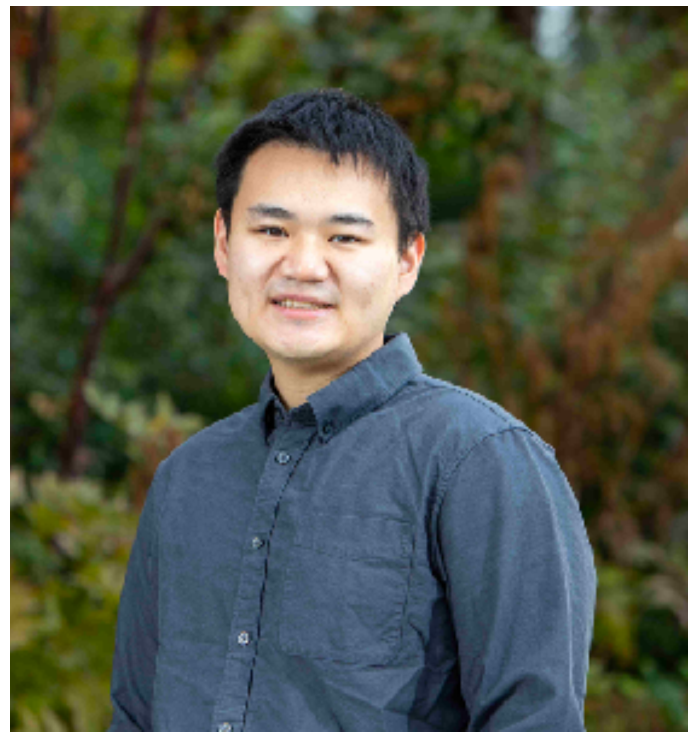
Linear Convergence of Natural Policy Gradient Methods with Log-Linear Policies

Rui Yuan[1,4], Simon S. Du[2], Robert M. Gower[3], Alessandro Lazaric[1], Lin Xiao[1]

[1]Meta AI, [2]University of Washington, [3]Flatiron Institute, [4]Télécom Paris

International Conference on Learning Representations (ICLR), 2023

Thank you to



Simon S. Du[2] Robert M. Gower[3] Alessandro Lazaric[1] Lin Xiao[1]

[1]Meta AI [2]University of Washington [3]Flatiron Institute

Impressive Reinforcement Learning (RL) Results

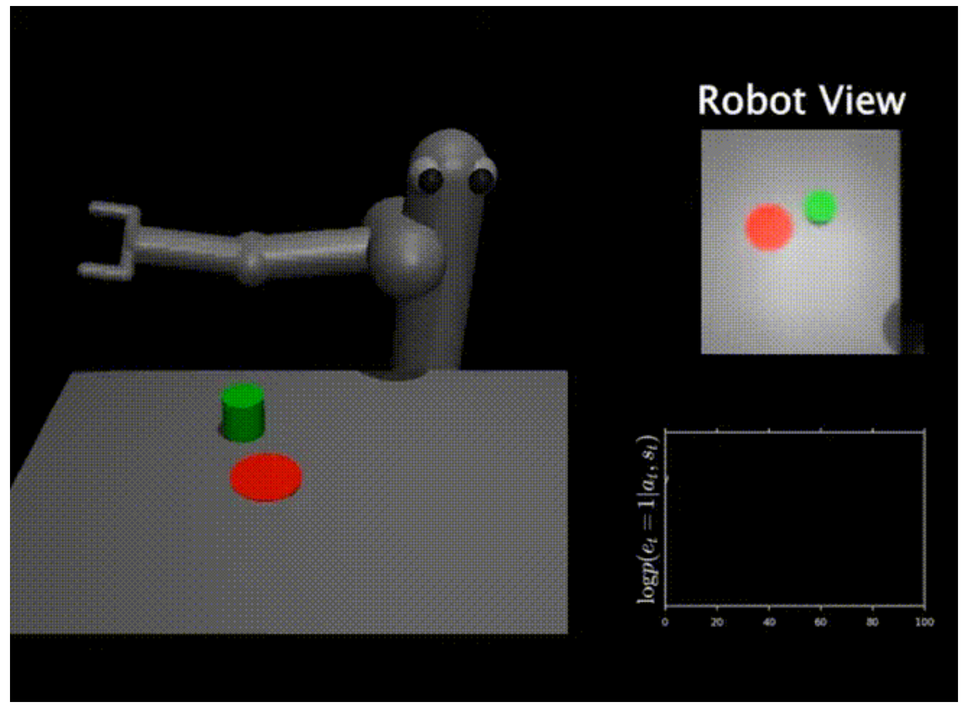
Impressive Reinforcement Learning (RL) Results

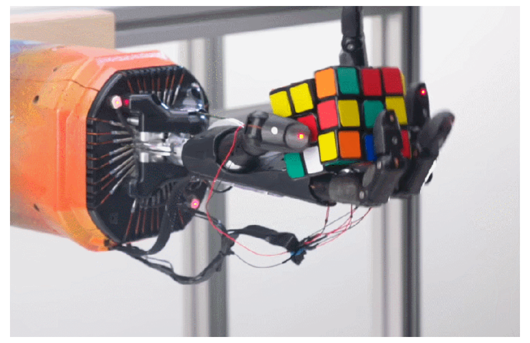
Board Game

Impressive Reinforcement Learning (RL) Results

Board Game

Robotic Manipulation

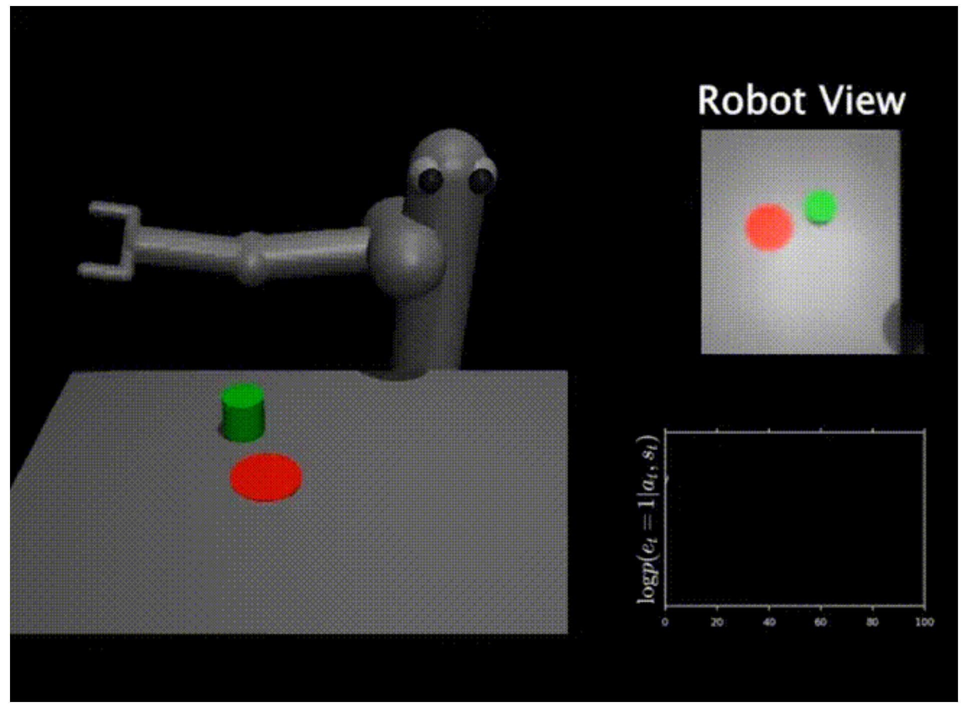


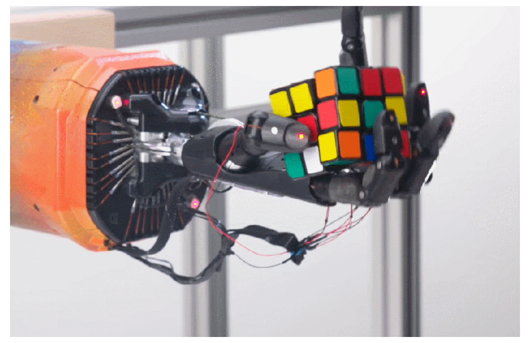


Impressive Reinforcement Learning (RL) Results

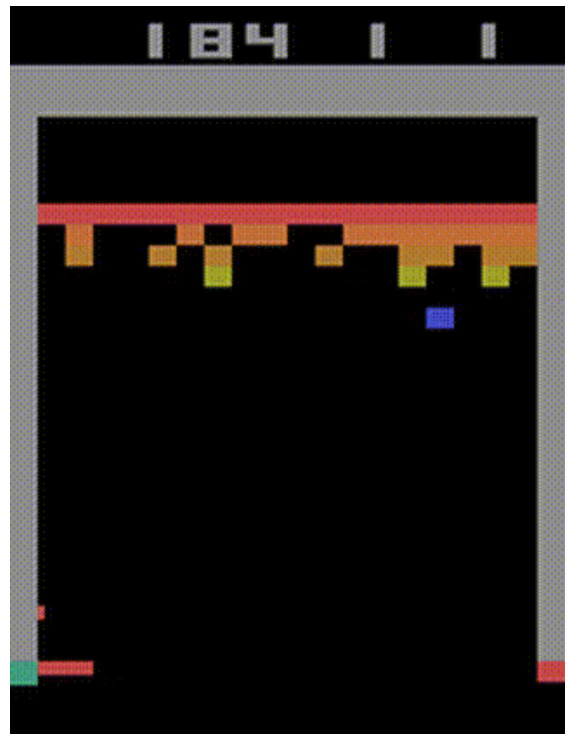
Board Game

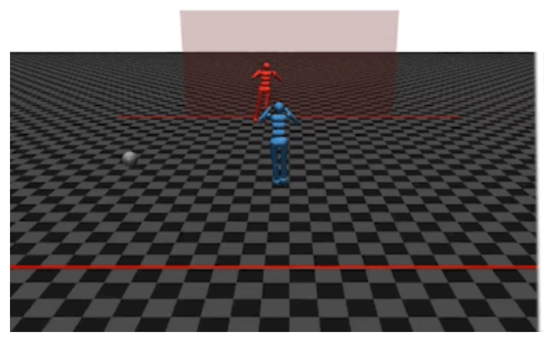
Robotic Manipulation

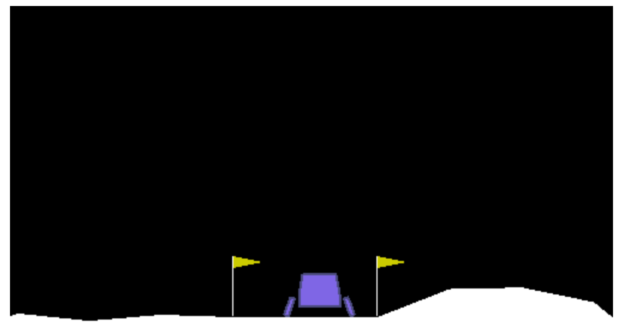




Game Playing

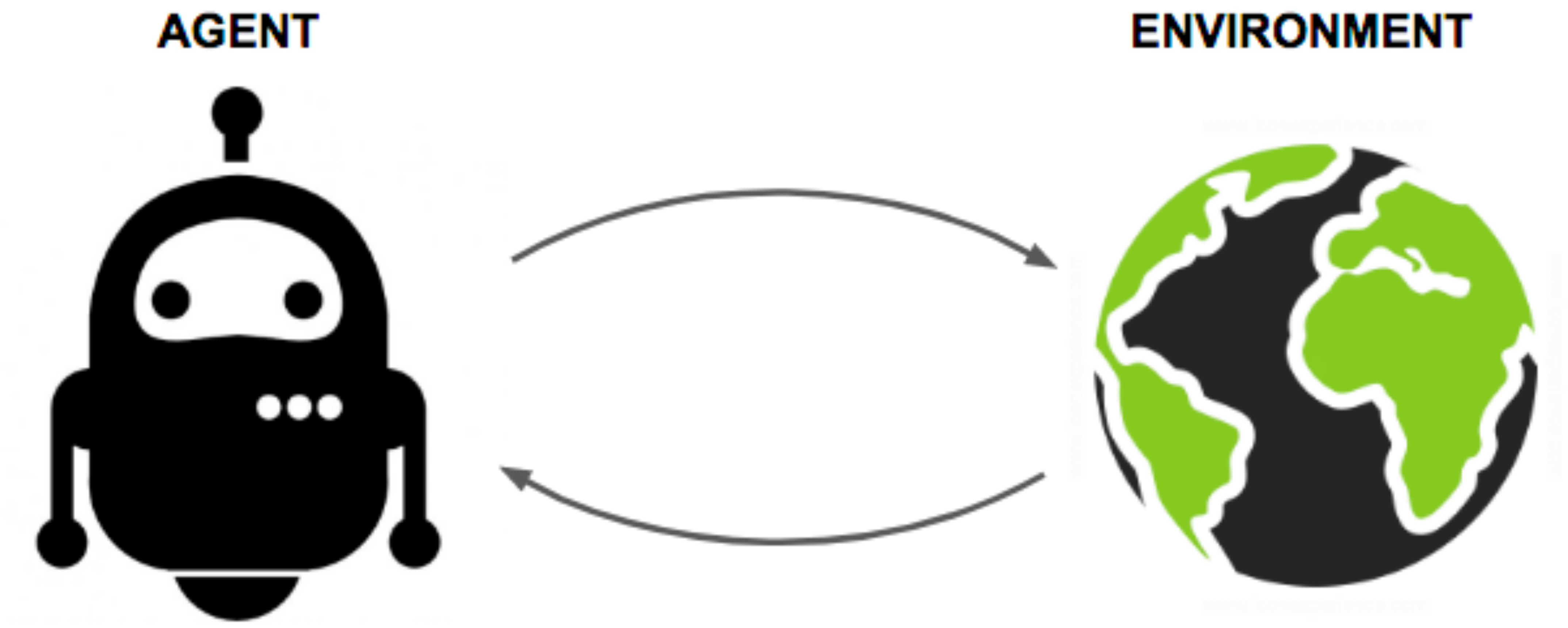






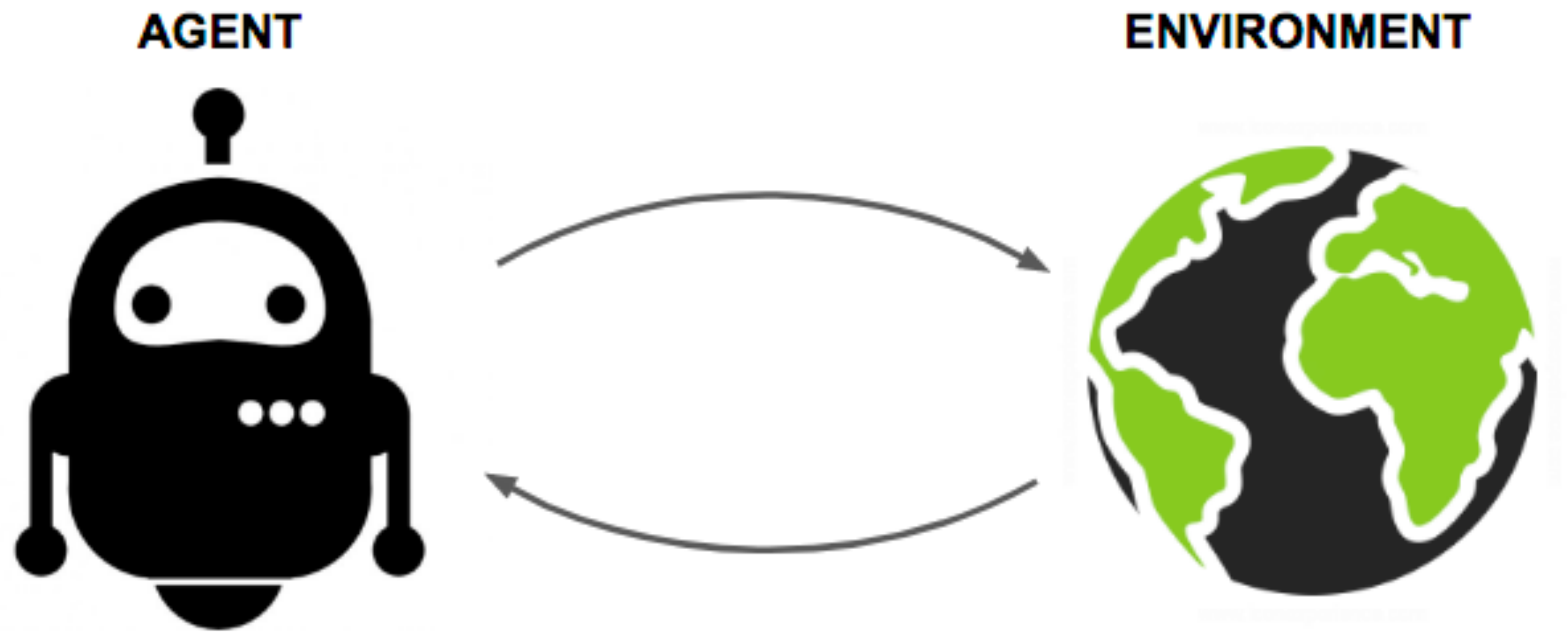
Reinforcement Learning

Sequential decision making problems



Reinforcement Learning

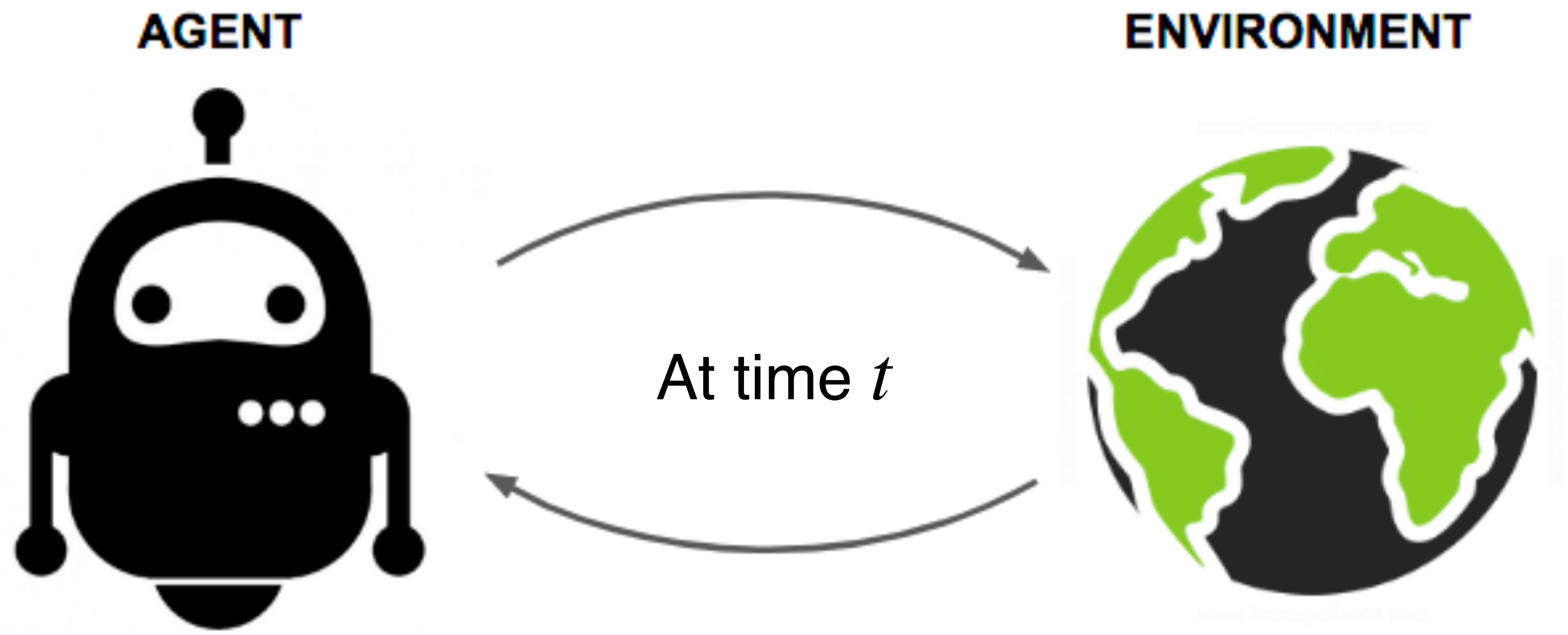
Sequential decision making problems



Markov decision Process (MDP)

Reinforcement Learning

Sequential decision making problems

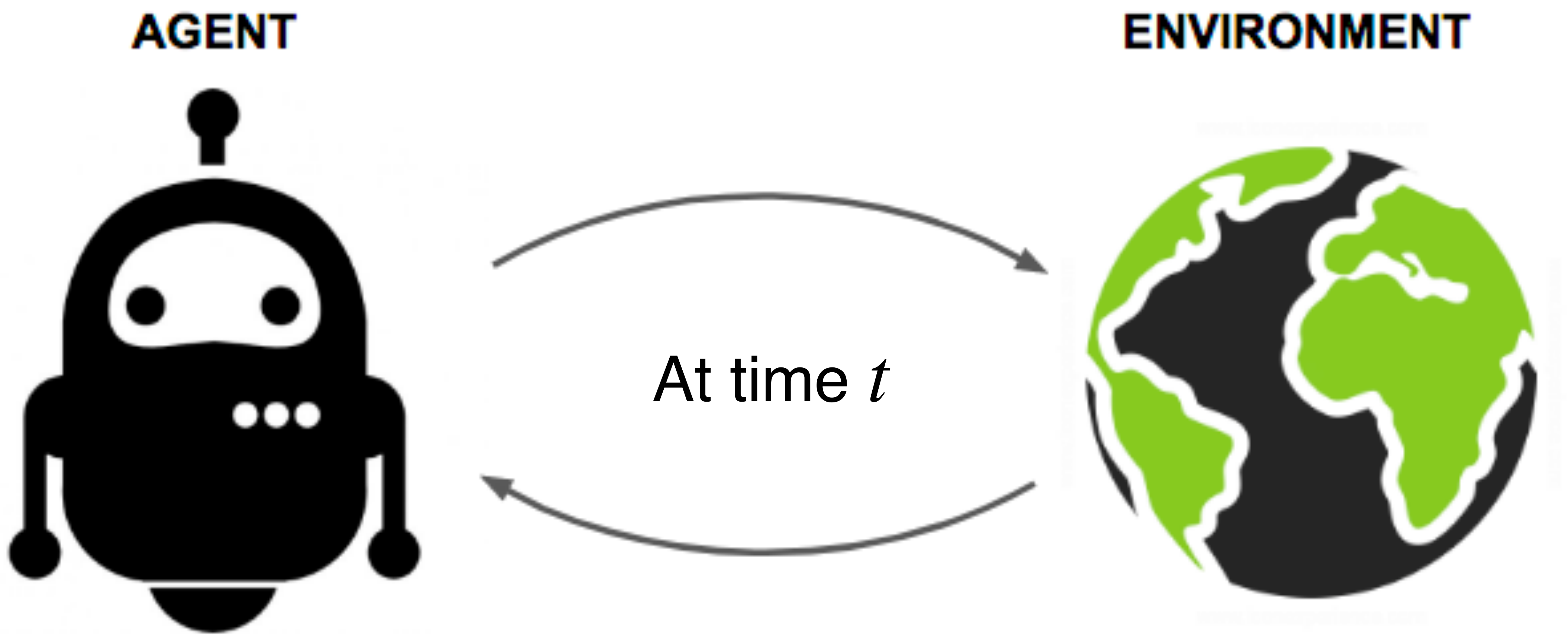


Markov decision Process (MDP)

Reinforcement Learning

Sequential decision making problems

- State $s_t \in \mathcal{S}$

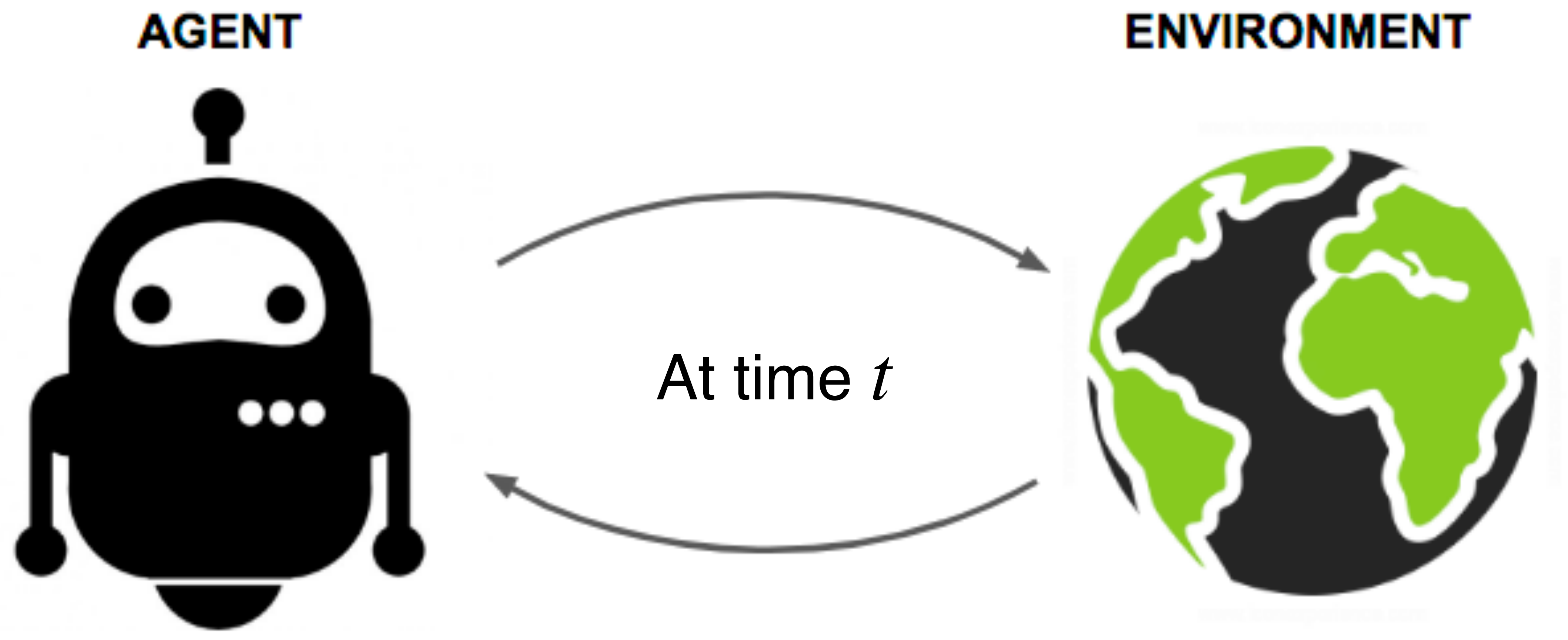


Markov decision Process (MDP)

Reinforcement Learning

Sequential decision making problems

- State $s_t \in \mathcal{S}$

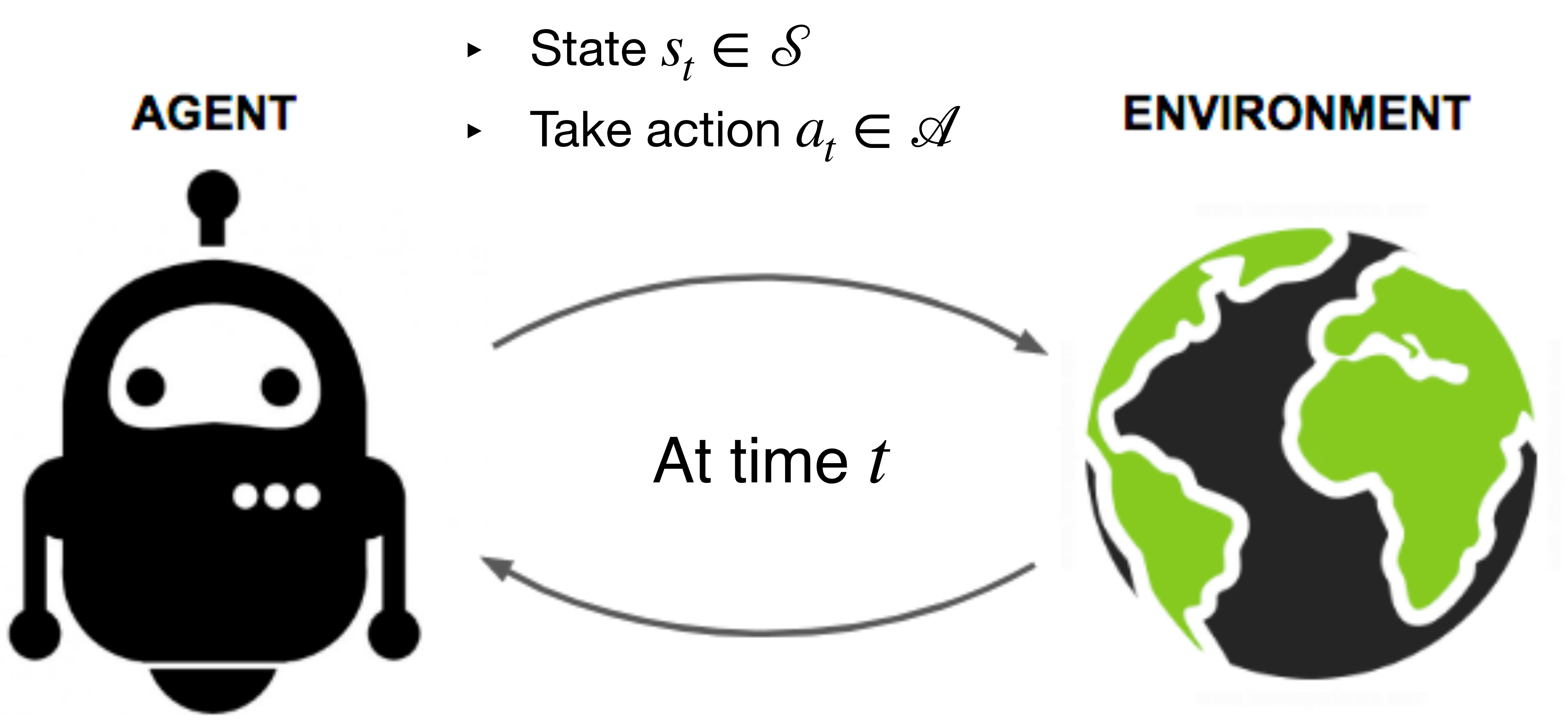


Markov decision Process (MDP)

- State space $\mathcal{S}$

Reinforcement Learning

Sequential decision making problems



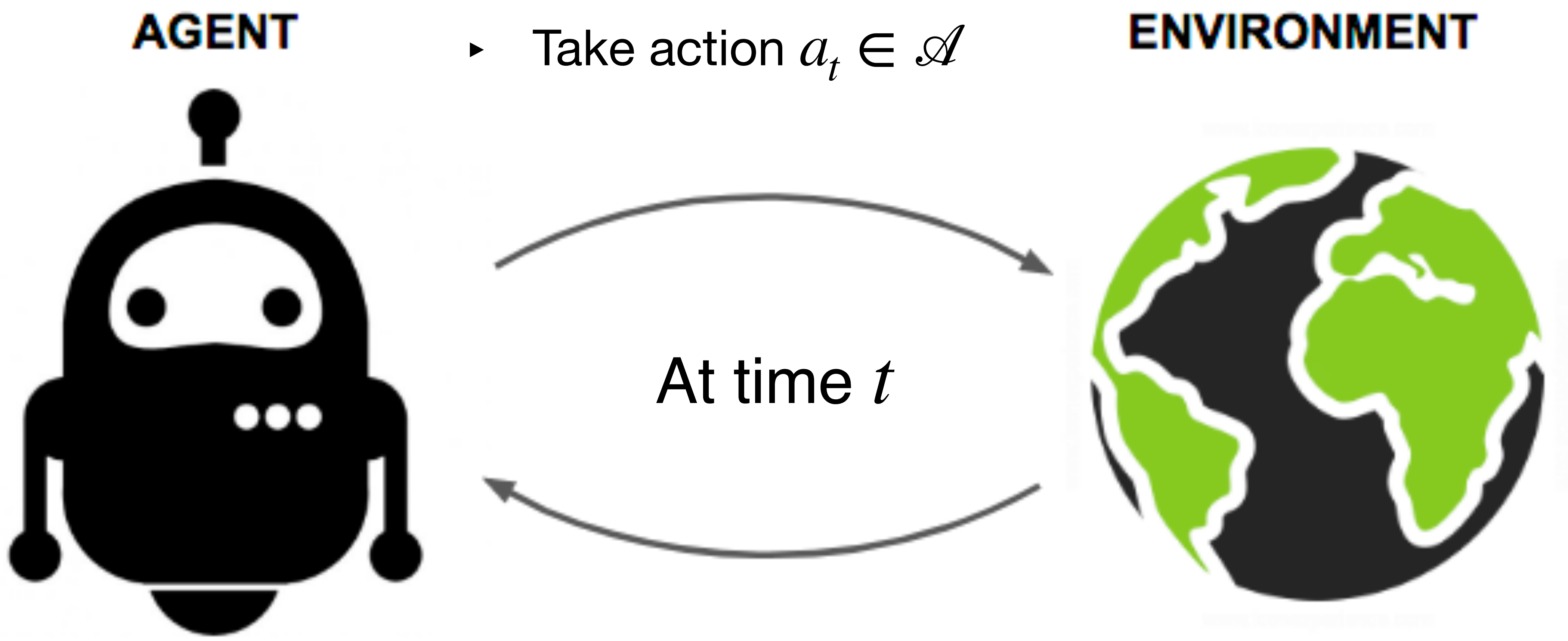
Markov decision Process (MDP)

- State space $\mathcal{S}$

Reinforcement Learning

Sequential decision making problems

- State $s_t \in \mathcal{S}$
- Take action $a_t \in \mathcal{A}$

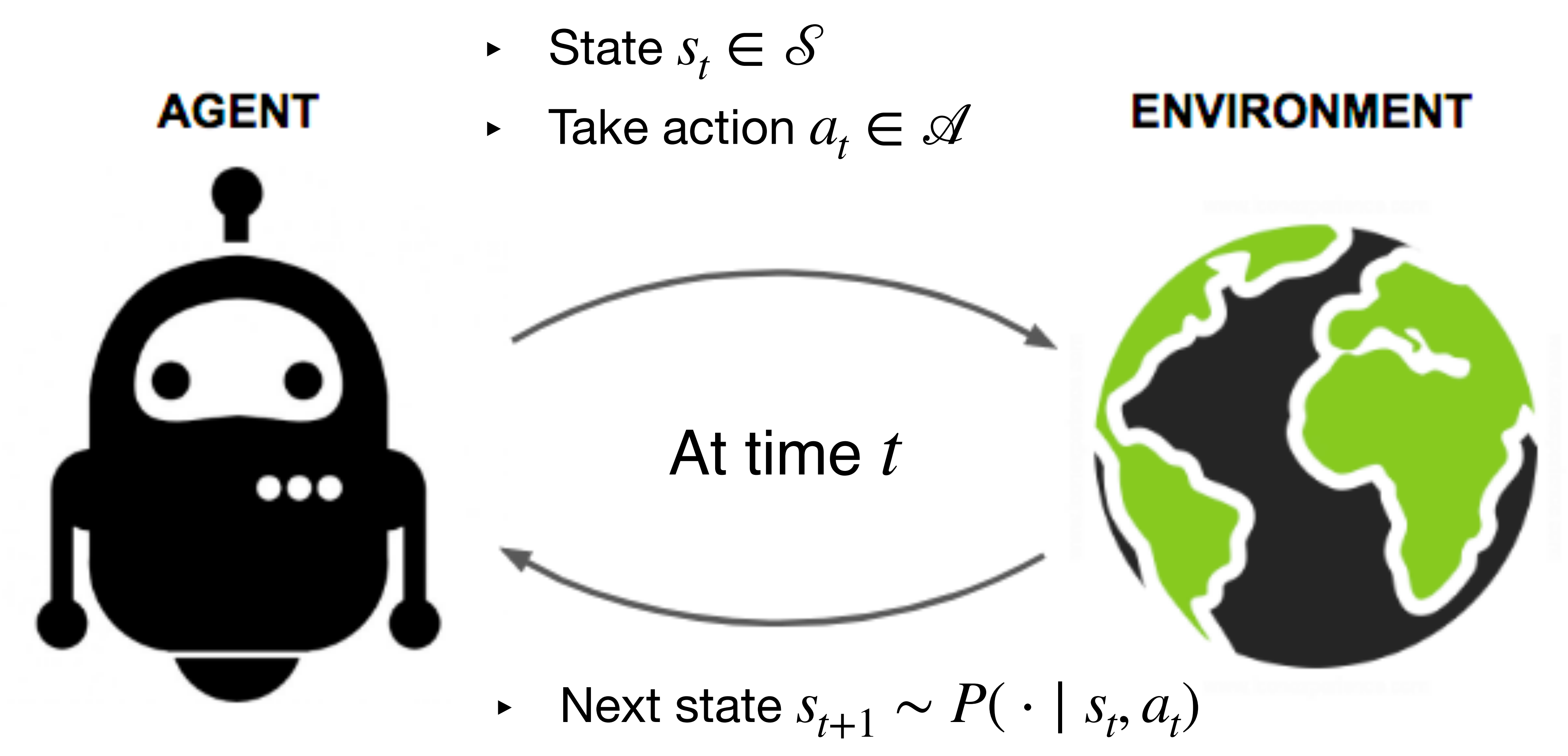


Markov decision Process (MDP)

- State space $\mathcal{S}$
- Action space $\mathcal{A}$

Reinforcement Learning

Sequential decision making problems



Markov decision Process (MDP)

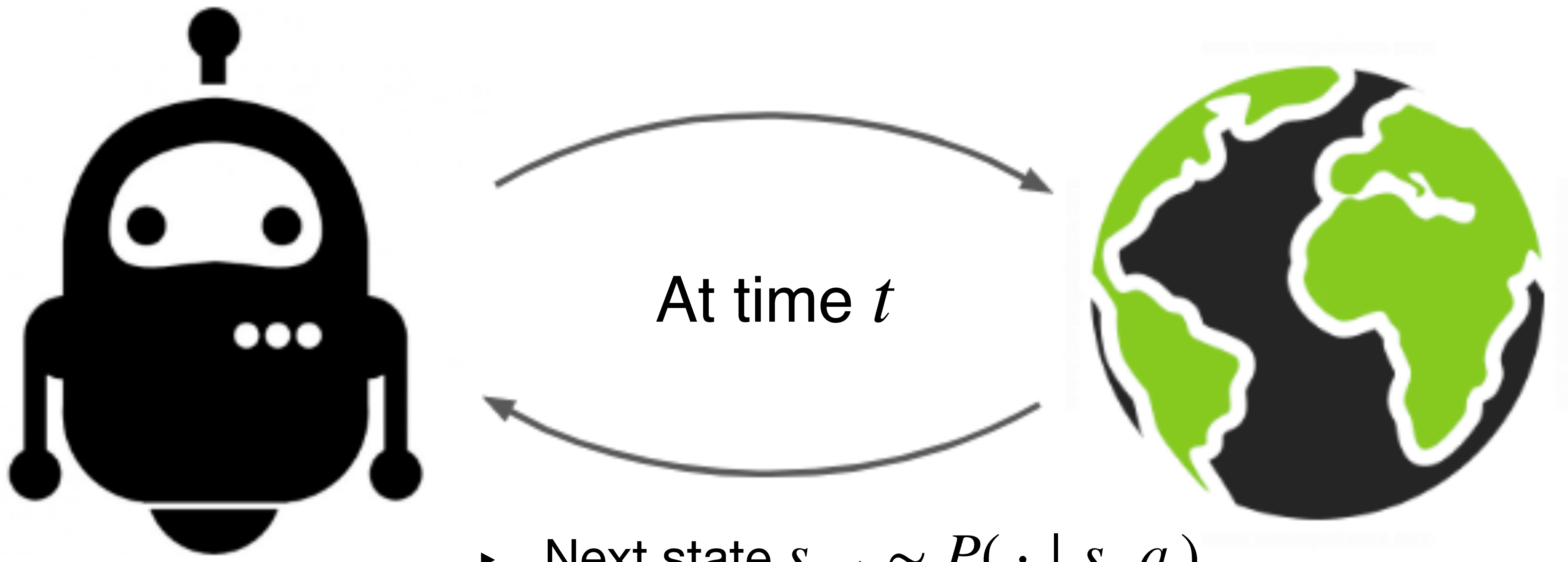
- State space $\mathcal{S}$
- Action space $\mathcal{A}$

Reinforcement Learning

Sequential decision making problems

- State $s_t \in \mathcal{S}$
- Take action $a_t \in \mathcal{A}$

AGENT **ENVIRONMENT**



- Next state $s_{t+1} \sim P(\cdot \mid s_t, a_t)$

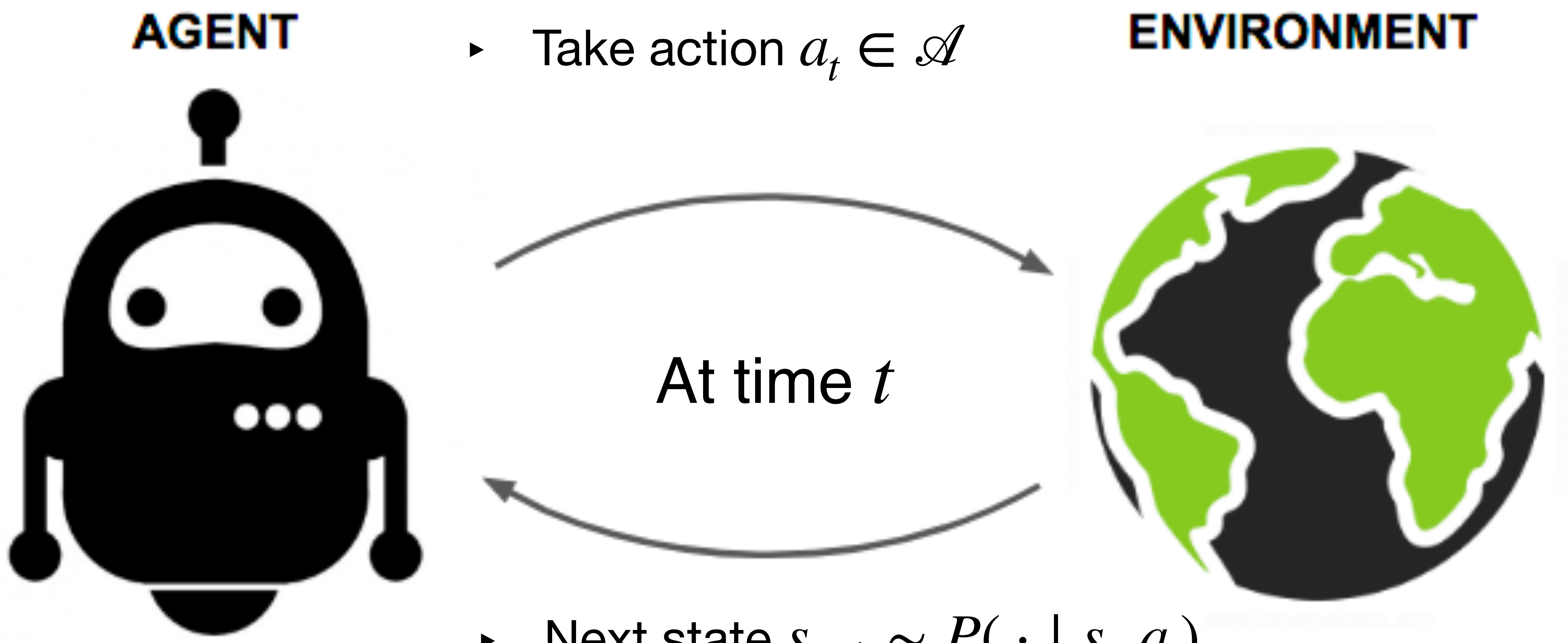
Markov decision Process (MDP)

- State space $\mathcal{S}$
- Action space $\mathcal{A}$
- Transition probabilities P

Reinforcement Learning

Sequential decision making problems

- State $s_t \in \mathcal{S}$
- Take action $a_t \in \mathcal{A}$



- Next state $s_{t+1} \sim P(\cdot \mid s_t, a_t)$
- Get a cost $c(s_t, a_t)$

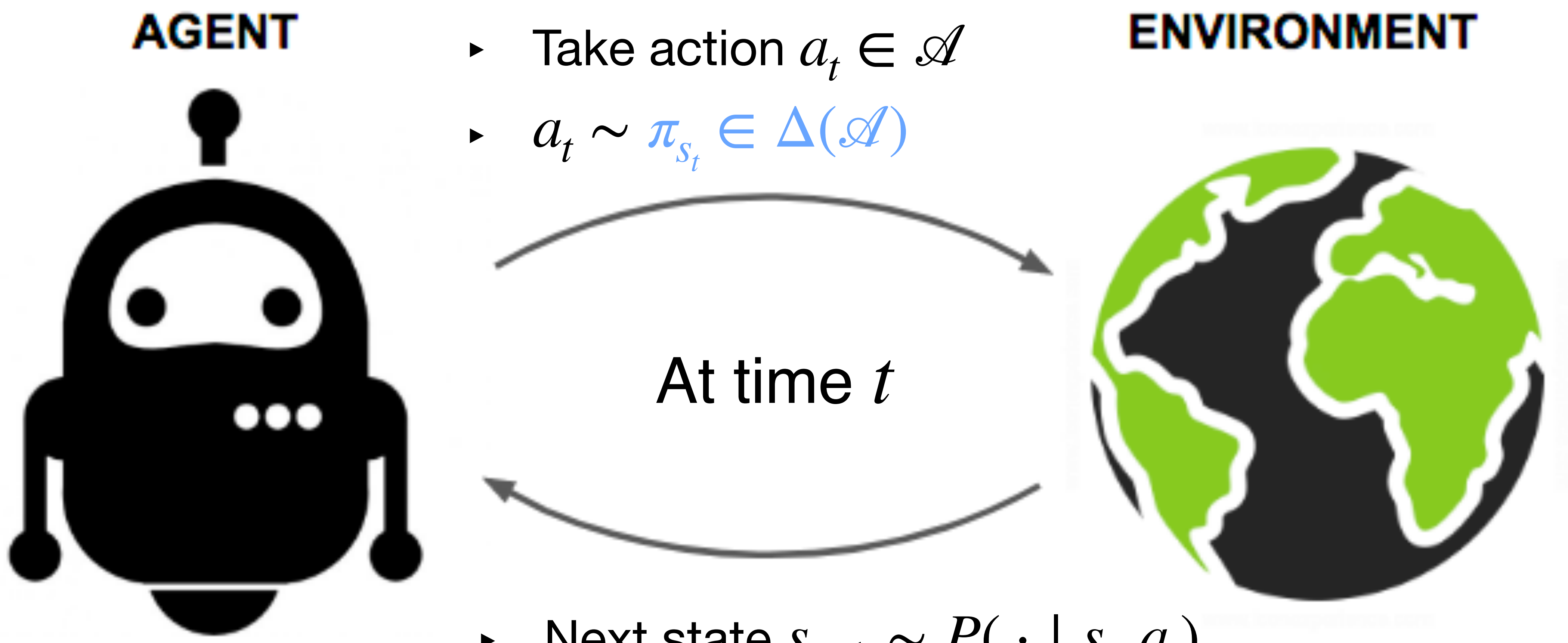
Markov decision Process (MDP)

- State space $\mathcal{S}$
- Action space $\mathcal{A}$
- Transition probabilities P

Reinforcement Learning

Sequential decision making problems

- State $s_t \in \mathcal{S}$
- Take action $a_t \in \mathcal{A}$
- $a_t \sim \pi_{s_t} \in \Delta(\mathcal{A})$



- Next state $s_{t+1} \sim P(\cdot \mid s_t, a_t)$
- Get a cost $c(s_t, a_t)$

Markov decision Process (MDP)

- State space $\mathcal{S}$
- Action space $\mathcal{A}$
- Transition probabilities P

Reinforcement Learning

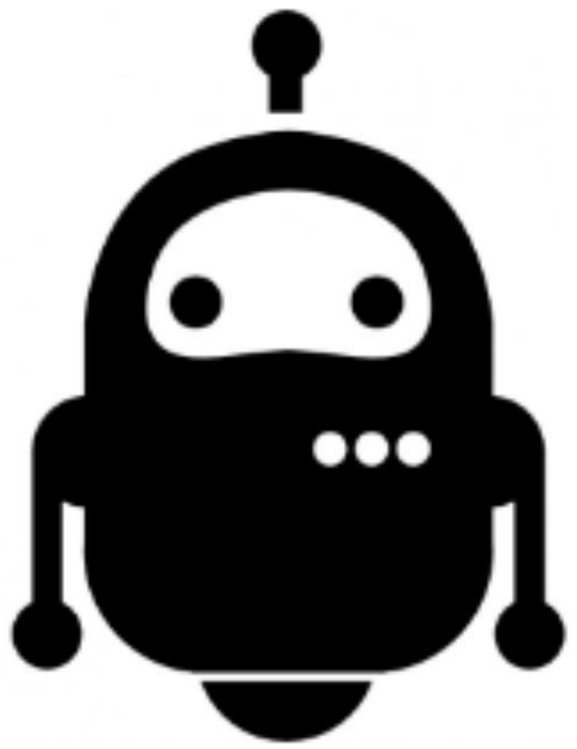
Sequential decision making problems

Policy $\pi : \mathcal{S} \to \Delta(\mathcal{A})$,

$\pi_{s_t,a_t} \in \mathbb{R}$ is the density of the distribution over actions at $s_t \in \mathcal{S}$

- State $s_t \in \mathcal{S}$
- Take action $a_t \in \mathcal{A}$
- $a_t \sim \boxed{\pi_{s_t} \in \Delta(\mathcal{A})}$

AGENT



At time t

ENVIRONMENT

- Next state $s_{t+1} \sim P(\cdot \mid s_t, a_t)$
- Get a cost $c(s_t, a_t)$

Markov decision Process (MDP)

- State space $\mathcal{S}$
- Action space $\mathcal{A}$
- Transition probabilities P

Reinforcement Learning

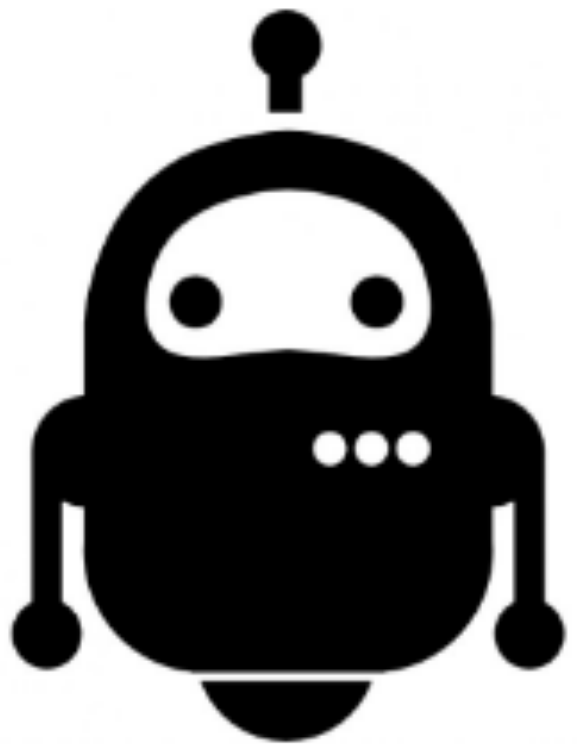
Sequential decision making problems

Policy $\pi : \mathscr{S} \to \Delta(\mathscr{A})$,

$\pi_{s_t,a_t} \in \mathbb{R}$ is the density of the distribution over actions at $s_t \in \mathscr{S}$

- State $s_t \in \mathscr{S}$
- Take action $a_t \in \mathscr{A}$
- $a_t \sim \boxed{\pi_{s_t} \in \Delta(\mathscr{A})}$

AGENT



At time t

ENVIRONMENT

- Next state $s_{t+1} \sim P(\cdot \mid s_t, a_t)$
- Get a cost $c(s_t, a_t)$

Markov decision Process (MDP)

- State space $\mathscr{S}$
- Action space $\mathscr{A}$
- Transition probabilities P

Solve an MDP to minimize total expected cost (a.k.a. policy optimization)

$$\arg\min_{\pi} V_\rho(\pi) := \mathbb{E}_{s_0 \sim \rho,\ a_t \sim \pi_{s_t},\ s_{t+1} \sim P(\cdot|s_t,a_t)} \left[\sum_{t=0}^{\infty} \gamma^t c(s_t, a_t) \right]$$

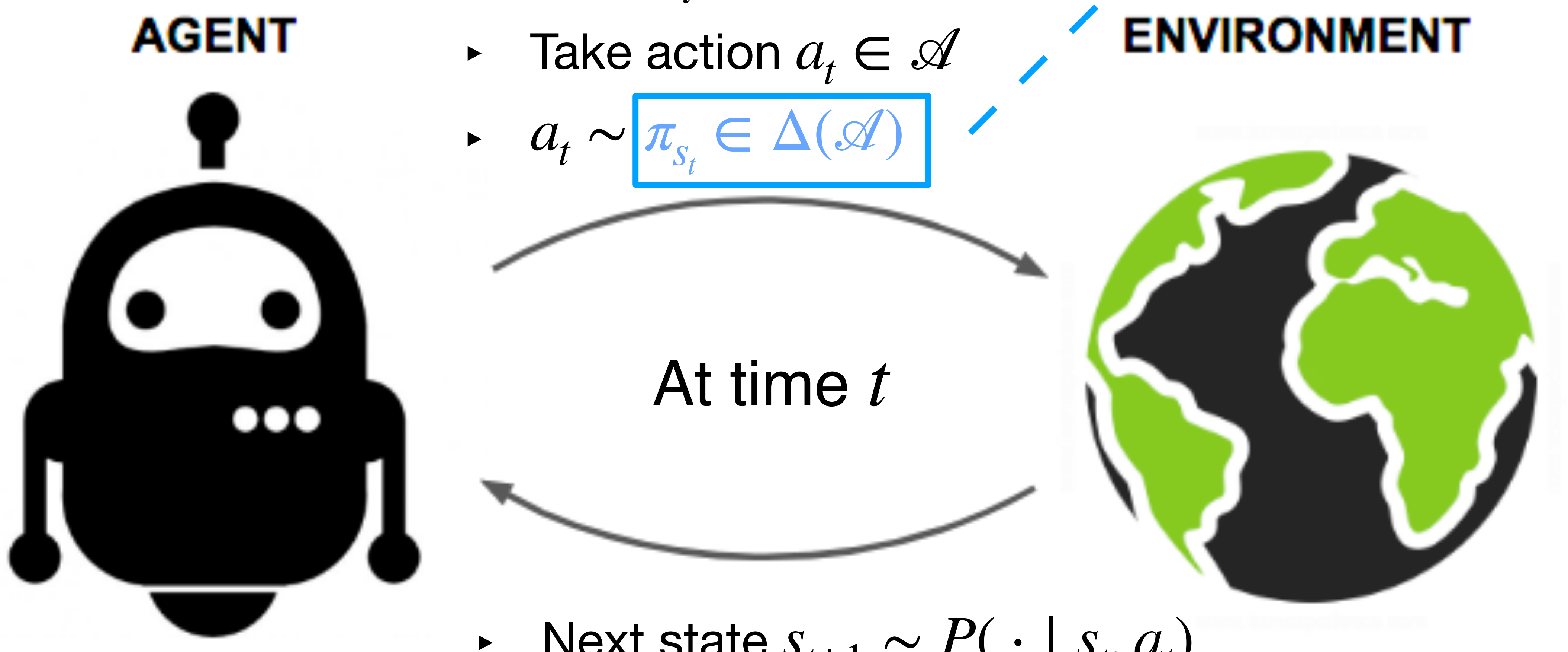
Reinforcement Learning

Sequential decision making problems

Policy $\pi : \mathcal{S} \to \Delta(\mathcal{A})$,

$\pi_{s_t, a_t} \in \mathbb{R}$ is the density of the distribution over actions at $s_t \in \mathcal{S}$

- State $s_t \in \mathcal{S}$
- Take action $a_t \in \mathcal{A}$
- $a_t \sim \boxed{\pi_{s_t} \in \Delta(\mathcal{A})}$



- Next state $s_{t+1} \sim P(\cdot \mid s_t, a_t)$
- Get a cost $c(s_t, a_t)$

Markov decision Process (MDP)

- State space $\mathcal{S}$
- Action space $\mathcal{A}$
- Transition probabilities P

Solve an MDP to minimize total expected cost (a.k.a. policy optimization)

$$\arg\min_{\pi} V_\rho(\pi) := \mathbb{E}_{s_0 \sim \rho,\ a_t \sim \pi_{s_t},\ s_{t+1} \sim P(\cdot|s_t, a_t)} \left[\boxed{\sum_{t=0}^{\infty} \gamma^t c(s_t, a_t)} \right]$$

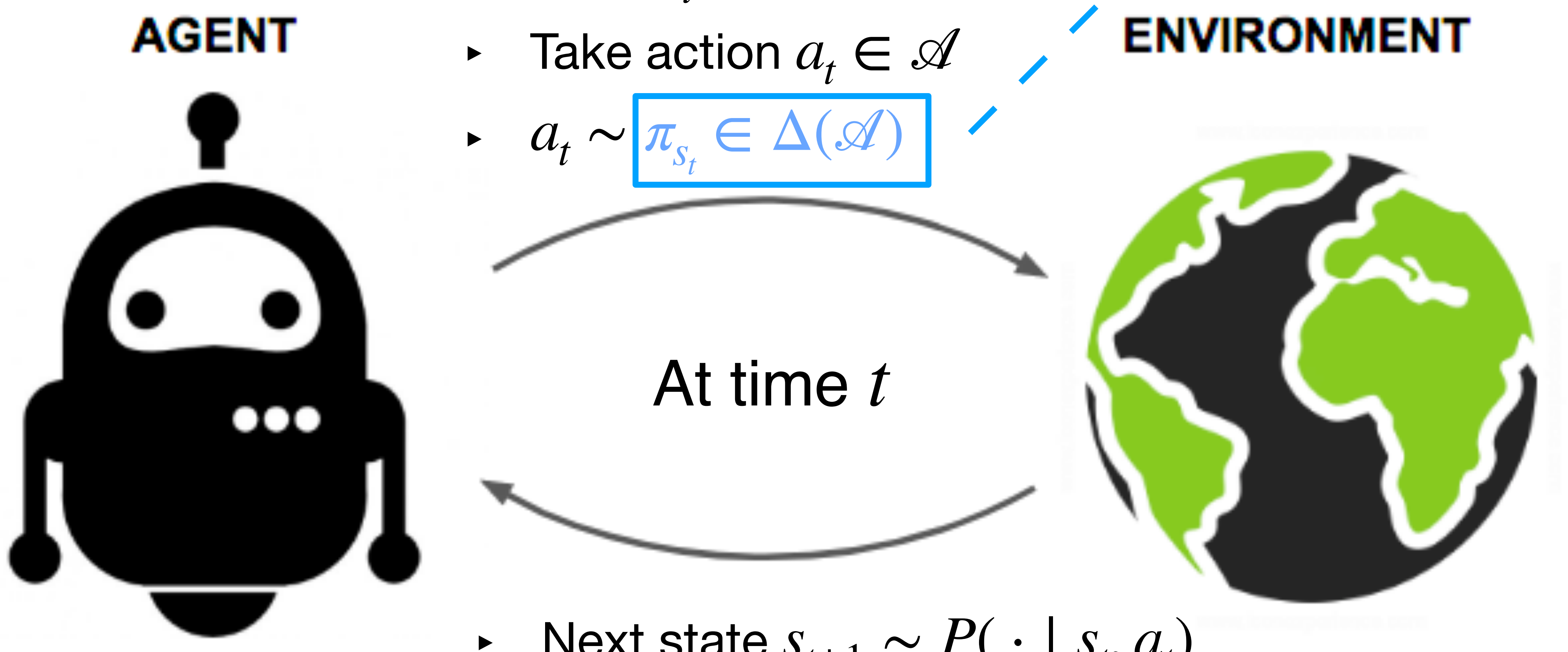
Cost function

Reinforcement Learning

Sequential decision making problems

Policy $\pi : \mathcal{S} \to \Delta(\mathcal{A})$,
$\pi_{s_t,a_t} \in \mathbb{R}$ is the density of the distribution over actions at $s_t \in \mathcal{S}$

- State $s_t \in \mathcal{S}$
- Take action $a_t \in \mathcal{A}$
- $a_t \sim \boxed{\pi_{s_t} \in \Delta(\mathcal{A})}$



- Next state $s_{t+1} \sim P(\cdot \mid s_t, a_t)$
- Get a cost $c(s_t, a_t)$

Markov decision Process (MDP)

- State space $\mathcal{S}$
- Action space $\mathcal{A}$
- Transition probabilities P

Solve an MDP to minimize total expected cost (a.k.a. policy optimization)

$$\arg\min_{\pi} V_\rho(\pi) := \mathbb{E}_{\boxed{s_0\sim\rho,}\, a_t\sim\pi_{s_t},\, s_{t+1}\sim P(\cdot|s_t,a_t)} \left[\boxed{\sum_{t=0}^{\infty} \gamma^t c(s_t, a_t)} \right] \longrightarrow \text{Cost function}$$

Reinforcement Learning

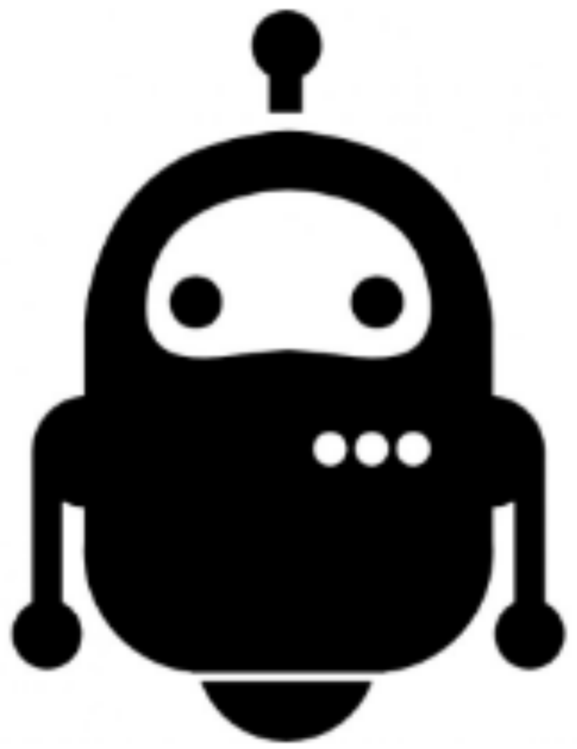
Sequential decision making problems

Policy $\pi : \mathcal{S} \rightarrow \Delta(\mathcal{A})$,

$\pi_{s_t, a_t} \in \mathbb{R}$ is the density of the distribution over actions at $s_t \in \mathcal{S}$

- State $s_t \in \mathcal{S}$
- Take action $a_t \in \mathcal{A}$
- $a_t \sim \boxed{\pi_{s_t} \in \Delta(\mathcal{A})}$

AGENT



At time t

ENVIRONMENT

- Next state $s_{t+1} \sim P(\cdot \mid s_t, a_t)$
- Get a cost $c(s_t, a_t)$

Markov decision Process (MDP)

- State space $\mathcal{S}$
- Action space $\mathcal{A}$
- Transition probabilities P
- Initial state distribution ρ

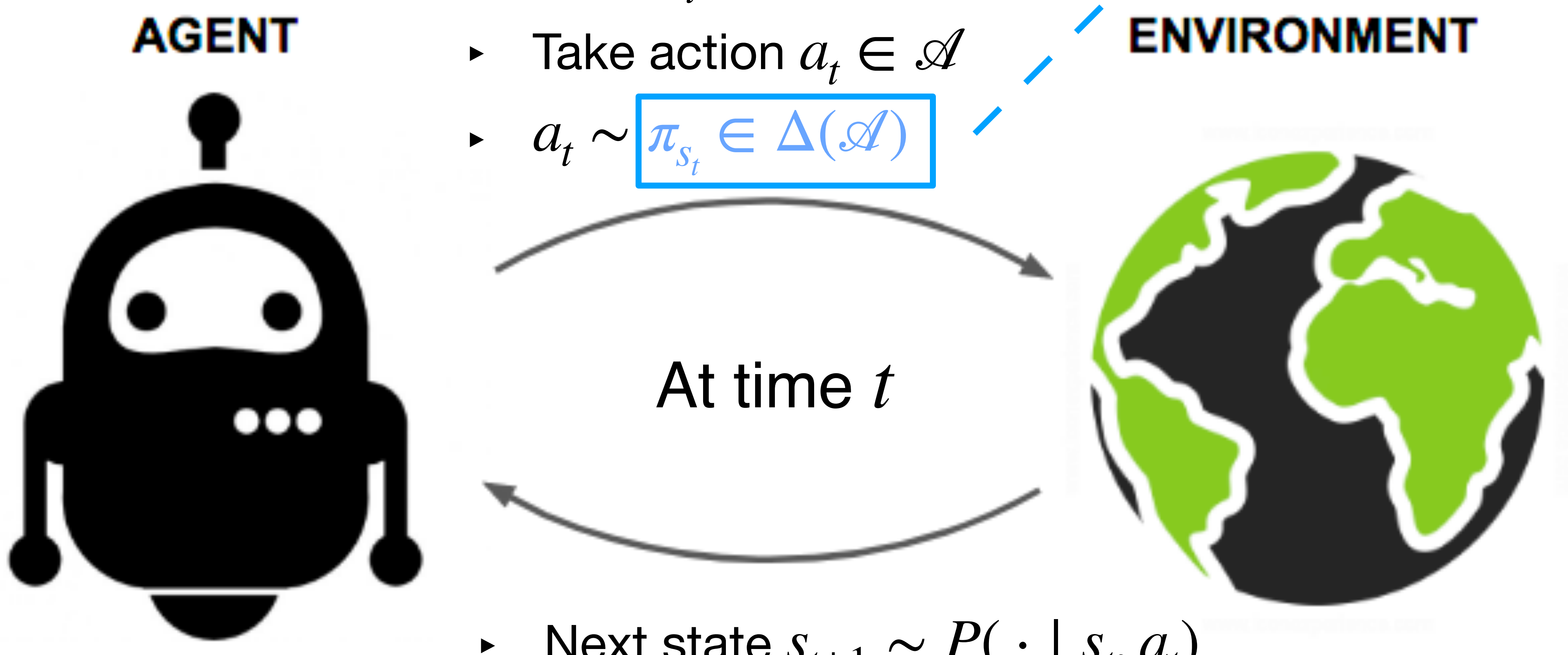
Solve an MDP to minimize total expected cost (a.k.a. policy optimization)

$$\arg\min_{\pi} V_\rho(\pi) := \mathbb{E}_{\boxed{s_0 \sim \rho,}\ a_t \sim \pi_{s_t},\ s_{t+1} \sim P(\cdot | s_t, a_t)} \left[\boxed{\sum_{t=0}^{\infty} \gamma^t c(s_t, a_t)} \right] \longrightarrow \text{Cost function}$$

Reinforcement Learning

Sequential decision making problems

Policy $\pi : \mathcal{S} \to \Delta(\mathcal{A})$,
$\pi_{s_t,a_t} \in \mathbb{R}$ is the density of the distribution over actions at $s_t \in \mathcal{S}$



- State $s_t \in \mathcal{S}$
- Take action $a_t \in \mathcal{A}$
- $a_t \sim \boxed{\pi_{s_t} \in \Delta(\mathcal{A})}$

At time t

- Next state $s_{t+1} \sim P(\cdot \mid s_t, a_t)$
- Get a cost $c(s_t, a_t)$

Markov decision Process (MDP)

- State space $\mathcal{S}$
- Action space $\mathcal{A}$
- Transition probabilities P
- Initial state distribution ρ

Solve an MDP to minimize total expected cost (a.k.a. policy optimization)

$$\arg\min_{\pi} V_\rho(\pi) := \mathbb{E}_{\boxed{s_0\sim\rho},\, a_t\sim\pi_{s_t},\, s_{t+1}\sim P(\cdot|s_t,a_t)}\left[\boxed{\sum_{t=0}^{\infty}\boxed{\gamma^t} c(s_t,a_t)}\right]$$

→ Cost function

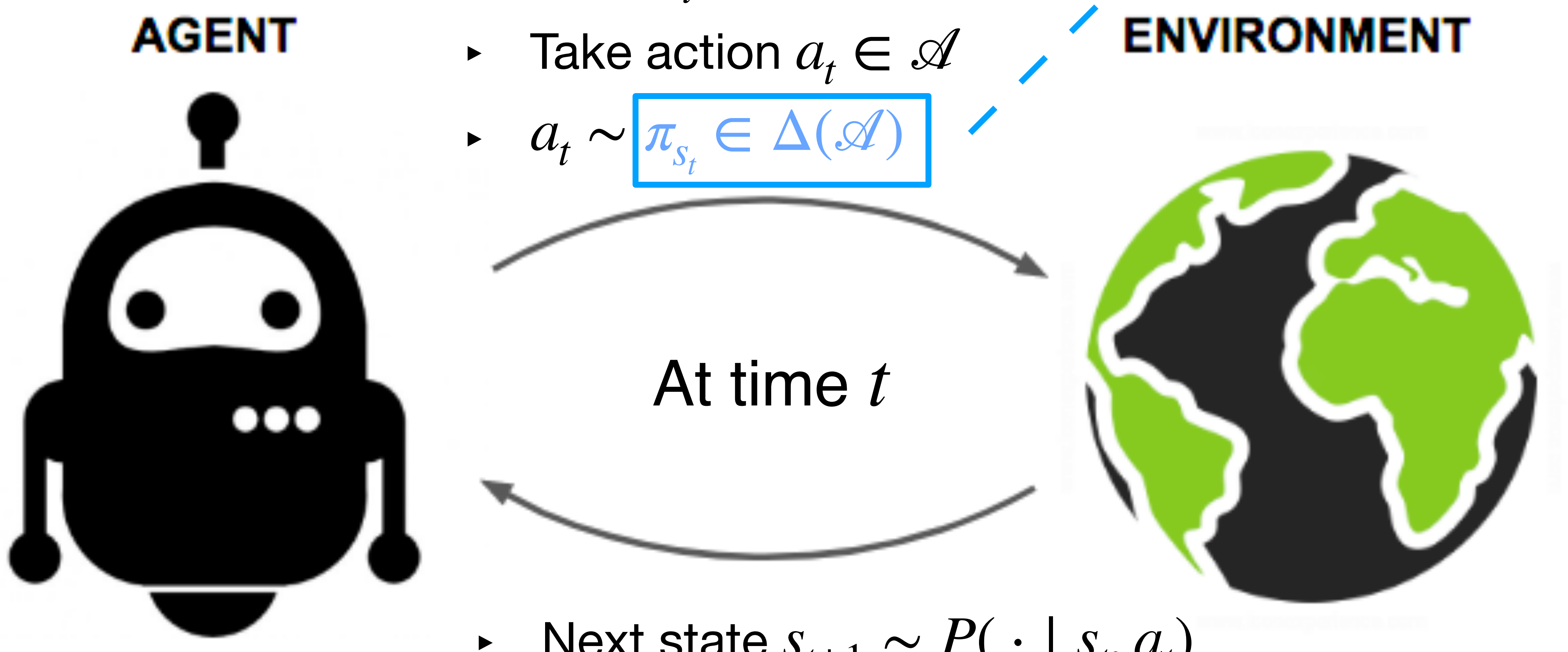
Reinforcement Learning

Sequential decision making problems

Policy $\pi : \mathscr{S} \to \Delta(\mathscr{A})$,

$\pi_{s_t, a_t} \in \mathbb{R}$ is the density of the distribution over actions at $s_t \in \mathscr{S}$

- State $s_t \in \mathscr{S}$
- Take action $a_t \in \mathscr{A}$
- $a_t \sim \boxed{\pi_{s_t} \in \Delta(\mathscr{A})}$



- Next state $s_{t+1} \sim P(\cdot \mid s_t, a_t)$
- Get a cost $c(s_t, a_t)$

Markov decision Process (MDP)

- State space $\mathscr{S}$
- Action space $\mathscr{A}$
- Transition probabilities P
- Initial state distribution ρ
- Discounted factor $\gamma \in (0,1)$

Solve an MDP to minimize total expected cost (a.k.a. policy optimization)

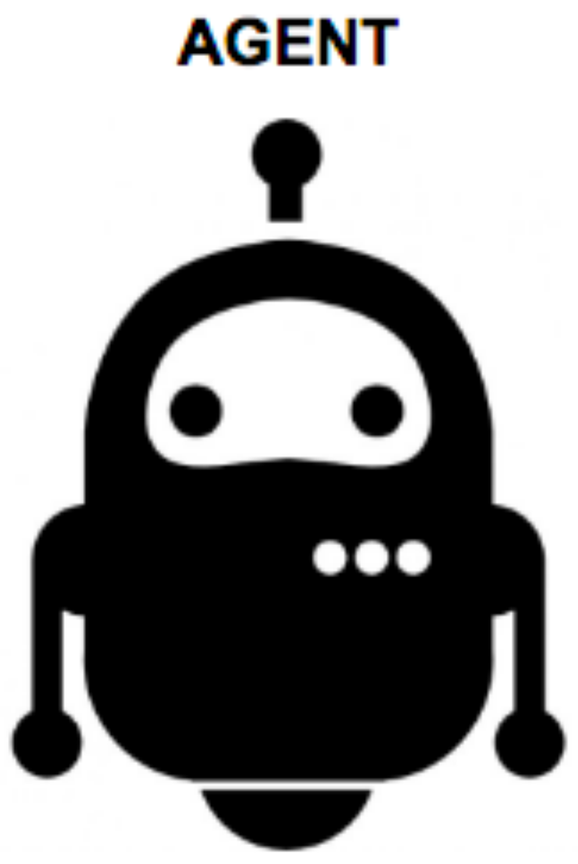
$$\arg\min_{\pi} V_\rho(\pi) := \mathbb{E}_{\boxed{s_0 \sim \rho,}\, a_t \sim \pi_{s_t},\, s_{t+1} \sim P(\cdot | s_t, a_t)} \left[\boxed{\sum_{t=0}^{\infty} \boxed{\gamma^t} c(s_t, a_t)} \right] \rightarrow \text{Cost function}$$

Reinforcement Learning

Sequential decision making problems

Policy $\pi : \mathcal{S} \to \Delta(\mathcal{A})$,
$\pi_{s_t,a_t} \in \mathbb{R}$ is the density of the distribution over actions at $s_t \in \mathcal{S}$

- State $s_t \in \mathcal{S}$
- Take action $a_t \in \mathcal{A}$
- $a_t \sim \boxed{\pi_{s_t} \in \Delta(\mathcal{A})}$



At time t

- Next state $s_{t+1} \sim P(\cdot \mid s_t, a_t)$
- Get a cost $c(s_t, a_t)$

Markov decision Process (MDP)

- State space $\mathcal{S}$
- Action space $\mathcal{A}$
- Transition probabilities P
- Initial state distribution ρ
- Discounted factor $\gamma \in (0,1)$

Solve an MDP to minimize total expected cost (a.k.a. policy optimization)

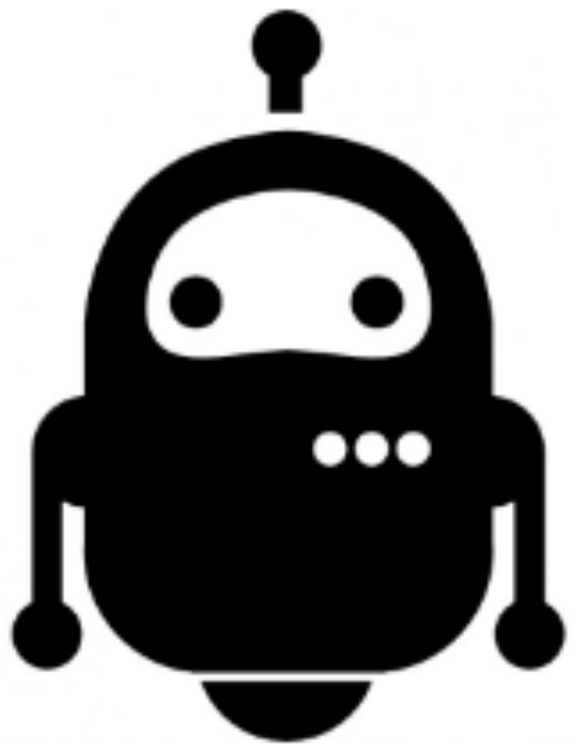
$$\arg\min_{\theta \in \mathbb{R}^d} V_\rho(\theta) := \mathbb{E}_{s_0 \sim \rho,\ a_t \sim \pi_{s_t}(\theta),\ s_{t+1} \sim P(\cdot|s_t,a_t)} \left[\sum_{t=0}^{\infty} \gamma^t c(s_t, a_t) \right]$$

Reinforcement Learning

Sequential decision making problems

Policy $\pi : \mathscr{S} \to \Delta(\mathscr{A})$,
$\pi_{s_t,a_t} \in \mathbb{R}$ is the density of the distribution over actions at $s_t \in \mathscr{S}$

AGENT



- State $s_t \in \mathscr{S}$
- Take action $a_t \in \mathscr{A}$
- $a_t \sim \boxed{\pi_{s_t} \in \Delta(\mathscr{A})}$

At time t

ENVIRONMENT

- Next state $s_{t+1} \sim P(\cdot \mid s_t, a_t)$
- Get a cost $c(s_t, a_t)$

Markov decision Process (MDP)

- State space $\mathscr{S}$
- Action space $\mathscr{A}$
- Transition probabilities P
- Initial state distribution ρ
- Discounted factor $\gamma \in (0,1)$

Solve an MDP to minimize total expected cost (a.k.a. policy optimization)

$$\arg\min_{\theta \in \mathbb{R}^d} V_\rho(\theta) := \mathbb{E}_{s_0 \sim \rho,\ \boxed{a_t \sim \pi_{s_t}(\theta),}\ s_{t+1} \sim P(\cdot|s_t,a_t)} \left[\sum_{t=0}^{\infty} \gamma^t c(s_t, a_t) \right]$$

Objective: $\arg\min_{\theta \in \mathbb{R}^d} V_\rho(\theta)$

Objective: $\arg\min_{\theta \in \mathbb{R}^d} V_\rho(\theta)$

$$\boxed{\theta^{(k+1)} = \theta^{(k)} - \eta_k \nabla_\theta V_\rho(\theta^{(k)})}$$

Objective: $\arg\min_{\theta \in \mathbb{R}^d} V_\rho(\theta)$

$$\boxed{\theta^{(k+1)} = \theta^{(k)} - \boxed{\eta_k} \nabla_\theta V_\rho(\theta^{(k)})}$$

Step size

Objective: $\arg\min_{\theta \in \mathbb{R}^d} V_\rho(\theta)$

$$\boxed{\theta^{(k+1)} = \theta^{(k)} - \underbrace{\boxed{\eta_k}}_{\text{Step size}} \boxed{\nabla_\theta V_\rho(\theta^{(k)})}}$$

Gradient of $V_\rho(\theta)$

Policy gradient (PG) methods

Objective: $\arg\min_{\theta \in \mathbb{R}^d} V_\rho(\theta)$

$$\boxed{\theta^{(k+1)} = \theta^{(k)} - \underbrace{\eta_k}_{\text{Step size}} \underbrace{\nabla_\theta V_\rho(\theta^{(k)})}_{\text{Gradient of } V_\rho(\theta)}}$$

Policy gradient (PG) methods

Objective: $\arg\min_{\theta \in \mathbb{R}^d} V_\rho(\theta)$

$$\boxed{\theta^{(k+1)} = \theta^{(k)} - \underbrace{\boxed{\eta_k}}_{\text{Step size}} \boxed{\nabla_\theta V_\rho(\theta^{(k)})}}$$

Gradient of $V_\rho(\theta)$

Step size

- Simplicity

Policy gradient (PG) methods

Objective: $\arg\min_{\theta \in \mathbb{R}^d} V_\rho(\theta)$

$$\boxed{\theta^{(k+1)} = \theta^{(k)} - \underbrace{\boxed{\eta_k}}_{\text{Step size}} \underbrace{\boxed{\nabla_\theta V_\rho(\theta^{(k)})}}_{\text{Gradient of } V_\rho(\theta)}}$$

- Simplicity
  - Easy to implement and use in practice

Policy gradient (PG) methods

Objective: $\arg\min_{\theta \in \mathbb{R}^d} V_\rho(\theta)$

$$\boxed{\theta^{(k+1)} = \theta^{(k)} - \underbrace{\boxed{\eta_k}}_{\text{Step size}} \underbrace{\boxed{\nabla_\theta V_\rho(\theta^{(k)})}}_{\text{Gradient of } V_\rho(\theta)}}$$

- Simplicity
  - Easy to implement and use in practice
  - Can solve a wide range of problems (e.g. partially-observable environments)

Policy gradient (PG) methods

Objective: $\arg\min_{\theta \in \mathbb{R}^d} V_\rho(\theta)$

$$\boxed{\theta^{(k+1)} = \theta^{(k)} - \underbrace{\eta_k}_{\text{Step size}} \underbrace{\nabla_\theta V_\rho(\theta^{(k)})}_{\text{Gradient of } V_\rho(\theta)}}$$

- Simplicity
  - Easy to implement and use in practice
  - Can solve a wide range of problems (e.g. partially-observable environments)
- Versatility

Policy gradient (PG) methods

Objective: $\arg\min_{\theta \in \mathbb{R}^d} V_\rho(\theta)$

$$\boxed{\theta^{(k+1)} = \theta^{(k)} - \underbrace{\boxed{\eta_k}}_{\text{Step size}} \underbrace{\boxed{\nabla_\theta V_\rho(\theta^{(k)})}}_{\text{Gradient of } V_\rho(\theta)}}$$

- Simplicity
  - Easy to implement and use in practice
  - Can solve a wide range of problems (e.g. partially-observable environments)
- Versatility
  - REINFORCE [Williams, 1992], PGT, GPOMDP [Baxter and Bartlett, 2001], actor-critic [Konda and Tsitsiklis, 2000]

Policy gradient (PG) methods

Objective: $\arg\min_{\theta \in \mathbb{R}^d} V_\rho(\theta)$

$$\boxed{\theta^{(k+1)} = \theta^{(k)} - \underbrace{\boxed{\eta_k}}_{\text{Step size}} \underbrace{\boxed{\nabla_\theta V_\rho(\theta^{(k)})}}_{\text{Gradient of } V_\rho(\theta)}}$$

- Simplicity
  - Easy to implement and use in practice
  - Can solve a wide range of problems (e.g. partially-observable environments)
- Versatility
  - REINFORCE [Williams, 1992], PGT, GPOMDP [Baxter and Bartlett, 2001], actor-critic [Konda and Tsitsiklis, 2000]
  - Natural PG[Kakade, 2001], policy mirror descent [Lan, 2022; Xiao, 2022], variance reduction techniques [Papini et al., 2018; Shen et al., 2019; Xu et al., 2020; Huang et al., 2020]

Policy gradient (PG) methods

Objective: $\arg\min_{\theta \in \mathbb{R}^d} V_\rho(\theta)$

$$\boxed{\theta^{(k+1)} = \theta^{(k)} - \underbrace{\boxed{\eta_k}}_{\text{Step size}} \underbrace{\boxed{\nabla_\theta V_\rho(\theta^{(k)})}}_{\text{Gradient of } V_\rho(\theta)}}$$

- Simplicity
  - Easy to implement and use in practice
  - Can solve a wide range of problems (e.g. partially-observable environments)
- Versatility
  - REINFORCE [Williams, 1992], PGT, GPOMDP [Baxter and Bartlett, 2001], actor-critic [Konda and Tsitsiklis, 2000]
  - Natural PG[Kakade, 2001], policy mirror descent [Lan, 2022; Xiao, 2022], variance reduction techniques [Papini et al., 2018; Shen et al., 2019; Xu et al., 2020; Huang et al., 2020]
  - Trust-region (e.g. TRPO [Schulman et al., 2015;]), proximal (e.g. PPO [Schulman et al., 2017])

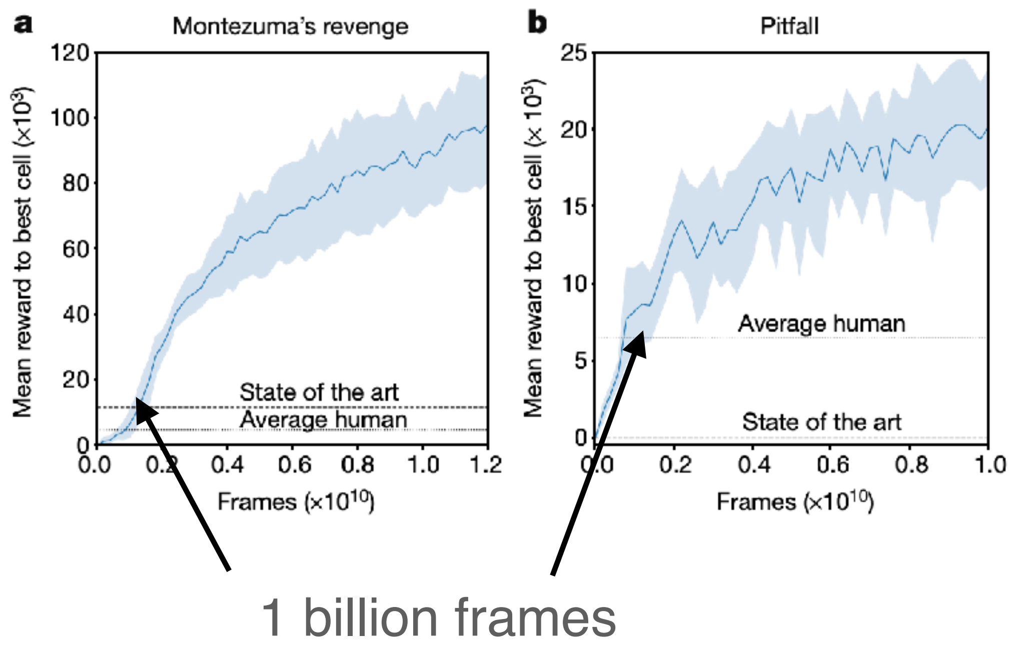
Context

Objective: $\arg\min_{\theta \in \mathbb{R}^d} V_\rho(\theta)$

Context

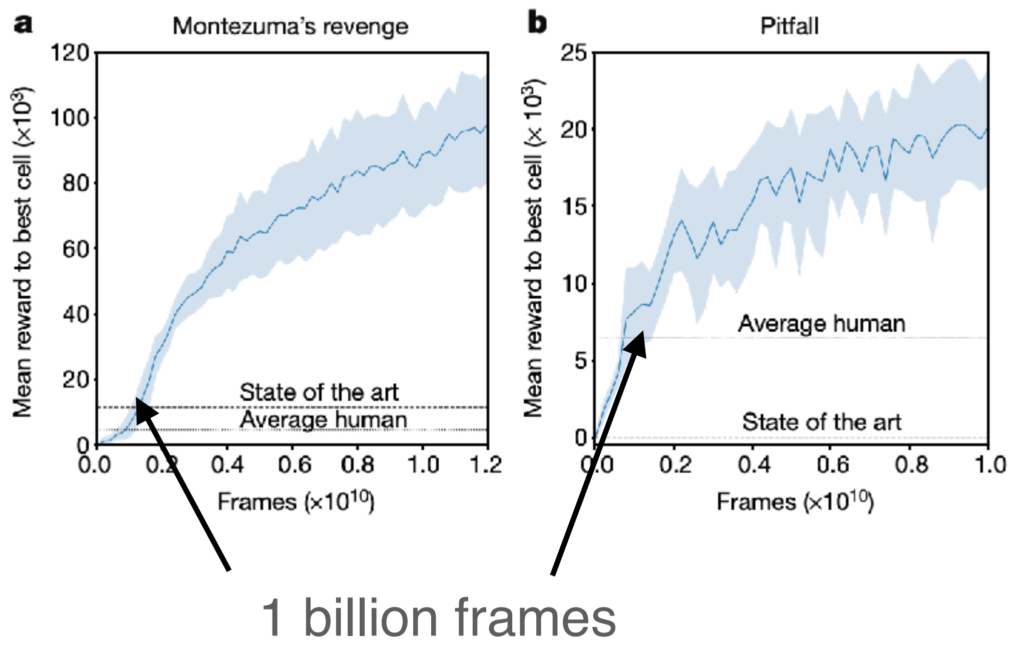
Objective: $\arg\min_{\theta \in \mathbb{R}^d} V_\rho(\theta)$

- Vanilla PG is not sample efficient



Context

Objective: $\arg\min_{\theta\in\mathbb{R}^d} V_\rho(\theta)$



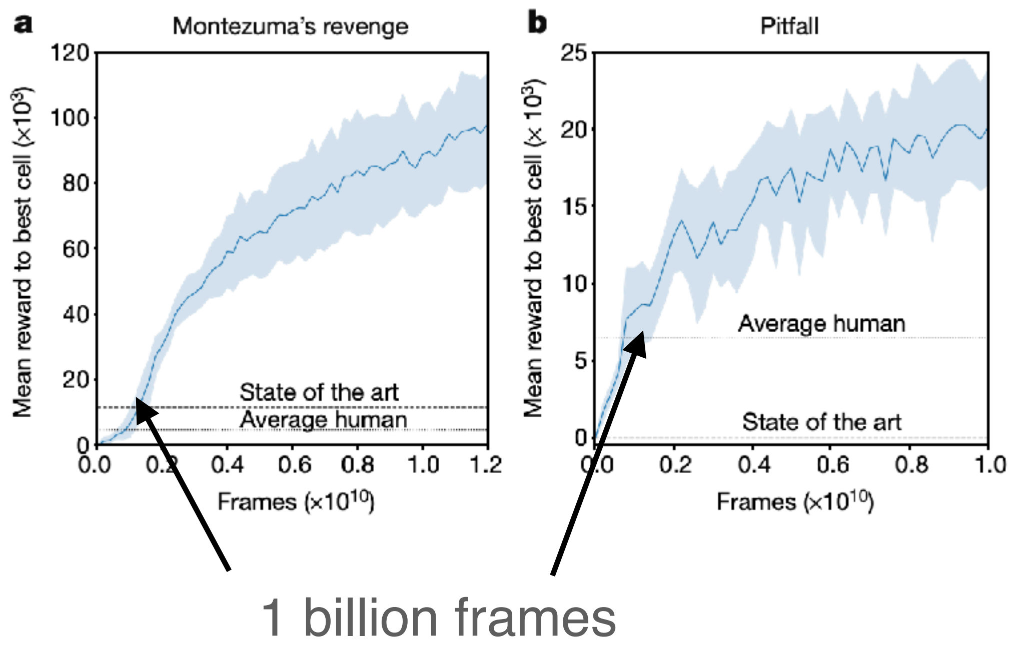
- Vanilla PG is not sample efficient

- Natural PG (NPG)[Kakade, 2001] uses a preconditioner to improve PG direction

Context

Objective: $\arg\min_{\theta \in \mathbb{R}^d} V_\rho(\theta)$

- Vanilla PG is not sample efficient

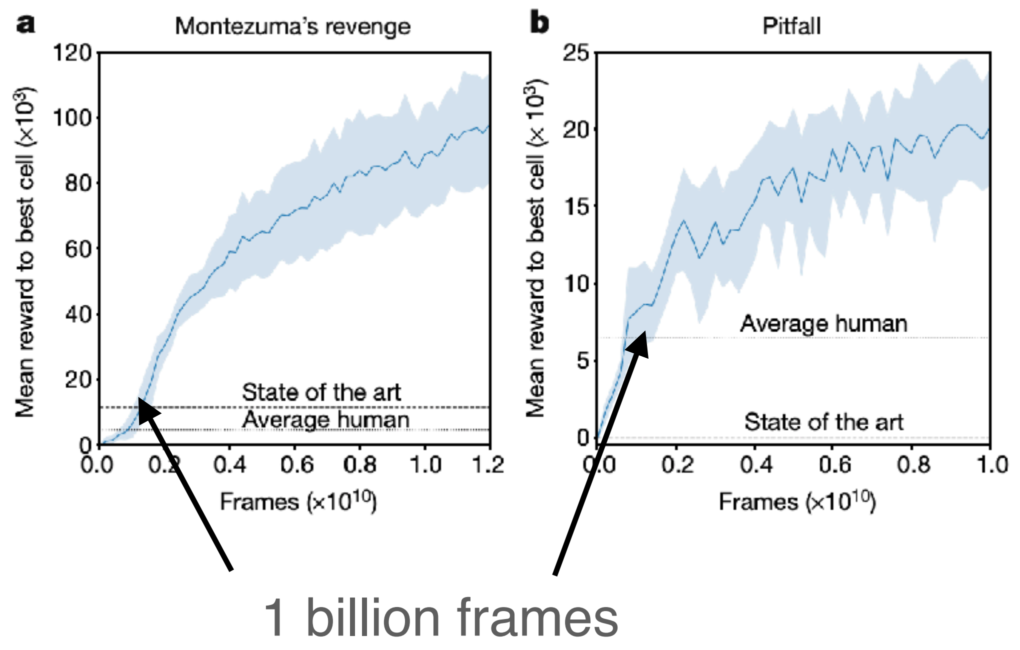


- Natural PG (NPG)[Kakade, 2001] uses a preconditioner to improve PG direction

- NPG is the building block of several state-of-the-art algorithms (TRPO, PPO)

Context

Objective: $\arg\min_{\theta \in \mathbb{R}^d} V_\rho(\theta)$



- Vanilla PG is not sample efficient

- Natural PG (NPG)[Kakade, 2001] uses a preconditioner to improve PG direction

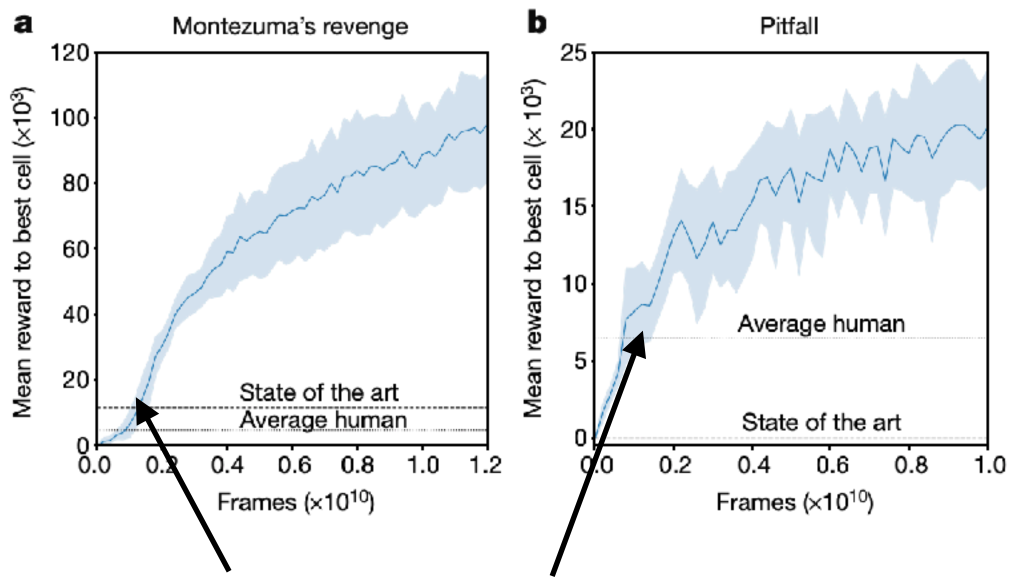
- NPG is the building block of several state-of-the-art algorithms (TRPO, PPO)

- Linear convergence of NPG is established for tabular case [Xiao, 2022]

Context

Objective: $\arg\min_{\theta \in \mathbb{R}^d} V_\rho(\theta)$

- Vanilla PG is not sample efficient

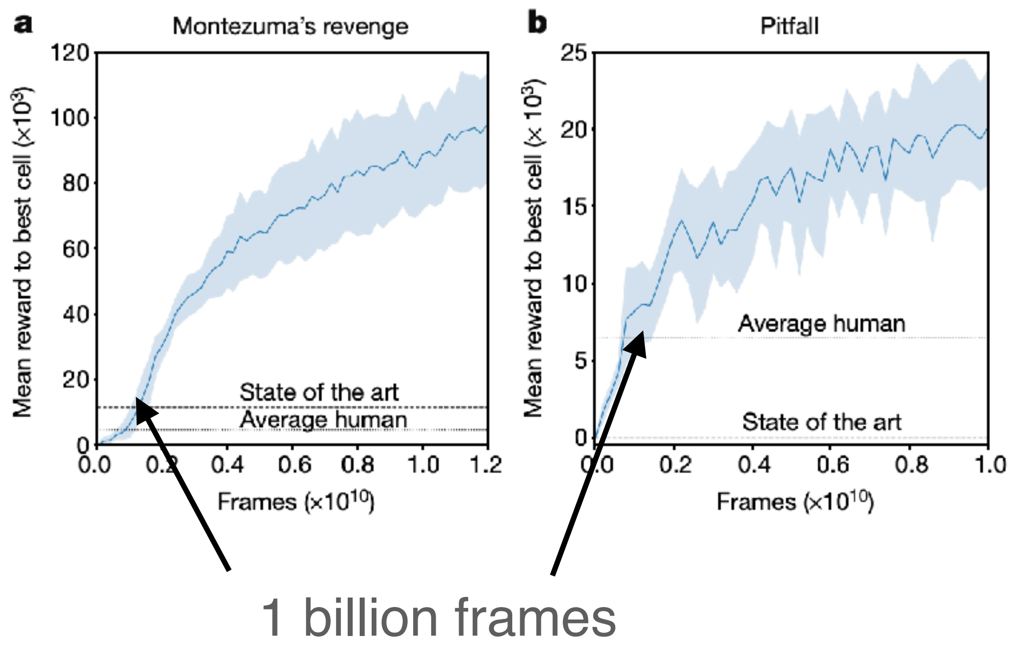


1 billion frames

- Natural PG (NPG)[Kakade, 2001] uses a preconditioner to improve PG direction
- NPG is the building block of several state-of-the-art algorithms (TRPO, PPO)
- Linear convergence of NPG is established for $\boxed{\text{tabular case}}$ [Xiao, 2022]

Context

Objective: $\arg\min_{\theta \in \mathbb{R}^d} V_\rho(\theta)$



- Vanilla PG is not sample efficient
- Natural PG (NPG)[Kakade, 2001] uses a preconditioner to improve PG direction
- NPG is the building block of several state-of-the-art algorithms (TRPO, PPO)
- Linear convergence of NPG is established for tabular case [Xiao, 2022]

Motivations

- Extend linear convergence of NPG from tabular to function approximation regime.

PG method: $\theta^{(k+1)} = \theta^{(k)} - \eta_k \nabla_\theta V_\rho(\theta^{(k)})$

Policy gradient theorem (PGT) [Sutton et al., 2000]

PG method: $\theta^{(k+1)} = \theta^{(k)} - \eta_k \nabla_\theta V_\rho(\theta^{(k)})$

Policy gradient theorem (PGT) [Sutton et al., 2000]

PG method: $\theta^{(k+1)} = \theta^{(k)} - \eta_k \nabla_\theta V_\rho(\theta^{(k)})$

- State-action cost function (a.k.a Q-function) & advantage function

Policy gradient theorem (PGT) [Sutton et al., 2000]

PG method: $\theta^{(k+1)} = \theta^{(k)} - \eta_k \nabla_\theta V_\rho(\theta^{(k)})$

- State-action cost function (a.k.a Q-function) & advantage function

$$Q_{s,a}(\theta) := \mathbb{E}_{a_t \sim \pi_{s_t}(\theta),\ s_{t+1} \sim P(\cdot|s_t,a_t)} \left[\sum_{t=0}^{\infty} \gamma^t c(s_t, a_t) \mid s_0 = s, a_0 = a \right]$$

Policy gradient theorem (PGT) [Sutton et al., 2000]

PG method: $\theta^{(k+1)} = \theta^{(k)} - \eta_k \nabla_\theta V_\rho(\theta^{(k)})$

- State-action cost function (a.k.a Q-function) & advantage function

$$Q_{s,a}(\theta) := \mathbb{E}_{a_t \sim \pi_{s_t}(\theta),\ s_{t+1} \sim P(\cdot|s_t,a_t)} \left[\sum_{t=0}^{\infty} \gamma^t c(s_t, a_t) \mid s_0 = s, a_0 = a \right]$$

$$\begin{aligned} V_\rho(\theta) &= \mathbb{E}_{s\sim\rho}[V_s(\theta)] \\ &= \mathbb{E}_{s\sim\rho, a\sim\pi_s(\theta)}[Q_{s,a}(\theta)] \end{aligned}$$

Policy gradient theorem (PGT) [Sutton et al., 2000]

PG method: $\theta^{(k+1)} = \theta^{(k)} - \eta_k \nabla_\theta V_\rho(\theta^{(k)})$

- State-action cost function (a.k.a Q-function) & advantage function

$$Q_{s,a}(\theta) := \mathbb{E}_{\,a_t \sim \pi_{s_t}(\theta),\ s_{t+1} \sim P(\cdot|s_t,a_t)} \left[\sum_{t=0}^{\infty} \gamma^t c(s_t, a_t) \mid s_0 = s, a_0 = a \right]$$

$$A_{s,a}(\theta) := Q_{s,a}(\theta) - \mathbb{E}_{a' \sim \pi_s(\theta)}[Q_{s,a'}(\theta)]$$

$$\boxed{\begin{aligned} V_\rho(\theta) &= \mathbb{E}_{s\sim\rho}[V_s(\theta)] \\ &= \mathbb{E}_{s\sim\rho, a\sim\pi_s(\theta)}[Q_{s,a}(\theta)] \end{aligned}}$$

Policy gradient theorem (PGT) [Sutton et al., 2000]

PG method: $\theta^{(k+1)} = \theta^{(k)} - \eta_k \nabla_\theta V_\rho(\theta^{(k)})$

- State-action cost function (a.k.a Q-function) & advantage function

$$Q_{s,a}(\theta) := \mathbb{E}_{\ a_t \sim \pi_{s_t}(\theta),\ s_{t+1} \sim P(\cdot | s_t, a_t)} \left[\sum_{t=0}^{\infty} \gamma^t c(s_t, a_t) \mid s_0 = s, a_0 = a \right]$$
$$A_{s,a}(\theta) := Q_{s,a}(\theta) - \mathbb{E}_{a' \sim \pi_s(\theta)}[Q_{s,a'}(\theta)]$$

$$\boxed{\begin{aligned} V_\rho(\theta) &= \mathbb{E}_{s \sim \rho}[V_s(\theta)] \\ &= \mathbb{E}_{s \sim \rho, a \sim \pi_s(\theta)}[Q_{s,a}(\theta)] \end{aligned}}$$

- State visitation distribution of the MDP $d^\pi(\rho) \in \Delta(\mathcal{S})$

Policy gradient theorem (PGT) [Sutton et al., 2000]

PG method: $\theta^{(k+1)} = \theta^{(k)} - \eta_k \nabla_\theta V_\rho(\theta^{(k)})$

- State-action cost function (a.k.a Q-function) & advantage function

$$Q_{s,a}(\theta) := \mathbb{E}_{\,a_t \sim \pi_{s_t}(\theta),\; s_{t+1} \sim P(\cdot|s_t, a_t)} \left[\sum_{t=0}^{\infty} \gamma^t c(s_t, a_t) \mid s_0 = s, a_0 = a \right]$$
$$A_{s,a}(\theta) := Q_{s,a}(\theta) - \mathbb{E}_{a' \sim \pi_s(\theta)}[Q_{s,a'}(\theta)]$$

$$\boxed{\begin{aligned} V_\rho(\theta) &= \mathbb{E}_{s \sim \rho}[V_s(\theta)] \\ &= \mathbb{E}_{s \sim \rho, a \sim \pi_s(\theta)}[Q_{s,a}(\theta)] \end{aligned}}$$

- State visitation distribution of the MDP $d^\pi(\rho) \in \Delta(\mathcal{S})$

$$d_s^\pi(\rho) := (1 - \gamma)\mathbb{E}_{s_0 \sim \rho} \left[\sum_{t=0}^{\infty} \gamma^t P(s_t = s \mid s_0, \pi) \right]$$

Policy gradient theorem (PGT) [Sutton et al., 2000]

PG method: $\theta^{(k+1)} = \theta^{(k)} - \eta_k \nabla_\theta V_\rho(\theta^{(k)})$

- State-action cost function (a.k.a Q-function) & advantage function

$$Q_{s,a}(\theta) := \mathbb{E}_{\,a_t \sim \pi_{s_t}(\theta),\ s_{t+1} \sim P(\cdot|s_t, a_t)} \left[\sum_{t=0}^{\infty} \gamma^t c(s_t, a_t) \mid s_0 = s, a_0 = a \right]$$

$$A_{s,a}(\theta) := Q_{s,a}(\theta) - \mathbb{E}_{a' \sim \pi_s(\theta)}[Q_{s,a'}(\theta)]$$

$$\boxed{\begin{aligned} V_\rho(\theta) &= \mathbb{E}_{s\sim\rho}[V_s(\theta)] \\ &= \mathbb{E}_{s\sim\rho, a\sim\pi_s(\theta)}[Q_{s,a}(\theta)] \end{aligned}}$$

- State visitation distribution of the MDP $d^\pi(\rho) \in \Delta(\mathcal{S})$

$$d_s^\pi(\rho) := (1-\gamma)\mathbb{E}_{s_0 \sim \rho} \left[\sum_{t=0}^{\infty} \gamma^t P(s_t = s \mid s_0, \pi) \right]$$

- PGT

Policy gradient theorem (PGT) [Sutton et al., 2000]

PG method: $\theta^{(k+1)} = \theta^{(k)} - \eta_k \nabla_\theta V_\rho(\theta^{(k)})$

- State-action cost function (a.k.a Q-function) & advantage function

$$Q_{s,a}(\theta) := \mathbb{E}_{\,a_t \sim \pi_{s_t}(\theta),\ s_{t+1} \sim P(\cdot | s_t, a_t)} \left[\sum_{t=0}^{\infty} \gamma^t c(s_t, a_t) \mid s_0 = s, a_0 = a \right]$$
$$A_{s,a}(\theta) := Q_{s,a}(\theta) - \mathbb{E}_{a' \sim \pi_s(\theta)}[Q_{s,a'}(\theta)]$$

$$\boxed{\begin{aligned} V_\rho(\theta) &= \mathbb{E}_{s \sim \rho}[V_s(\theta)] \\ &= \mathbb{E}_{s \sim \rho, a \sim \pi_s(\theta)}[Q_{s,a}(\theta)] \end{aligned}}$$

- State visitation distribution of the MDP $d^\pi(\rho) \in \Delta(\mathcal{S})$

$$d_s^\pi(\rho) := (1 - \gamma)\mathbb{E}_{s_0 \sim \rho} \left[\sum_{t=0}^{\infty} \gamma^t P(s_t = s \mid s_0, \pi) \right]$$

- PGT

$$\nabla_\theta V_\rho(\theta) = \frac{1}{1-\gamma} \mathbb{E}_{s \sim d^{\pi(\theta)}(\rho), a \sim \pi_s(\theta)} \left[Q_{s,a}(\theta) \nabla_\theta \log \pi_{s,a}(\theta) \right]$$

Derivation of Policy Gradient Theorem

$$\nabla_\theta V_\rho(\theta) = \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\pi(\theta)}(\rho),a\sim\pi_s(\theta)}\left[Q_{s,a}(\theta)\,\nabla_\theta \log \pi_{s,a}(\theta)\right]$$

Derivation of Policy Gradient Theorem

$$\nabla_\theta V_\rho(\theta) = \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\pi(\theta)}(\rho), a\sim\pi_s(\theta)}\left[Q_{s,a}(\theta)\,\nabla_\theta \log \pi_{s,a}(\theta)\right]$$

- *Proof:*

Derivation of Policy Gradient Theorem

$$\nabla_\theta V_\rho(\theta) = \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\pi(\theta)}(\rho), a\sim\pi_s(\theta)}\left[Q_{s,a}(\theta)\,\nabla_\theta \log \pi_{s,a}(\theta)\right]$$

- *Proof:* $\nabla_\theta V_\rho(\theta) = \nabla_\theta \sum_{s_0\in\mathcal{S},\ a_0\in\mathcal{A}} \rho(s_0)\pi_{s_0,a_0}(\theta)Q_{s_0,a_0}(\theta)$

Derivation of Policy Gradient Theorem

$$\nabla_\theta V_\rho(\theta) = \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\pi(\theta)}(\rho), a\sim\pi_s(\theta)}\left[Q_{s,a}(\theta)\,\nabla_\theta \log \pi_{s,a}(\theta)\right]$$

- *Proof:*

$$\begin{aligned}\nabla_\theta V_\rho(\theta) &= \nabla_\theta \sum\nolimits_{s_0\in\mathcal{S},\ a_0\in\mathcal{A}} \rho(s_0)\pi_{s_0,a_0}(\theta)Q_{s_0,a_0}(\theta)\\ &= \sum\nolimits_{s_0,a_0} \rho(s_0)\big(\nabla_\theta \pi_{s_0,a_0}(\theta)\big)Q_{s_0,a_0}(\theta) + \sum\nolimits_{s_0,a_0}\rho(s_0)\pi_{s_0,a_0}(\theta)\,\nabla_\theta Q_{s_0,a_0}(\theta)\end{aligned}$$

Derivation of Policy Gradient Theorem

$$\nabla_\theta V_\rho(\theta) = \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\pi(\theta)}(\rho), a\sim\pi_s(\theta)}\left[Q_{s,a}(\theta)\,\nabla_\theta \log \pi_{s,a}(\theta)\right]$$

- *Proof:*

$$\begin{aligned}
\nabla_\theta V_\rho(\theta) &= \nabla_\theta \sum_{s_0\in\mathcal{S},\ a_0\in\mathcal{A}} \rho(s_0)\pi_{s_0,a_0}(\theta)Q_{s_0,a_0}(\theta) \\
&= \sum_{s_0,a_0} \rho(s_0)\left(\nabla_\theta \pi_{s_0,a_0}(\theta)\right)Q_{s_0,a_0}(\theta) + \sum_{s_0,a_0} \rho(s_0)\pi_{s_0,a_0}(\theta)\,\nabla_\theta Q_{s_0,a_0}(\theta) \\
&= \sum_{s_0,a_0} \rho(s_0)\pi_{s_0,a_0}(\theta)\left(\nabla_\theta \log \pi_{s_0,a_0}(\theta)\right)Q_{s_0,a_0}(\theta) \\
&\quad + \sum_{s_0,a_0} \rho(s_0)\pi_{s_0,a_0}(\theta)\,\nabla_\theta\left(c(s_0,a_0) + \gamma \sum_{s_1} P(s_1 \mid s_0,a_0)V_{s_1}(\theta)\right)
\end{aligned}$$

Derivation of Policy Gradient Theorem

$$\nabla_\theta V_\rho(\theta) = \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\pi(\theta)}(\rho), a\sim\pi_s(\theta)}\left[Q_{s,a}(\theta)\,\nabla_\theta \log \pi_{s,a}(\theta)\right]$$

- *Proof:*

$$\begin{aligned}
\nabla_\theta V_\rho(\theta) &= \nabla_\theta \sum_{s_0\in\mathcal{S},\ a_0\in\mathcal{A}} \rho(s_0)\pi_{s_0,a_0}(\theta)Q_{s_0,a_0}(\theta) \\
&= \sum_{s_0,a_0} \rho(s_0)\left(\nabla_\theta \pi_{s_0,a_0}(\theta)\right)Q_{s_0,a_0}(\theta) + \sum_{s_0,a_0} \rho(s_0)\pi_{s_0,a_0}(\theta)\,\nabla_\theta Q_{s_0,a_0}(\theta) \\
&= \sum_{s_0,a_0} \rho(s_0)\pi_{s_0,a_0}(\theta)\left(\nabla_\theta \log \pi_{s_0,a_0}(\theta)\right)Q_{s_0,a_0}(\theta) \\
&\quad + \sum_{s_0,a_0} \rho(s_0)\pi_{s_0,a_0}(\theta)\,\nabla_\theta\left(\boxed{c(s_0,a_0) + \gamma\sum_{s_1} P(s_1 \mid s_0,a_0)V_{s_1}(\theta)}\right)
\end{aligned}$$

Bellman Equation

Derivation of Policy Gradient Theorem

$$\nabla_\theta V_\rho(\theta) = \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\pi(\theta)}(\rho), a\sim \pi_s(\theta)}\left[Q_{s,a}(\theta)\,\nabla_\theta \log \pi_{s,a}(\theta)\right]$$

- *Proof:*

$$\begin{aligned}
\nabla_\theta V_\rho(\theta) &= \nabla_\theta \sum_{s_0\in\mathcal{S},\ a_0\in\mathcal{A}} \rho(s_0)\pi_{s_0,a_0}(\theta)Q_{s_0,a_0}(\theta) \\
&= \sum_{s_0,a_0} \rho(s_0)\left(\nabla_\theta \pi_{s_0,a_0}(\theta)\right)Q_{s_0,a_0}(\theta) + \sum_{s_0,a_0} \rho(s_0)\pi_{s_0,a_0}(\theta)\,\nabla_\theta Q_{s_0,a_0}(\theta) \\
&= \sum_{s_0,a_0} \rho(s_0)\pi_{s_0,a_0}(\theta)\left(\nabla_\theta \log \pi_{s_0,a_0}(\theta)\right)Q_{s_0,a_0}(\theta) \\
&\quad + \sum_{s_0,a_0} \rho(s_0)\pi_{s_0,a_0}(\theta)\,\nabla_\theta\left(\boxed{c(s_0,a_0) + \gamma\sum_{s_1} P(s_1 \mid s_0,a_0)V_{s_1}(\theta)}\right) \quad \text{Bellman Equation} \\
&= \sum_{s_0,a_0} \rho(s_0)\pi_{s_0,a_0}(\theta)\left(\nabla_\theta \log \pi_{s_0,a_0}(\theta)\right)Q_{s_0,a_0}(\theta) \\
&\quad + \gamma\sum_{s_0,a_0,s_1} \rho(s_0)\pi_{s_0,a_0}(\theta)P(s_1 \mid s_0,a_0)\,\nabla_\theta V_{s_1}(\theta)
\end{aligned}$$

Derivation of Policy Gradient Theorem

$$\nabla_\theta V_\rho(\theta) = \frac{1}{1-\gamma} \mathbb{E}_{s \sim d^{\pi(\theta)}(\rho), a \sim \pi_s(\theta)} \left[Q_{s,a}(\theta) \nabla_\theta \log \pi_{s,a}(\theta) \right]$$

- *Proof:*

$$\begin{aligned}
\nabla_\theta V_\rho(\theta) &= \nabla_\theta \sum_{s_0 \in \mathcal{S},\ a_0 \in \mathcal{A}} \rho(s_0) \pi_{s_0,a_0}(\theta) Q_{s_0,a_0}(\theta) \\
&= \sum_{s_0,a_0} \rho(s_0) \left(\nabla_\theta \pi_{s_0,a_0}(\theta) \right) Q_{s_0,a_0}(\theta) + \sum_{s_0,a_0} \rho(s_0) \pi_{s_0,a_0}(\theta) \nabla_\theta Q_{s_0,a_0}(\theta) \\
&= \sum_{s_0,a_0} \rho(s_0) \pi_{s_0,a_0}(\theta) \left(\nabla_\theta \log \pi_{s_0,a_0}(\theta) \right) Q_{s_0,a_0}(\theta) \\
&\quad + \sum_{s_0,a_0} \rho(s_0) \pi_{s_0,a_0}(\theta) \nabla_\theta \left(\boxed{c(s_0, a_0) + \gamma \sum_{s_1} P(s_1 \mid s_0, a_0) V_{s_1}(\theta)} \right) \quad \text{Bellman Equation} \\
&= \sum_{s_0,a_0} \rho(s_0) \pi_{s_0,a_0}(\theta) \left(\nabla_\theta \log \pi_{s_0,a_0}(\theta) \right) Q_{s_0,a_0}(\theta) \\
&\quad + \gamma \sum_{s_0,a_0,s_1} \rho(s_0) \pi_{s_0,a_0}(\theta) P(s_1 \mid s_0, a_0) \nabla_\theta V_{s_1}(\theta) \\
&= \mathbb{E}\left[Q_{s_0,a_0}(\theta) \nabla_\theta \log \pi_{s_0,a_0}(\theta) \right] + \gamma \mathbb{E}\left[\nabla_\theta V_{s_1}(\theta) \right]
\end{aligned}$$

Derivation of Policy Gradient Theorem

$$\nabla_\theta V_\rho(\theta) = \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\pi(\theta)}(\rho), a\sim\pi_s(\theta)}\left[Q_{s,a}(\theta)\nabla_\theta \log \pi_{s,a}(\theta)\right]$$

- *Proof:*

$$\begin{aligned}
\nabla_\theta V_\rho(\theta) &= \nabla_\theta \sum_{s_0\in\mathcal{S},\ a_0\in\mathcal{A}} \rho(s_0)\pi_{s_0,a_0}(\theta)Q_{s_0,a_0}(\theta) \\
&= \sum_{s_0,a_0} \rho(s_0)\left(\nabla_\theta \pi_{s_0,a_0}(\theta)\right)Q_{s_0,a_0}(\theta) + \sum_{s_0,a_0} \rho(s_0)\pi_{s_0,a_0}(\theta)\nabla_\theta Q_{s_0,a_0}(\theta) \\
&= \sum_{s_0,a_0} \rho(s_0)\pi_{s_0,a_0}(\theta)\left(\nabla_\theta \log \pi_{s_0,a_0}(\theta)\right)Q_{s_0,a_0}(\theta) \\
&\quad + \sum_{s_0,a_0} \rho(s_0)\pi_{s_0,a_0}(\theta)\nabla_\theta\left(\boxed{c(s_0,a_0) + \gamma\sum_{s_1} P(s_1 \mid s_0,a_0)V_{s_1}(\theta)}\right) \quad \text{Bellman Equation} \\
&= \sum_{s_0,a_0} \rho(s_0)\pi_{s_0,a_0}(\theta)\left(\nabla_\theta \log \pi_{s_0,a_0}(\theta)\right)Q_{s_0,a_0}(\theta) \\
&\quad + \gamma\sum_{s_0,a_0,s_1} \rho(s_0)\pi_{s_0,a_0}(\theta)P(s_1 \mid s_0,a_0)\nabla_\theta V_{s_1}(\theta) \\
&= \mathbb{E}\left[Q_{s_0,a_0}(\theta)\nabla_\theta \log \pi_{s_0,a_0}(\theta)\right] + \gamma\mathbb{E}\left[\nabla_\theta V_{s_1}(\theta)\right] \\
&= \mathbb{E}\left[Q_{s_0,a_0}(\theta)\nabla_\theta \log \pi_{s_0,a_0}(\theta)\right] + \gamma\mathbb{E}\left[Q_{s_1,a_1}(\theta)\nabla_\theta \log \pi_{s_1,a_1}(\theta)\right] + \cdots
\end{aligned}$$

Derivation of Policy Gradient Theorem

$$\nabla_\theta V_\rho(\theta) = \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\pi(\theta)}(\rho),a\sim\pi_s(\theta)}\left[Q_{s,a}(\theta)\,\nabla_\theta \log \pi_{s,a}(\theta)\right]$$

- *Proof:*

$$\begin{aligned}
\nabla_\theta V_\rho(\theta) &= \nabla_\theta \sum_{s_0\in\mathcal{S},\ a_0\in\mathcal{A}} \rho(s_0)\pi_{s_0,a_0}(\theta)Q_{s_0,a_0}(\theta) \\
&= \sum_{s_0,a_0} \rho(s_0)\big(\nabla_\theta \pi_{s_0,a_0}(\theta)\big)Q_{s_0,a_0}(\theta) + \sum_{s_0,a_0} \rho(s_0)\pi_{s_0,a_0}(\theta)\,\nabla_\theta Q_{s_0,a_0}(\theta) \\
&= \sum_{s_0,a_0} \rho(s_0)\pi_{s_0,a_0}(\theta)\big(\nabla_\theta \log \pi_{s_0,a_0}(\theta)\big)Q_{s_0,a_0}(\theta) \\
&\quad + \sum_{s_0,a_0} \rho(s_0)\pi_{s_0,a_0}(\theta)\,\nabla_\theta\Big(\boxed{c(s_0,a_0) + \gamma\sum_{s_1} P(s_1\mid s_0,a_0)V_{s_1}(\theta)}\Big) \quad \text{Bellman Equation} \\
&= \sum_{s_0,a_0} \rho(s_0)\pi_{s_0,a_0}(\theta)\Big(\nabla_\theta \log \pi_{s_0,a_0}(\theta)\Big)\,Q_{s_0,a_0}(\theta) \\
&\quad + \gamma\sum_{s_0,a_0,s_1} \rho(s_0)\pi_{s_0,a_0}(\theta)P(s_1\mid s_0,a_0)\,\nabla_\theta V_{s_1}(\theta) \\
&= \mathbb{E}\left[Q_{s_0,a_0}(\theta)\,\nabla_\theta \log \pi_{s_0,a_0}(\theta)\right] + \gamma\mathbb{E}\left[\nabla_\theta V_{s_1}(\theta)\right] \\
&= \mathbb{E}\left[Q_{s_0,a_0}(\theta)\,\nabla_\theta \log \pi_{s_0,a_0}(\theta)\right] + \gamma\mathbb{E}\left[Q_{s_1,a_1}(\theta)\,\nabla_\theta \log \pi_{s_1,a_1}(\theta)\right] + \cdots \\
&= \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\pi(\theta)}(\rho),a\sim\pi_s(\theta)}\left[Q_{s,a}(\theta)\,\nabla_\theta \log \pi_{s,a}(\theta)\right]
\end{aligned}$$

Natural policy gradient

Natural policy gradient

- Natural policy gradient

Natural policy gradient

- Natural policy gradient

$$\theta^{(k+1)} = \theta^{(k)} - \eta_k F_\rho(\theta^{(k)})^\dagger \nabla_\theta V_\rho(\theta^{(k)})$$

Natural policy gradient

- Natural policy gradient

$$\theta^{(k+1)} = \theta^{(k)} - \eta_k F_\rho(\theta^{(k)})^\dagger \nabla_\theta V_\rho(\theta^{(k)})$$

- Fisher information matrix

Natural policy gradient

- Natural policy gradient

$$\theta^{(k+1)} = \theta^{(k)} - \eta_k F_\rho(\theta^{(k)})^\dagger \nabla_\theta V_\rho(\theta^{(k)})$$

- Fisher information matrix

$$F_\rho(\theta) = \mathbb{E}_{s \sim d^{\pi(\theta)}(\rho), a \sim \pi_s(\theta)} \left[\nabla_\theta \log \pi_{s,a}(\theta) (\nabla_\theta \log \pi_{s,a}(\theta))^\top \right]$$

Natural policy gradient

With log-linear policies

- Natural policy gradient

$$\theta^{(k+1)} = \theta^{(k)} - \eta_k F_\rho(\theta^{(k)})^\dagger \nabla_\theta V_\rho(\theta^{(k)})$$

- Fisher information matrix

$$F_\rho(\theta) = \mathbb{E}_{s\sim d^{\pi(\theta)}(\rho), a\sim\pi_s(\theta)}\left[\nabla_\theta \log \pi_{s,a}(\theta)(\nabla_\theta \log \pi_{s,a}(\theta))^\top\right]$$

Natural policy gradient

With log-linear policies

Log-linear policy:

$$\pi_{s,a}(\theta) = \frac{\exp \phi_{s,a}^\top \theta}{\sum_{a' \in \mathcal{A}} \exp \phi_{s,a'}^\top \theta}$$

- Natural policy gradient

$$\theta^{(k+1)} = \theta^{(k)} - \eta_k F_\rho(\theta^{(k)})^\dagger \nabla_\theta V_\rho(\theta^{(k)})$$

- Fisher information matrix

$$F_\rho(\theta) = \mathbb{E}_{s \sim d^{\pi(\theta)}(\rho), a \sim \pi_s(\theta)} \left[\nabla_\theta \log \pi_{s,a}(\theta) (\nabla_\theta \log \pi_{s,a}(\theta))^\top \right]$$

Natural policy gradient

With log-linear policies

Log-linear policy:

$$\pi_{s,a}(\theta) = \frac{\exp \phi_{s,a}^\top \theta}{\sum_{a' \in \mathcal{A}} \exp \boxed{\phi_{s,a'}^\top} \theta}$$

Feature map $\phi_{s,a'} \in \mathbb{R}^d$ over $\mathcal{S} \times \mathcal{A}$

- Natural policy gradient

$$\theta^{(k+1)} = \theta^{(k)} - \eta_k F_\rho(\theta^{(k)})^\dagger \nabla_\theta V_\rho(\theta^{(k)})$$

- Fisher information matrix

$$F_\rho(\theta) = \mathbb{E}_{s \sim d^{\pi(\theta)}(\rho), a \sim \pi_s(\theta)} \left[\nabla_\theta \log \pi_{s,a}(\theta) (\nabla_\theta \log \pi_{s,a}(\theta))^\top \right]$$

NPG with compatible function approximation

NPG with compatible function approximation

- Compatible function approximation

NPG with compatible function approximation

- Compatible function approximation

$$L(w, \theta, \zeta) = \mathbb{E}_{(s,a)\sim\zeta}\left[(w^\top \nabla_\theta \log \pi_{s,a}(\theta) - A_{s,a}(\theta))^2\right]$$

NPG with compatible function approximation

- Compatible function approximation

$$L(w, \theta, \zeta) = \mathbb{E}_{(s,a)\sim\zeta}\left[(w^\top \nabla_\theta \log \pi_{s,a}(\theta) - A_{s,a}(\theta))^2\right]$$

- NPG can be rewritten as

NPG with compatible function approximation

- Compatible function approximation

$$L(w,\theta,\zeta) = \mathbb{E}_{(s,a)\sim\zeta}\Big[(w^\top \nabla_\theta \log \pi_{s,a}(\theta) - A_{s,a}(\theta))^2\Big]$$

- NPG can be rewritten as

$$\theta^{(k+1)} = \theta^{(k)} - \eta_k w_\star^{(k)}, \qquad w_\star^{(k)} \in \arg\min_{w\in\mathbb{R}^d} L(w, \theta^{(k)}, d^{\pi(\theta^{(k)})}(\rho)\cdot \pi_s(\theta^{(k)}))$$

NPG with compatible function approximation

- Compatible function approximation

$$L(w, \theta, \zeta) = \mathbb{E}_{(s,a)\sim\zeta}\left[(w^\top \nabla_\theta \log \pi_{s,a}(\theta) - A_{s,a}(\theta))^2\right]$$

- NPG can be rewritten as

$$\theta^{(k+1)} = \theta^{(k)} - \eta_k w_\star^{(k)}, \qquad \boxed{w_\star^{(k)} \in \arg\min_{w\in\mathbb{R}^d} L(w, \theta^{(k)}, d^{\pi(\theta^{(k)})}(\rho) \cdot \pi_s(\theta^{(k)}))}$$

NPG with compatible function approximation

- Compatible function approximation

$$L(w,\theta,\zeta) = \mathbb{E}_{(s,a)\sim\zeta}\Big[\underbrace{(w^\top \nabla_\theta \log \pi_{s,a}(\theta) - A_{s,a}(\theta))^2}\Big]$$

Linear approximation of the advantage function

- NPG can be rewritten as

$$\theta^{(k+1)} = \theta^{(k)} - \eta_k w_\star^{(k)}, \qquad \boxed{w_\star^{(k)} \in \arg\min_{w\in\mathbb{R}^d} L(w, \theta^{(k)}, d^{\pi(\theta^{(k)})}(\rho)\cdot \pi_s(\theta^{(k)}))}$$

NPG with log-linear as policy mirror descent

NPG with log-linear as policy mirror descent

Log-linear policy:

$$\pi_{s,a}(\theta) = \frac{\exp \phi_{s,a}^\top \theta}{\sum_{a' \in \mathcal{A}} \exp \phi_{s,a'}^\top \theta}$$

NPG with log-linear as policy mirror descent

Log-linear policy:

$$\pi_{s,a}(\theta) = \frac{\exp \phi_{s,a}^\top \theta}{\sum_{a' \in \mathcal{A}} \exp \phi_{s,a'}^\top \theta}$$

- NPG with log-linear can also be written as

NPG with log-linear as policy mirror descent

Log-linear policy:

$$\pi_{s,a}(\theta) = \frac{\exp \phi_{s,a}^\top \theta}{\sum_{a' \in \mathscr{A}} \exp \phi_{s,a'}^\top \theta}$$

- NPG with log-linear can also be written as

$$\pi_s(\theta^{(k+1)}) = \arg \min_{p \in \Delta(\mathscr{A})} \left\{ \eta_k \langle \bar{\Phi}_s^{(k)} w_\star^{(k)}, p \rangle + \mathrm{KL}(p, \pi_s(\theta^{(k)})) \right\}$$

NPG with log-linear as policy mirror descent

Log-linear policy:

$$\pi_{s,a}(\theta) = \frac{\exp \phi_{s,a}^\top \theta}{\sum_{a' \in \mathscr{A}} \exp \phi_{s,a'}^\top \theta}$$

- NPG with log-linear can also be written as

$$\pi_s(\theta^{(k+1)}) = \boxed{\arg\min_{p \in \Delta(\mathscr{A})} \left\{ \eta_k \langle \bar{\Phi}_s^{(k)} w_\star^{(k)}, p \rangle + \mathrm{KL}(p, \pi_s(\theta^{(k)})) \right\}}$$

→ Policy mirror descent

NPG with log-linear as policy mirror descent

Log-linear policy:

$$\pi_{s,a}(\theta) = \frac{\exp \phi_{s,a}^\top \theta}{\sum_{a' \in \mathcal{A}} \exp \phi_{s,a'}^\top \theta}$$

- NPG with log-linear can also be written as

$$\pi_s(\theta^{(k+1)}) = \boxed{\arg\min_{p \in \Delta(\mathcal{A})} \left\{ \eta_k \langle \bar{\Phi}_s^{(k)} w_\star^{(k)}, p \rangle + \mathrm{KL}(p, \pi_s(\theta^{(k)})) \right\}} \rightarrow \text{Policy mirror descent}$$

$\bar{\Phi}_s^{(k)} \in \mathbb{R}^{|\mathcal{A}| \times d}$ is a matrix whose rows consist of the *centered feature maps*

NPG with log-linear as policy mirror descent

Log-linear policy:

$$\pi_{s,a}(\theta) = \frac{\exp \phi_{s,a}^\top \theta}{\sum_{a' \in \mathcal{A}} \exp \phi_{s,a'}^\top \theta}$$

- NPG with log-linear can also be written as

$$\pi_s(\theta^{(k+1)}) = \boxed{\arg\min_{p \in \Delta(\mathcal{A})} \left\{ \eta_k \langle \bar{\Phi}_s^{(k)} w_\star^{(k)}, p \rangle + \mathrm{KL}(p, \pi_s(\theta^{(k)})) \right\}}$$ → Policy mirror descent

$\bar{\Phi}_s^{(k)} \in \mathbb{R}^{|\mathcal{A}| \times d}$ is a matrix whose rows consist of the *centered feature maps*

$$\bar{\phi}_{s,a}(\theta^{(k)}) := \nabla_\theta \log \pi_{s,a}(\theta^{(k)}) = \phi_{s,a} - \mathbb{E}_{a' \sim \pi_s(\theta^{(k)})}[\phi_{s,a'}]$$

NPG with log-linear as policy mirror descent

Log-linear policy:

$$\pi_{s,a}(\theta) = \frac{\exp \phi_{s,a}^\top \theta}{\sum_{a' \in \mathcal{A}} \exp \phi_{s,a'}^\top \theta}$$

- NPG with log-linear can also be written as

$$\pi_s(\theta^{(k+1)}) = \boxed{\arg \min_{p \in \Delta(\mathcal{A})} \left\{ \eta_k \langle \bar{\Phi}_s^{(k)} w_\star^{(k)}, p \rangle + \mathrm{KL}(p, \pi_s(\theta^{(k)})) \right\}}$$ → Policy mirror descent

$\bar{\Phi}_s^{(k)} \in \mathbb{R}^{|\mathcal{A}| \times d}$ is a matrix whose rows consist of the *centered feature maps*

$$\bar{\phi}_{s,a}(\theta^{(k)}) := \nabla_\theta \log \pi_{s,a}(\theta^{(k)}) = \phi_{s,a} - \mathbb{E}_{a' \sim \pi_s(\theta^{(k)})}[\phi_{s,a'}]$$

$\mathrm{KL}(p, q) = \sum_{a \in \mathcal{A}} p_a \log(p_a / q_a)$ is the Kullback-Leibler (KL) divergence for $p, q \in \Delta(\mathcal{A})$

NPG with log-linear as policy mirror descent

Log-linear policy:

$$\pi_{s,a}(\theta) = \frac{\exp \phi_{s,a}^\top \theta}{\sum_{a' \in \mathcal{A}} \exp \phi_{s,a'}^\top \theta}$$

- NPG with log-linear can also be written as

$$\pi_s(\theta^{(k+1)}) = \boxed{\arg\min_{p \in \Delta(\mathcal{A})} \left\{ \eta_k \langle \bar{\Phi}_s^{(k)} w_\star^{(k)}, p \rangle + \mathrm{KL}(p, \pi_s(\theta^{(k)})) \right\}}$$ → Policy mirror descent

$\bar{\Phi}_s^{(k)} \in \mathbb{R}^{|\mathcal{A}| \times d}$ is a matrix whose rows consist of the *centered feature maps*

$$\bar{\phi}_{s,a}(\theta^{(k)}) := \nabla_\theta \log \pi_{s,a}(\theta^{(k)}) = \phi_{s,a} - \mathbb{E}_{a' \sim \pi_s(\theta^{(k)})}[\phi_{s,a'}]$$

$\mathrm{KL}(p, q) = \sum_{a \in \mathcal{A}} p_a \log(p_a / q_a)$ is the Kullback-Leibler (KL) divergence for $p, q \in \Delta(\mathcal{A})$

- Connection with Policy Iteration

NPG with log-linear as policy mirror descent

Log-linear policy:

$$\pi_{s,a}(\theta) = \frac{\exp \phi_{s,a}^\top \theta}{\sum_{a' \in \mathcal{A}} \exp \phi_{s,a'}^\top \theta}$$

- NPG with log-linear can also be written as

$$\pi_s(\theta^{(k+1)}) = \boxed{\arg\min_{p \in \Delta(\mathcal{A})} \left\{ \eta_k \langle \bar{\Phi}_s^{(k)} w_\star^{(k)}, p \rangle + \mathrm{KL}(p, \pi_s(\theta^{(k)})) \right\}} \rightarrow \text{Policy mirror descent}$$

$\bar{\Phi}_s^{(k)} \in \mathbb{R}^{|\mathcal{A}| \times d}$ is a matrix whose rows consist of the *centered feature maps*

$$\bar{\phi}_{s,a}(\theta^{(k)}) := \nabla_\theta \log \pi_{s,a}(\theta^{(k)}) = \phi_{s,a} - \mathbb{E}_{a' \sim \pi_s(\theta^{(k)})}[\phi_{s,a'}]$$

$\mathrm{KL}(p, q) = \sum_{a \in \mathcal{A}} p_a \log(p_a / q_a)$ is the Kullback-Leibler (KL) divergence for $p, q \in \Delta(\mathcal{A})$

- Connection with Policy Iteration

$$\pi_s(\theta^{(k+1)}) = \arg\min_{p \in \Delta(\mathcal{A})} \left\{ \eta_k \langle A_s(\theta^{(k)}), p \rangle \right\} \quad \text{with} \quad A_s(\theta^{(k)}) := [A_{s,a}(\theta^{(k)})]_a \in \mathbb{R}^{|\mathcal{A}|}$$

NPG with log-linear as policy mirror descent

Log-linear policy:

$$\pi_{s,a}(\theta) = \frac{\exp \phi_{s,a}^\top \theta}{\sum_{a' \in \mathcal{A}} \exp \phi_{s,a'}^\top \theta}$$

- NPG with log-linear can also be written as

$$\pi_s(\theta^{(k+1)}) = \boxed{\arg \min_{p \in \Delta(\mathcal{A})} \left\{ \eta_k \langle \bar{\Phi}_s^{(k)} w_\star^{(k)}, p \rangle + \underbrace{\boxed{\mathrm{KL}(p, \pi_s(\theta^{(k)}))}}_{\text{Regularization}} \right\}} \rightarrow \text{Policy mirror descent}$$

$\bar{\Phi}_s^{(k)} \in \mathbb{R}^{|\mathcal{A}| \times d}$ is a matrix whose rows consist of the *centered feature maps*

$$\bar{\phi}_{s,a}(\theta^{(k)}) := \nabla_\theta \log \pi_{s,a}(\theta^{(k)}) = \phi_{s,a} - \mathbb{E}_{a' \sim \pi_s(\theta^{(k)})}[\phi_{s,a'}]$$

$\mathrm{KL}(p, q) = \sum_{a \in \mathcal{A}} p_a \log(p_a / q_a)$ is the Kullback-Leibler (KL) divergence for $p, q \in \Delta(\mathcal{A})$

- Connection with Policy Iteration

$$\pi_s(\theta^{(k+1)}) = \arg \min_{p \in \Delta(\mathcal{A})} \left\{ \eta_k \langle A_s(\theta^{(k)}), p \rangle \right\} \quad \text{with} \quad A_s(\theta^{(k)}) := [A_{s,a}(\theta^{(k)})]_a \in \mathbb{R}^{|\mathcal{A}|}$$

NPG with log-linear as policy mirror descent

Log-linear policy:

$$\pi_{s,a}(\theta) = \frac{\exp \phi_{s,a}^\top \theta}{\sum_{a' \in \mathcal{A}} \exp \phi_{s,a'}^\top \theta}$$

- NPG with log-linear can also be written as

$$\pi_s(\theta^{(k+1)}) = \boxed{\arg\min_{p \in \Delta(\mathcal{A})} \left\{ \eta_k \langle \underbrace{\boxed{\bar{\Phi}_s^{(k)} w_\star^{(k)}}}_{\text{Linear approximation}}, p \rangle + \underbrace{\boxed{\mathrm{KL}(p, \pi_s(\theta^{(k)}))}}_{\text{Regularization}} \right\}} \rightarrow \text{Policy mirror descent}$$

$\bar{\Phi}_s^{(k)} \in \mathbb{R}^{|\mathcal{A}| \times d}$ is a matrix whose rows consist of the *centered feature maps*

$$\bar{\phi}_{s,a}(\theta^{(k)}) := \nabla_\theta \log \pi_{s,a}(\theta^{(k)}) = \phi_{s,a} - \mathbb{E}_{a' \sim \pi_s(\theta^{(k)})}[\phi_{s,a'}]$$

$\mathrm{KL}(p, q) = \sum_{a \in \mathcal{A}} p_a \log(p_a / q_a)$ is the Kullback-Leibler (KL) divergence for $p, q \in \Delta(\mathcal{A})$

- Connection with Policy Iteration

$$\pi_s(\theta^{(k+1)}) = \arg\min_{p \in \Delta(\mathcal{A})} \left\{ \eta_k \langle \boxed{A_s(\theta^{(k)})}, p \rangle \right\} \quad \text{with} \quad A_s(\theta^{(k)}) := [A_{s,a}(\theta^{(k)})]_a \in \mathbb{R}^{|\mathcal{A}|}$$

Convergence theory

Convergence theory

- Three-point descent lemma [Chen and Teboulle, 1993]:

Convergence theory

- Three-point descent lemma [Chen and Teboulle, 1993]:

  For any $p \in \Delta(\mathscr{A})$,

Convergence theory

- Three-point descent lemma [Chen and Teboulle, 1993]:

  For any $p \in \Delta(\mathscr{A})$,

$$
\begin{aligned}
&\eta_k \langle \bar{\Phi}_s^{(k)} w_\star^{(k)}, \pi_s(\theta^{(k+1)}) \rangle + \mathrm{KL}(\pi_s(\theta^{(k+1)}), \pi_s(\theta^{(k)})) \\
&\leq \eta_k \langle \bar{\Phi}_s^{(k)} w_\star^{(k)}, p \rangle + \mathrm{KL}(p, \pi_s(\theta^{(k)})) - \mathrm{KL}(p, \pi_s(\theta^{(k+1)}))
\end{aligned}
$$

Convergence theory

- Three-point descent lemma [Chen and Teboulle, 1993]:

  For any $p \in \Delta(\mathscr{A})$,

$$\eta_k \langle \bar{\Phi}_s^{(k)} w_\star^{(k)}, \pi_s(\theta^{(k+1)}) \rangle + \mathrm{KL}(\pi_s(\theta^{(k+1)}), \pi_s(\theta^{(k)}))$$
$$\leq \eta_k \langle \bar{\Phi}_s^{(k)} w_\star^{(k)}, p \rangle + \mathrm{KL}(p, \pi_s(\theta^{(k)})) - \mathrm{KL}(p, \pi_s(\theta^{(k+1)}))$$

  One can let $p = \pi_s(\theta^{(k)})$ or be the optimal policy to derive a telescoping sum

Convergence theory

- Three-point descent lemma [Chen and Teboulle, 1993]:

  For any $p \in \Delta(\mathscr{A})$,

$$\eta_k \langle \bar{\Phi}_s^{(k)} w_\star^{(k)}, \pi_s(\theta^{(k+1)}) \rangle + \mathrm{KL}(\pi_s(\theta^{(k+1)}), \pi_s(\theta^{(k)}))$$
$$\leq \eta_k \langle \bar{\Phi}_s^{(k)} w_\star^{(k)}, p \rangle + \mathrm{KL}(p, \pi_s(\theta^{(k)})) - \mathrm{KL}(p, \pi_s(\theta^{(k+1)}))$$

  One can let $p = \pi_s(\theta^{(k)})$ or be the optimal policy to derive a telescoping sum

- Linear convergence to the global optimum by increasing step size by $1/\gamma$

Convergence theory

- Three-point descent lemma [Chen and Teboulle, 1993]:

  For any $p \in \Delta(\mathscr{A})$,

$$\eta_k \langle \bar{\Phi}_s^{(k)} w_\star^{(k)}, \pi_s(\theta^{(k+1)}) \rangle + \mathrm{KL}(\pi_s(\theta^{(k+1)}), \pi_s(\theta^{(k)}))$$
$$\leq \eta_k \langle \bar{\Phi}_s^{(k)} w_\star^{(k)}, p \rangle + \mathrm{KL}(p, \pi_s(\theta^{(k)})) - \mathrm{KL}(p, \pi_s(\theta^{(k+1)}))$$

  One can let $p = \pi_s(\theta^{(k)})$ or be the optimal policy to derive a telescoping sum

- Linear convergence to the global optimum by increasing step size by $1/\gamma$

$$\pi_s(\theta^{(k+1)}) = \arg\min_{p \in \Delta(\mathscr{A})} \left\{ \eta_k \langle \bar{\Phi}_s^{(k)} w_\star^{(k)}, p \rangle + \mathrm{KL}(p, \pi_s(\theta^{(k)})) \right\}$$

Convergence theory

- Three-point descent lemma [Chen and Teboulle, 1993]:

  For any $p \in \Delta(\mathscr{A})$,

$$\eta_k \langle \bar{\Phi}_s^{(k)} w_\star^{(k)}, \pi_s(\theta^{(k+1)}) \rangle + \mathrm{KL}(\pi_s(\theta^{(k+1)}), \pi_s(\theta^{(k)}))$$
$$\leq \eta_k \langle \bar{\Phi}_s^{(k)} w_\star^{(k)}, p \rangle + \mathrm{KL}(p, \pi_s(\theta^{(k)})) - \mathrm{KL}(p, \pi_s(\theta^{(k+1)}))$$

  One can let $p = \pi_s(\theta^{(k)})$ or be the optimal policy to derive a telescoping sum

- Linear convergence to the global optimum by increasing step size by $1/\gamma$

$$\pi_s(\theta^{(k+1)}) = \arg \min_{p \in \Delta(\mathscr{A})} \left\{ \eta_k \langle \bar{\Phi}_s^{(k)} w_\star^{(k)}, p \rangle + \mathrm{KL}(p, \pi_s(\theta^{(k)})) \right\} \qquad \eta_k \longrightarrow \infty$$

Convergence theory

- Three-point descent lemma [Chen and Teboulle, 1993]:

  For any $p \in \Delta(\mathscr{A})$,

$$\eta_k \langle \bar{\Phi}_s^{(k)} w_\star^{(k)}, \pi_s(\theta^{(k+1)}) \rangle + \mathrm{KL}(\pi_s(\theta^{(k+1)}), \pi_s(\theta^{(k)}))$$
$$\leq \eta_k \langle \bar{\Phi}_s^{(k)} w_\star^{(k)}, p \rangle + \mathrm{KL}(p, \pi_s(\theta^{(k)})) - \mathrm{KL}(p, \pi_s(\theta^{(k+1)}))$$

  One can let $p = \pi_s(\theta^{(k)})$ or be the optimal policy to derive a telescoping sum

- Linear convergence to the global optimum by increasing step size by $1/\gamma$

$$\pi_s(\theta^{(k+1)}) = \arg \min_{p \in \Delta(\mathscr{A})} \left\{ \eta_k \langle \bar{\Phi}_s^{(k)} w_\star^{(k)}, p \rangle + \cancel{\mathrm{KL}(p, \pi_s(\theta^{(k)}))} \right\} \qquad \eta_k \longrightarrow \infty$$

Convergence theory

- Three-point descent lemma [Chen and Teboulle, 1993]:

  For any $p \in \Delta(\mathscr{A})$,

$$\eta_k \langle \bar{\Phi}_s^{(k)} w_\star^{(k)}, \pi_s(\theta^{(k+1)}) \rangle + \mathrm{KL}(\pi_s(\theta^{(k+1)}), \pi_s(\theta^{(k)}))$$
$$\leq \eta_k \langle \bar{\Phi}_s^{(k)} w_\star^{(k)}, p \rangle + \mathrm{KL}(p, \pi_s(\theta^{(k)})) - \mathrm{KL}(p, \pi_s(\theta^{(k+1)}))$$

  One can let $p = \pi_s(\theta^{(k)})$ or be the optimal policy to derive a telescoping sum

- Linear convergence to the global optimum by increasing step size by $1/\gamma$

$$\pi_s(\theta^{(k+1)}) = \arg\min_{p \in \Delta(\mathscr{A})} \left\{ \eta_k \langle \boxed{\bar{\Phi}_s^{(k)} w_\star^{(k)}}, p \rangle + \cancel{\mathrm{KL}(p, \pi_s(\theta^{(k)}))} \right\} \qquad \eta_k \longrightarrow \infty$$

Convergence theory

- Three-point descent lemma [Chen and Teboulle, 1993]:

  For any $p \in \Delta(\mathscr{A})$,

$$\eta_k \langle \bar{\Phi}_s^{(k)} w_\star^{(k)}, \pi_s(\theta^{(k+1)}) \rangle + \mathrm{KL}(\pi_s(\theta^{(k+1)}), \pi_s(\theta^{(k)}))$$
$$\leq \eta_k \langle \bar{\Phi}_s^{(k)} w_\star^{(k)}, p \rangle + \mathrm{KL}(p, \pi_s(\theta^{(k)})) - \mathrm{KL}(p, \pi_s(\theta^{(k+1)}))$$

  One can let $p = \pi_s(\theta^{(k)})$ or be the optimal policy to derive a telescoping sum

- Linear convergence to the global optimum by increasing step size by $1/\gamma$

$$\pi_s(\theta^{(k+1)}) = \arg\min_{p \in \Delta(\mathscr{A})} \left\{ \eta_k \langle \boxed{\bar{\Phi}_s^{(k)} w_\star^{(k)}}, p \rangle + \cancel{\mathrm{KL}(p, \pi_s(\theta^{(k)}))} \right\} \qquad \eta_k \longrightarrow \infty$$

Behave more and more like policy iteration

Convergence theory 2

Convergence theory 2

- Consequently, we obtain an $\tilde{O}(\epsilon^{-2})$ sample complexity for NPG

Convergence theory 2

- Consequently, we obtain an $\tilde{O}(\epsilon^{-2})$ sample complexity for NPG
- Similar linear convergence and $\tilde{O}(\epsilon^{-2})$ sample complexity results are also established for Q-NPG

Convergence theory 2

- Consequently, we obtain an $\tilde{O}(\epsilon^{-2})$ sample complexity for NPG
- Similar linear convergence and $\tilde{O}(\epsilon^{-2})$ sample complexity results are also established for Q-NPG
- Sublinear convergence for both NPG and Q-NPG with arbitrary large constant step size

Discussion & Conclusion

Discussion & Conclusion

- We derive sample efficient policy gradient-based RL convergence theory

Discussion & Conclusion

- We derive sample efficient policy gradient-based RL convergence theory
- The linear convergence analysis of NPG with log-linear policy can be extended to general parametrization [Alfano et al., 2023]

Thank you !

References

- Vijay Konda and John Tsitsiklis. Actor-critic algorithms. In Advances in Neural Information Processing Systems, volume 12. MIT Press, 2000.
- Sham M Kakade. A natural policy gradient. In Advances in Neural Information Processing Systems, volume 14. MIT Press, 2001.
- John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In Proceedings of the 32nd International Conference on Machine Learning, volume 37 of Proceedings of Machine Learning Research, pages 1889–1897, Lille, France, 07–09 Jul 2015. PMLR.
- John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017.
- Matteo Papini, Damiano Binaghi, Giuseppe Canonaco, Matteo Pirotta, and Marcello Restelli. Stochastic variance-reduced policy gradient. In Proceedings of the 35th International Conference on Machine Learning, volume 80, pages 4026–4035. PMLR, 2018.
- Zebang Shen, Alejandro Ribeiro, Hamed Hassani, Hui Qian, and Chao Mi. Hessian aided policy gradient. In Proceedings of the 36th International Conference on Machine Learning, volume 97 of Proceedings of Machine Learning Research, pages 5729– 5738. PMLR, 09–15 Jun 2019
- Pan Xu, Felicia Gao, and Quanquan Gu. Sample efficient policy gradient methods with recursive variance reduction. In International Conference on Learning Representations, 2020.
- Feihu Huang, Shangqian Gao, Jian Pei, and Heng Huang. Momentum-based policy gradient methods, 2020.
- Lin Xiao. On the convergence rates of policy gradient methods. Journal of Machine Learning Research, 23(282):1–36, 2022.
- Richard S Sutton, David A. McAllester, Satinder P. Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. In Advances in Neural Information Processing Systems 12, pages 1057–1063. MIT Press, 2000.
- Gong Chen and Marc Teboulle. Convergence analysis of a proximal-like minimization algorithm using bregman functions. SIAM Journal on Optimization, 3(3):538–543, 1993.
- Rui Yuan, Simon S. Du, Robert M. Gower, Alessandro Lazaric, Lin Xiao. Linear Convergence of Natural Policy Gradient Methods with Log-Linear Policies. In International Conference on Learning Representations, 2023.