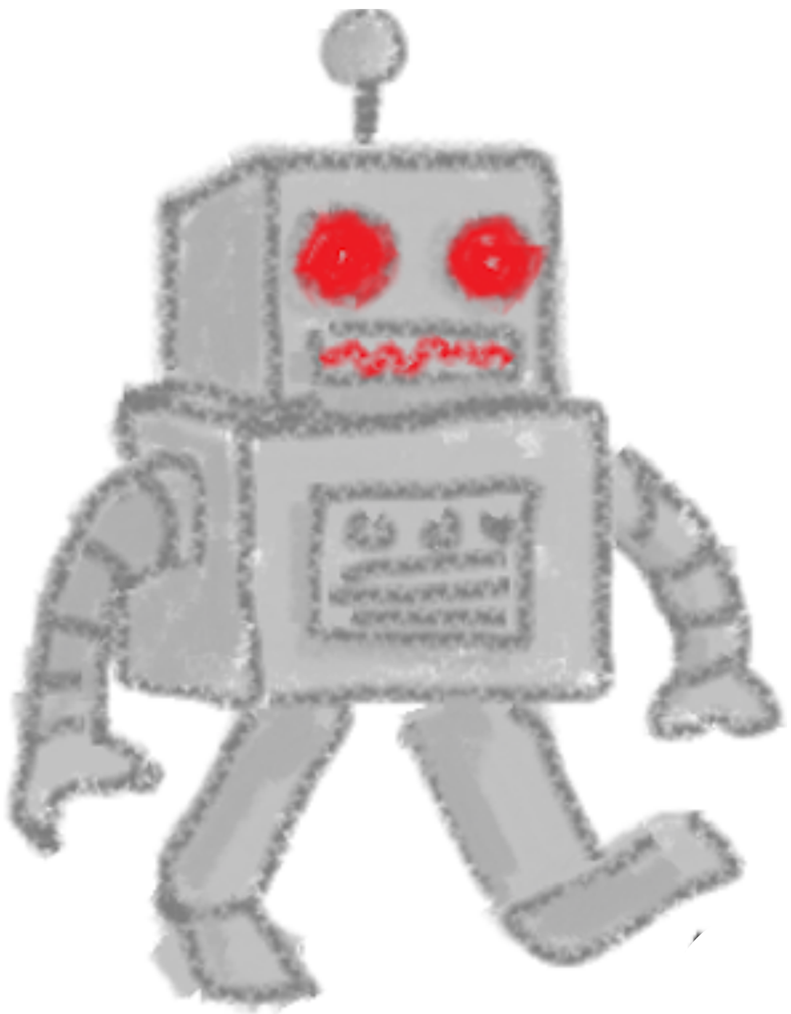
Multiple Steps

Send **reward** and **next state**

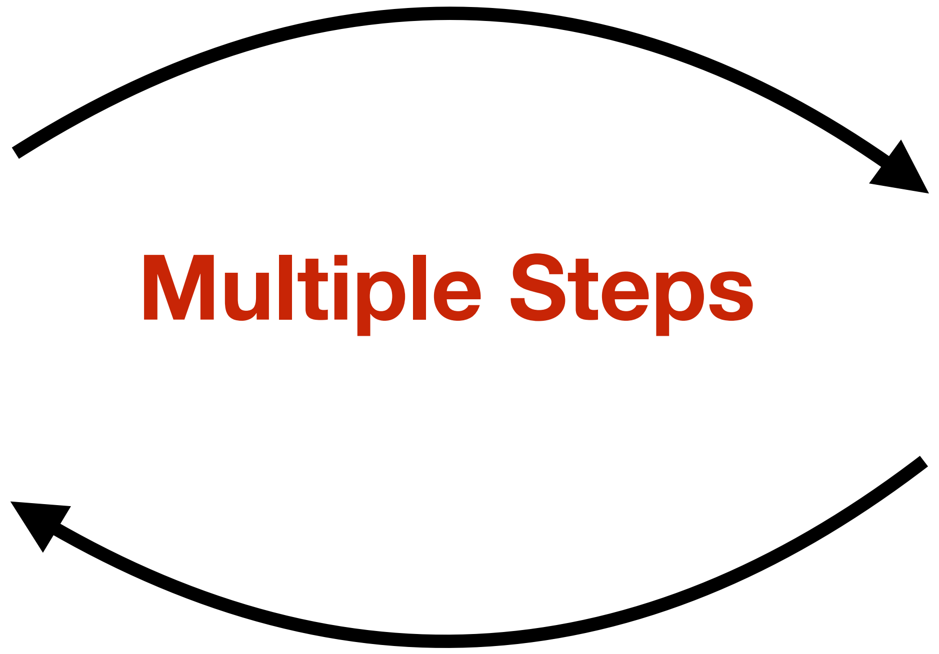
What is reinforcement learning?

Learning Agent

Environment

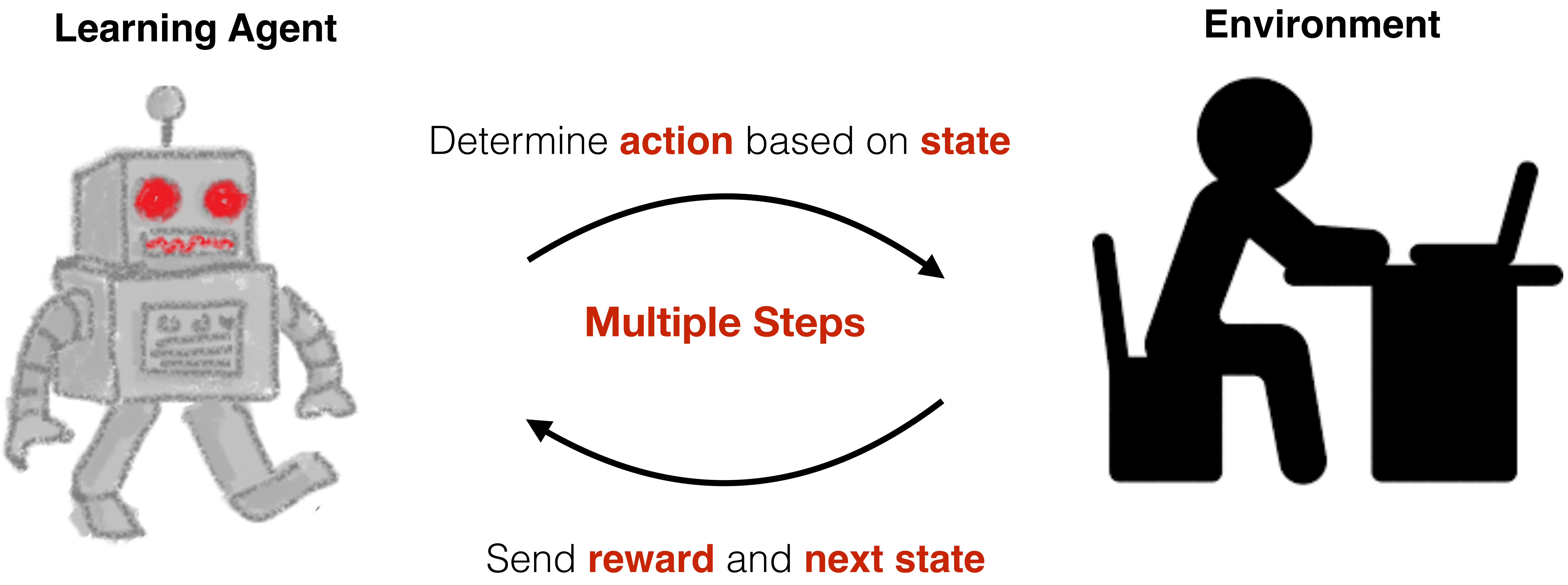


Determine **action** based on **state**

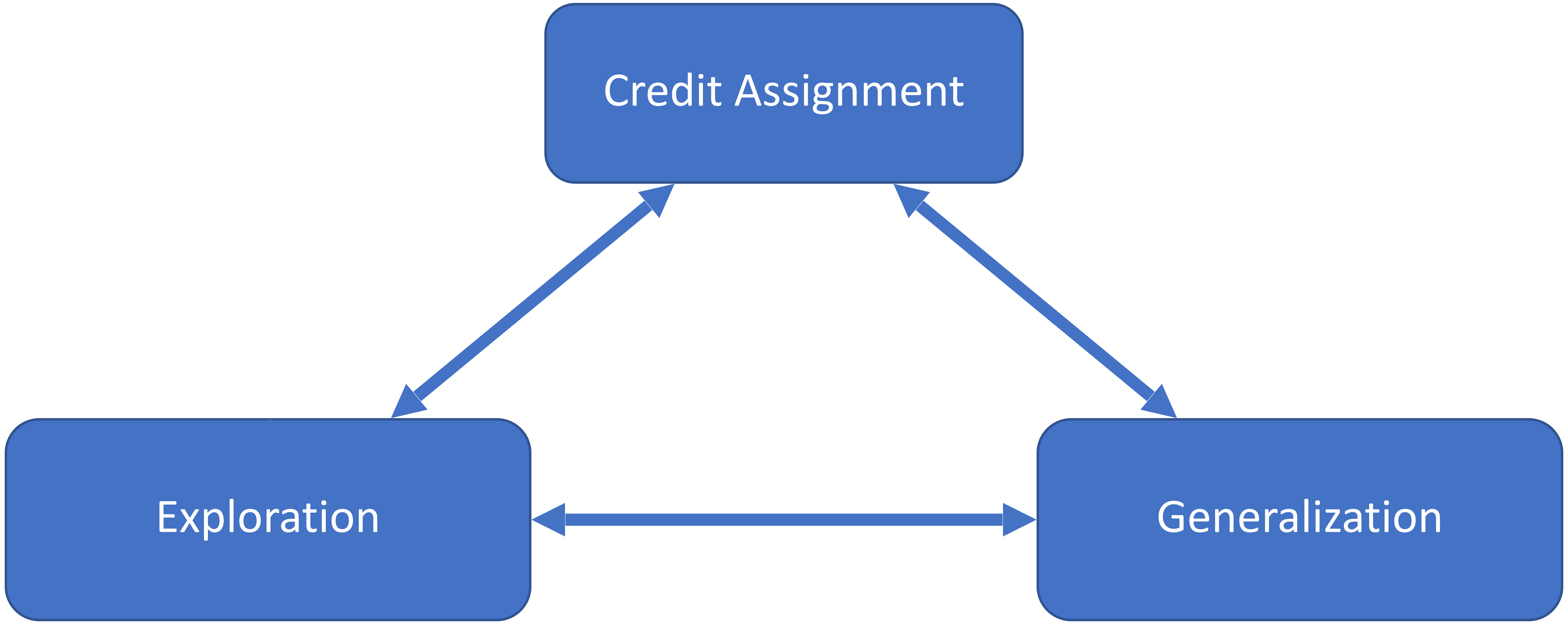


Send **reward** and **next state**

What is reinforcement learning?

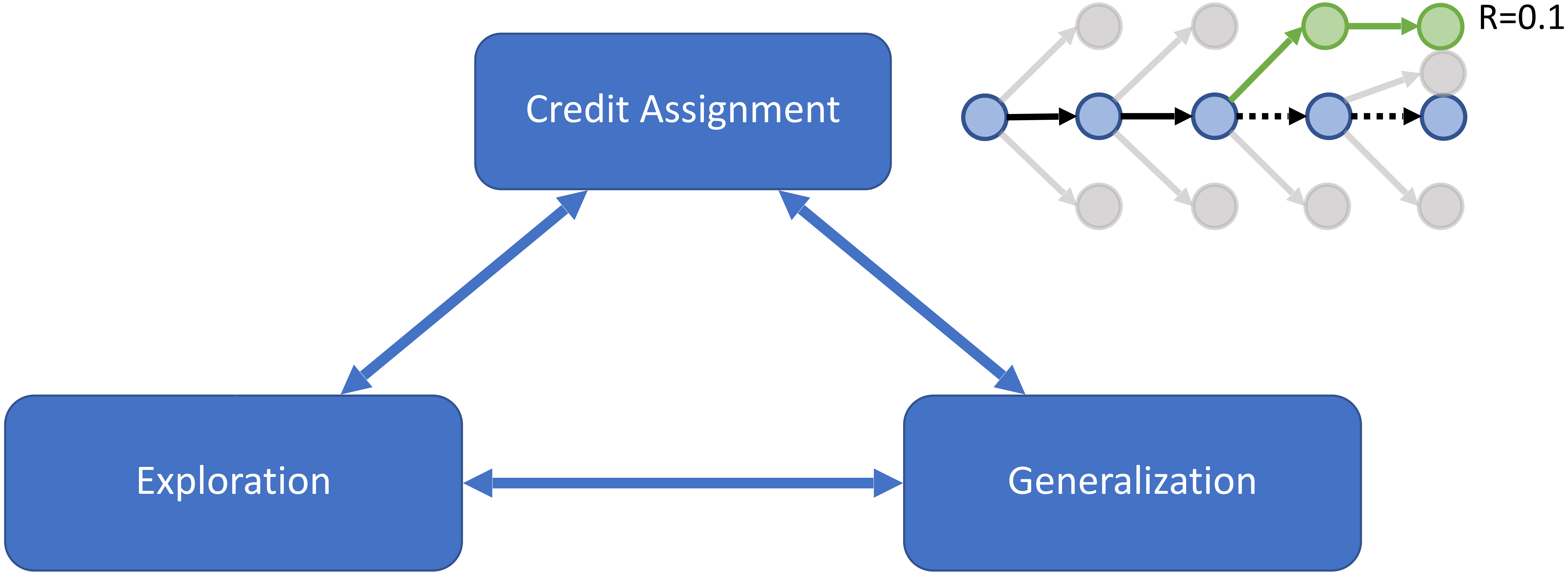


Why is RL hard?



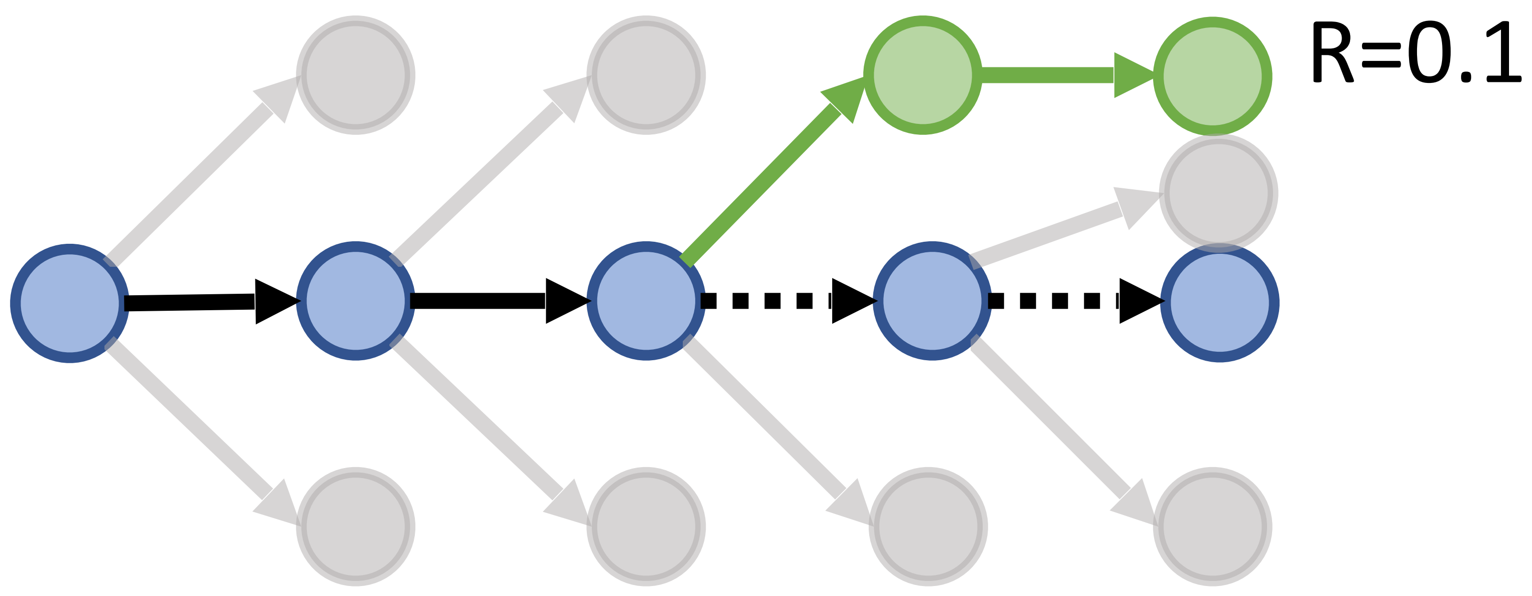
Why is RL hard?

Policy search methods;
Structured prediction;
Imitation learning.



Why is RL hard?

Policy search methods;
Structured prediction;
Imitation learning.

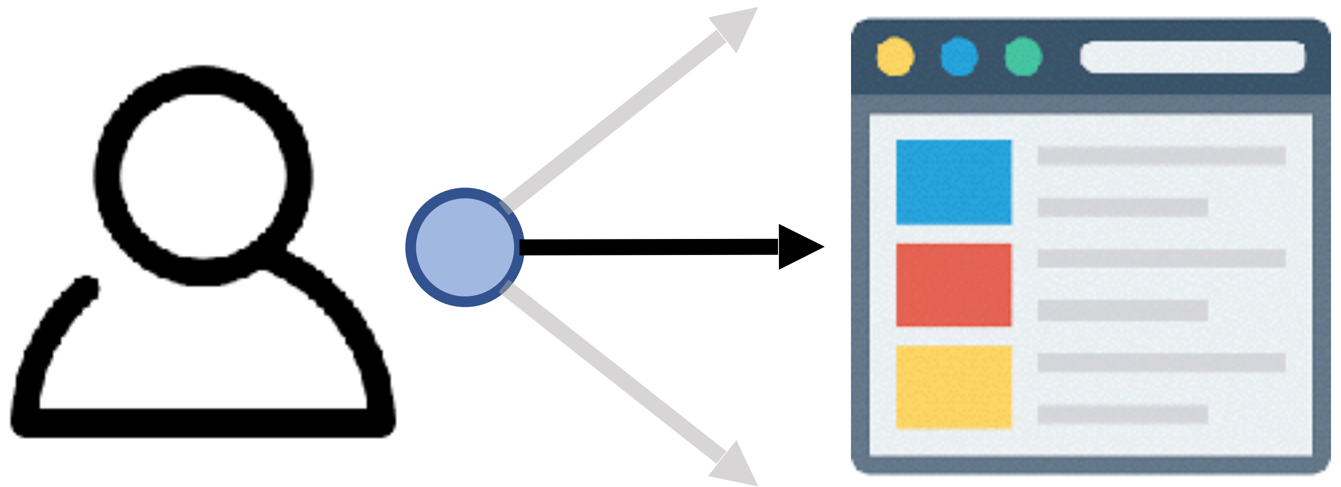


Credit Assignment

Exploration

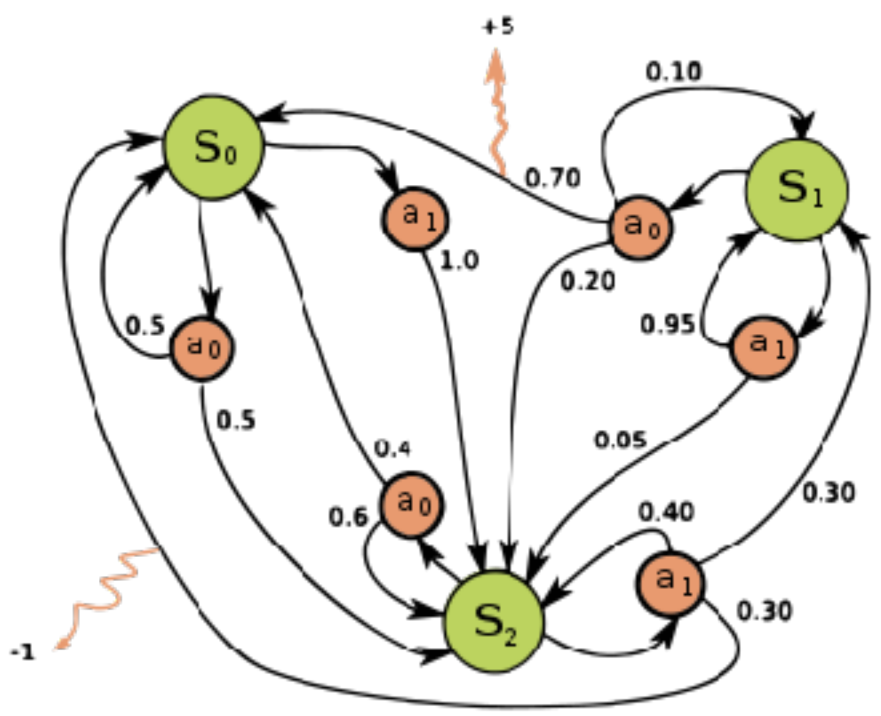
Generalization

Contextual Bandits

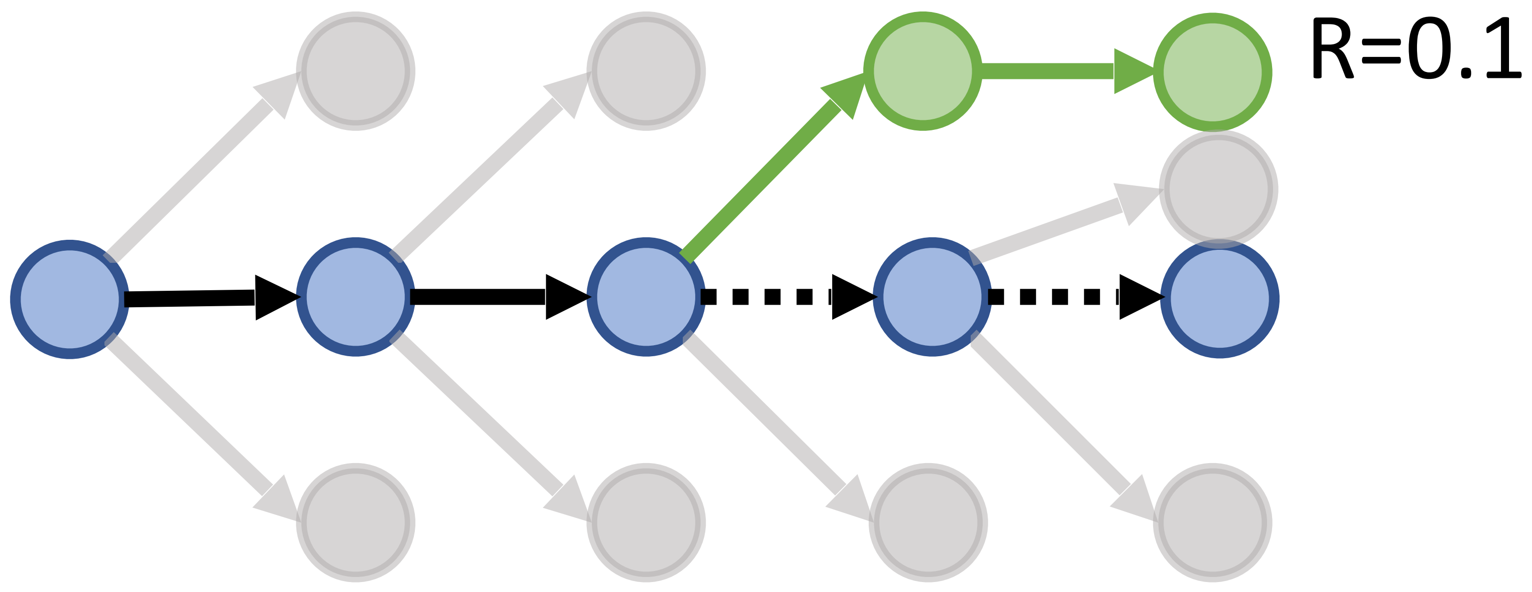


Policy search methods;
Structured prediction;
Imitation learning.

Why is RL hard?



Tabular RL

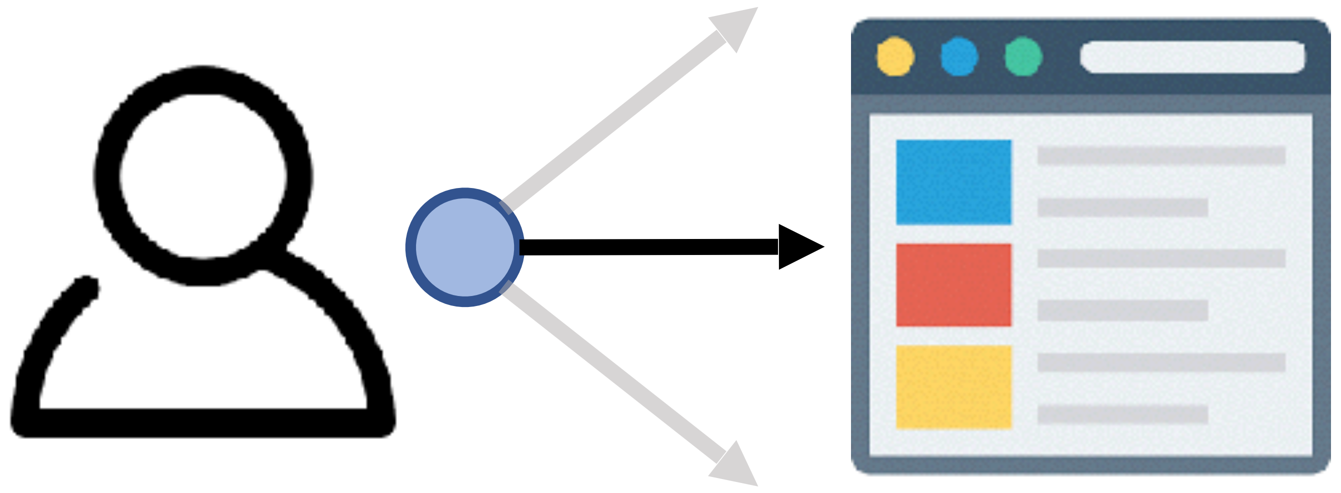


Credit Assignment

Exploration

Generalization

Contextual Bandits



Plan for the tutorial

Part 1: Tabular setting

1. Basics and key concepts
2. Policy optimization and Natural Policy Gradient
3. UCB-Value Iteration

Part 2: Problem set

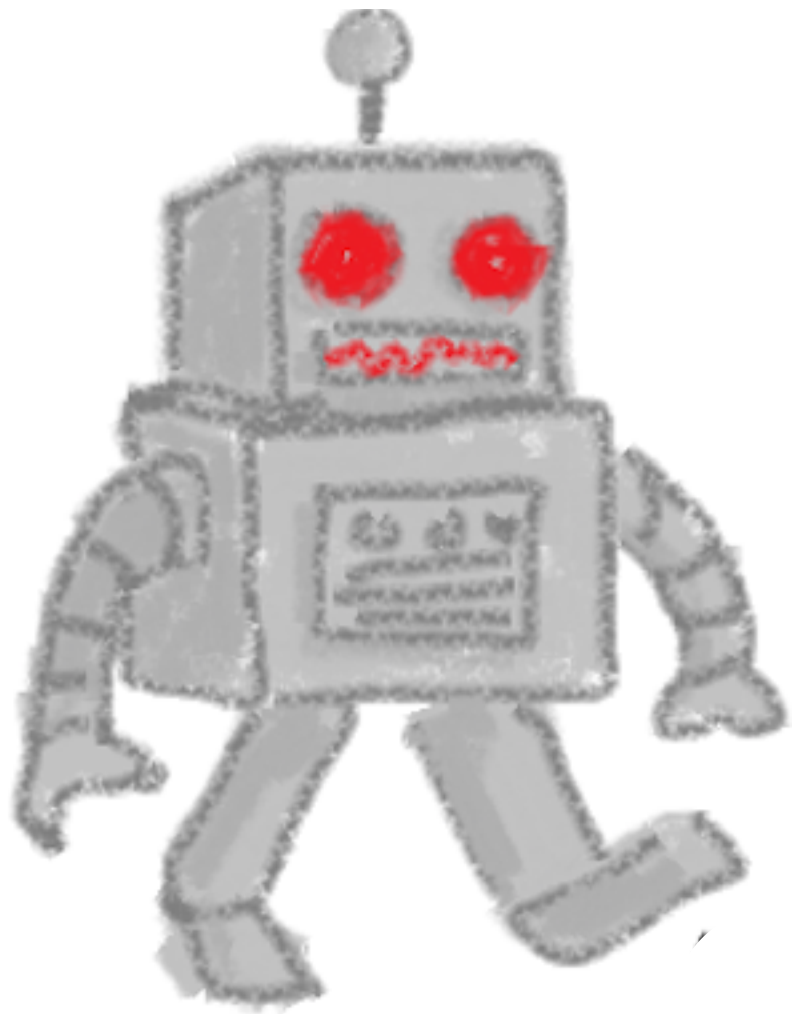
Part 3: Function approximation + Exploration

1. Linear methods and complexity
2. Nonlinear methods, bellman rank, bilinear classes, representation learning

Part 1A: MDP Basics

Markov Decision Processes (Discounted version)

Learning Agent



policy $\pi(a \mid s)$

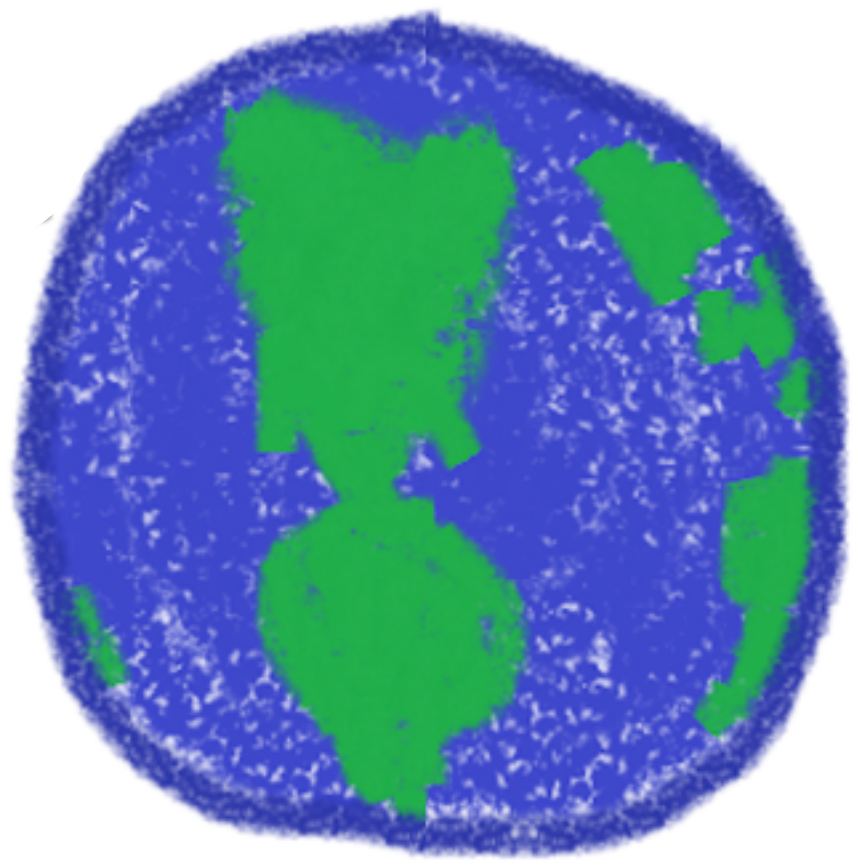
Determine **action** based on **state**

Infinitely many steps

Send **reward** and **next state**

$r(s, a),\ s' \sim P(\cdot \mid s, a)$

Environment



$\mathcal{M} = \{S, A, P, r, \gamma, \mu\}$

$\mu \in \Delta(S)$

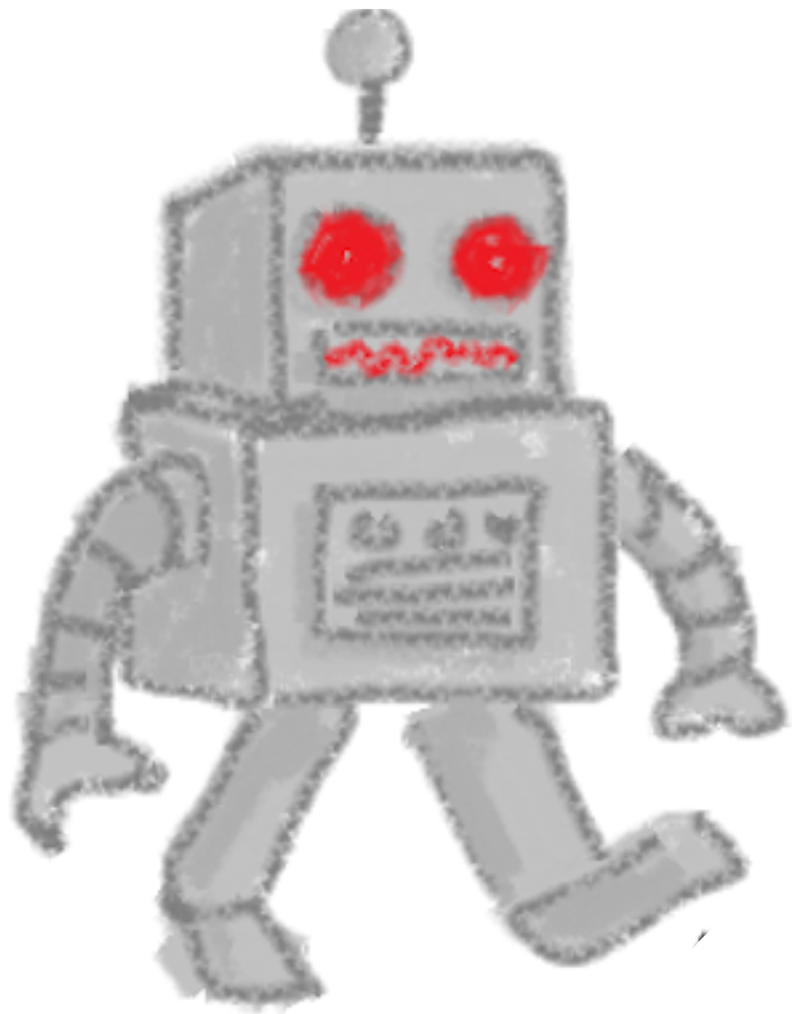
$P : S \times A \mapsto \Delta(S)$

$r : S \times A \to [0,1]$

$\gamma \in [0,1)$

Markov Decision Processes (Discounted version)

Learning Agent



policy $\pi(a \mid s)$

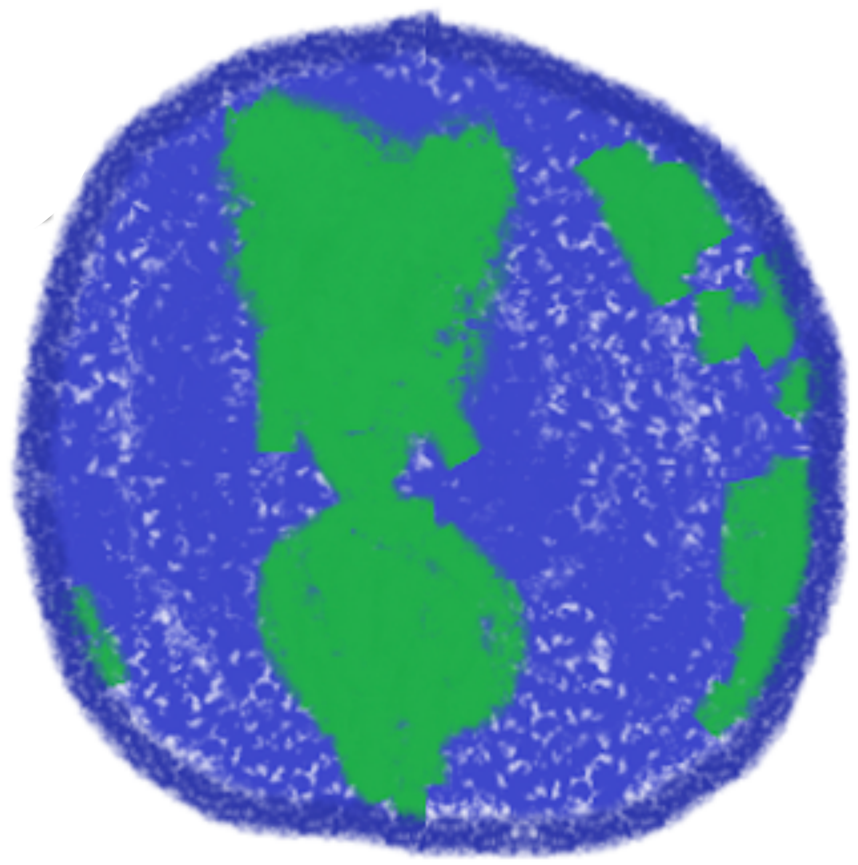
Determine **action** based on **state**

Infinitely many steps

Send **reward** and **next state**

$r(s, a),\ s' \sim P(\cdot \mid s, a)$

Environment



$\mathcal{M} = \{S, A, P, r, \gamma, \mu\}$

$\mu \in \Delta(S)$

$P : S \times A \mapsto \Delta(S)$

$r : S \times A \to [0,1]$

$\gamma \in [0,1)$

Objective:

$$\max_{\pi} \mathbb{E}\left[\sum_{h=0}^{\infty} \gamma^h r(s_h, a_h) \,|\, s_0 \sim \mu, a_h \sim \pi(\,.\,|\,s_h), s_{h+1} \sim P(\,.\,|\,s_h, a_h)\right]$$

Average State-action Distributions

Given a policy $\pi : S \mapsto \Delta(A)$

Denote $d^{\pi}_{\mu,h}(s,a) := \mathrm{P}^{\pi}\left((s_h, a_h) = (s,a)\right)$, i.e., probability of π hitting (s,a) at time step h

Average State-action Distributions

Given a policy $\pi : S \mapsto \Delta(A)$

Denote $d^{\pi}_{\mu,h}(s,a) := \mathrm{P}^{\pi}\left((s_h, a_h) = (s,a)\right)$, i.e., probability of π hitting (s,a) at time step h

Denote $d^{\pi}_{\mu}(s,a) := (1-\gamma)\sum_{h=0}^{\infty}\gamma^h d^{\pi}_h(s,a)$ as the average state-action distribution

Average State-action Distributions

Given a policy $\pi : S \mapsto \Delta(A)$

Denote $d^{\pi}_{\mu,h}(s, a) := \mathbb{P}^{\pi}\left((s_h, a_h) = (s, a)\right)$, i.e., probability of π hitting (s, a) at time step h

Denote $d^{\pi}_{\mu}(s, a) := (1 - \gamma)\sum_{h=0}^{\infty} \gamma^h d^{\pi}_h(s, a)$ as the average state-action distribution

We will abuse notation a bit and denote $d^{\pi}_{\mu}(s) := \sum_a d^{\pi}_{\mu}(s, a)$ as the average state-distribution

Value functions and Bellman equations

Value function $V^{\pi}(s)$: total reward when starting in state s and following π afterwards

Value functions and Bellman equations

Value function $V^\pi(s)$: total reward when starting in state s and following π afterwards

$$V^\pi(s) = \mathbb{E}\left[\sum_{h=0}^{\infty} \gamma^h r(s_h, a_h) \,\middle|\, s_0 = s, a_h \sim \pi(s_h), s_{h+1} \sim P(\,\cdot \mid s_h, a_h)\right]$$

Value functions and Bellman equations

Value function $V^\pi(s)$: total reward when starting in state s and following π afterwards

$$V^\pi(s) = \mathbb{E}\left[\sum_{h=0}^{\infty} \gamma^h r(s_h, a_h) \,\middle|\, s_0 = s, a_h \sim \pi(s_h), s_{h+1} \sim P(\cdot \mid s_h, a_h)\right]$$

$$= \mathbb{E}_{a\sim\pi(\cdot|s)}\left[r(s,a) + \gamma \mathbb{E}_{s'\sim P(\cdot|s,a)} V^\pi(s')\right]$$

(Bellman equation)

Value functions and Bellman equations

Value function $V^\pi(s)$: total reward when starting in state s and following π afterwards

$$V^\pi(s) = \mathbb{E}\left[\sum_{h=0}^{\infty} \gamma^h r(s_h, a_h) \,\middle|\, s_0 = s, a_h \sim \pi(s_h), s_{h+1} \sim P(\cdot \mid s_h, a_h)\right]$$

$$= \mathbb{E}_{a \sim \pi(\cdot|s)}\left[r(s,a) + \gamma \mathbb{E}_{s' \sim P(\cdot|s,a)} V^\pi(s')\right]$$

(Bellman equation)

Q function $Q^\pi(s,a)$: total reward when starting in state s and action a and following π afterwards

Value functions and Bellman equations

Value function $V^\pi(s)$: total reward when starting in state s and following π afterwards

$$V^\pi(s) = \mathbb{E}\left[\sum_{h=0}^{\infty} \gamma^h r(s_h, a_h) \,\middle|\, s_0 = s, a_h \sim \pi(s_h), s_{h+1} \sim P(\cdot \mid s_h, a_h)\right]$$

$$= \mathbb{E}_{a\sim\pi(\cdot|s)}\left[r(s,a) + \gamma \mathbb{E}_{s'\sim P(\cdot|s,a)} V^\pi(s')\right]$$ (Bellman equation)

Q function $Q^\pi(s,a)$: total reward when starting in state s and action a and following π afterwards

$$Q^\pi(s,a) = \mathbb{E}\left[\sum_{h=0}^{\infty} \gamma^h r(s_h, a_h) \,\middle|\, (s_0, a_0) = (s,a), a_h \sim \pi(s_h), s_{h+1} \sim P(\cdot \mid s_h, a_h)\right]$$

Value functions and Bellman equations

Value function $V^\pi(s)$: total reward when starting in state s and following π afterwards

$$V^\pi(s) = \mathbb{E}\left[\sum_{h=0}^{\infty} \gamma^h r(s_h, a_h) \middle| s_0 = s, a_h \sim \pi(s_h), s_{h+1} \sim P(\cdot \mid s_h, a_h)\right]$$

$$= \mathbb{E}_{a\sim\pi(\cdot|s)}\left[r(s,a) + \gamma \mathbb{E}_{s'\sim P(\cdot|s,a)} V^\pi(s')\right]$$ (Bellman equation)

Q function $Q^\pi(s,a)$: total reward when starting in state s and action a and following π afterwards

$$Q^\pi(s,a) = \mathbb{E}\left[\sum_{h=0}^{\infty} \gamma^h r(s_h, a_h) \middle| (s_0, a_0) = (s,a), a_h \sim \pi(s_h), s_{h+1} \sim P(\cdot \mid s_h, a_h)\right]$$

$$= r(s,a) + \gamma \mathbb{E}_{s'\sim P(\cdot|s,a)} V^\pi(s')$$ (Bellman equation)

Optimality

There exists a deterministic stationary policy $\pi^\star : S \mapsto A$, s.t.,

$$V^{\pi^\star}(s) \geq V^{\pi}(s), \forall s, \pi$$

Optimality

There exists a deterministic stationary policy $\pi^\star : S \mapsto A$, s.t.,

$$V^{\pi^\star}(s) \geq V^{\pi}(s), \forall s, \pi$$

We denote $V^\star := V^{\pi^\star}, Q^\star := Q^{\pi^\star}$

Optimality

There exists a deterministic stationary policy $\pi^\star : S \mapsto A$, s.t.,

$$V^{\pi^\star}(s) \geq V^\pi(s), \forall s, \pi$$

We denote $V^\star := V^{\pi^\star}, Q^\star := Q^{\pi^\star}$

Theorem 1: Bellman Optimality

$$\forall s, a: \;\; Q^\star(s,a) = r(s,a) + \gamma \mathbb{E}_{s' \sim P(\cdot|s,a)} \max_{a'} Q^\star(s', a')$$

Optimality

There exists a deterministic stationary policy $\pi^\star : S \mapsto A$, s.t.,
$$V^{\pi^\star}(s) \geq V^{\pi}(s), \forall s, \pi$$

We denote $V^\star := V^{\pi^\star}, Q^\star := Q^{\pi^\star}$

Theorem 1: Bellman Optimality

$$\forall s, a: \;\; Q^\star(s,a) = r(s,a) + \gamma \mathbb{E}_{s' \sim P(\cdot|s,a)} \max_{a'} Q^\star(s',a')$$

Theorem 2: Bellman Optimality

For any $Q : S \times A \rightarrow \mathbb{R}$, if $Q(s,a) = r(s,a) + \gamma \mathbb{E}_{s' \sim P(\cdot|s,a)} \max_{a'} Q(s',a')$

for all s, a, then $Q(s,a) = Q^\star(s,a), \forall s, a$

Planning in MDP with known transition P and reward r

i.e., how to compute $\pi^\star$ (and $V^\star$ / $Q^\star$) given the MDP (P, r)

MDP Planning: Value iteration

Idea: fixed point iteration

Define: Bellman operator $\mathcal{T} : (S \times A \to \mathbb{R}) \to (S \times A \to \mathbb{R})$

$$(\mathcal{T}f)_{s,a} := r(s,a) + \gamma \mathbb{E}_{s' \sim P(\cdot|s,a)}[\max_{a'} f(s',a')]$$

MDP Planning: Value iteration

Idea: fixed point iteration

Define: Bellman operator $\mathcal{T} : (S \times A \to \mathbb{R}) \to (S \times A \to \mathbb{R})$

$$(\mathcal{T}f)_{s,a} := r(s,a) + \gamma \mathbb{E}_{s' \sim P(\cdot|s,a)}[\max_{a'} f(s',a')]$$

VI Algorithm: Initialize $Q^{(0)} \text{s.t.}, Q^{(0)}(s,a) \in [0,1/(1-\gamma))$

Iterate $Q^{(t+1)} \leftarrow \mathcal{T} Q^{(t)}$

MDP Planning: Value iteration

Idea: fixed point iteration

Define: Bellman operator $\mathcal{T} : (S \times A \to \mathbb{R}) \to (S \times A \to \mathbb{R})$

$$(\mathcal{T} f)_{s,a} := r(s,a) + \gamma \mathbb{E}_{s' \sim P(\cdot|s,a)}[\max_{a'} f(s', a')]$$

VI Algorithm: Initialize $Q^{(0)} \text{s.t.}, Q^{(0)}(s,a) \in [0, 1/(1-\gamma))$

Iterate $Q^{(t+1)} \leftarrow \mathcal{T} Q^{(t)}$

Theorem: Induced policy $\pi^{(t)} : s \mapsto \arg\max_a Q^{(t)}(s,a)$ satisfies

$$V^{\pi^{(t)}}(s) \geq V^\star(s) - \frac{2\gamma^t}{1-\gamma} \|Q^{(0)} - Q^\star\|_\infty \quad \forall s \in S$$

MDP Planning: Value iteration

Idea: fixed point iteration

Define: Bellman operator $\mathcal{T} : (S \times A \to \mathbb{R}) \to (S \times A \to \mathbb{R})$

$$(\mathcal{T} f)_{s,a} := r(s,a) + \gamma \mathbb{E}_{s' \sim P(\cdot|s,a)}[\max_{a'} f(s', a')]$$

VI Algorithm: Initialize $Q^{(0)} \text{ s.t.}, Q^{(0)}(s,a) \in [0, 1/(1-\gamma))$

Iterate $Q^{(t+1)} \leftarrow \mathcal{T} Q^{(t)}$

Contraction lemma

$\|\mathcal{T} Q - \mathcal{T} Q'\|_\infty \leq \gamma \|Q - Q'\|_\infty$

Theorem: Induced policy $\pi^{(t)} : s \mapsto \arg\max_a Q^{(t)}(s,a)$ satisfies

$$V^{\pi^{(t)}}(s) \geq V^\star(s) - \frac{2\gamma^t}{1-\gamma} \|Q^{(0)} - Q^\star\|_\infty \quad \forall s \in S$$

MDP Planning: Policy iteration

Idea: Alternate between policy evaluation and policy improvement

Initialize $\pi^{(0)} : S \to A$

Repeat:

- Compute $Q^{\pi^{(t)}}$ (evaluation)
- Update $\pi^{(t+1)} : \pi^{(t+1)}(s) = \arg\max_a Q^{\pi^{(t)}}(s, a)$ (improvement)

MDP Planning: Policy iteration

Idea: Alternate between policy evaluation and policy improvement

Initialize $\pi^{(0)} : S \rightarrow A$

Repeat:

- Compute $Q^{\pi^{(t)}}$ (evaluation)
- Update $\pi^{(t+1)} : \pi^{(t+1)}(s) = \arg\max_a Q^{\pi^{(t)}}(s, a)$ (improvement)

Linear system solve

MDP Planning: Policy iteration

Idea: Alternate between policy evaluation and policy improvement

Initialize $\pi^{(0)} : S \to A$

Repeat:

- Compute $Q^{\pi^{(t)}}$ (evaluation)
- Update $\pi^{(t+1)} : \pi^{(t+1)}(s) = \arg\max_a Q^{\pi^{(t)}}(s, a)$ (improvement)

Linear system solve

Theorem: Geometric convergence:

$$\|V^{\pi^{(t+1)}} - V^\star\|_\infty \leq \gamma \|V^{\pi^{(t)}} - V^\star\|_\infty$$

Finite Horizon MDPs

$$\mathcal{M} = \{S, A, P, r, \mu, H\}$$

$$P : S \times A \mapsto \Delta(S), \quad r : S \times A \to [0,1], \quad H \in \mathbb{N}^+, \quad \mu \in \Delta(S)$$

time-dependent policies: $\pi^\star := \{\pi_0^\star, \ldots, \pi_{H-1}^\star\}$

time-dependent V/Q functions: $\{V_h^\star\}_{h=0}^{H-1}, \{Q_h^\star\}_{h=0}^{H-1}$

Finite Horizon MDPs

$$\mathcal{M} = \{S, A, P, r, \mu, H\}$$

$$P : S \times A \mapsto \Delta(S), \quad r : S \times A \to [0,1], \quad H \in \mathbb{N}^+, \quad \mu \in \Delta(S)$$

Episode:

$s_0 \sim \mu$

For $h = 0,\ldots,H-1$:

- Take action a_h
- Collect reward $r(s_h, a_h)$
- Transition $s_{h+1} \sim P(\cdot \mid s_h, a_h)$

time-dependent policies: $\pi^\star := \{\pi_0^\star, \ldots, \pi_{H-1}^\star\}$

time-dependent V/Q functions: $\{V_h^\star\}_{h=0}^{H-1}, \{Q_h^\star\}_{h=0}^{H-1}$

Finite Horizon MDPs

$$\mathcal{M} = \{S, A, P, r, \mu, H\}$$

$$P : S \times A \mapsto \Delta(S), \quad r : S \times A \to [0,1], \quad H \in \mathbb{N}^+, \quad \mu \in \Delta(S)$$

Episode:

$s_0 \sim \mu$

For $h = 0,\ldots,H-1$:

- Take action a_h
- Collect reward $r(s_h, a_h)$
- Transition $s_{h+1} \sim P(\cdot \mid s_h, a_h)$

Objective function: $V(\pi) = \mathbb{E}\left[\sum_{h=0}^{H-1} r(s_h, a_h)\right]$

time-dependent policies: $\pi^\star := \{\pi_0^\star, \ldots, \pi_{H-1}^\star\}$

time-dependent V/Q functions: $\{V_h^\star\}_{h=0}^{H-1}, \{Q_h^\star\}_{h=0}^{H-1}$

Summary so far:

MDP definitions (discounted infinite horizon & finite horizon);

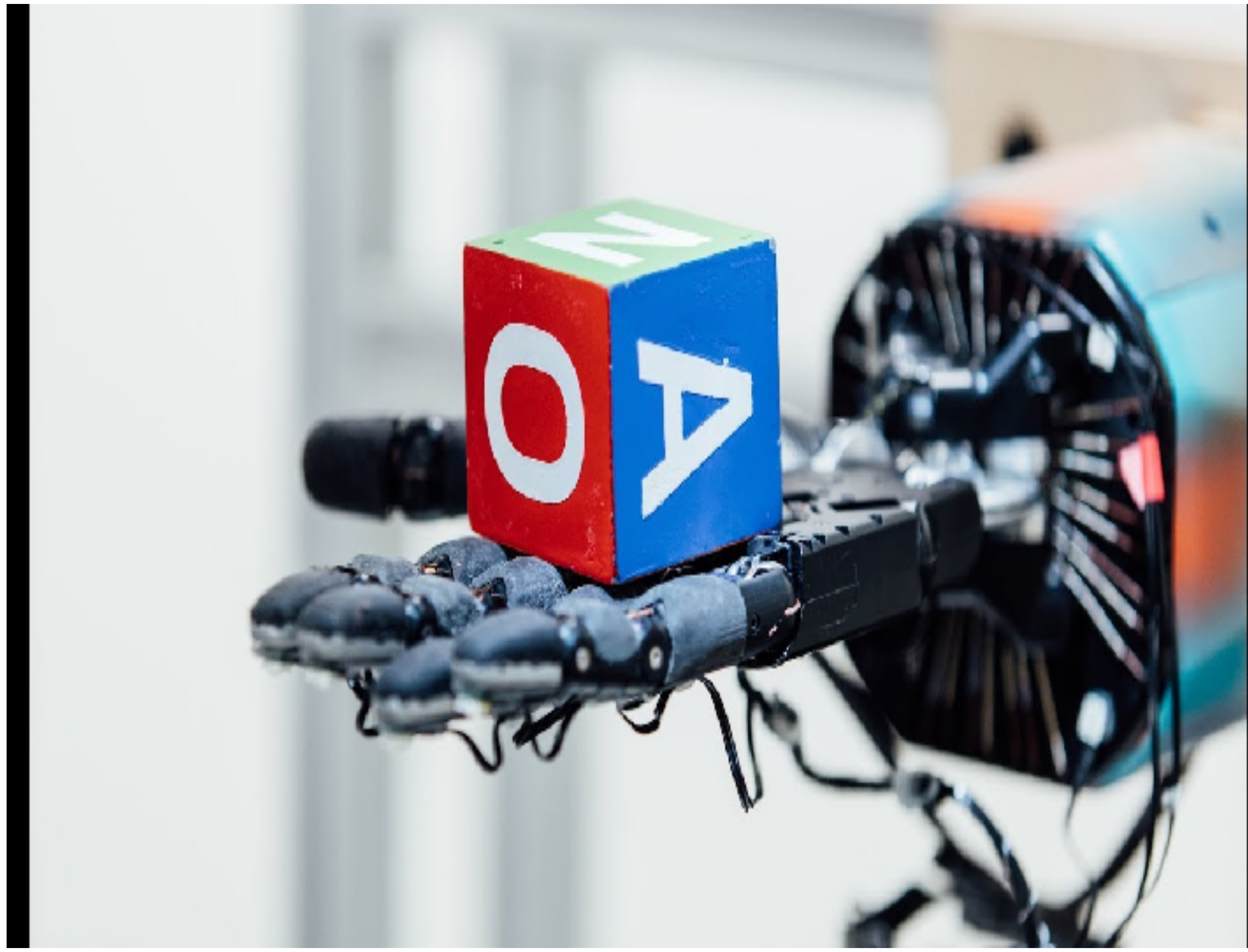
State-action distributions, value and Q functions, and two planning algorithms

Part 1B: Policy Gradient & Natural Policy Gradient

Policy Optimization Motivation: Practical

[AlphaZero, Silver et.al, 17]

[OpenAI Five, 18]



[OpenAI,19]

Policy Optimization Motivation: Simple

$$\pi_\theta(a \mid s) := \pi(a \mid s; \theta) \qquad V^{\pi_\theta} = \mathbb{E}_{\pi_\theta}\left[\sum_{h=0}^{\infty} \gamma^h r_h\right]$$

$$\theta_{t+1} = \theta_t + \eta \nabla_\theta V^{\pi_\theta}|_{\theta=\theta_t}$$

Policy Optimization Motivation: Simple

$$\pi_\theta(a \mid s) := \pi(a \mid s; \theta) \qquad V^{\pi_\theta} = \mathbb{E}_{\pi_\theta}\left[\sum_{h=0}^{\infty} \gamma^h r_h\right]$$

$$\theta_{t+1} = \theta_t + \eta \nabla_\theta V^{\pi_\theta}|_{\theta=\theta_t}$$

We can have a closed-form expression for PG:

Policy Gradient Theorem [Sutton, McAllester, Singh, Mansour]:

Define advantage function $A^{\pi_\theta}(s,a) := Q^{\pi_\theta}(s,a) - V^{\pi_\theta}(s)$, we have:

$$\nabla_\theta V^{\pi_\theta} = \frac{1}{1-\gamma}\mathbb{E}_{s,a\sim d_\mu^{\pi_\theta}}\left[\nabla_\theta \ln \pi_\theta(a \mid s) A^{\pi_\theta}(s,a)\right]$$

Policy Optimization Motivation: Simple

$$\pi_\theta(a \mid s) := \pi(a \mid s; \theta) \qquad V^{\pi_\theta} = \mathbb{E}_{\pi_\theta}\left[\sum_{h=0}^{\infty} \gamma^h r_h\right]$$

$$\theta_{t+1} = \theta_t + \eta \nabla_\theta V^{\pi_\theta}\big|_{\theta=\theta_t}$$

We can have a closed-form expression for PG:

Policy Gradient Theorem [Sutton, McAllester, Singh, Mansour]:

Define advantage function $A^{\pi_\theta}(s, a) := Q^{\pi_\theta}(s, a) - V^{\pi_\theta}(s)$, we have:

$$\nabla_\theta V^{\pi_\theta} = \frac{1}{1-\gamma} \mathbb{E}_{s,a \sim d_\mu^{\pi_\theta}}\left[\nabla_\theta \ln \pi_\theta(a \mid s) A^{\pi_\theta}(s, a)\right]$$

Adjust the probability $\pi_\theta(a \mid s)$ proportional to $A^{\pi_\theta}(s, a) := Q^{\pi_\theta}(s, a) - V^{\pi_\theta}(s)$

Global optimality of Policy Gradient methods

Consider tabular MDPs, with $\pi_\theta(a \mid s) = \dfrac{\exp(\theta_{s,a})}{\sum_{a'} \exp(\theta_{s,a'})}$, $\theta_{s,a} \in \mathbb{R}$

Global optimality of Policy Gradient methods

Consider tabular MDPs, with $\pi_\theta(a \mid s) = \frac{\exp(\theta_{s,a})}{\sum_{a'} \exp(\theta_{s,a'})}, \theta_{s,a} \in \mathbb{R}$

PG formulation:

$\frac{\partial V(\theta)}{\partial \theta_{s,a}} = \frac{1}{1-\gamma} d_\mu^\pi(s) \pi_\theta(a \mid s) A^{\pi_\theta}(s, a)$, where $A^{\pi_\theta}(s, a) = Q^{\pi_\theta}(s, a) - V^{\pi_\theta}(s)$

Global optimality of Policy Gradient methods

Consider tabular MDPs, with $\pi_\theta(a \mid s) = \frac{\exp(\theta_{s,a})}{\sum_{a'} \exp(\theta_{s,a'})}$, $\theta_{s,a} \in \mathbb{R}$

PG formulation:

$$\frac{\partial V(\theta)}{\partial \theta_{s,a}} = \frac{1}{1-\gamma} d_\mu^\pi(s) \pi_\theta(a \mid s) A^{\pi_\theta}(s,a), \text{ where } A^{\pi_\theta}(s,a) = Q^{\pi_\theta}(s,a) - V^{\pi_\theta}(s)$$

Despite being non-concave, we have global convergence:

Global optimality of Policy Gradient methods

Consider tabular MDPs, with $\pi_\theta(a \mid s) = \dfrac{\exp(\theta_{s,a})}{\sum_{a'} \exp(\theta_{s,a'})}, \theta_{s,a} \in \mathbb{R}$

PG formulation:

$$\frac{\partial V(\theta)}{\partial \theta_{s,a}} = \frac{1}{1-\gamma} d_\mu^\pi(s) \pi_\theta(a \mid s) A^{\pi_\theta}(s,a), \text{ where } A^{\pi_\theta}(s,a) = Q^{\pi_\theta}(s,a) - V^{\pi_\theta}(s)$$

Despite being non-concave, we have global convergence:

Theorem (Informal) [Agarwal, Kakade, Lee, Mahajan 20; Mei, Xiao, Szepesvari, Schuurmans 20]:

Assume $\mu(s) > 0, \forall s,$ the PG algorithm $\theta^{t+1} := \theta^t + \eta \nabla_\theta V(\theta) \mid_{\theta=\theta^t}$ converges to global optimality

Policy optimization: Natural Policy Gradient

[Kakade 03]

Policy optimization: Natural Policy Gradient

[Kakade 03]

Define Fisher information matrix

$$F_\theta = \mathbb{E}_{s,a \sim d^{\pi_\theta}} \left[\nabla_\theta \ln \pi_\theta(a \mid s) \left(\nabla_\theta \ln \pi_\theta(a \mid s) \right)^\top \right] \in \mathbb{R}^{d_\theta \times d_\theta}$$

Policy optimization: Natural Policy Gradient

[Kakade 03]

Define Fisher information matrix

$$F_\theta = \mathbb{E}_{s,a\sim d^{\pi_\theta}}\left[\nabla_\theta \ln \pi_\theta(a\,|\,s)\left(\nabla_\theta \ln \pi_\theta(a\,|\,s)\right)^\top\right] \in \mathbb{R}^{d_\theta \times d_\theta}$$

Natural policy gradient uses F_θ to pre-condition PG:

$$\theta^{t+1} := \theta^t + \eta F_{\theta^t}^{-1} \nabla_\theta V(\theta)\,|_{\theta=\theta^t}$$

Policy optimization: Natural Policy Gradient

[Kakade 03]

Define Fisher information matrix

$$F_\theta = \mathbb{E}_{s,a \sim d^{\pi_\theta}} \left[\nabla_\theta \ln \pi_\theta(a \mid s) \left(\nabla_\theta \ln \pi_\theta(a \mid s) \right)^\top \right] \in \mathbb{R}^{d_\theta \times d_\theta}$$

Natural policy gradient uses F_θ to pre-condition PG:

$$\theta^{t+1} := \theta^t + \eta F_{\theta^t}^{-1} \nabla_\theta V(\theta) \,|_{\theta=\theta^t}$$

(For simplicity, assume F_θ is full rank —- otherwise use pseudo inverse)

The trust region optimization interpretation of NPG

[Bagnell & Schneider 03]

NPG as a Trust-region optimization procedure:

$$\max_{\theta} \langle \theta, \nabla_\theta V(\theta) |_{\theta=\theta^t} \rangle, \text{ s.t., } KL\left(\rho_{\theta^t} || \rho_\theta\right) \leq \delta$$

$$\left(\rho_\theta(\tau) := \mu(s_0) \prod_h \pi(a_h | s_h) P(s_{h+1} | s_h, a_h)\right)$$

The trust region optimization interpretation of NPG

[Bagnell & Schneider 03]

NPG as a Trust-region optimization procedure:

$$\max_{\theta} \langle \theta, \nabla_\theta V(\theta)|_{\theta=\theta^t} \rangle, \text{ s.t., } KL\left(\rho_{\theta^t} || \rho_\theta\right) \leq \delta$$

$$\left(\rho_\theta(\tau) := \mu(s_0) \prod_h \pi(a_h | s_h) P(s_{h+1} | s_h, a_h)\right)$$

i.e., optimize the **linearized objective** s.t. a KL constraint **forcing new policy's trajectory distribution staying close to old one's**

The trust region optimization interpretation of NPG

[Bagnell & Schneider 03]

NPG as a Trust-region optimization procedure:

$$\max_{\theta} \langle \theta, \nabla_\theta V(\theta)|_{\theta=\theta^t} \rangle, \text{ s.t., } KL\left(\rho_{\theta^t} || \rho_\theta\right) \leq \delta$$

$$\left(\rho_\theta(\tau) := \mu(s_0) \prod_h \pi(a_h | s_h) P(s_{h+1} | s_h, a_h)\right)$$

i.e., optimize the **linearized objective** s.t. a KL constraint **forcing new policy's trajectory distribution staying close to old one's**

Further perform second-order Taylor expansion on $KL\left(\rho_{\theta^t} || \rho_\theta\right)$ at θ^t:

The trust region optimization interpretation of NPG

[Bagnell & Schneider 03]

NPG as a Trust-region optimization procedure:

$$\max_{\theta} \langle \theta, \nabla_\theta V(\theta) |_{\theta=\theta^t} \rangle, \text{ s.t., } KL\left(\rho_{\theta^t} || \rho_\theta\right) \leq \delta$$

$$\left(\rho_\theta(\tau) := \mu(s_0) \prod_h \pi(a_h | s_h) P(s_{h+1} | s_h, a_h)\right)$$

i.e., optimize the **linearized objective** s.t. a KL constraint **forcing new policy's trajectory distribution staying close to old one's**

Further perform second-order Taylor expansion on $KL\left(\rho_{\theta^t} || \rho_\theta\right)$ at θ^t:

$$KL\left(\rho_{\theta^t} || \rho_\theta\right) \approx (\theta - \theta^t)^\top F_{\theta^t} (\theta - \theta^t)$$

The trust region optimization interpretation of NPG

[Bagnell & Schneider 03]

NPG as a Trust-region optimization procedure:

$$\max_{\theta} \langle \theta, \nabla_\theta V(\theta)|_{\theta=\theta^t} \rangle, \text{ s.t., } KL\left(\rho_{\theta^t} || \rho_\theta\right) \leq \delta$$

$$\left(\rho_\theta(\tau) := \mu(s_0) \prod_h \pi(a_h | s_h) P(s_{h+1} | s_h, a_h)\right)$$

i.e., optimize the **linearized objective** s.t. a KL constraint **forcing new policy's trajectory distribution staying close to old one's**

Further perform second-order Taylor expansion on $KL\left(\rho_{\theta^t} || \rho_\theta\right)$ at θ^t:

$$KL\left(\rho_{\theta^t} || \rho_\theta\right) \approx (\theta - \theta^t)^\top F_{\theta^t}(\theta - \theta^t)$$

NPG then is revealed by solving the convex program:

$$\max_{\theta} \langle \theta, \nabla_\theta V(\theta)|_{\theta=\theta^t} \rangle, \text{ s.t., } (\theta - \theta^t)^\top F_{\theta^t}(\theta - \theta^t) \leq \delta$$

Natural policy gradient in Tabular MDPs

Recall the softmax Policy for Tabular MDPs:

$$\theta_{s,a} \in \mathbb{R}, \forall s, a \in S \times A \quad \pi_\theta(a \mid s) = \frac{\exp(\theta_{s,a})}{\sum_{a'} \exp(\theta_{s,a'})}$$

Natural policy gradient in Tabular MDPs

Recall the softmax Policy for Tabular MDPs:

$$\theta_{s,a} \in \mathbb{R}, \forall s, a \in S \times A \quad \pi_\theta(a \mid s) = \frac{\exp(\theta_{s,a})}{\sum_{a'} \exp(\theta_{s,a'})}$$

We can show that the NPG update $\theta^{t+1} := \theta^t + \eta F_{\theta^t}^{-1} \nabla_\theta V(\theta^t)$ is equivalent to (see the exercise in recitation):

Natural policy gradient in Tabular MDPs

Recall the softmax Policy for Tabular MDPs:

$$\theta_{s,a} \in \mathbb{R}, \forall s, a \in S \times A \quad \pi_\theta(a \mid s) = \frac{\exp(\theta_{s,a})}{\sum_{a'} \exp(\theta_{s,a'})}$$

We can show that the NPG update $\theta^{t+1} := \theta^t + \eta F_{\theta^t}^{-1} \nabla_\theta V(\theta^t)$ is equivalent to (see the exercise in recitation):

$$(\pi^t := \pi_{\theta^t}) \quad \pi^{t+1}(a \mid s) \propto \pi^t(a \mid s) \cdot \exp\left(\eta A^{\pi^t}(s, a)\right)$$

Natural policy gradient in Tabular MDPs

Recall the softmax Policy for Tabular MDPs:

$$\theta_{s,a} \in \mathbb{R}, \forall s, a \in S \times A \quad \pi_\theta(a \mid s) = \frac{\exp(\theta_{s,a})}{\sum_{a'} \exp(\theta_{s,a'})}$$

We can show that the NPG update $\theta^{t+1} := \theta^t + \eta F_{\theta^t}^{-1} \nabla_\theta V(\theta^t)$ is equivalent to (see the exercise in recitation):

$$(\pi^t := \pi_{\theta^t}) \quad \pi^{t+1}(a \mid s) \propto \pi^t(a \mid s) \cdot \exp\left(\eta A^{\pi^t}(s, a)\right)$$

Proof sketch: $A^{\pi_{\theta^t}}(\cdot, \cdot) \propto \arg\min_x \|\nabla_\theta V(\theta^t) - F_{\theta^t} x\|_2^2$ (see recitation for details)

Natural policy gradient in Tabular MDPs

Recall the softmax Policy for Tabular MDPs:

$$\theta_{s,a} \in \mathbb{R}, \forall s, a \in S \times A \quad \pi_\theta(a \mid s) = \frac{\exp(\theta_{s,a})}{\sum_{a'} \exp(\theta_{s,a'})}$$

We can show that the NPG update $\theta^{t+1} := \theta^t + \eta F_{\theta^t}^{-1} \nabla_\theta V(\theta^t)$ is equivalent to (see the exercise in recitation):

$$(\pi^t := \pi_{\theta^t}) \quad \pi^{t+1}(a \mid s) \propto \pi^t(a \mid s) \cdot \exp\left(\eta A^{\pi^t}(s, a)\right)$$

Proof sketch: $A^{\pi_{\theta^t}}(\cdot, \cdot) \propto \arg\min_x \|\nabla_\theta V(\theta^t) - F_{\theta^t} x\|_2^2$ (see recitation for details)

Interpretation: for each state s, NPG runs online mirror ascent with $A^{\pi^t}(s, \cdot) \in \mathbb{R}^{|A|}$ as the reward vector at iter t

Global Convergence of the exact Natural policy gradient

$$\pi^{t+1}(a \mid s) \propto \pi^{t}(a \mid s) \cdot \exp\left(\eta A^{\pi^t}(s,a)\right)$$

(Note here we are studying the **idealized case where we have exact** $A^{\pi^t}(\cdot,\cdot)$.
We will look into learning/approximation in the recitation)

Global Convergence of the exact Natural policy gradient

$$\pi^{t+1}(a \mid s) \propto \pi^t(a \mid s) \cdot \exp\left(\eta A^{\pi^t}(s, a)\right)$$

(Note here we are studying the **idealized case where we have exact** $A^{\pi^t}(\cdot, \cdot)$.
We will look into learning/approximation in the recitation)

Theorem [Agarwal, Kakade, Lee, Mahajan 20]: Initialize $\pi^0(\cdot \mid s) = \text{Unif}(A)$. After T iterations, there exits a policy $\pi \in \{\pi^0, \ldots, \pi^{T-1}\}$, s.t.,

$$V^\pi \geq V^\star - \frac{\log A}{\eta T} - \frac{1}{(1-\gamma)^2 T}.$$

Global Convergence of the exact Natural policy gradient

$$\pi^{t+1}(a \mid s) \propto \pi^t(a \mid s) \cdot \exp\left(\eta A^{\pi^t}(s,a)\right)$$

(Note here we are studying the **idealized case where we have exact** $A^{\pi^t}(\cdot,\cdot)$.
We will look into learning/approximation in the recitation)

Theorem [Agarwal, Kakade, Lee, Mahajan 20]: Initialize $\pi^0(\cdot \mid s) = \text{Unif}(A)$. After T iterations, there exits a policy $\pi \in \{\pi^0, \ldots, \pi^{T-1}\}$, s.t.,

$$V^\pi \geq V^\star - \frac{\log A}{\eta T} - \frac{1}{(1-\gamma)^2 T}.$$

- Global optimality despite non-concavity in the objective
- No $|S|$ dependence at all; log-dependence on $|A|$
- No coverage requirement on the initial distribution μ

Proof Sketch for NPG's global optimality (a $1/\sqrt{T}$ rate)

Proof Sketch for NPG's global optimality (a $1/\sqrt{T}$ rate)

1. Since we run Mirror Ascent per state, we have that for all $s \in S$:

Proof Sketch for NPG's global optimality (a $1/\sqrt{T}$ rate)

1. Since we run Mirror Ascent per state, we have that for all $s \in S$:

$$\underbrace{\sum_{t=0}^{T-1} \langle \pi^\star(\cdot \mid s), A^{\pi^t}(s,\cdot) \rangle - \underbrace{\langle \pi^t(\cdot \mid s), A^{\pi^t}(s,\cdot) \rangle}_{=0}}_{\text{regret of mirror ascent on s}} \lesssim \sqrt{\ln(|A|)T}\,.$$

Proof Sketch for NPG's global optimality (a $1/\sqrt{T}$ rate)

1. Since we run Mirror Ascent per state, we have that for all $s \in S$:

$$\underbrace{\sum_{t=0}^{T-1} \langle \pi^\star(\cdot \mid s), A^{\pi^t}(s, \cdot) \rangle - \underbrace{\langle \pi^t(\cdot \mid s), A^{\pi^t}(s, \cdot) \rangle}_{=0}}_{\text{regret of mirror ascent on s}} \lesssim \sqrt{\ln(|A|)T}.$$

2. Add $\mathbb{E}_{s\sim d_\mu^{\pi^\star}}$ on both sides, and via performance difference lemma [Kakade & Langford 2003]:

$$\sum_{t=0}^{T-1} V^{\pi^\star} - V^{\pi^t} \propto \sum_{t=0}^{T-1} \mathbb{E}_{s\sim d_\mu^{\pi^\star}} \left[\mathbb{E}_{a\sim\pi^\star(\cdot|s)} A^{\pi^t}(s, a) \right] \lesssim \sqrt{\ln(|A|)T}.$$

Proof Sketch for NPG's global optimality (a $1/\sqrt{T}$ rate)

1. Since we run Mirror Ascent per state, we have that for all $s \in S$:

$$\underbrace{\sum_{t=0}^{T-1} \langle \pi^\star(\cdot \mid s), A^{\pi^t}(s, \cdot) \rangle - \underbrace{\langle \pi^t(\cdot \mid s), A^{\pi^t}(s, \cdot) \rangle}_{=0}}_{\text{regret of mirror ascent on s}} \lesssim \sqrt{\ln(|A|)T}\,.$$

2. Add $\mathbb{E}_{s \sim d_\mu^{\pi^\star}}$ on both sides, and via performance difference lemma [Kakade & Langford 2003]:

$$\sum_{t=0}^{T-1} V^{\pi^\star} - V^{\pi^t} \propto \sum_{t=0}^{T-1} \mathbb{E}_{s \sim d_\mu^{\pi^\star}} \left[\mathbb{E}_{a \sim \pi^\star(\cdot|s)} A^{\pi^t}(s, a) \right] \lesssim \sqrt{\ln(|A|)T}\,.$$

(see the exercise in recitation for a detailed proof with approximation on Q^{π^t},
and see chapter 10 in AJKS monograph for the proof for $1/T$ rate)

Summary so far:

Policy Gradient and NPG:

Global Convergence vanilla PG and NPG in tabular MDPs with softmax parameterization

NPG w/ approximation in Recitation

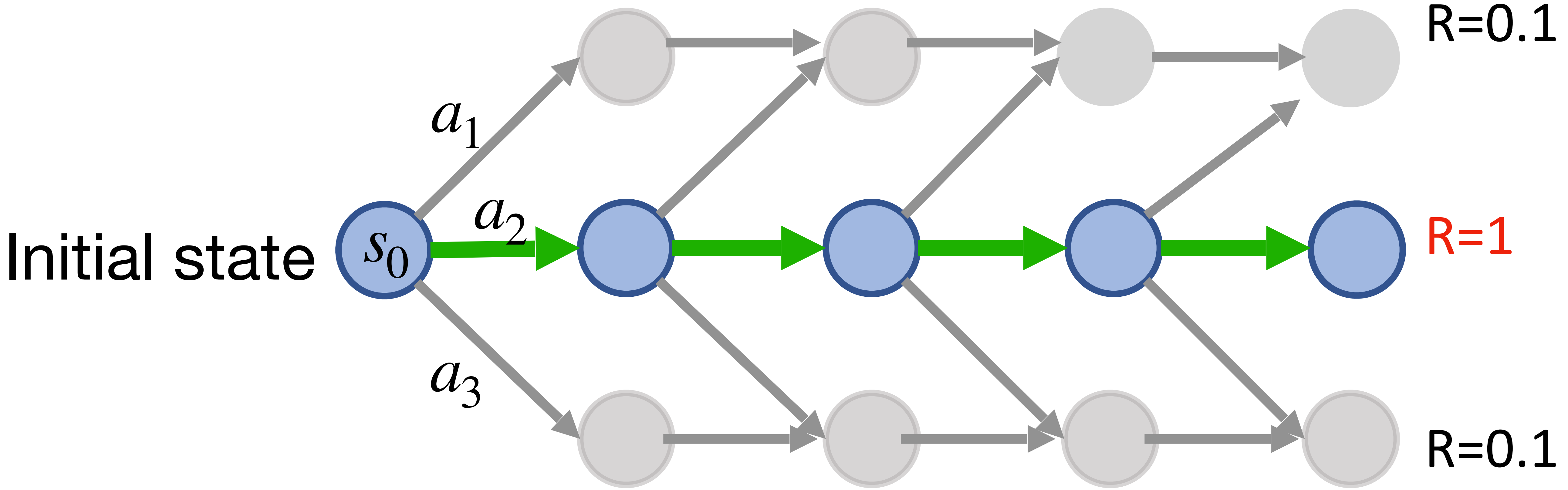
Part 1C: Exploration in tabular MDP w/ UCB-Value Iteration

In this part:

Question: how to explore efficient if we do not know (P, r)

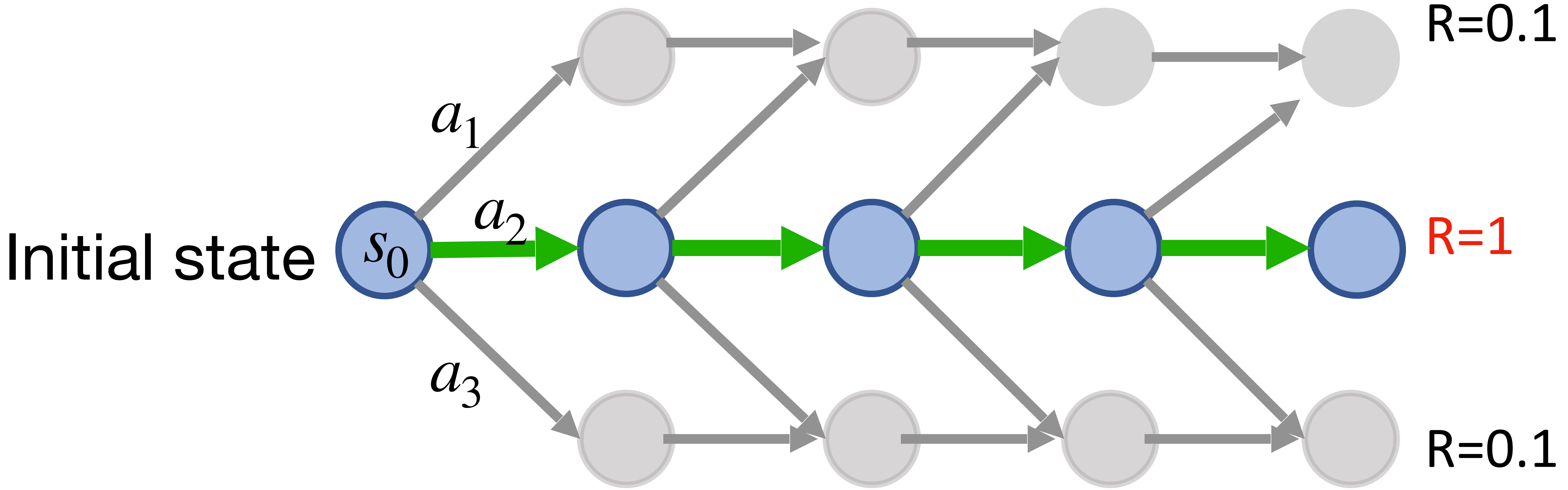
We need to perform efficient exploration when learning:

The combination lock problem:



We need to perform efficient exploration when learning:

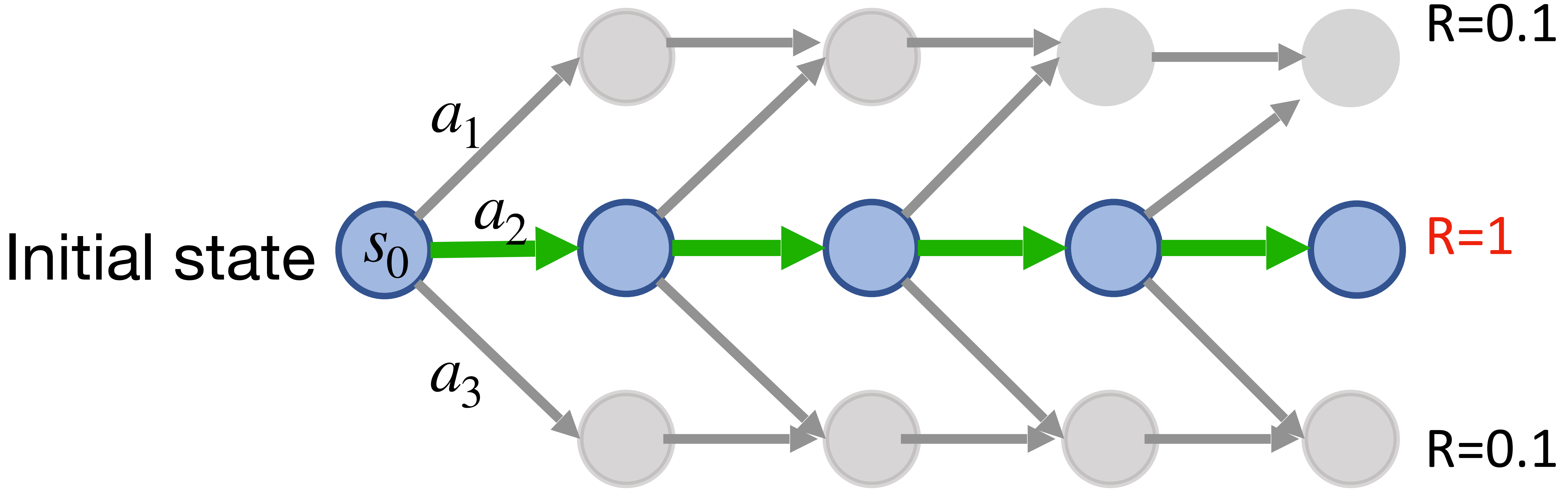
The combination lock problem:



The prob of a random walk reaching the goal is exponentially small wrt H

We need to perform efficient exploration when learning:

The combination lock problem:



The prob of a random walk reaching the goal is exponentially small wrt H

The principle behind UCB-VI: Optimism in the face of uncertainty

Problem setup, learning protocol, and goal

Setting: episodic finite horizon tabular MDP (horizon = H), fixed initial state s_0

transitions $\{P_h\}_{h=0}^{H-1}$ unknown, but reward $r(s, a)$ known

learning protocol:

Goal:

Problem setup, learning protocol, and goal

Setting: episodic finite horizon tabular MDP (horizon = H), fixed initial state s_0

transitions $\{P_h\}_{h=0}^{H-1}$ unknown, but reward $r(s, a)$ known

learning protocol:

1. Learner initializes a policy π^0

Goal:

Problem setup, learning protocol, and goal

Setting: episodic finite horizon tabular MDP (horizon = H), fixed initial state s_0

transitions$\{P_h\}_{h=0}^{H-1}$ unknown, but reward $r(s,a)$ known

learning protocol:

1. Learner initializes a policy π^0

2. At episode n, learner executes π^n to draw a trajectory starting at s_0:
$\{s_h^n, a_h^n, r_h^n\}_{h=0}^{H-1}$, with $a_h^n = \pi^n(s_h^n), r_h^n = r(s_h^n, a_h^n),\ s_{h+1}^n \sim P(\cdot \mid s_h^n, a_h^n)$

Goal:

Problem setup, learning protocol, and goal

Setting: episodic finite horizon tabular MDP (horizon = H), fixed initial state s_0

transitions $\{P_h\}_{h=0}^{H-1}$ unknown, but reward $r(s, a)$ known

learning protocol:

1. Learner initializes a policy π^0

2. At episode n, learner executes π^n to draw a trajectory starting at s_0:
$\{s_h^n, a_h^n, r_h^n\}_{h=0}^{H-1}$, with $a_h^n = \pi^n(s_h^n), r_h^n = r(s_h^n, a_h^n),\ s_{h+1}^n \sim P(\cdot \mid s_h^n, a_h^n)$

3. Learner updates policy to π^{n+1} using all prior information

Goal:

Problem setup, learning protocol, and goal

Setting: episodic finite horizon tabular MDP (horizon = H), fixed initial state s_0

transitions $\{P_h\}_{h=0}^{H-1}$ unknown, but reward $r(s,a)$ known

learning protocol:

1. Learner initializes a policy π^0

2. At episode n, learner executes π^n to draw a trajectory starting at s_0:
$\{s_h^n, a_h^n, r_h^n\}_{h=0}^{H-1}$, with $a_h^n = \pi^n(s_h^n), r_h^n = r(s_h^n, a_h^n),\ s_{h+1}^n \sim P(\cdot \mid s_h^n, a_h^n)$

3. Learner updates policy to π^{n+1} using all prior information

Goal:

Sub-linear regret:

$$\mathbb{E}\left[\sum_{n=1}^{N}\left(V^\star - V^{\pi^n}\right)\right] = \text{poly}(S, A, H)\sqrt{N}$$

UCBVI: Optimistic Model-based Learning

Inside iteration n :

UCBVI: Optimistic Model-based Learning

Inside iteration n :

Use all previous data to estimate transitions $\widehat{P}_0^n, \ldots, \widehat{P}_{H-1}^n$

UCBVI: Optimistic Model-based Learning

Inside iteration n :

Use all previous data to estimate transitions $\widehat{P}_0^n, \ldots, \widehat{P}_{H-1}^n$

Design reward bonus $b_h^n(s, a), \forall s, a, h$

UCBVI: Optimistic Model-based Learning

Inside iteration n :

Use all previous data to estimate transitions $\widehat{P}_0^n, \ldots, \widehat{P}_{H-1}^n$

Design reward bonus $b_h^n(s, a), \forall s, a, h$

Optimistic planning with learned model: $\pi^n = \text{Value-Iter}\left(\{ \widehat{P}_h^n, r_h + b_h^n \}_{h=1}^{H-1} \right)$

UCBVI: Optimistic Model-based Learning

Inside iteration n :

Use all previous data to estimate transitions $\widehat{P}_0^n, \ldots, \widehat{P}_{H-1}^n$

Design reward bonus $b_h^n(s, a), \forall s, a, h$

Optimistic planning with learned model: $\pi^n = \text{Value-Iter}\left(\{ \widehat{P}_h^n, r_h + b_h^n \}_{h=1}^{H-1} \right)$

Collect a new trajectory by executing π^n in the real world $\{P_h\}_{h=0}^{H-1}$ starting from s_0

UCBVI—Part 1: Model Estimation

Let us consider the **very beginning** of episode n:

$$\mathscr{D}_h^n = \{s_h^i, a_h^i, s_{h+1}^i\}_{i=1}^{n-1}, \forall h$$

UCBVI—Part 1: Model Estimation

Let us consider the **very beginning** of episode n:

$$\mathcal{D}_h^n = \{s_h^i, a_h^i, s_{h+1}^i\}_{i=1}^{n-1}, \forall h$$

Let's also maintain some statistics using these datasets:

UCBVI—Part 1: Model Estimation

Let us consider the **very beginning** of episode n:

$$\mathcal{D}_h^n = \{s_h^i, a_h^i, s_{h+1}^i\}_{i=1}^{n-1}, \forall h$$

Let's also maintain some statistics using these datasets:

$$N_h^n(s,a) = \sum_{i=1}^{n-1} \mathbf{1}\{(s_h^i, a_h^i) = (s,a)\}, \forall s,a,h, \quad N_h^n(s,a,s') = \sum_{i=1}^{n-1} \mathbf{1}\{(s_h^i, a_h^i, s_{h+1}^i) = (s,a,s')\}, \forall s,a,h$$

UCBVI—Part 1: Model Estimation

Let us consider the **very beginning** of episode n:

$$\mathcal{D}_h^n = \{s_h^i, a_h^i, s_{h+1}^i\}_{i=1}^{n-1}, \forall h$$

Let's also maintain some statistics using these datasets:

$$N_h^n(s,a) = \sum_{i=1}^{n-1} \mathbf{1}\{(s_h^i, a_h^i) = (s,a)\}, \forall s, a, h, \quad N_h^n(s,a,s') = \sum_{i=1}^{n-1} \mathbf{1}\{(s_h^i, a_h^i, s_{h+1}^i) = (s,a,s')\}, \forall s, a, h$$

Estimate model $\widehat{P}_h^n(s' | s, a), \forall s, a, s', h$ (i.e., MLE):

$$\widehat{P}_h^n(s' | s, a) = \frac{N_h^n(s,a,s')}{N_h^n(s,a)}$$

UCBVI—Part 2: Reward Bonus Design and Value Iteration

Let us consider the very beginning of episode n:

$$\mathcal{D}_h^n = \{s_h^i, a_h^i, s_{h+1}^i\}_{i=1}^{n-1}, \forall h, \quad N_h^n(s,a) = \sum_{i=1}^{n-1} \mathbf{1}\{(s_h^i, a_h^i) = (s,a)\}, \forall s, a, h,$$

UCBVI—Part 2: Reward Bonus Design and Value Iteration

Let us consider the very beginning of episode n:

$$\mathcal{D}_h^n = \{s_h^i, a_h^i, s_{h+1}^i\}_{i=1}^{n-1}, \forall h, \quad N_h^n(s,a) = \sum_{i=1}^{n-1} \mathbf{1}\{(s_h^i, a_h^i) = (s,a)\}, \forall s, a, h,$$

$$b_h^n(s,a) = cH\sqrt{\frac{\ln\left(SAHN/\delta\right)}{N_h^n(s,a)}}$$

UCBVI—Part 2: Reward Bonus Design and Value Iteration

Let us consider the very beginning of episode n:

$$\mathcal{D}_h^n = \{s_h^i, a_h^i, s_{h+1}^i\}_{i=1}^{n-1}, \forall h, \quad N_h^n(s,a) = \sum_{i=1}^{n-1} \mathbf{1}\{(s_h^i, a_h^i) = (s,a)\}, \forall s, a, h,$$

$$b_h^n(s,a) = cH\sqrt{\frac{\ln(SAHN/\delta)}{N_h^n(s,a)}}$$

Encourage to explore new state-actions

UCBVI—Part 2: Reward Bonus Design and Value Iteration

Let us consider the very beginning of episode n:

$$\mathcal{D}_h^n = \{s_h^i, a_h^i, s_{h+1}^i\}_{i=1}^{n-1}, \forall h, \quad N_h^n(s,a) = \sum_{i=1}^{n-1} \mathbf{1}\{(s_h^i, a_h^i) = (s,a)\}, \forall s, a, h,$$

$$b_h^n(s,a) = cH\sqrt{\frac{\ln\left(SAHN/\delta\right)}{N_h^n(s,a)}}$$

Encourage to explore new state-actions

Value Iteration (aka DP) at episode n using $\{\widehat{P}_h^n\}_h$ **and** $\{r_h + b_h^n\}_h$

UCBVI—Part 2: Reward Bonus Design and Value Iteration

Let us consider the very beginning of episode n:

$$\mathscr{D}_h^n = \{s_h^i, a_h^i, s_{h+1}^i\}_{i=1}^{n-1}, \forall h, \quad N_h^n(s,a) = \sum_{i=1}^{n-1} \mathbf{1}\{(s_h^i, a_h^i) = (s,a)\}, \forall s, a, h,$$

$$b_h^n(s,a) = cH\sqrt{\frac{\ln(SAHN/\delta)}{N_h^n(s,a)}}$$

Encourage to explore new state-actions

Value Iteration (aka DP) at episode n using $\{\widehat{P}_h^n\}_h$ **and** $\{r_h + b_h^n\}_h$

$$\widehat{V}_H^n(s) = 0, \forall s$$

UCBVI—Part 2: Reward Bonus Design and Value Iteration

Let us consider the very beginning of episode n:

$$\mathcal{D}_h^n = \{s_h^i, a_h^i, s_{h+1}^i\}_{i=1}^{n-1}, \forall h, \quad N_h^n(s,a) = \sum_{i=1}^{n-1} \mathbf{1}\{(s_h^i, a_h^i) = (s,a)\}, \forall s, a, h,$$

$$b_h^n(s,a) = cH\sqrt{\frac{\ln(SAHN/\delta)}{N_h^n(s,a)}}$$

Encourage to explore new state-actions

Value Iteration (aka DP) at episode n using $\{\widehat{P}_h^n\}_h$ **and** $\{r_h + b_h^n\}_h$

$$\widehat{V}_H^n(s) = 0, \forall s \qquad \widehat{Q}_h^n(s,a) = \min\left\{ r_h(s,a) + b_h^n(s,a) + \widehat{P}_h^n(\cdot \mid s,a) \cdot \widehat{V}_{h+1}^n, \quad H \right\}, \forall s, a$$

UCBVI—Part 2: Reward Bonus Design and Value Iteration

Let us consider the very beginning of episode n:

$$\mathscr{D}_h^n = \{s_h^i, a_h^i, s_{h+1}^i\}_{i=1}^{n-1}, \forall h, \quad N_h^n(s,a) = \sum_{i=1}^{n-1} \mathbf{1}\{(s_h^i, a_h^i) = (s,a)\}, \forall s, a, h,$$

$$b_h^n(s,a) = cH\sqrt{\frac{\ln(SAHN/\delta)}{N_h^n(s,a)}}$$

Encourage to explore new state-actions

Value Iteration (aka DP) at episode n using $\{\widehat{P}_h^n\}_h$ **and** $\{r_h + b_h^n\}_h$

$$\widehat{V}_H^n(s) = 0, \forall s \qquad \widehat{Q}_h^n(s,a) = \min\left\{r_h(s,a) + b_h^n(s,a) + \widehat{P}_h^n(\cdot \mid s,a) \cdot \widehat{V}_{h+1}^n, \quad H\right\}, \forall s, a$$

$$\widehat{V}_h^n(s) = \max_a \widehat{Q}_h^n(s,a), \quad \pi_h^n(s) = \arg\max_a \widehat{Q}_h^n(s,a), \forall s$$

UCBVI—Part 2: Reward Bonus Design and Value Iteration

Let us consider the very beginning of episode n:

$$\mathcal{D}_h^n = \{s_h^i, a_h^i, s_{h+1}^i\}_{i=1}^{n-1}, \forall h, \quad N_h^n(s,a) = \sum_{i=1}^{n-1} \mathbf{1}\{(s_h^i, a_h^i) = (s,a)\}, \forall s, a, h,$$

$$b_h^n(s,a) = cH\sqrt{\frac{\ln(SAHN/\delta)}{N_h^n(s,a)}}$$

Encourage to explore new state-actions

Value Iteration (aka DP) at episode n using $\{\widehat{P}_h^n\}_h$ **and** $\{r_h + b_h^n\}_h$

$$\widehat{V}_H^n(s) = 0, \forall s \qquad \widehat{Q}_h^n(s,a) = \min\left\{r_h(s,a) + b_h^n(s,a) + \widehat{P}_h^n(\cdot \mid s,a)\cdot \widehat{V}_{h+1}^n, \quad H\right\}, \forall s, a$$

$$\widehat{V}_h^n(s) = \max_a \widehat{Q}_h^n(s,a), \quad \pi_h^n(s) = \arg\max_a \widehat{Q}_h^n(s,a), \forall s$$

$$\left\| \widehat{V}_h^n \right\|_\infty \le H, \forall h, n$$

UCBVI: Put All Together

For $n = 1 \to N$:

1. Set $N_h^n(s,a) = \sum_{i=1}^{n-1} \mathbf{1}\{(s_h^i, a_h^i) = (s,a)\}, \forall s, a, h$
2. Set $N_h^n(s,a,s') = \sum_{i=1}^{n-1} \mathbf{1}\{(s_h^i, a_h^i, s_{h+1}^i) = (s,a,s')\}, \forall s, a, a', h$
3. Estimate $\widehat{P}^n$: $\widehat{P}_h^n(s' \mid s,a) = \dfrac{N_h^n(s,a,s')}{N_h^n(s,a)}, \forall s, a, s', h$
4. Plan: $\pi^n = VI\left(\{\widehat{P}_h^n, r_h + b_h^n\}_h\right)$, with $b_h^n(s,a) = cH\sqrt{\dfrac{\ln(SAHN/\delta)}{N_h^n(s,a)}}$
5. Execute $\pi^n : \{s_0^n, a_0^n, r_0^n, \ldots, s_{H-1}^n, a_{H-1}^n, r_{H-1}^n, s_H^n\}$

Theorem: UCBVI Regret Bound

We will prove the following in the recitation:

$$\mathbb{E}\left[\text{Regret}_N\right] := \mathbb{E}\left[\sum_{n=1}^{N}\left(V^\star - V^{\pi^n}\right)\right] \leq \widetilde{O}\left(H^2\sqrt{S^2AN}\right)$$

Theorem: UCBVI Regret Bound

We will prove the following in the recitation:

$$\mathbb{E}\left[\text{Regret}_N\right] := \mathbb{E}\left[\sum_{n=1}^{N}\left(V^\star - V^{\pi^n}\right)\right] \leq \widetilde{O}\left(H^2\sqrt{S^2AN}\right)$$

Remarks:

Note that we consider expected regret here (policy π^n is a random quantity).
High probability version is not hard to get (need to do a martingale argument)

Theorem: UCBVI Regret Bound

We will prove the following in the recitation:

$$\mathbb{E}\left[\text{Regret}_N\right] := \mathbb{E}\left[\sum_{n=1}^{N}\left(V^\star - V^{\pi^n}\right)\right] \leq \widetilde{O}\left(H^2\sqrt{S^2AN}\right)$$

Remarks:

Note that we consider expected regret here (policy π^n is a random quantity).
High probability version is not hard to get (need to do a martingale argument)

Dependency on H and S are suboptimal; but the **same** algorithm can achieve $H^2\sqrt{SAN}$ in the leading term [Azar et.al 17 ICML]

Key Intuition behind the theorem:

VI at episode n under $\{\widehat{P}^n_h\}_h$ **and** $\{r_h + b^n_h\}_h$

$$\widehat{V}^n_H(s) = 0, \forall s \quad \widehat{Q}^n_h(s,a) = \min\left\{ r_h(s,a) + b^n_h(s,a) + \widehat{P}^n_h(\cdot \mid s,a) \cdot \widehat{V}^n_{h+1}, \quad H \right\}, \forall s, a$$

$$\widehat{V}^n_h(s) = \max_a \widehat{Q}^n_h(s,a), \quad \pi^n_h(s) = \arg\max_a \widehat{Q}^n_h(s,a), \forall s$$

Key Intuition behind the theorem:

VI at episode n under $\{ \widehat{P}_h^n \}_h$ **and** $\{ r_h + b_h^n \}_h$

$$\widehat{V}_H^n(s) = 0, \forall s \quad \widehat{Q}_h^n(s,a) = \min\left\{ r_h(s,a) + b_h^n(s,a) + \widehat{P}_h^n(\cdot \mid s,a) \cdot \widehat{V}_{h+1}^n, \quad H \right\}, \forall s, a$$

$$\widehat{V}_h^n(s) = \max_a \widehat{Q}_h^n(s,a), \quad \pi_h^n(s) = \arg\max_a \widehat{Q}_h^n(s,a), \forall s$$

Key lemma 1: optimism — our bonus is large enough s.t. $\widehat{V}_h^n(s) \geq V_h^\star(s), \forall s, h$

Key Intuition behind the theorem:

VI at episode n under $\{\widehat{P}_h^n\}_h$ and $\{r_h + b_h^n\}_h$

$$\widehat{V}_H^n(s) = 0, \forall s \quad \widehat{Q}_h^n(s,a) = \min\left\{ r_h(s,a) + b_h^n(s,a) + \widehat{P}_h^n(\cdot \mid s,a) \cdot \widehat{V}_{h+1}^n, \quad H \right\}, \forall s,a$$

$$\widehat{V}_h^n(s) = \max_a \widehat{Q}_h^n(s,a), \quad \pi_h^n(s) = \arg\max_a \widehat{Q}_h^n(s,a), \forall s$$

Key lemma 1: optimism — our bonus is large enough s.t. $\widehat{V}_h^n(s) \geq V_h^\star(s), \forall s, h$

Key lemma 2: regret decomposition:

Regret at iter n $= V_0^\star(s_0) - V_0^{\pi^n}(s_0) \leq \widehat{V}_0^n(s_0) - V_0^{\pi^n}(s_0)$

Key Intuition behind the theorem:

VI at episode n under $\{\widehat{P}_h^n\}_h$ and $\{r_h + b_h^n\}_h$

$$\widehat{V}_H^n(s) = 0, \forall s \quad \widehat{Q}_h^n(s,a) = \min\left\{ r_h(s,a) + b_h^n(s,a) + \widehat{P}_h^n(\cdot \mid s,a) \cdot \widehat{V}_{h+1}^n, \quad H \right\}, \forall s,a$$

$$\widehat{V}_h^n(s) = \max_a \widehat{Q}_h^n(s,a), \quad \pi_h^n(s) = \arg\max_a \widehat{Q}_h^n(s,a), \forall s$$

Key lemma 1: optimism — our bonus is large enough s.t. $\widehat{V}_h^n(s) \geq V_h^\star(s), \forall s, h$

Key lemma 2: regret decomposition:

Regret at iter n $= V_0^\star(s_0) - V_0^{\pi^n}(s_0) \leq \widehat{V}_0^n(s_0) - V_0^{\pi^n}(s_0)$

$$\leq \sum_h \mathbb{E}_{s,a\sim d_h^{\pi^n}} \left[b_h^n(s,a) + (\widehat{P}_h^n(\cdot \mid s,a) - P_h^\star(\cdot \mid s,a))^\top \widehat{V}_{h+1}^n \right]$$

Key Intuition behind the theorem:

VI at episode n under $\{\widehat{P}_h^n\}_h$ and $\{r_h + b_h^n\}_h$

$$\widehat{V}_H^n(s) = 0, \forall s \quad \widehat{Q}_h^n(s,a) = \min\left\{ r_h(s,a) + b_h^n(s,a) + \widehat{P}_h^n(\cdot \mid s,a) \cdot \widehat{V}_{h+1}^n, \quad H \right\}, \forall s, a$$

$$\widehat{V}_h^n(s) = \max_a \widehat{Q}_h^n(s,a), \quad \pi_h^n(s) = \arg\max_a \widehat{Q}_h^n(s,a), \forall s$$

Key lemma 1: optimism — our bonus is large enough s.t. $\widehat{V}_h^n(s) \geq V_h^\star(s), \forall s, h$

Key lemma 2: regret decomposition:

Regret at iter n $= V_0^\star(s_0) - V_0^{\pi^n}(s_0) \leq \widehat{V}_0^n(s_0) - V_0^{\pi^n}(s_0)$

$$\leq \sum_h \mathbb{E}_{s,a\sim d_h^{\pi^n}} \left[b_h^n(s,a) + (\widehat{P}_h^n(\cdot \mid s,a) - P_h^\star(\cdot \mid s,a))^\top \widehat{V}_{h+1}^n \right]$$

If π^n is suboptimal, i.e., $V^\star(s_0) - V^{\pi^n}(s_0)$ is large, then π^n must visit some (s,a) pairs with large bonus $b(s,a)$ or wrong $\widehat{P}(\cdot \mid s,a)$

Summary

1. Basics of MDPs:

Bellman Equation / Optimality; two planning algs: Value Iteration and Policy Iteration

2. Policy Gradient:

Vanilla PG formulation & Natural Policy Gradient with their global convergence

3. Efficient exploration in tabular MDPs:

The UCB-VI algorithm via the principle of optimism in the face of uncertainty