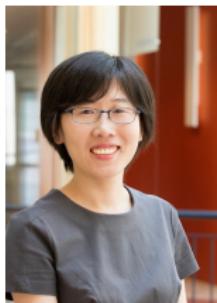


# **Information-theoretic, statistical and algorithmic foundations of reinforcement learning**



Yuejie Chi  
CMU



Yuxin Chen  
UPenn



Yuting Wei  
UPenn

Tutorial, ISIT 2024  
Part 1

# Our wonderful collaborators

---



Gen Li  
UPenn → CUHK



Zihan Zhang  
Princeton



Laixi Shi  
CMU → Caltech



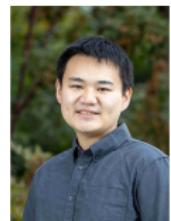
Yuling Yan  
Princeton → MIT



Shicong Cen  
CMU



Changxiao Cai  
UPenn → UMich



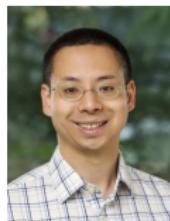
Simon Du  
UWashington



Jianqing Fan  
Princeton



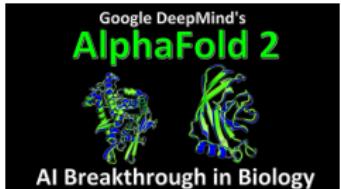
Matthieu Geist  
Google → Cohere



Jason Lee  
Princeton

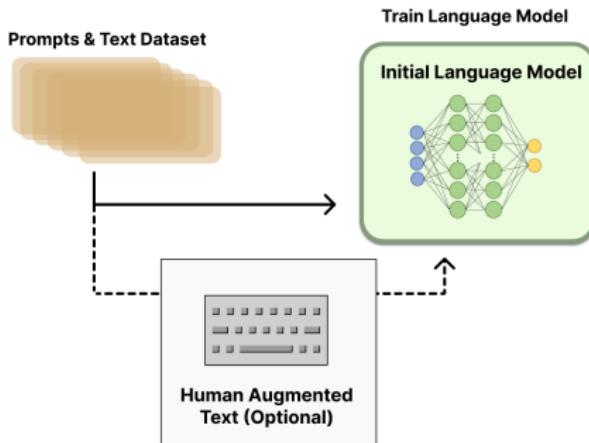
# Recent successes in reinforcement learning (RL)

---



*RL holds great promise in the era of AI*

# One more recent success: RLHF



You

What is RLHF?



ChatGPT

RLHF stands for **Reinforcement Learning from Human Feedback**. It's a technique used in machine learning and artificial intelligence where a model learns to perform tasks or make decisions based on feedback from human trainers, rather than solely relying on pre-existing data sets or explicit programming. This approach allows the

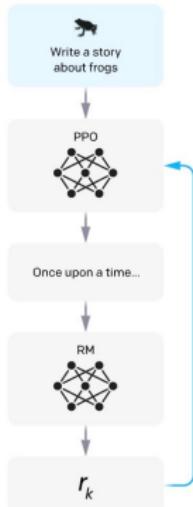


ChatGPT



Optimize a policy against the reward model using reinforcement learning.

A new prompt is sampled from the dataset.



The policy generates an output.

The reward model calculates a reward for the output.

The reward is used to update the policy using PPO.

# Reinforcement Learning



In RL, agent(s) often learn by probing the environment

## Reinforcement Learning



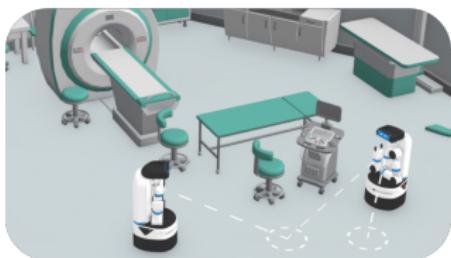
In RL, agent(s) often learn by probing the environment

- unknown environment
- explosion of dimensionality
- delayed feedback
- nonconvexity

# Data efficiency

---

Data collection might be expensive, time-consuming, or high-stakes



clinical trials



self-driving cars

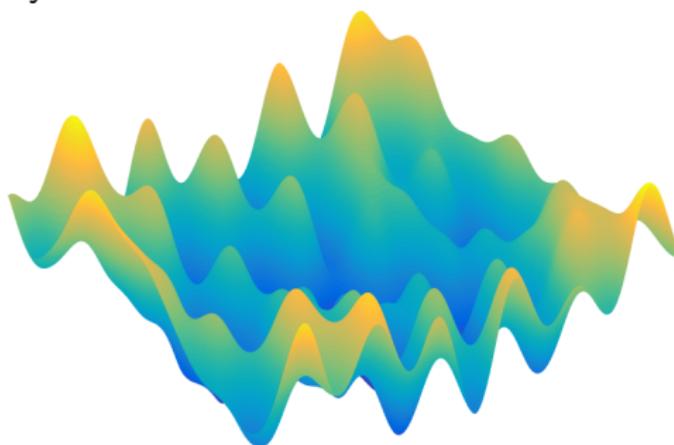
**Calls for design of sample-efficient RL algorithms!**

# Computational efficiency

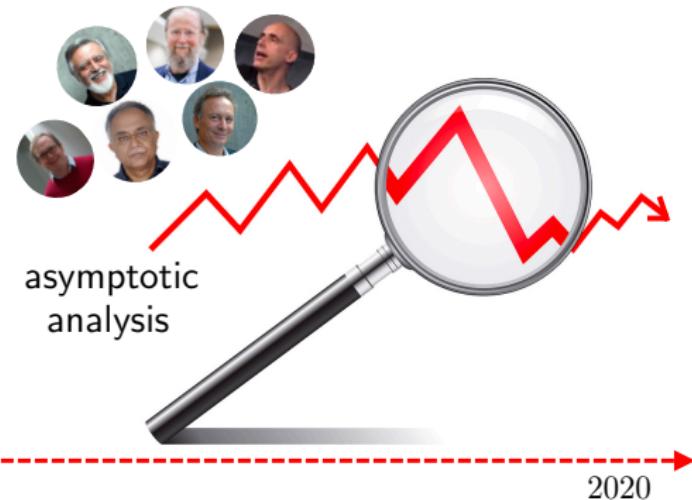
---

Running RL algorithms might take a long time ...

- enormous state-action space
- nonconvexity



Calls for computationally efficient RL algorithms!

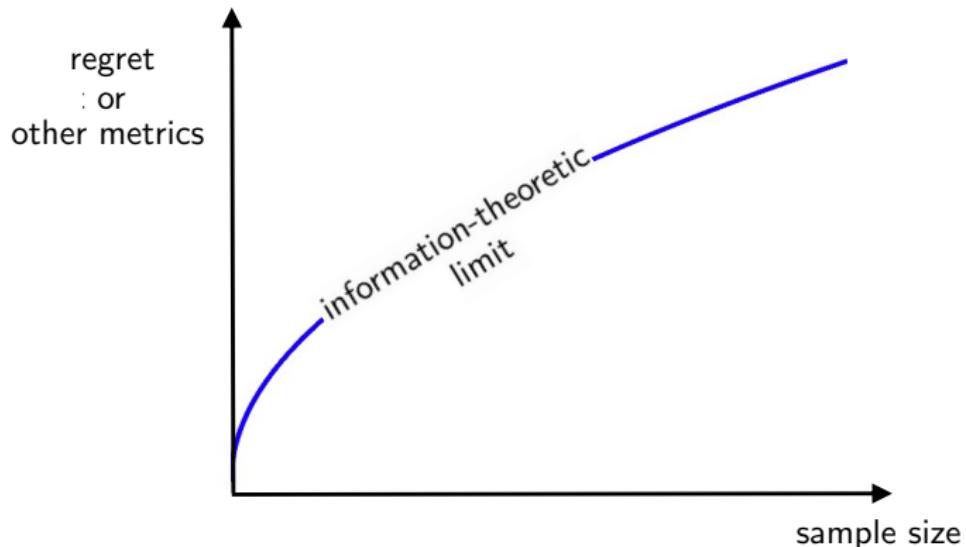




Understanding efficiency of contemporary RL requires a modern suite of non-asymptotic analysis

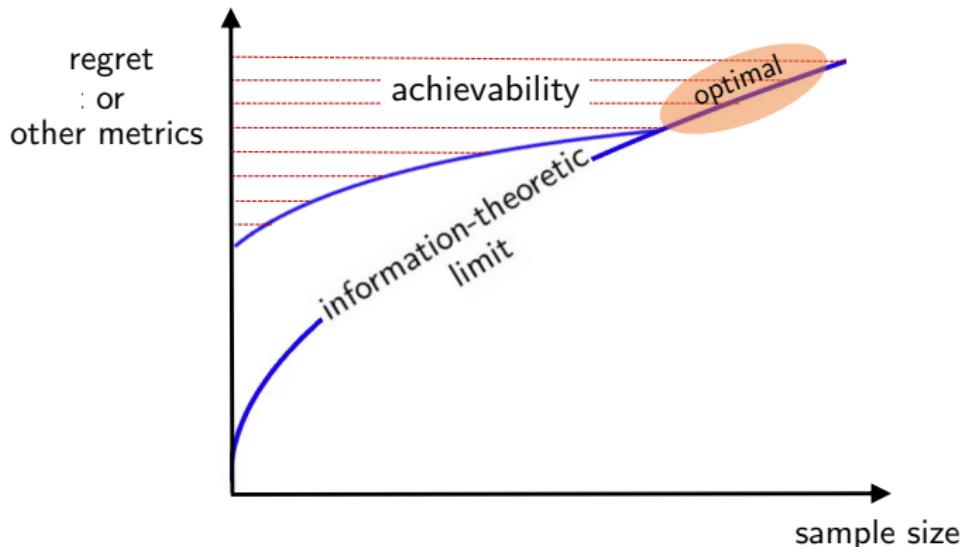
# Sample complexity issues that permeate state-of-the-art RL theory

---

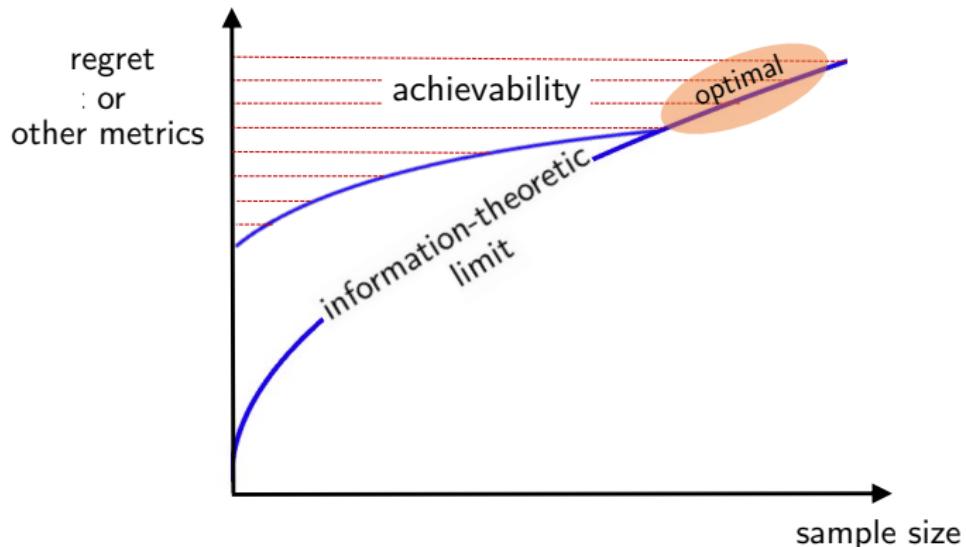


# Sample complexity issues that permeate state-of-the-art RL theory

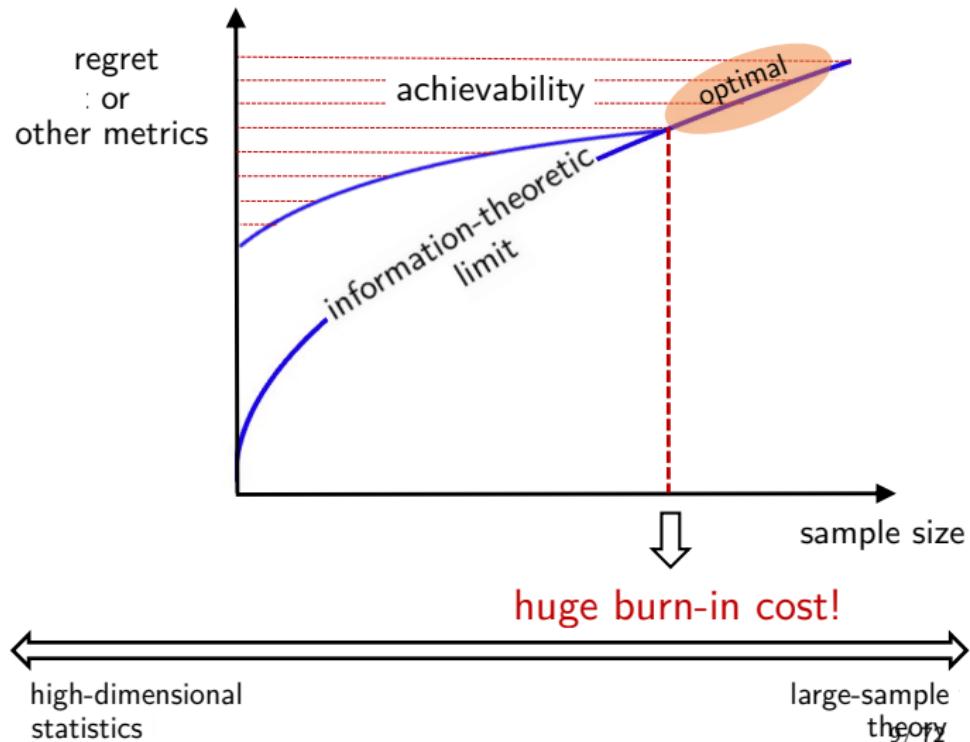
---



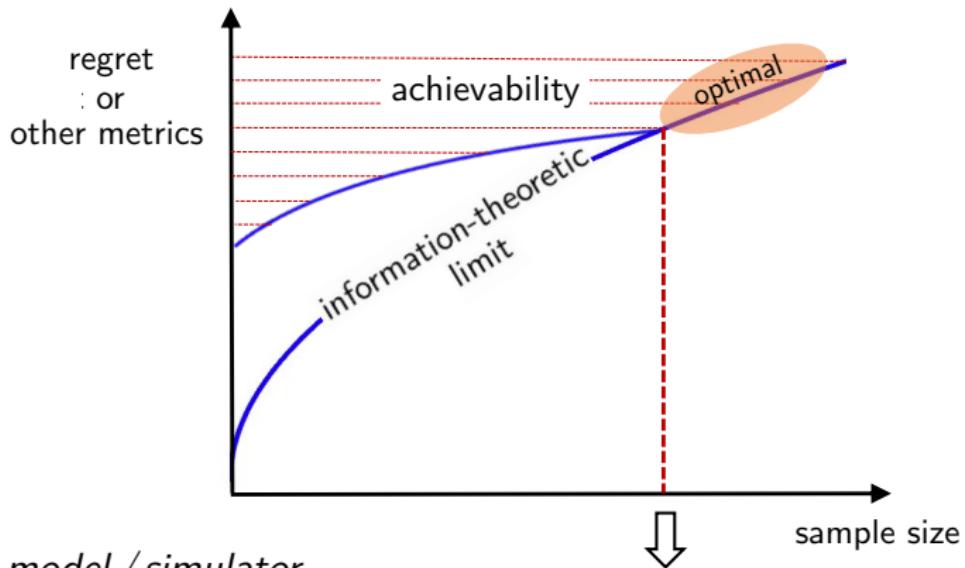
# Sample complexity issues that permeate state-of-the-art RL theory



# Sample complexity issues that permeate state-of-the-art RL theory



# Sample complexity issues that permeate state-of-the-art RL theory

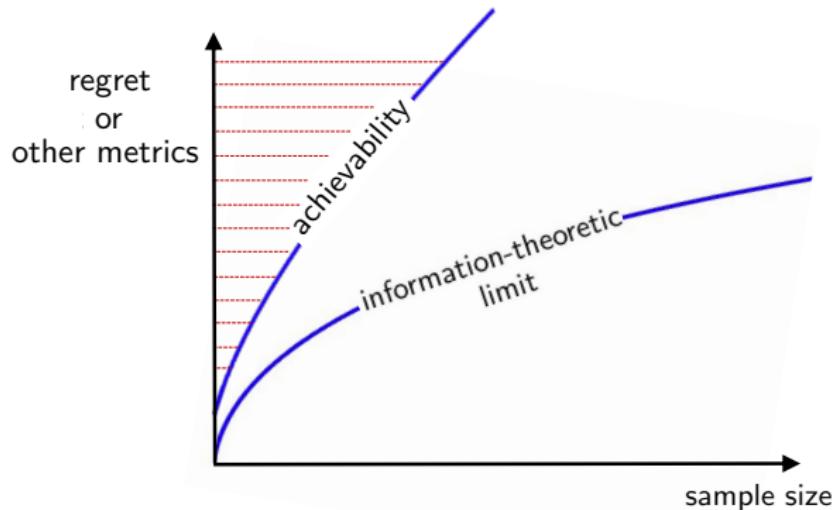


- generative model / simulator
- online RL
- offline RL
- ...

huge burn-in cost!

# Sample complexity issues that permeate state-of-the-art RL theory

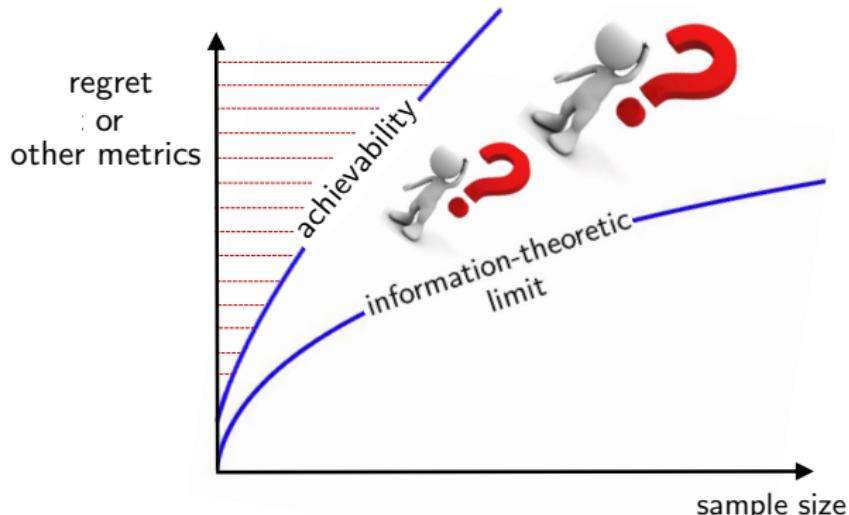
---



- *multi-agent RL*
- *partially observable MDPs*
- ...

# Sample complexity issues that permeate state-of-the-art RL theory

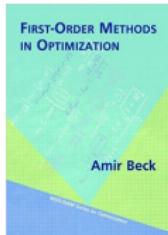
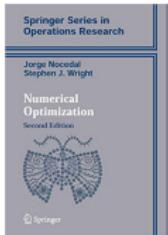
---



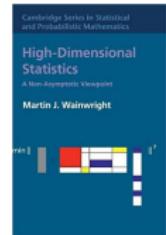
- *multi-agent RL*
- *partially observable MDPs*
- ...

# This tutorial

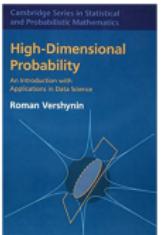
---



(large-scale) optimization



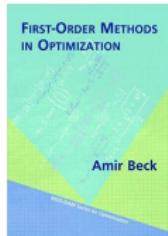
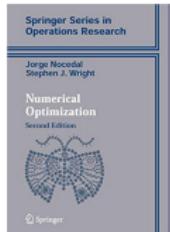
(high-dimensional) statistics



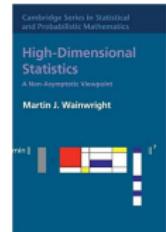
Design **sample-** and **computationally**-efficient RL algorithms

# This tutorial

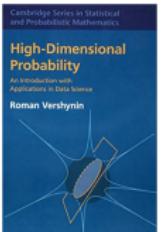
---



(large-scale) optimization



(high-dimensional) statistics



Design **sample-** and **computationally**-efficient RL algorithms

Part 1. basics, RL w/ a generative model

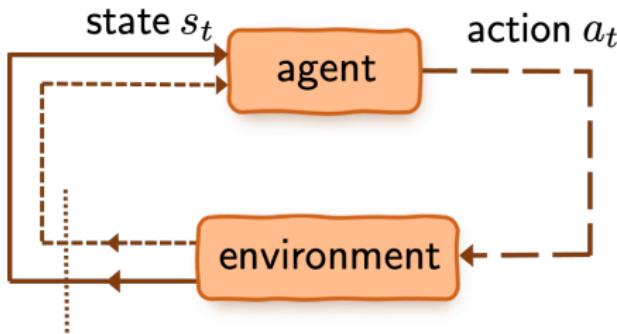
Part 2. online / offline RL, multi-agent / robust RL

# Part 1

1. Basics: Markov decision processes
2. RL w/ a generative model (simulator)
  - o model-based algorithms (a “plug-in” approach)
  - o model-free algorithms

# Markov decision process (MDP)

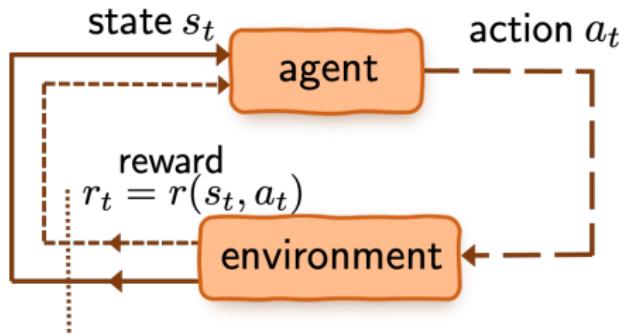
---



- $\mathcal{S} = \{1, \dots, S\}$ : state space (containing  $S$  states)
- $\mathcal{A} = \{1, \dots, A\}$ : action space (containing  $A$  actions)

# Markov decision process (MDP)

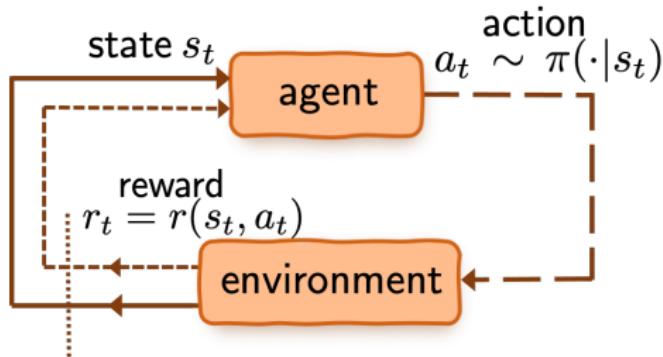
---



- $\mathcal{S} = \{1, \dots, S\}$ : state space (containing  $S$  states)
- $\mathcal{A} = \{1, \dots, A\}$ : action space (containing  $A$  actions)
- $r(s, a) \in [0, 1]$ : immediate reward

# Discounted infinite-horizon MDPs

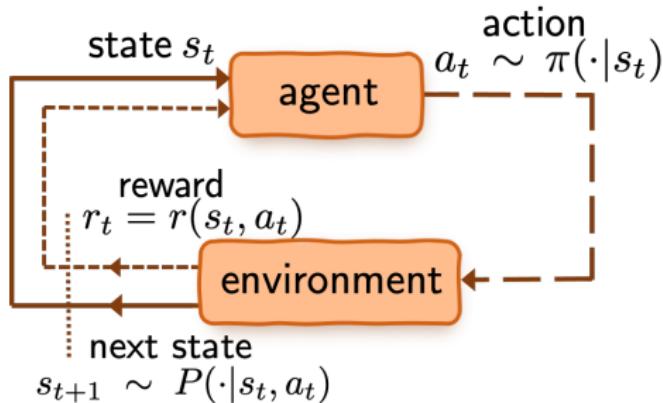
---



- $\mathcal{S} = \{1, \dots, S\}$ : state space (containing  $S$  states)
- $\mathcal{A} = \{1, \dots, A\}$ : action space (containing  $A$  actions)
- $r(s, a) \in [0, 1]$ : immediate reward
- $\pi(\cdot | s)$ : policy (or action selection rule)

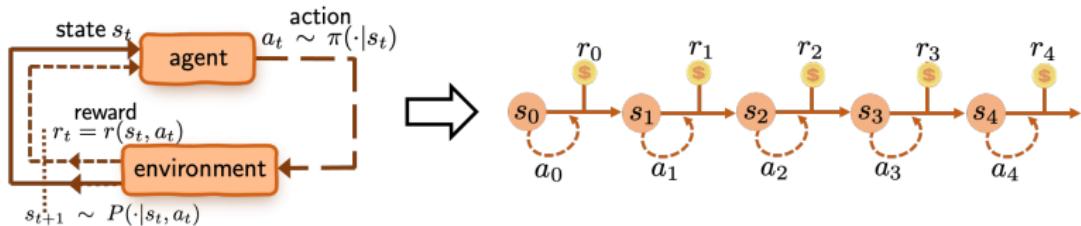
# Discounted infinite-horizon MDPs

---



- $\mathcal{S} = \{1, \dots, S\}$ : state space (containing  $S$  states)
- $\mathcal{A} = \{1, \dots, A\}$ : action space (containing  $A$  actions)
- $r(s, a) \in [0, 1]$ : immediate reward
- $\pi(\cdot | s)$ : policy (or action selection rule)
- $P(\cdot | s, a)$ : **unknown** transition probabilities

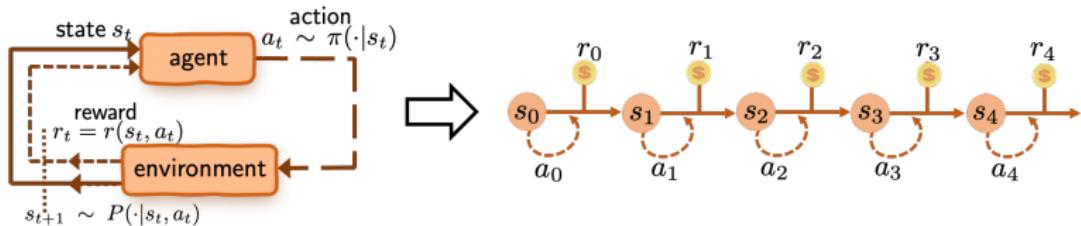
# Value function



Value of policy  $\pi$ : cumulative **discounted** reward

$$\forall s \in \mathcal{S} : V^\pi(s) := \mathbb{E} \left[ \sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \mid s_0 = s \right]$$

# Value function



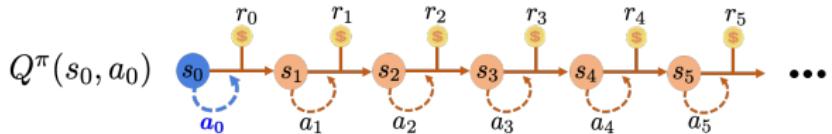
Value of policy  $\pi$ : cumulative **discounted** reward

$$\forall s \in \mathcal{S} : V^\pi(s) := \mathbb{E} \left[ \sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \mid s_0 = s \right]$$

- $\gamma \in [0, 1)$ : discount factor
  - take  $\gamma \rightarrow 1$  to approximate **long-horizon** MDPs
  - **effective horizon**:  $\frac{1}{1-\gamma}$

# Q-function (action-value function)

---



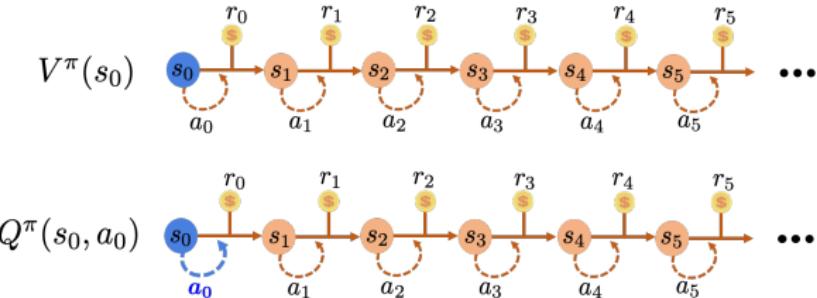
Q-function of policy  $\pi$ :

$$\forall (s, a) \in \mathcal{S} \times \mathcal{A} : Q^\pi(s, a) := \mathbb{E} \left[ \sum_{t=0}^{\infty} \gamma^t r_t \mid s_0 = s, \textcolor{red}{a_0 = a} \right]$$

- $(\textcolor{red}{a_0}, s_1, a_1, s_2, a_2, \dots)$ : induced by policy  $\pi$

# Q-function (action-value function)

---



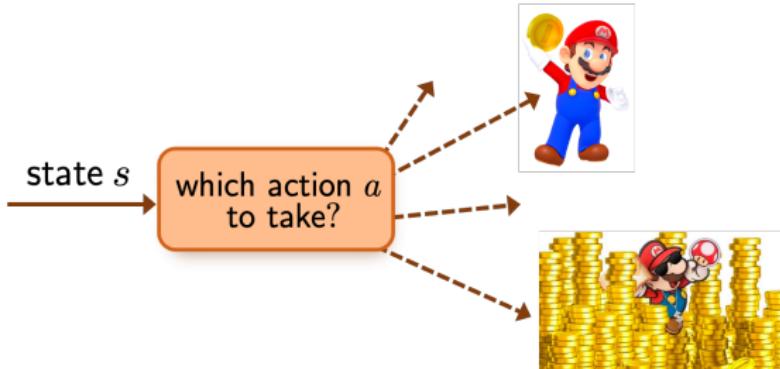
Q-function of policy  $\pi$ :

$$\forall (s, a) \in \mathcal{S} \times \mathcal{A} : \quad Q^\pi(s, a) := \mathbb{E} \left[ \sum_{t=0}^{\infty} \gamma^t r_t \mid s_0 = s, \textcolor{red}{a_0 = a} \right]$$

- $(\textcolor{red}{a_0}, s_1, a_1, s_2, a_2, \dots)$ : induced by policy  $\pi$

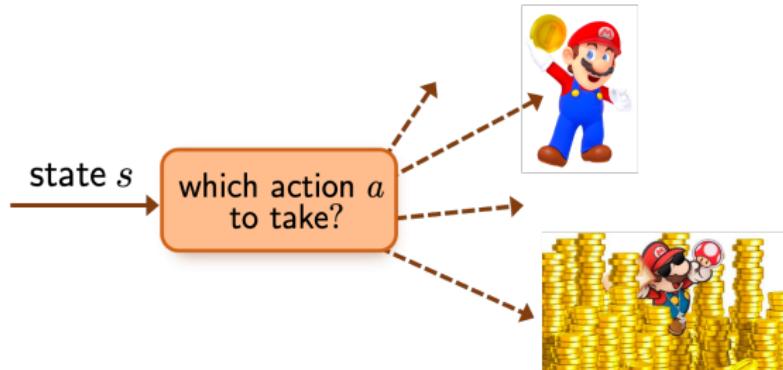
# Optimal policy and optimal value

---



- **optimal policy**  $\pi^*$ : maximizing value function  $\max_{\pi} V^{\pi}$

# Optimal policy and optimal value



- **optimal policy**  $\pi^*$ : maximizing value function  $\max_{\pi} V^{\pi}$

## Theorem (Puterman'94)

For infinite horizon discounted MDP, there always exists a deterministic policy  $\pi^*$ , such that

$$V^{\pi^*}(s) \geq V^{\pi}(s), \quad \forall s, \text{ and } \pi.$$

# Optimal policy and optimal value

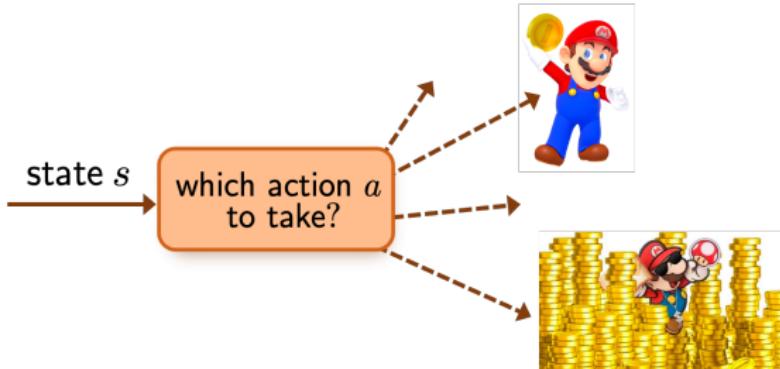
---



- **optimal policy**  $\pi^*$ : maximizing value function  $\max_{\pi} V^{\pi}$
- **optimal value / Q function**:  $V^* := V^{\pi^*}$ ,  $Q^* := Q^{\pi^*}$

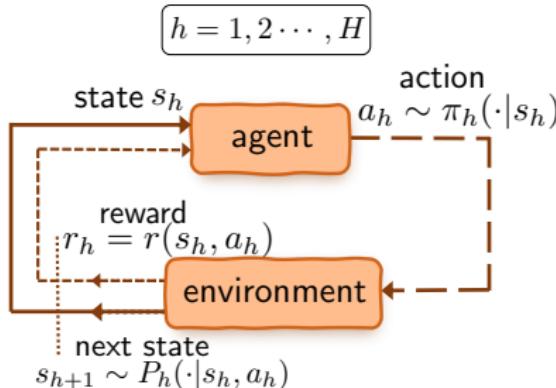
# Optimal policy and optimal value

---



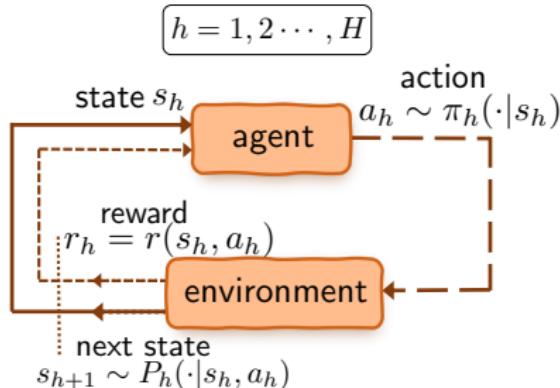
- **optimal policy**  $\pi^*$ : maximizing value function  $\max_{\pi} V^{\pi}$
- **optimal value / Q function**:  $V^* := V^{\pi^*}$ ,  $Q^* := Q^{\pi^*}$
- A question to keep in mind: *how to find optimal  $\pi^*$ ?*

# Finite-horizon MDPs (nonstationary)



- $H$ : horizon length
- $\mathcal{S}$ : state space with size  $S$
- $\mathcal{A}$ : action space with size  $A$
- $r_h(s_h, a_h) \in [0, 1]$ : immediate reward in step  $h$
- $\pi = \{\pi_h\}_{h=1}^H$ : policy (or action selection rule)
- $P_h(\cdot | s, a)$ : transition probabilities in step  $h$

# Finite-horizon MDPs (nonstationary)



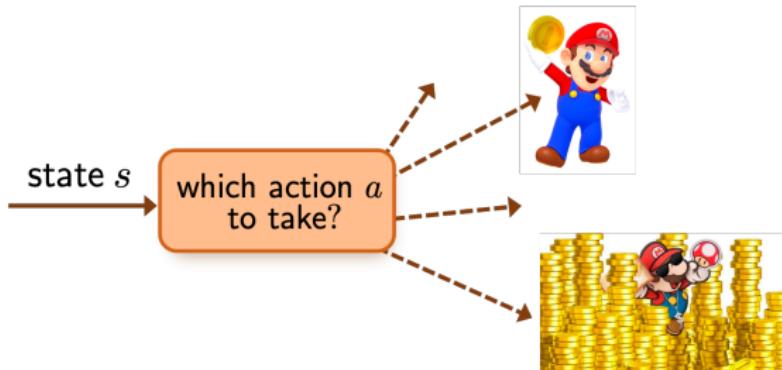
value function:  $V_h^\pi(s) := \mathbb{E} \left[ \sum_{t=h}^H r_h(s_h, a_h) \mid s_h = s \right]$

Q-function:  $Q_h^\pi(s, a) := \mathbb{E} \left[ \sum_{t=h}^H r_h(s_h, a_h) \mid s_h = s, a_h = a \right]$



# Optimal policy and optimal value

---



- **optimal policy**  $\pi^*$ : maximizing value function at all steps
- **optimal value / Q function**:  $V_h^* := V_h^{\pi^*}$ ,  $Q_h^* := Q_h^{\pi^*}$ ,  $\forall h$
- **Question:** *how to find optimal  $\pi^*$ ?*

*Basic dynamic programming algorithms  
when MDP specification is known*

A simpler problem: **policy evaluation**

— given MDP  $\mathcal{M}$  and policy  $\pi$ , how to compute  $V^\pi$ ,  $Q^\pi$ ?

A simpler problem: **policy evaluation**

— given MDP  $\mathcal{M}$  and policy  $\pi$ , how to compute  $V^\pi$ ,  $Q^\pi$ ?

A simpler problem: **policy evaluation**

— given MDP  $\mathcal{M}$  and policy  $\pi$ , how to compute  $V^\pi$ ,  $Q^\pi$ ?

**solution: Bellman's consistency equation**

$$V^\pi(s) = \mathbb{E}_{a \sim \pi(\cdot|s)} [Q^\pi(s, a)]$$
$$Q^\pi(s, a) = \underbrace{r(s, a)}_{\text{immediate reward}} + \gamma \mathbb{E}_{s' \sim P(\cdot|s, a)} \left[ \underbrace{V^\pi(s')}_{\text{next state's value}} \right]$$

- one-step look-ahead



*Richard Bellman*

A simpler problem: **policy evaluation**

— given MDP  $\mathcal{M}$  and policy  $\pi$ , how to compute  $V^\pi$ ,  $Q^\pi$ ?

**solution: Bellman's consistency equation**

$$V^\pi(s) = \mathbb{E}_{a \sim \pi(\cdot|s)} [Q^\pi(s, a)]$$
$$Q^\pi(s, a) = \underbrace{r(s, a)}_{\text{immediate reward}} + \gamma \mathbb{E}_{s' \sim P(\cdot|s, a)} \left[ \underbrace{V^\pi(s')}_{\text{next state's value}} \right]$$

- one-step look-ahead
- $P^\pi$ : state-action transition matrix induced by  $\pi$ :

$$Q^\pi = r + \gamma P^\pi Q^\pi \implies Q^\pi = (I - \gamma P^\pi)^{-1} r$$



*Richard Bellman*

**Back to main question:** how to find optimal policy  $\pi^*$ ?

**solution: Bellman's optimality principle**

- Bellman operator:

$$\mathcal{T}(Q)(s, a) := \underbrace{r(s, a)}_{\text{immediate reward}} + \gamma \mathbb{E}_{s' \sim P(\cdot | s, a)} \left[ \underbrace{\max_{a' \in \mathcal{A}} Q(s', a')}_{\text{next state's value}} \right]$$

- one-step look-ahead
- $\gamma$ -contraction:  $\|\mathcal{T}(Q_1) - \mathcal{T}(Q_2)\|_\infty \leq \gamma \|Q_1 - Q_2\|_\infty$

**Back to main question:** how to find optimal policy  $\pi^*$ ?

**solution: Bellman's optimality principle**

- Bellman operator:

$$\mathcal{T}(Q)(s, a) := \underbrace{r(s, a)}_{\text{immediate reward}} + \gamma \mathbb{E}_{s' \sim P(\cdot | s, a)} \left[ \underbrace{\max_{a' \in \mathcal{A}} Q(s', a')}_{\text{next state's value}} \right]$$

- one-step look-ahead
- $\gamma$ -contraction:  $\|\mathcal{T}(Q_1) - \mathcal{T}(Q_2)\|_\infty \leq \gamma \|Q_1 - Q_2\|_\infty$

- Bellman equation:  $Q^*$  is *unique* solution to

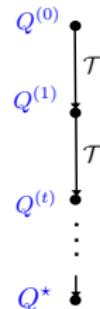
$$\mathcal{T}(Q^*) = Q^*$$

# Two dynamic programming algorithms

## Value iteration (VI)

For  $t = 0, 1, \dots$

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

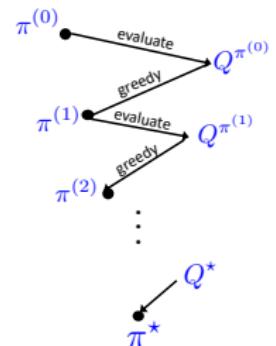


## Policy iteration (PI)

For  $t = 0, 1, \dots$

**policy evaluation:**  $Q^{(t)} = Q^{\pi^{(t)}}$

**policy improvement:**  $\pi^{(t+1)}(s) = \underset{a \in \mathcal{A}}{\operatorname{argmax}} Q^{(t)}(s, a)$



# Iteration complexity

---

**Theorem (Linear convergence of policy/value iteration)**

$$\|Q^{(t)} - Q^*\|_\infty \leq \gamma^t \|Q^{(0)} - Q^*\|_\infty$$

# Iteration complexity

---

**Theorem (Linear convergence of policy/value iteration)**

$$\|Q^{(t)} - Q^*\|_\infty \leq \gamma^t \|Q^{(0)} - Q^*\|_\infty$$

**Implications:** to achieve  $\|Q^{(t)} - Q^*\|_\infty \leq \varepsilon$ , it takes no more than

$$\frac{1}{1-\gamma} \log \left( \frac{\|Q^{(0)} - Q^*\|_\infty}{\varepsilon} \right) \text{ iterations}$$

# Iteration complexity

**Theorem (Linear convergence of policy/value iteration)**

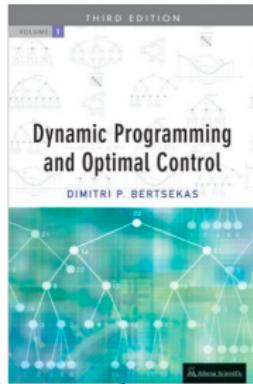
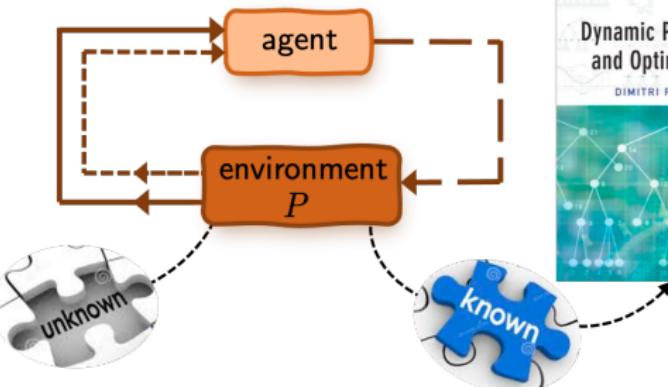
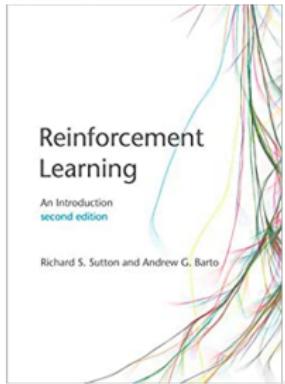
$$\|Q^{(t)} - Q^*\|_\infty \leq \gamma^t \|Q^{(0)} - Q^*\|_\infty$$

**Implications:** to achieve  $\|Q^{(t)} - Q^*\|_\infty \leq \varepsilon$ , it takes no more than

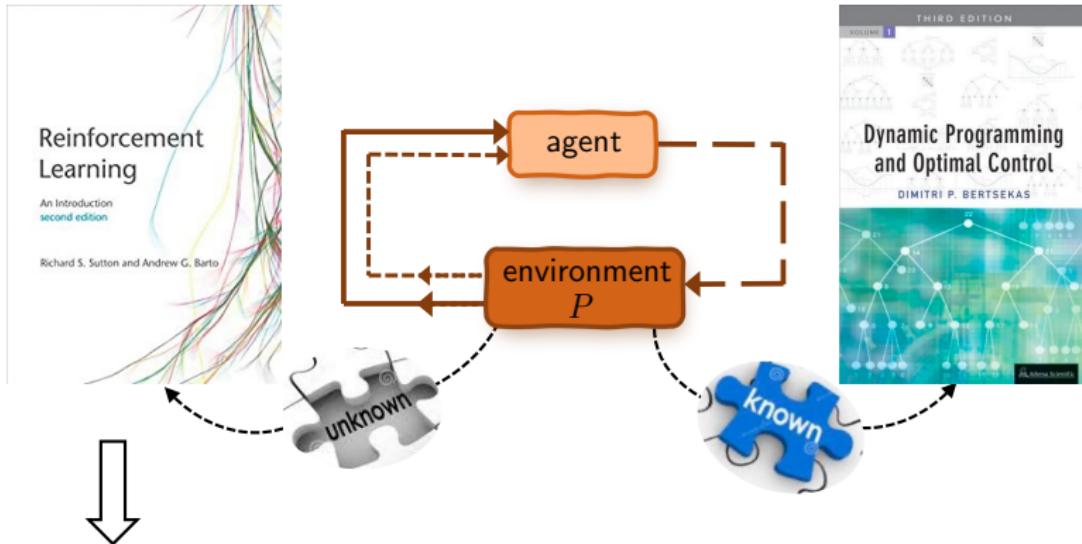
$$\frac{1}{1-\gamma} \log \left( \frac{\|Q^{(0)} - Q^*\|_\infty}{\varepsilon} \right) \text{ iterations}$$

Linear convergence at a **dimension-free** rate!

# When the model is unknown ...



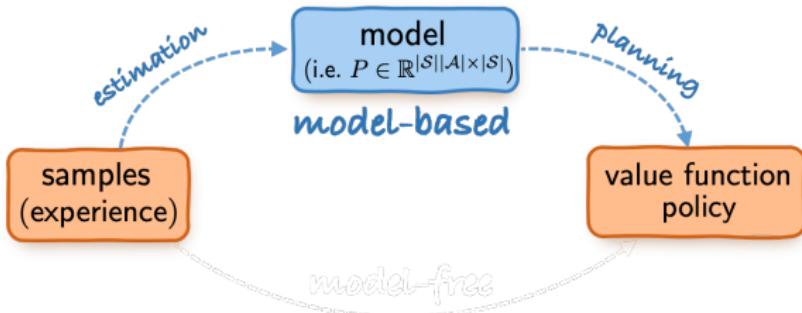
# When the model is unknown ...



Need to learn optimal policy from samples w/o model specification

# Two approaches

---

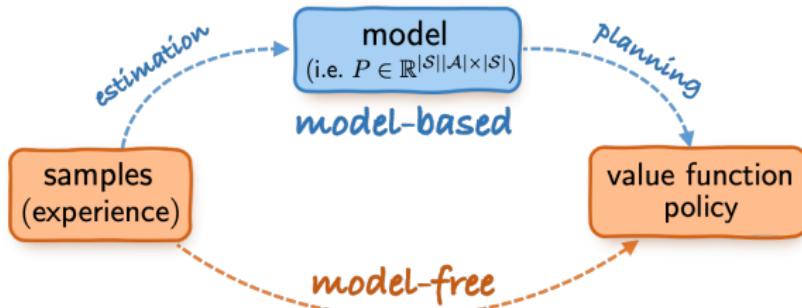


## Model-based approach (“plug-in”)

1. build an empirical estimate  $\hat{P}$  for  $P$
2. planning based on the empirical  $\hat{P}$

# Two approaches

---



## Model-based approach (“plug-in”)

1. build an empirical estimate  $\hat{P}$  for  $P$
2. planning based on the empirical  $\hat{P}$

## Model-free approach

— learning w/o estimating the model explicitly

# Sampling mechanisms

---

1. RL w/ a generative model (a.k.a. simulator)
  - o can query arbitrary state-action pairs to draw samples

# Sampling mechanisms

---

1. RL w/ a generative model (a.k.a. simulator)
  - can query arbitrary state-action pairs to draw samples
2. online RL
  - execute MDP in real time to obtain sample trajectories

# Sampling mechanisms

---

1. RL w/ a generative model (a.k.a. simulator)
  - can query arbitrary state-action pairs to draw samples
2. online RL
  - execute MDP in real time to obtain sample trajectories
3. offline RL
  - use pre-collected historical data

# Sampling mechanisms

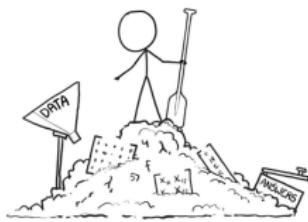
---

1. RL w/ a generative model (a.k.a. simulator)
  - can query arbitrary state-action pairs to draw samples
2. online RL
  - execute MDP in real time to obtain sample trajectories
3. offline RL
  - use pre-collected historical data

**Question:** how many samples are sufficient to learn an  $\varepsilon$ -optimal policy?

$$\hat{V^\pi} \geq V^* - \varepsilon$$

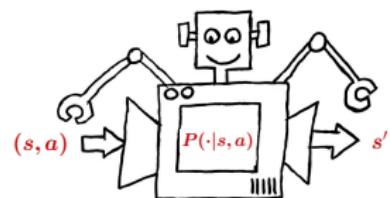
# Exploration vs exploitation



offline RL

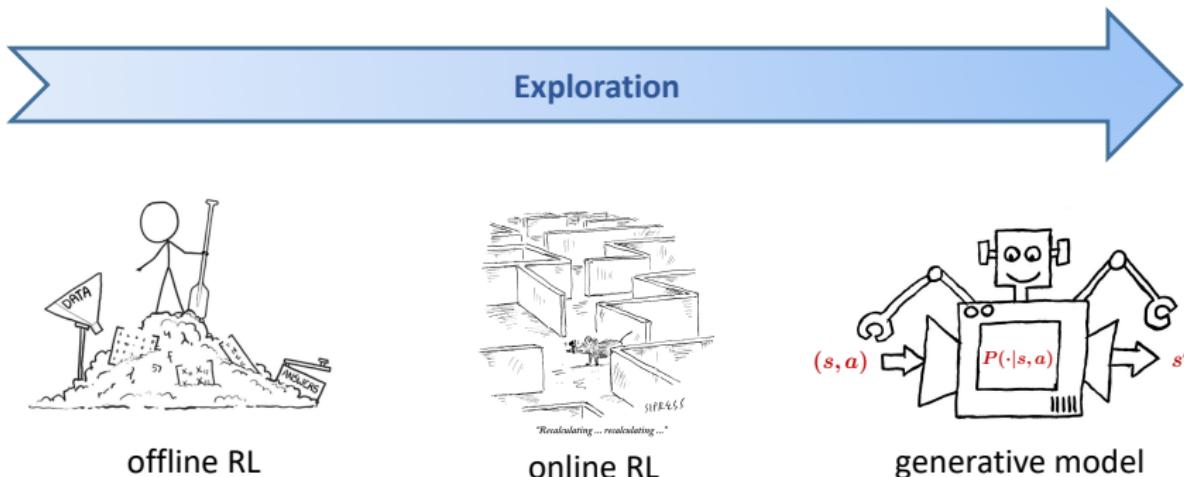


online RL



generative model

# Exploration vs exploitation



Varying levels of trade-offs between exploration and exploitation.

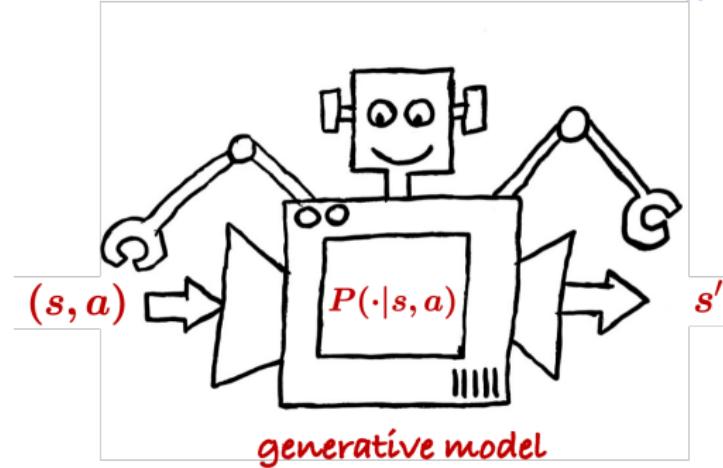
# Part 1

1. Basics: Markov decision processes
2. RL w/ a generative model (simulator)
  - o model-based algorithms (a “plug-in” approach)
  - o model-free algorithms

# A generative model / simulator

---

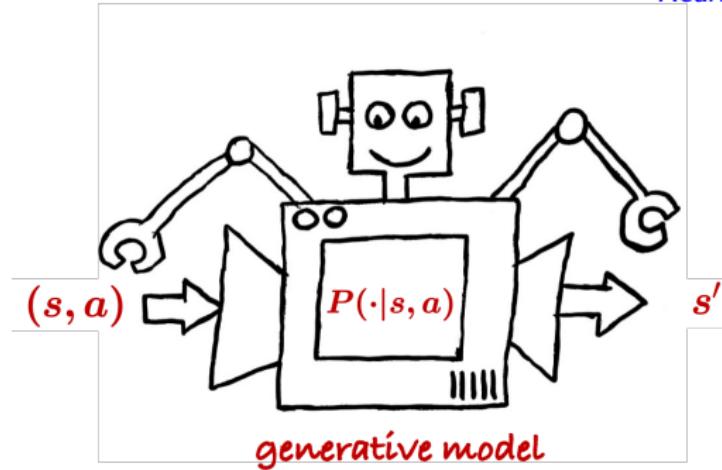
— Kearns and Singh, 1999



- **sampling:** for each  $(s, a)$ , collect  $N$  samples  $\{(s, a, s'_{(i)})\}_{1 \leq i \leq N}$

# A generative model / simulator

— Kearns and Singh, 1999



- **sampling:** for each  $(s, a)$ , collect  $N$  samples  $\{(s, a, s'_{(i)})\}_{1 \leq i \leq N}$
- construct  $\hat{\pi}$  based on samples (in total  $SA \times N$ )

$\ell_\infty$ -sample complexity: how many samples are required to  
learn an  $\varepsilon$ -optimal policy ?

$$\forall s: \hat{V^\pi}(s) \geq V^*(s) - \varepsilon$$

## Minimax lower bound

---

### Theorem (minimax lower bound; Azar et al., 2013)

For all  $\varepsilon \in [0, \frac{1}{1-\gamma})$ , there exists some MDP such that the total number of samples need to be *at least*

$$\tilde{\Omega} \left( \frac{|\mathcal{S}||\mathcal{A}|}{(1-\gamma)^3 \varepsilon^2} \right)$$

to achieve  $V^* - V^{\widehat{\pi}} \leq \varepsilon$ , where  $\widehat{\pi}$  is the output of any RL algorithm.

## Minimax lower bound

### Theorem (minimax lower bound; Azar et al., 2013)

For all  $\varepsilon \in [0, \frac{1}{1-\gamma})$ , there exists some MDP such that the total number of samples need to be *at least*

$$\tilde{\Omega}\left(\frac{|\mathcal{S}||\mathcal{A}|}{(1-\gamma)^3\varepsilon^2}\right)$$

to achieve  $V^* - V^{\widehat{\pi}} \leq \varepsilon$ , where  $\widehat{\pi}$  is the output of any RL algorithm.

- holds for both finding the optimal Q-function and the optimal policy over the entire range of  $\varepsilon$
- much smaller than the model dimension  $|\mathcal{S}|^2|\mathcal{A}|$

# An incomplete list of works

---

- Kearns and Singh, 1999
- Kakade, 2003
- Kearns et al., 2002
- Azar et al., 2013
- Sidford et al., 2018a, 2018b
- Wang, 2019
- Agarwal et al., 2019
- Wainwright, 2019a, 2019b
- Pananjady and Wainwright, 2019
- Yang and Wang, 2019
- Khamaru et al., 2020
- Mou et al., 2020
- Cui and Yang, 2021
- ...

## An even shorter list of prior art

---

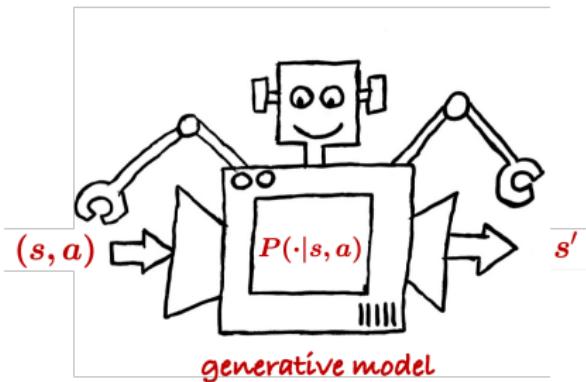
algorithm	sample size range	sample complexity	$\varepsilon$ -range
Empirical QVI Azar et al., 2013	$\left[ \frac{S^2 A}{(1-\gamma)^2}, \infty \right)$	$\frac{SA}{(1-\gamma)^3 \varepsilon^2}$	$(0, \frac{1}{\sqrt{(1-\gamma)S}}]$
Sublinear randomized VI Sidford et al., 2018b	$\left[ \frac{SA}{(1-\gamma)^2}, \infty \right)$	$\frac{SA}{(1-\gamma)^4 \varepsilon^2}$	$(0, \frac{1}{1-\gamma}]$
Variance-reduced QVI Sidford et al., 2018a	$\left[ \frac{SA}{(1-\gamma)^3}, \infty \right)$	$\frac{SA}{(1-\gamma)^3 \varepsilon^2}$	$(0, 1]$
Randomized primal-dual Wang 2019	$\left[ \frac{SA}{(1-\gamma)^2}, \infty \right)$	$\frac{SA}{(1-\gamma)^4 \varepsilon^2}$	$(0, \frac{1}{1-\gamma}]$
Empirical MDP + planning Agarwal et al., 2019	$\left[ \frac{SA}{(1-\gamma)^2}, \infty \right)$	$\frac{SA}{(1-\gamma)^3 \varepsilon^2}$	$(0, \frac{1}{\sqrt{1-\gamma}}]$

important parameters  
 $\implies$

- # states  $S$ , # actions  $A$
- the discounted complexity  $\frac{1}{1-\gamma}$
- approximation error  $\varepsilon \in (0, \frac{1}{1-\gamma}]$

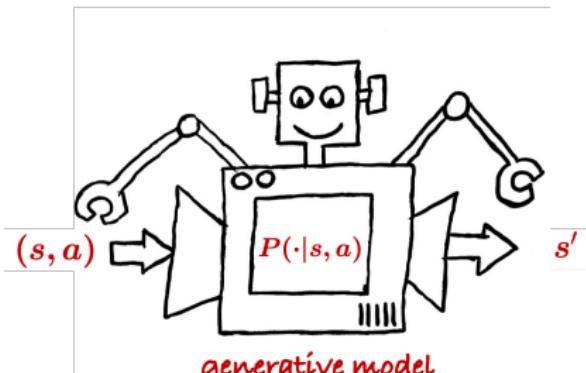
# Model estimation

---



**Sampling:** for each  $(s, a)$ ,  
collect  $N$  ind. samples  
 $\{(s, a, s'_{(i)})\}_{1 \leq i \leq N}$

# Model estimation



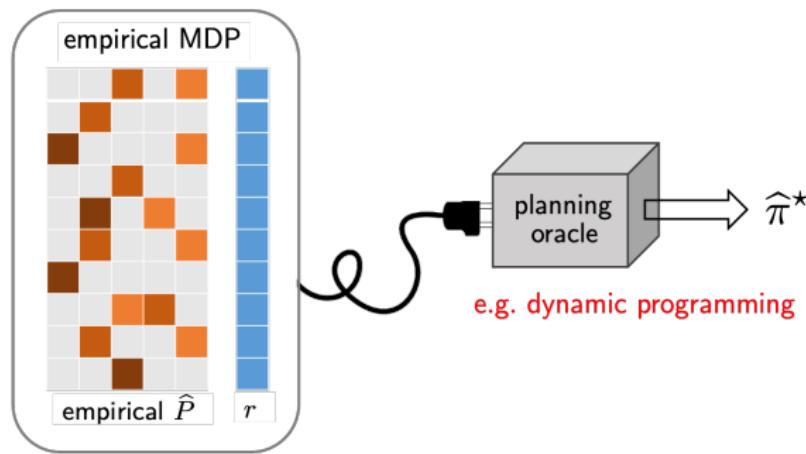
**Sampling:** for each  $(s, a)$ ,  
collect  $N$  ind. samples  
 $\{(s, a, s'_{(i)})\}_{1 \leq i \leq N}$

**Empirical estimates:**

$$\widehat{P}(s'|s, a) = \underbrace{\frac{1}{N} \sum_{i=1}^N \mathbb{1}\{s'_{(i)} = s'\}}_{\text{empirical frequency}}$$

# Empirical MDP + planning

— Azar et al., 2013, Agarwal et al., 2019

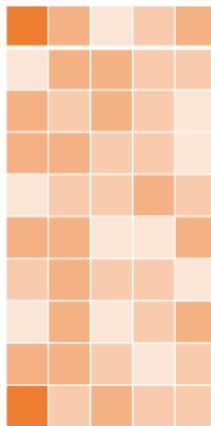


Find policy based on the empirical MDP (*empirical maximizer*)  
using, e.g., policy iteration

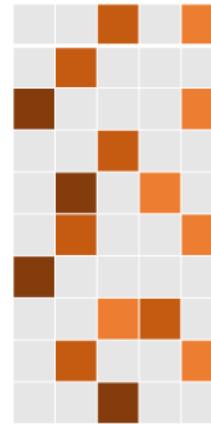
$$(\hat{P}, r)$$

# Challenges in the sample-starved regime

---



truth:  $P \in \mathbb{R}^{SA \times S}$

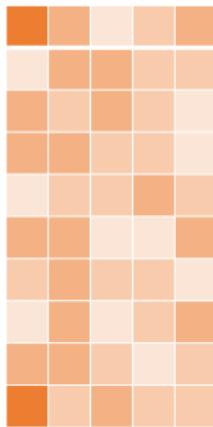


empirical estimate:  
 $\hat{P}$

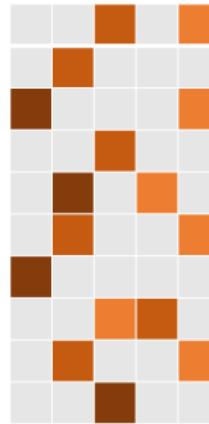
- Can't recover  $P$  faithfully if sample size  $\ll S^2 A$ !

# Challenges in the sample-starved regime

---



truth:  $P \in \mathbb{R}^{SA \times S}$



empirical estimate:  
 $\hat{P}$

- Can't recover  $P$  faithfully if sample size  $\ll S^2 A$ !
- Can we trust our policy estimate when reliable model estimation is infeasible?

## $\ell_\infty$ -based sample complexity

### Theorem (Agarwal, Kakade, Yang '19)

For any  $0 < \varepsilon \leq \frac{1}{\sqrt{1-\gamma}}$ , the optimal policy  $\hat{\pi}^*$  of empirical MDP achieves

$$\|V^{\hat{\pi}^*} - V^*\|_\infty \leq \varepsilon$$

with high prob., with sample complexity at most

$$\tilde{O}\left(\frac{SA}{(1-\gamma)^3\varepsilon^2}\right)$$

## $\ell_\infty$ -based sample complexity

### Theorem (Agarwal, Kakade, Yang '19)

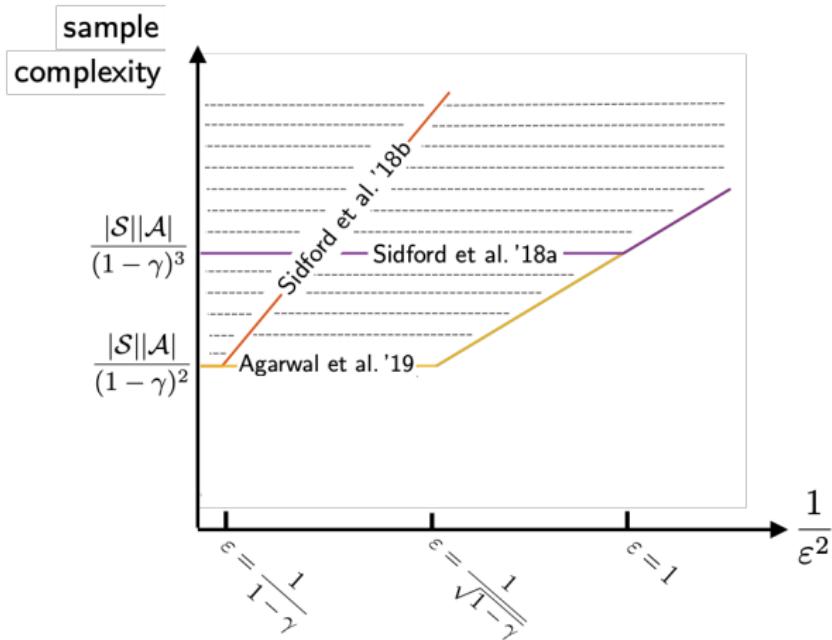
For any  $0 < \varepsilon \leq \frac{1}{\sqrt{1-\gamma}}$ , the optimal policy  $\hat{\pi}^*$  of empirical MDP achieves

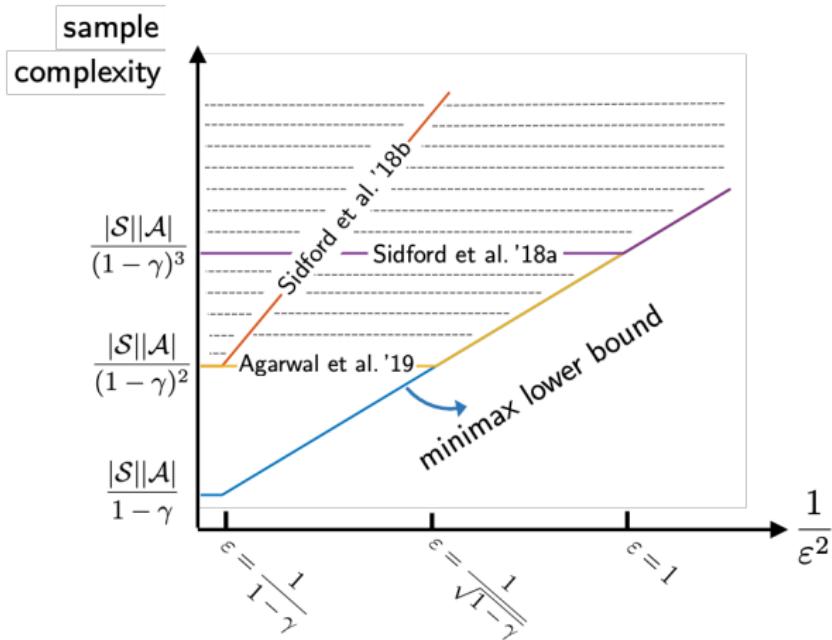
$$\|V^{\hat{\pi}^*} - V^*\|_\infty \leq \varepsilon$$

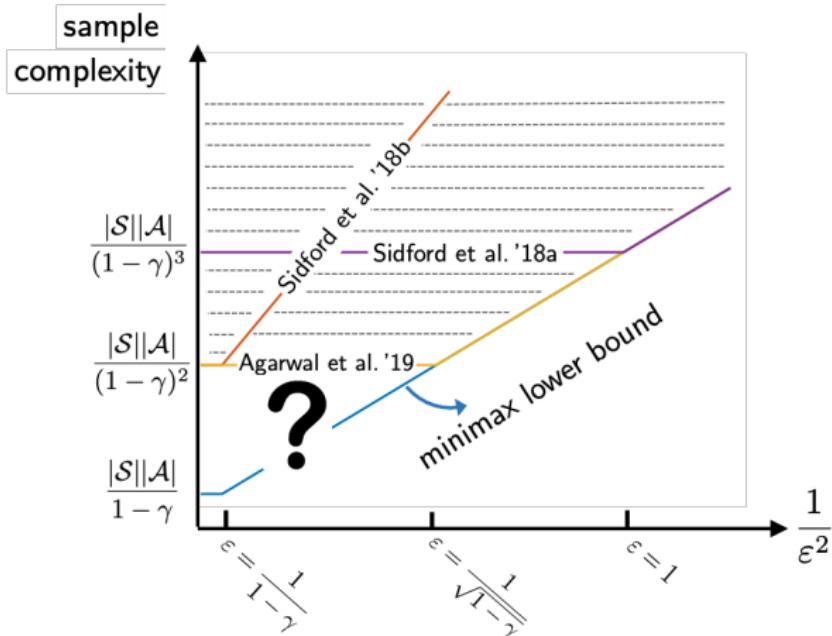
with high prob., with sample complexity at most

$$\tilde{O}\left(\frac{SA}{(1-\gamma)^3\varepsilon^2}\right)$$

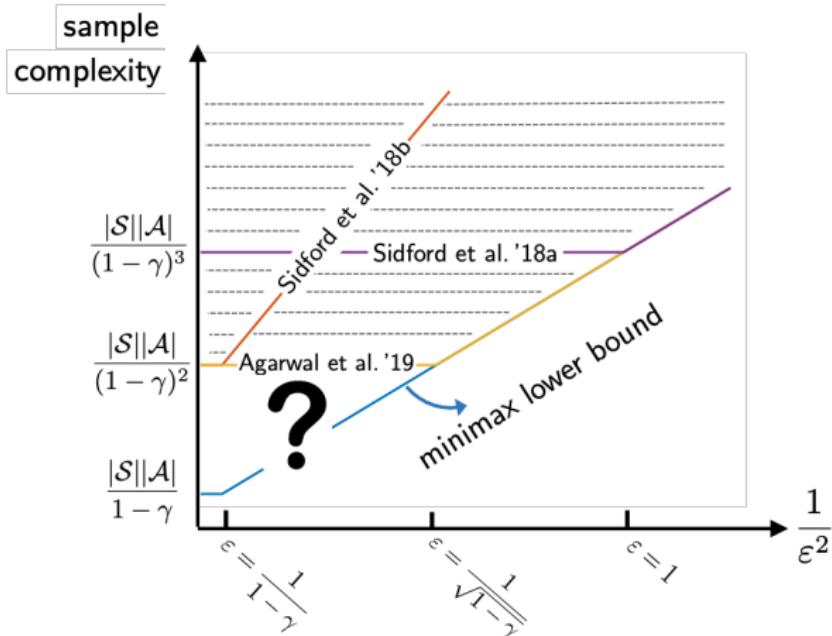
- matches minimax lower bound:  $\tilde{\Omega}\left(\frac{SA}{(1-\gamma)^3\varepsilon^2}\right)$  when  $\varepsilon \leq \frac{1}{\sqrt{1-\gamma}}$   
(equivalently, when sample size exceeds  $\frac{SA}{(1-\gamma)^2}$ ) Azar et al., 2013







Agarwal et al., 2019 still requires a **burn-in sample size**  $\gtrsim \frac{SA}{(1-\gamma)^2}$

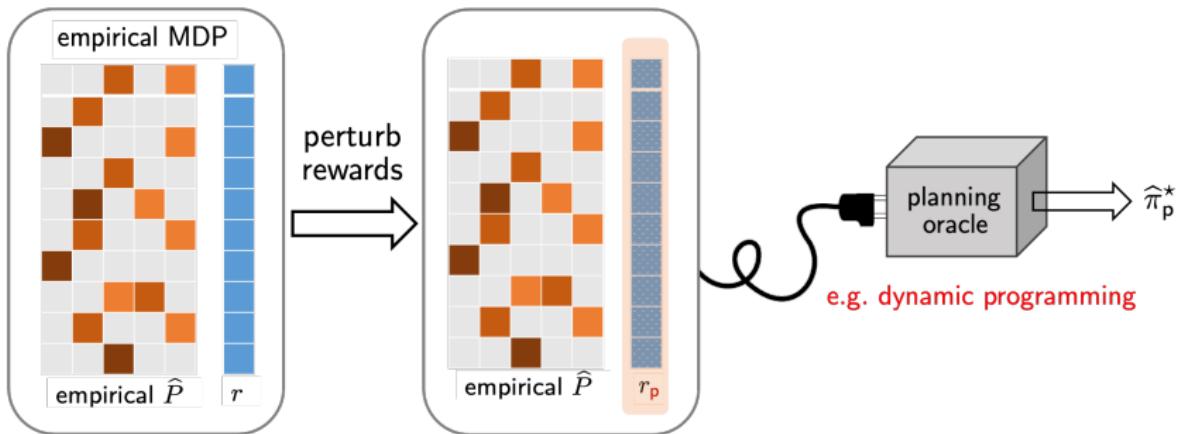


Agarwal et al., 2019 still requires a **burn-in sample size**  $\gtrsim \frac{SA}{(1-\gamma)^2}$

**Question:** is it possible to break this sample size barrier?

# Perturbed model-based approach (Li et al. '24)

— Li, Wei, Chi, Chen, 2024



Find policy based on empirical MDP w/ slightly perturbed rewards

# Optimal $\ell_\infty$ -based sample complexity

**Theorem (Li, Wei, Chi, Chen '20; OR '24)**

For any  $0 < \varepsilon \leq \frac{1}{1-\gamma}$ , the optimal policy  $\widehat{\pi}_p^*$  of perturbed empirical MDP achieves

$$\|V^{\widehat{\pi}_p^*} - V^*\|_\infty \leq \varepsilon$$

with high prob., with sample complexity at most

$$\tilde{O}\left(\frac{SA}{(1-\gamma)^3\varepsilon^2}\right)$$

# Optimal $\ell_\infty$ -based sample complexity

**Theorem (Li, Wei, Chi, Chen '20; OR '24)**

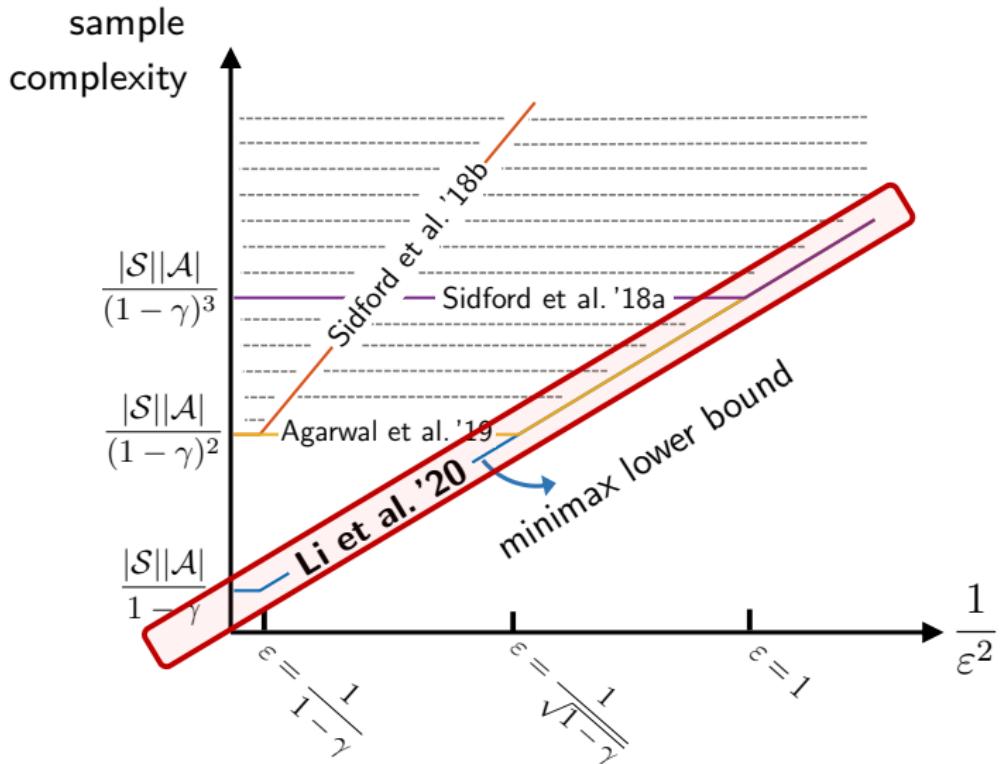
For any  $0 < \varepsilon \leq \frac{1}{1-\gamma}$ , the optimal policy  $\widehat{\pi}_p^*$  of perturbed empirical MDP achieves

$$\|V^{\widehat{\pi}_p^*} - V^*\|_\infty \leq \varepsilon$$

with high prob., with sample complexity at most

$$\tilde{O}\left(\frac{SA}{(1-\gamma)^3\varepsilon^2}\right)$$

- matches minimax lower bound:  $\widetilde{\Omega}\left(\frac{SA}{(1-\gamma)^3\varepsilon^2}\right)$  Azar et al., 2013
- full  $\varepsilon$ -range:  $\varepsilon \in (0, \frac{1}{1-\gamma}] \rightarrow$  no burn-in cost



# Notation and Bellman equation

---

**Bellman equation:**  $V^\pi = r_\pi + \gamma P_\pi V^\pi$

- $V^\pi$ : value function under policy  $\pi$ 
  - Bellman equation:  $V^\pi = (I - \gamma P_\pi)^{-1} r_\pi$
- $\hat{V}^\pi$ : empirical version value function under policy  $\pi$ 
  - Bellman equation:  $\hat{V}^\pi = (I - \gamma \hat{P}_\pi)^{-1} r_\pi$

# Notation and Bellman equation

---

**Bellman equation:**  $V^\pi = r_\pi + \gamma P_\pi V^\pi$

- $V^\pi$ : value function under policy  $\pi$ 
  - Bellman equation:  $V^\pi = (I - \gamma P_\pi)^{-1} r_\pi$
- $\hat{V}^\pi$ : empirical version value function under policy  $\pi$ 
  - Bellman equation:  $\hat{V}^\pi = (I - \gamma \hat{P}_\pi)^{-1} r_\pi$
- $\pi^*$ : optimal policy for  $V^\pi$
- $\hat{\pi}^*$ : optimal policy for  $\hat{V}^\pi$

# Main steps

---

Elementary decomposition:

$$\begin{aligned} V^* - V^{\widehat{\pi}^*} &= (V^* - \widehat{V}^{\pi^*}) + (\widehat{V}^{\pi^*} - \widehat{V}^{\widehat{\pi}^*}) + (\widehat{V}^{\widehat{\pi}^*} - V^{\widehat{\pi}^*}) \\ &\leq (V^{\pi^*} - \widehat{V}^{\pi^*}) + \textcolor{red}{0} + (\widehat{V}^{\widehat{\pi}^*} - V^{\widehat{\pi}^*}) \end{aligned}$$

# Main steps

---

Elementary decomposition:

$$\begin{aligned} V^* - V^{\widehat{\pi}^*} &= (V^* - \widehat{V}^{\pi^*}) + (\widehat{V}^{\pi^*} - \widehat{V}^{\widehat{\pi}^*}) + (\widehat{V}^{\widehat{\pi}^*} - V^{\widehat{\pi}^*}) \\ &\leq (V^{\pi^*} - \widehat{V}^{\pi^*}) + 0 + (\widehat{V}^{\widehat{\pi}^*} - V^{\widehat{\pi}^*}) \end{aligned}$$

- **Step 1:** control  $V^\pi - \widehat{V}^\pi$  for a fixed  $\pi$  (called “policy evaluation”)  
(Bernstein inequality + a peeling argument)

# Main steps

---

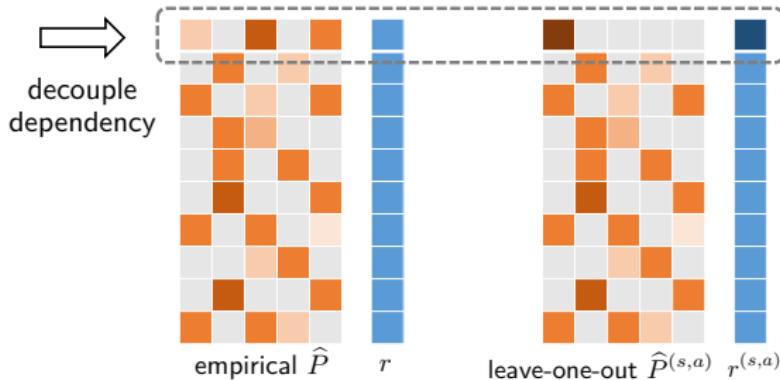
Elementary decomposition:

$$\begin{aligned} V^* - V^{\hat{\pi}^*} &= (V^* - \hat{V}^{\pi^*}) + (\hat{V}^{\pi^*} - \hat{V}^{\hat{\pi}^*}) + (\hat{V}^{\hat{\pi}^*} - V^{\hat{\pi}^*}) \\ &\leq (V^{\pi^*} - \hat{V}^{\pi^*}) + 0 + (\hat{V}^{\hat{\pi}^*} - V^{\hat{\pi}^*}) \end{aligned}$$

- **Step 1:** control  $V^\pi - \hat{V}^\pi$  for a fixed  $\pi$  (called “policy evaluation”)  
(Bernstein inequality + a peeling argument)
- **Step 2:** extend it to control  $\hat{V}^{\hat{\pi}^*} - V^{\hat{\pi}^*}$  ( $\hat{\pi}^*$  depends on samples)  
(decouple statistical dependency)

# A glimpse of key analysis ideas

1. leave-one-out analysis: decouple statistical dependency



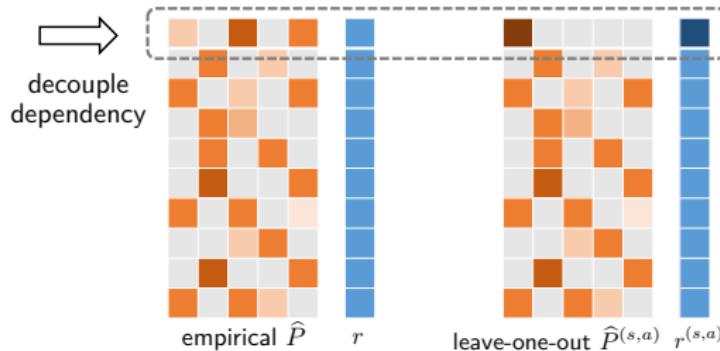
2. tie-breaking via random perturbation



# Key idea 1: leave-one-out analysis

---

Decouple dependency by introducing auxiliary state-action absorbing MDPs by dropping randomness for each  $(s, a)$



— inspired by Agarwal et al. '19 but quite different ...

# Key idea 1: leave-one-out analysis

---

- El Karoui, Bean, Bickel, Lim, Yu, 2013
- El Karoui, 2015
- Javanmard, Montanari, 2015
- Zhong, Boumal, 2017
- Lei, Bickel, El Karoui, 2017
- Sur, Chen, Candès, 2017
- Abbe, Fan, Wang, Zhong, 2017
- Chen, Fan, Ma, Wang, 2017
- Ma, Wang, Chi, Chen, 2017
- Chen, Chi, Fan, Ma, 2018
- Ding, Chen, 2018
- Dong, Shi, 2018
- Chen, Chi, Fan, Ma, Yan, 2019
- Chen, Fan, Ma, Yan, 2019
- Cai, Li, Poor, Chen, 2019
- Agarwal, Kakade, Yang, 2019
- Pananjady, Wainwright, 2019
- Ling, 2020
- Yan, Chen, Fan, 2024

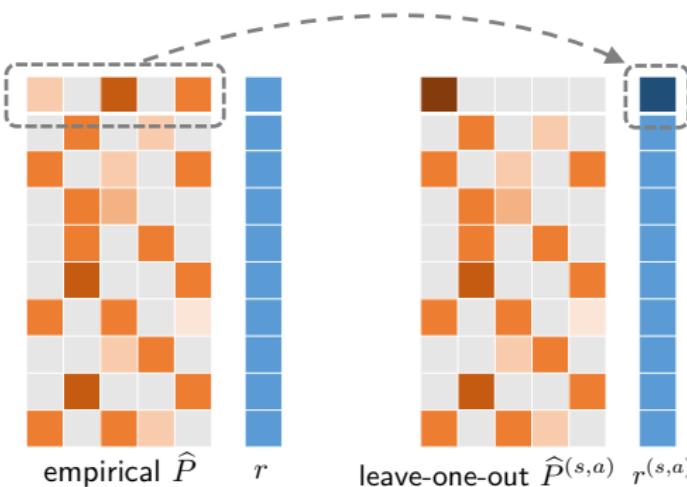
Foundations and Trends® in Machine Learning  
**Spectral Methods for Data Science: A Statistical Perspective**

Suggested Citation: Yuxin Chen, Yuejie Chi, Jianqing Fan and Cong Ma (2020), "Spectral Methods for Data Science: A Statistical Perspective", Foundations and Trends® in

4 Fine-grained analysis: $\ell_\infty$ and $\ell_{2,\infty}$ perturbation theory	126
4.1 Leave-one-out-analysis: An illustrative example . . . . .	127

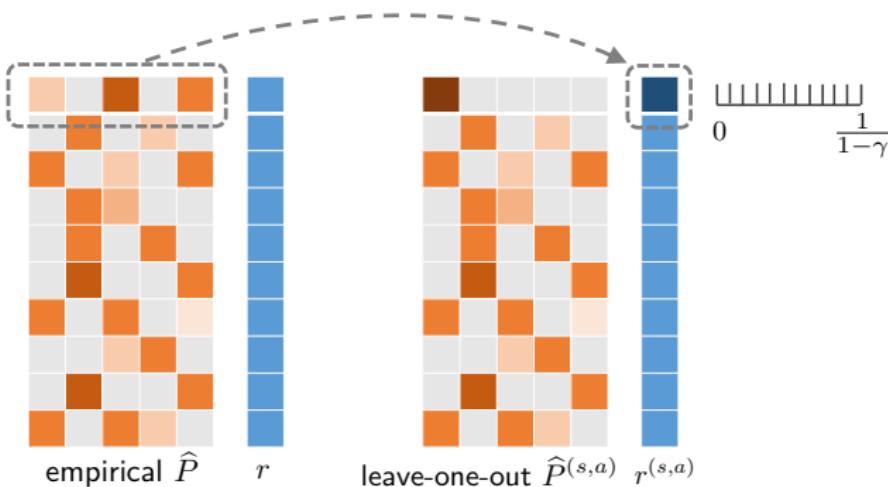
# Key idea 1: leave-one-out analysis

---



1. embed all randomness from  $\hat{P}_{s,a}$  into a single scalar (i.e.  $r_{s,a}^{(s,a)}$ )

# Key idea 1: leave-one-out analysis

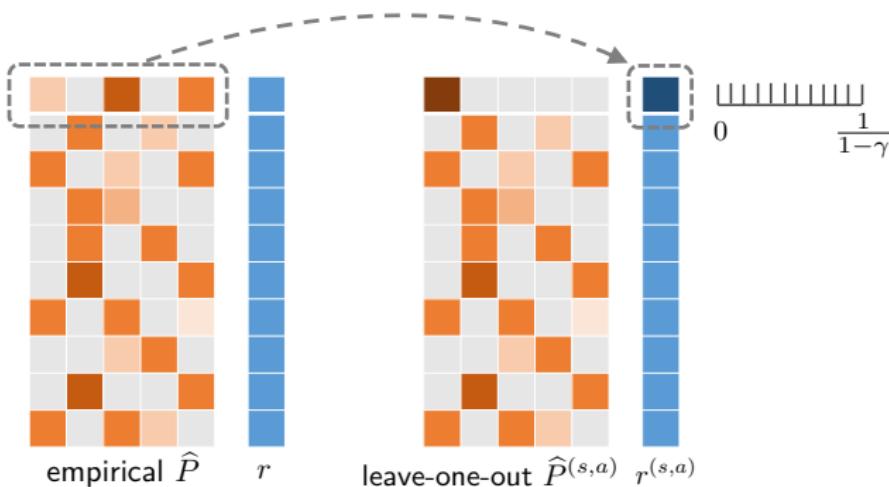


1. embed all randomness from  $\hat{P}_{s,a}$  into a single scalar (i.e.  $r_{s,a}^{(s,a)}$ )
2. build an  $\epsilon$ -net for this scalar

*works under a separation condition*

$$\forall s, \quad \hat{Q}^*(s, \hat{\pi}^*(s)) - \max_{a: a \neq \hat{\pi}^*(s)} \hat{Q}^*(s, a) > 0$$

# Key idea 1: leave-one-out analysis



1. embed all randomness from  $\hat{P}_{s,a}$  into a single scalar (i.e.  $r_{s,a}^{(s,a)}$ )
2. build an  **$\epsilon$ -net** for this scalar

*works under a separation condition*

$$\forall s, \quad \hat{Q}^*(s, \hat{\pi}^*(s)) - \max_{a: a \neq \hat{\pi}^*(s)} \hat{Q}^*(s, a) > 0$$

## Key idea 2: tie-breaking via perturbation

---

- How to ensure separation between the optimal policy and others?

$$\forall s, \quad \hat{Q}^*(s, \hat{\pi}^*(s)) - \max_{a: a \neq \hat{\pi}^*(s)} \hat{Q}^*(s, a) > 0$$

## Key idea 2: tie-breaking via perturbation

---

- How to ensure separation between the optimal policy and others?

$$\forall s, \quad \hat{Q}^*(s, \hat{\pi}^*(s)) - \max_{a: a \neq \hat{\pi}^*(s)} \hat{Q}^*(s, a) > 0$$

- **Solution:** *slightly perturb rewards r*  $\implies \hat{\pi}_p^*$ 
  - ensures  $\hat{\pi}_p^*$  can be differentiated from others with high prob.



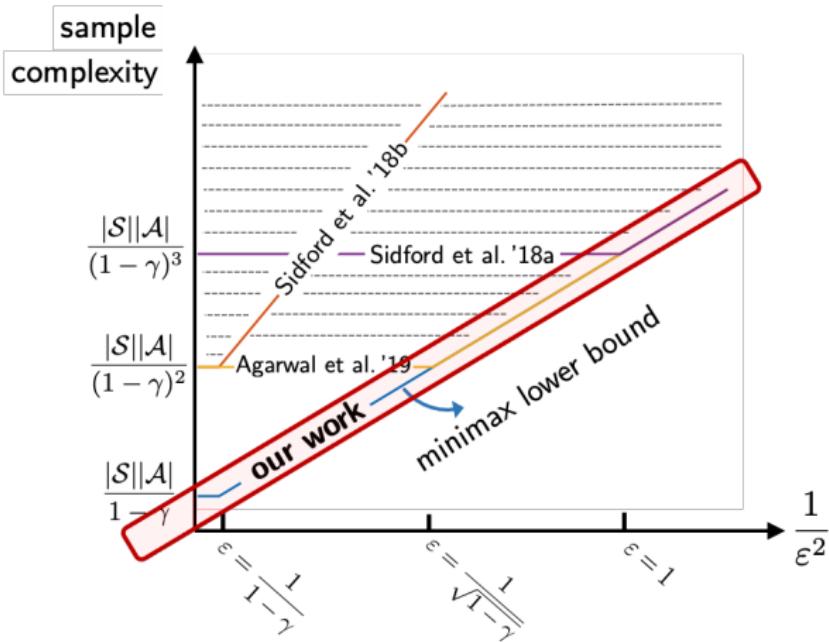
## Key idea 2: tie-breaking via perturbation

- How to ensure separation between the optimal policy and others?

$$\forall s, \quad \hat{Q}^*(s, \hat{\pi}^*(s)) - \max_{a: a \neq \hat{\pi}^*(s)} \hat{Q}^*(s, a) > \frac{(1-\gamma)\varepsilon}{S^5 A^5}$$

- **Solution:** slightly perturb rewards  $r$   $\implies \hat{\pi}_p^*$ 
  - ensures  $\hat{\pi}_p^*$  can be differentiated from others with high prob.



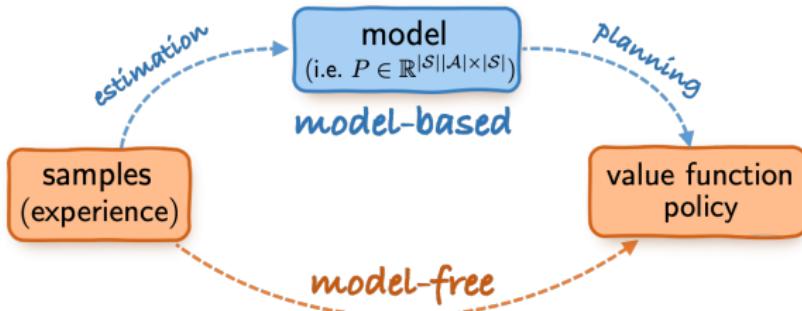


Model based RL is minimax optimal under generative models  
and does NOT suffer from a sample size barrier

# Part 1

1. Basics: Markov decision processes
2. RL w/ a generative model (simulator)
  - o model-based algorithms (a “plug-in” approach)
  - o model-free algorithms

# Model-based vs. model-free RL

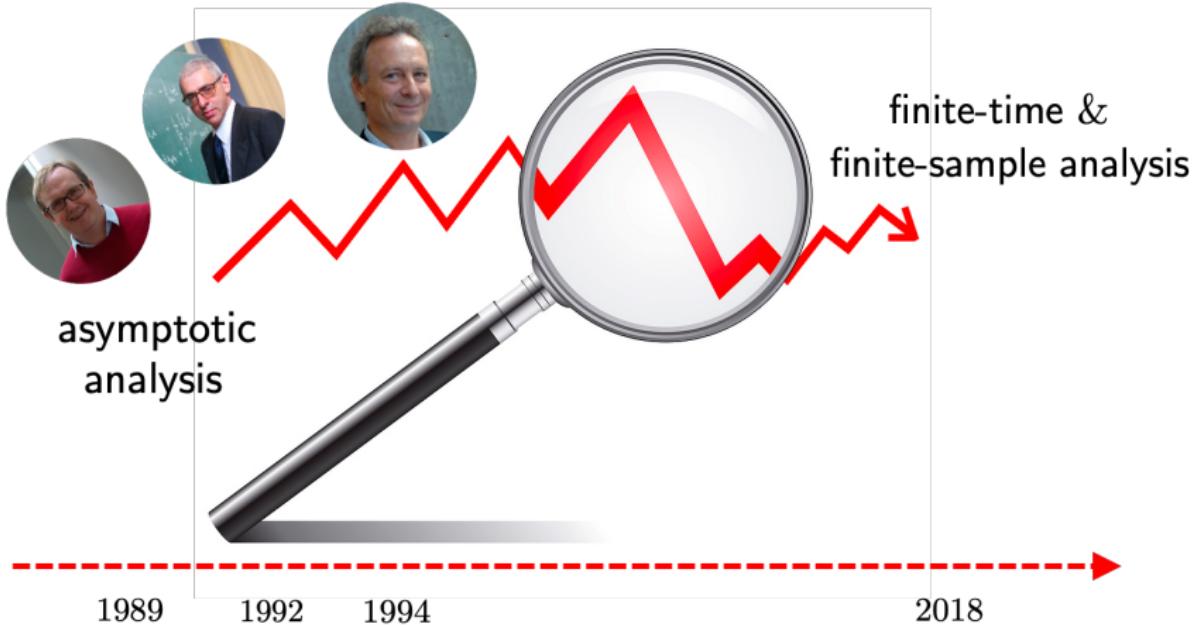


## Model-based approach (“plug-in”)

1. build empirical estimate  $\hat{P}$  for  $P$
2. planning based on empirical  $\hat{P}$

## Model-free / value-based approach

- learning w/o modeling & estimating environment explicitly
- memory-efficient, online, ...



Focus of this part: classical **Q-learning** algorithm and its variants

# A starting point: Bellman optimality principle

---

## Bellman operator

$$\mathcal{T}(Q)(s, a) := \underbrace{r(s, a)}_{\text{immediate reward}} + \gamma \mathbb{E}_{s' \sim P(\cdot | s, a)} \left[ \underbrace{\max_{a' \in \mathcal{A}} Q(s', a')}_{\text{next state's value}} \right]$$

- one-step look-ahead

# A starting point: Bellman optimality principle

---

## Bellman operator

$$\mathcal{T}(Q)(s, a) := \underbrace{r(s, a)}_{\text{immediate reward}} + \gamma \mathbb{E}_{s' \sim P(\cdot | s, a)} \left[ \underbrace{\max_{a' \in \mathcal{A}} Q(s', a')}_{\text{next state's value}} \right]$$

- one-step look-ahead

**Bellman equation:**  $Q^*$  is *unique* solution to

$$\mathcal{T}(Q^*) = Q^*$$

# A starting point: Bellman optimality principle

## Bellman operator

$$\mathcal{T}(Q)(s, a) := \underbrace{r(s, a)}_{\text{immediate reward}} + \gamma \mathbb{E}_{s' \sim P(\cdot | s, a)} \left[ \underbrace{\max_{a' \in \mathcal{A}} Q(s', a')}_{\text{next state's value}} \right]$$

- one-step look-ahead

**Bellman equation:**  $Q^*$  is *unique* solution to

$$\mathcal{T}(Q^*) = Q^*$$

- **takeaway message:** it suffices to solve the Bellman equation
- **challenge:** how to solve it using stochastic samples?



Richard Bellman

# Q-learning: a stochastic approximation algorithm



Chris Watkins



Peter Dayan

Stochastic approximation for solving the **Bellman equation**

Robbins & Monro, 1951

$$\mathcal{T}(Q) - Q = 0$$

where

$$\mathcal{T}(Q)(s, a) := \underbrace{r(s, a)}_{\text{immediate reward}} + \gamma \mathbb{E}_{s' \sim P(\cdot | s, a)} \left[ \underbrace{\max_{a' \in \mathcal{A}} Q(s', a')}_{\text{next state's value}} \right].$$

# Q-learning: a stochastic approximation algorithm

---



Chris Watkins



Peter Dayan

Stochastic approximation for solving Bellman equation  $\mathcal{T}(Q) - Q = 0$

$$\underbrace{Q_{t+1}(s, a) = Q_t(s, a) + \eta_t(\mathcal{T}_t(Q_t)(s, a) - Q_t(s, a))}_{\text{sample transition } (s, a, s')} , \quad t \geq 0$$

# Q-learning: a stochastic approximation algorithm

---



Chris Watkins



Peter Dayan

Stochastic approximation for solving Bellman equation  $\mathcal{T}(Q) - Q = 0$

$$\underbrace{Q_{t+1}(s, a) = Q_t(s, a) + \eta_t (\mathcal{T}_t(Q_t)(s, a) - Q_t(s, a))}_{\text{sample transition } (s, a, s')} , \quad t \geq 0$$

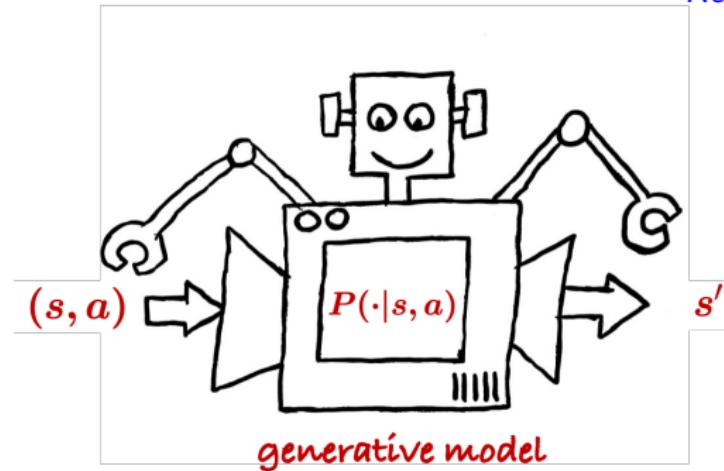
$$\mathcal{T}_t(Q)(s, a) = r(s, a) + \gamma \max_{a'} Q(s', a')$$

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

# A generative model / simulator

---

— Kearns, Singh, 1999



Each iteration, draw an independent sample  $(s, a, s')$  for given  $(s, a)$

# Synchronous Q-learning

---



Chris Watkins



Peter Dayan

**for**  $t = 0, 1, \dots, T$

**for** each  $(s, a) \in \mathcal{S} \times \mathcal{A}$

draw a sample  $(s, a, s')$ , run

$$Q_{t+1}(s, a) = (1 - \eta_t)Q_t(s, a) + \eta_t \left\{ r(s, a) + \gamma \max_{a'} Q_t(s', a') \right\}$$

**synchronous:** all state-action pairs are updated simultaneously

- total sample size:  $TSA$

# Sample complexity of synchronous Q-learning

Theorem (Li, Cai, Chen, Wei, Chi '21, OR'24)

For any  $0 < \varepsilon \leq 1$ , synchronous Q-learning yields  $\|\hat{Q} - Q^*\|_\infty \leq \varepsilon$  with high prob. and  $\mathbb{E}[\|\hat{Q} - Q^*\|_\infty] \leq \varepsilon$ , with sample size **at most**

$$\begin{cases} \tilde{O}\left(\frac{SA}{(1-\gamma)^4\varepsilon^2}\right) & \text{if } A \geq 2 \\ \tilde{O}\left(\frac{S}{(1-\gamma)^3\varepsilon^2}\right) & \text{if } A = 1 \end{cases} \quad (\text{TD learning})$$

# Sample complexity of synchronous Q-learning

Theorem (Li, Cai, Chen, Wei, Chi '21, OR'24)

For any  $0 < \varepsilon \leq 1$ , synchronous Q-learning yields  $\|\hat{Q} - Q^*\|_\infty \leq \varepsilon$  with high prob. and  $\mathbb{E}[\|\hat{Q} - Q^*\|_\infty] \leq \varepsilon$ , with sample size **at most**

$$\begin{cases} \tilde{O}\left(\frac{SA}{(1-\gamma)^4\varepsilon^2}\right) & \text{if } A \geq 2 \\ \tilde{O}\left(\frac{S}{(1-\gamma)^3\varepsilon^2}\right) & \text{if } A = 1 \end{cases} \quad (\text{TD learning})$$

- Covers both *constant* and *rescaled linear* learning rates:

$$\eta_t \equiv \frac{1}{1 + \frac{c_1(1-\gamma)T}{\log^2 T}} \quad \text{or} \quad \eta_t = \frac{1}{1 + \frac{c_2(1-\gamma)t}{\log^2 T}}$$

# Sample complexity of synchronous Q-learning

Theorem (Li, Cai, Chen, Wei, Chi '21, OR'24)

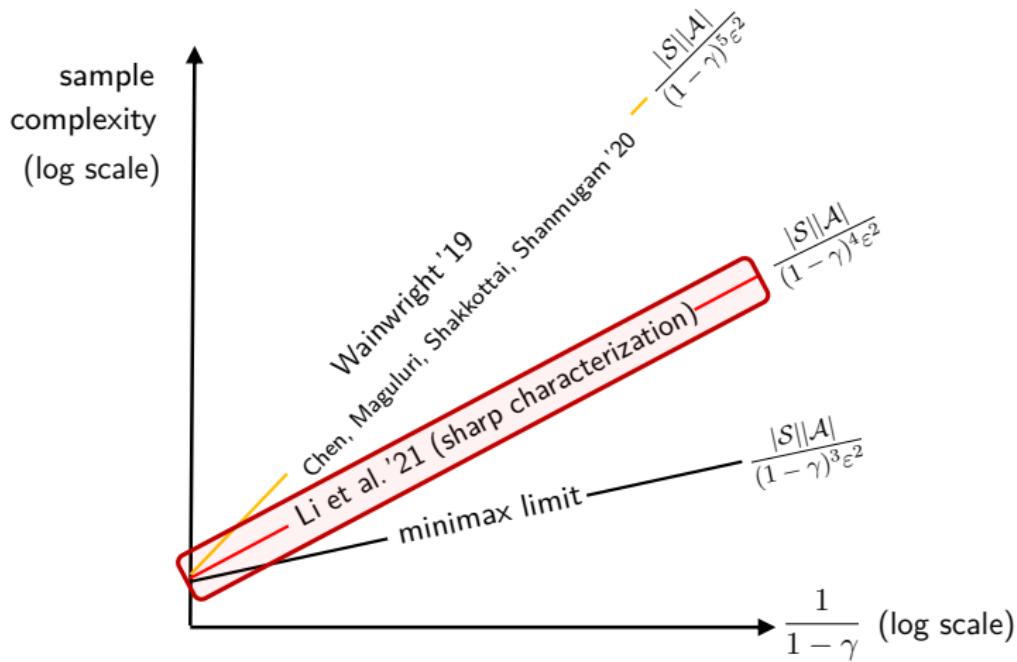
For any  $0 < \varepsilon \leq 1$ , synchronous Q-learning yields  $\|\widehat{Q} - Q^*\|_\infty \leq \varepsilon$  with high prob. and  $\mathbb{E}[\|\widehat{Q} - Q^*\|_\infty] \leq \varepsilon$ , with sample size **at most**

$$\begin{cases} \tilde{O}\left(\frac{SA}{(1-\gamma)^4\varepsilon^2}\right) & \text{if } A \geq 2 \\ \tilde{O}\left(\frac{S}{(1-\gamma)^3\varepsilon^2}\right) & \text{if } A = 1 \end{cases} \quad (?)$$

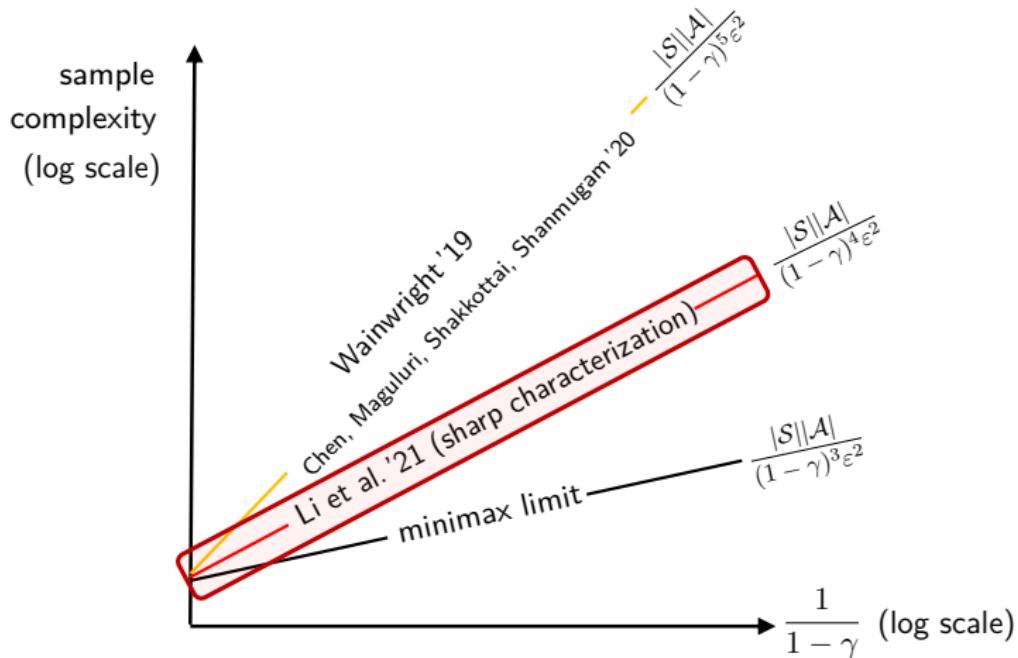
(minimax optimal)

other papers	sample complexity
Even-Dar & Mansour, 2003	$2^{\frac{1}{1-\gamma}} \frac{SA}{(1-\gamma)^4\varepsilon^2}$
Beck, Srikant, 2012	$\frac{S^2A^2}{(1-\gamma)^5\varepsilon^2}$
Wainwright, 2019	$\frac{SA}{(1-\gamma)^5\varepsilon^2}$
Chen, Maguluri, Shakkottai, Shanmugam, 2020	$\frac{SA}{(1-\gamma)^5\varepsilon^2}$

All this requires sample size at least  $\frac{|S||A|}{(1-\gamma)^4 \varepsilon^2} (A \geq 2) \dots$



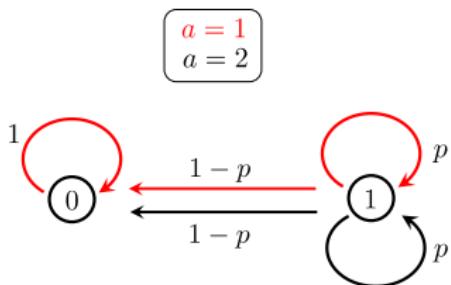
All this requires sample size at least  $\frac{|S||A|}{(1-\gamma)^4 \varepsilon^2} (A \geq 2) \dots$



**Question:** Is Q-learning sub-optimal, or is it an analysis artifact?

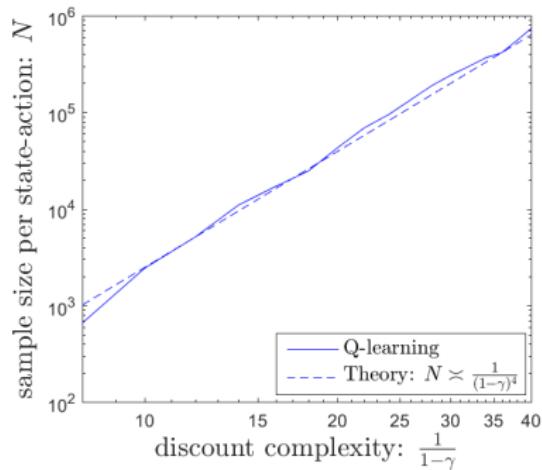
**A numerical example:**  $\frac{SA}{(1-\gamma)^4 \varepsilon^2}$  samples seem necessary . . .

— observed in Wainwright '19



$$p = \frac{4\gamma - 1}{3\gamma}$$

$$r(0, 1) = 0, \quad r(1, 1) = r(1, 2) = 1$$



# Q-learning is NOT minimax optimal

Theorem (Li, Cai, Chen, Wei, Chi '21, OR'24)

For any  $0 < \varepsilon \leq 1$ , there exists an MDP with  $A \geq 2$  such that to achieve  $\|\hat{Q} - Q^*\|_\infty \leq \varepsilon$ , synchronous Q-learning needs *at least*

$$\tilde{\Omega}\left(\frac{SA}{(1-\gamma)^4\varepsilon^2}\right) \text{ samples}$$

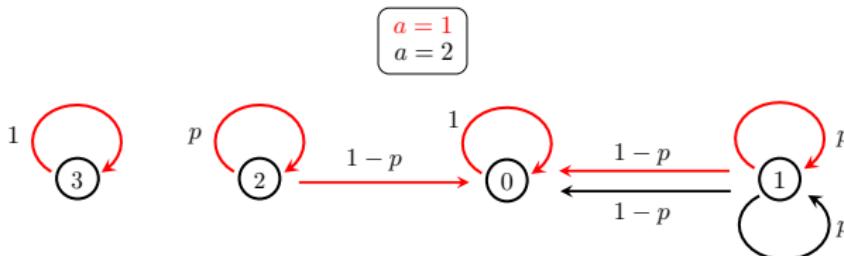
# Q-learning is NOT minimax optimal

Theorem (Li, Cai, Chen, Wei, Chi '21, OR'24)

For any  $0 < \varepsilon \leq 1$ , there exists an MDP with  $A \geq 2$  such that to achieve  $\|\hat{Q} - Q^*\|_\infty \leq \varepsilon$ , synchronous Q-learning needs **at least**

$$\tilde{\Omega}\left(\frac{SA}{(1-\gamma)^4\varepsilon^2}\right) \text{ samples}$$

- Tight **algorithm-dependent** lower bound
- Holds for both constant and rescaled linear learning rates

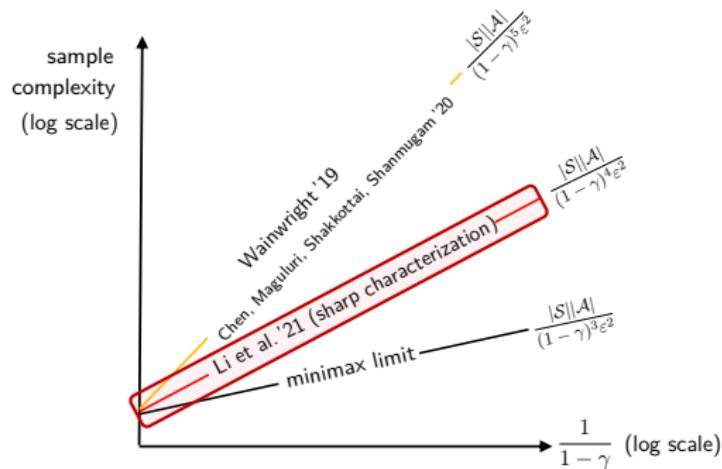


# Q-learning is NOT minimax optimal

Theorem (Li, Cai, Chen, Wei, Chi '21, OR'24)

For any  $0 < \varepsilon \leq 1$ , there exists an MDP with  $A \geq 2$  such that to achieve  $\|\hat{Q} - Q^*\|_\infty \leq \varepsilon$ , synchronous Q-learning needs **at least**

$$\tilde{\Omega}\left(\frac{SA}{(1-\gamma)^4\varepsilon^2}\right) \text{ samples}$$



# Why is Q-learning sub-optimal?

## Over-estimation of Q-functions (Thrun & Schwartz '93; Hasselt '10)

- $\max_{a \in \mathcal{A}} \mathbb{E}[X(a)]$  tends to be over-estimated (high positive bias) when  $\mathbb{E}[X(a)]$  is replaced by its empirical estimates using a small sample size
- often gets worse with a large number of actions (Hasselt, Guez, Silver '15)

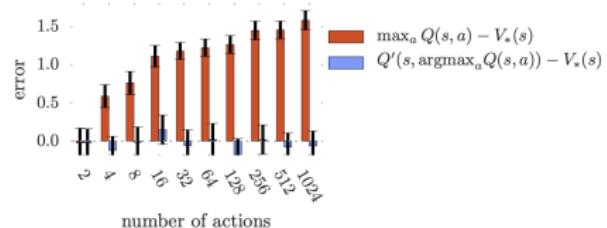


Figure 1: The orange bars show the bias in a single Q-learning update when the action values are  $Q(s, a) = V_*(s) + \epsilon_a$  and the errors  $\{\epsilon_a\}_{a=1}^m$  are independent standard normal random variables. The second set of action values  $Q'$ , used for the blue bars, was generated identically and independently. All bars are the average of 100 repetitions.

# Why is Q-learning sub-optimal?

## Over-estimation of Q-functions (Thrun & Schwartz '93; Hasselt '10)

- $\max_{a \in \mathcal{A}} \mathbb{E}[X(a)]$  tends to be over-estimated (high positive bias) when  $\mathbb{E}[X(a)]$  is replaced by its empirical estimates using a small sample size
- often gets worse with a large number of actions (Hasselt, Guez, Silver '15)

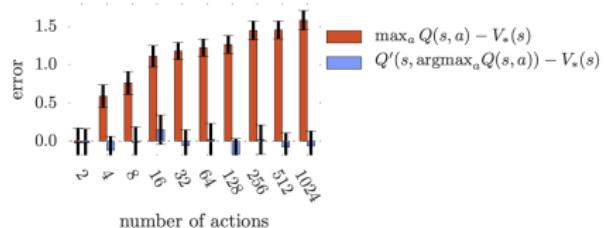


Figure 1: The orange bars show the bias in a single Q-learning update when the action values are  $Q(s, a) = V_*(s) + \epsilon_a$  and the errors  $\{\epsilon_a\}_{a=1}^m$  are independent standard normal random variables. The second set of action values  $Q'$ , used for the blue bars, was generated identically and independently. All bars are the average of 100 repetitions.

## A provable improvement: Q-learning with variance reduction

(Wainwright 2019)

*Improving sample complexity via **variance reduction***

— *a powerful idea from finite-sum stochastic optimization*

## Variance-reduced Q-learning updates ([Wainwright, 2019](#))

— *inspired by SVRG ([Johnson & Zhang, 2013](#))*

$$Q_t(s, a) = (1 - \eta)Q_{t-1}(s, a) + \eta \left( \mathcal{T}_t(Q_{t-1}) \underbrace{- \mathcal{T}_t(\overline{Q}) + \tilde{\mathcal{T}}(\overline{Q})}_{\text{use } \overline{Q} \text{ to help reduce variability}} \right)(s, a)$$

## Variance-reduced Q-learning updates ([Wainwright, 2019](#))

— inspired by SVRG ([Johnson & Zhang, 2013](#))

$$Q_t(s, a) = (1 - \eta)Q_{t-1}(s, a) + \eta \left( \mathcal{T}_t(Q_{t-1}) \underbrace{- \mathcal{T}_t(\bar{Q}) + \tilde{\mathcal{T}}(\bar{Q})}_{\text{use } \bar{Q} \text{ to help reduce variability}} \right)(s, a)$$

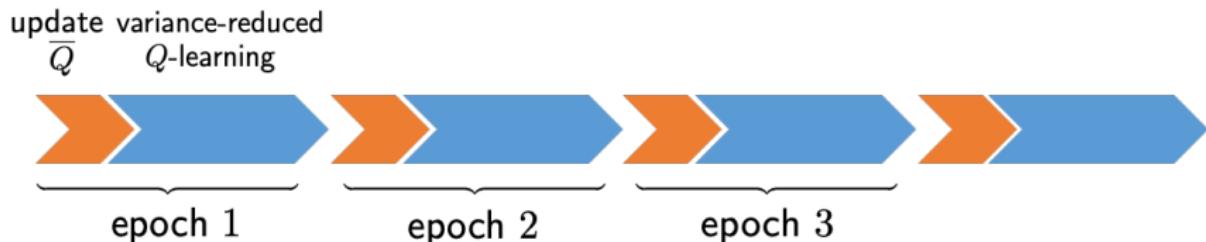
- $\bar{Q}$ : some reference Q-estimate
- $\tilde{\mathcal{T}}$ : empirical Bellman operator (using a batch of samples)

$$\mathcal{T}_t(Q)(s, a) = r(s, a) + \gamma \max_{a'} Q(s', a')$$

$$\tilde{\mathcal{T}}(Q)(s, a) = r(s, a) + \gamma \mathbb{E}_{s' \sim \tilde{\mathbf{P}}(\cdot | s, a)} \left[ \max_{a'} Q(s', a') \right]$$

## An epoch-based stochastic algorithm

— inspired by Johnson & Zhang, 2013



**for** each epoch

1. update  $\bar{Q}$  and  $\tilde{\mathcal{T}}(\bar{Q})$  (which stay fixed in the rest of the epoch)
  2. run variance-reduced Q-learning updates iteratively

# Sample complexity of variance-reduced Q-learning

## Theorem (Wainwright '19)

For any  $0 < \varepsilon \leq 1$ , sample complexity for **variance-reduced synchronous Q-learning** to yield  $\|\hat{Q} - Q^*\|_\infty \leq \varepsilon$  is at most

$$\tilde{O}\left(\frac{SA}{(1-\gamma)^3\varepsilon^2}\right)$$

- allows for more aggressive learning rates

# Sample complexity of variance-reduced Q-learning

## Theorem (Wainwright '19)

For any  $0 < \varepsilon \leq 1$ , sample complexity for **variance-reduced synchronous Q-learning** to yield  $\|\hat{Q} - Q^*\|_\infty \leq \varepsilon$  is at most

$$\tilde{O}\left(\frac{SA}{(1-\gamma)^3\varepsilon^2}\right)$$

- allows for more aggressive learning rates
- minimax-optimal for  $0 < \varepsilon \leq 1$ 
  - remains suboptimal if  $1 < \varepsilon < \frac{1}{1-\gamma}$

## Reference: general RL textbooks I

---

- “*Reinforcement learning: An introduction*,” R. S. Sutton, A. G. Barto, MIT Press, 2018
- “*Reinforcement learning: Theory and algorithms*,” A. Agarwal, N. Jiang, S. Kakade, W. Sun, 2019
- “*Reinforcement learning and optimal control*,” D. Bertsekas, Athena Scientific, 2019
- “*Algorithms for reinforcement learning*,” C. Szepesvari, Springer, 2022
- “*Bandit algorithms*,” T. Lattimore, C. Szepesvari, Cambridge University Press, 2020

# Reference: model-based algorithms I

---

- “*Finite-sample convergence rates for Q-learning and indirect algorithms,*” M. Kearns, S. Satinder, *NeurIPS*, 1998
- “*On the sample complexity of reinforcement learning,*” S. Kakade, 2003
- “*A sparse sampling algorithm for near-optimal planning in large Markov decision processes,*” M. Kearns, Y. Mansour, A. Y. Ng, *Machine learning*, 2002
- “*Minimax PAC bounds on the sample complexity of reinforcement learning with a generative model,*” M. G. Azar, R. Munos, H. J. Kappen, *Machine learning*, 2013
- “*Randomized linear programming solves the Markov decision problem in nearly linear (sometimes sublinear) time,*” *Mathematics of Operations Research*, 2020
- “*Near-optimal time and sample complexities for solving Markov decision processes with a generative model,*” A. Sidford, M. Wang, X. Wu, L. Yang, Y. Ye, *NeurIPS*, 2018

## Reference: model-based algorithms II

---

- “*Variance reduced value iteration and faster algorithms for solving Markov decision processes,*” A. Sidford, M. Wang, X. Wu, Y. Ye, *SODA*, 2018
- “*Model-based reinforcement learning with a generative model is minimax optimal,*” A. Agarwal, S. Kakade, L. Yang, *COLT*, 2020
- “*Instance-dependent  $\ell_\infty$ -bounds for policy evaluation in tabular reinforcement learning,*” A. Pananjady, M. J. Wainwright, *IEEE Trans. on Information Theory*, 2020
- “*Spectral methods for data science: A statistical perspective,*” Y. Chen, Y. Chi, J. Fan, C. Ma, *Foundations and Trends® in Machine Learning*, 2021
- “*Breaking the sample size barrier in model-based reinforcement learning with a generative model,*” G. Li, Y. Wei, Y. Chi, Y. Chen, *Operations Research*, 2024

# Reference: model-free algorithms I

---

- "A stochastic approximation method," H. Robbins, S. Monro, *Annals of Mathematical Statistics*, 1951
- "Robust stochastic approximation approach to stochastic programming," A. Nemirovski, A. Juditsky, G. Lan, A. Shapiro, *SIAM Journal on optimization*, 2009
- "Q-learning," C. Watkins, P. Dayan, *Machine Learning*, 1992
- "Learning rates for Q-learning," E. Even-Dar, Y. Mansour, *Journal of Machine Learning Research*, 2003
- "The asymptotic convergence-rate of Q-learning," C. Szepesvari, *NeurIPS*, 1998
- "Error bounds for constant step-size Q-learning," C. Beck, R. Srikant, *Systems & Control Letters*, 2012
- "Stochastic approximation with cone-contractive operators: Sharp  $\ell_\infty$  bounds for Q-learning," M. Wainwright, 2019

## Reference: model-free algorithms II

---

- “*Is Q-learning minimax optimal? a tight sample complexity analysis,*” G. Li, C. Cai, Y. Chen, Y. Wei, Y. Chi, *Operations Research*, 2024
- “*Variance-reduced Q-learning is minimax optimal,*” M. Wainwright, 2019
- “*Sample-optimal parametric Q-learning using linearly additive features,*” L. Yang, M. Wang, *ICML*, 2019
- “*Asynchronous stochastic approximation and Q-learning,*” J. Tsitsiklis, *Machine learning*, 1994
- “*Finite-time analysis of asynchronous stochastic approximation and Q-learning,*” G. Qu, A. Wierman, *COLT*, 2020
- “*Finite-sample analysis of contractive stochastic approximation using smooth convex envelopes,*” Z. Chen, S. T. Maguluri, S. Shakkottai, K. Shanmugam, *NeurIPS*, 2020
- “*Sample complexity of asynchronous Q-learning: Sharper analysis and variance reduction,*” G. Li, Y. Wei, Y. Chi, Y. Gu, Y. Chen, *IEEE Trans. on Information Theory*, 2022

# **Information-theoretic, statistical and algorithmic foundations of reinforcement learning**



Yuejie Chi  
CMU



Yuxin Chen  
UPenn



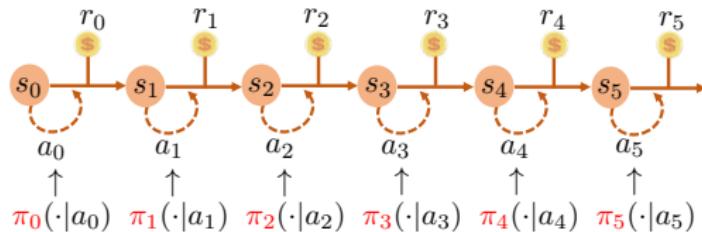
Yuting Wei  
UPenn

Tutorial, ISIT 2024  
Part 2

## **Part 2**

1. **Online RL**
2. Offline RL
3. Multi-agent RL
4. Robust RL

# Online RL: interacting with real environment



## exploration via adaptive policies

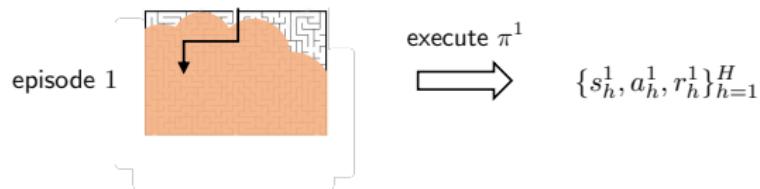
- trial-and-error
- sequential and online
- adaptive learning from data



"Recalculating ... recalculating ..."

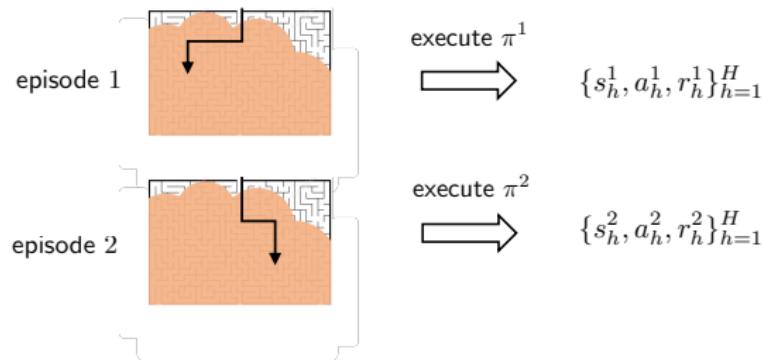
# Online episodic RL

*Sequentially* execute MDP for  $K$  episodes, each consisting of  $H$  steps



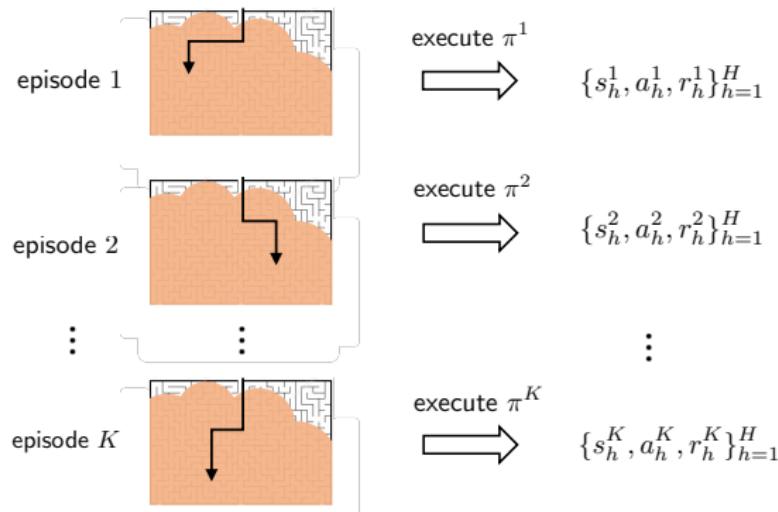
# Online episodic RL

Sequentially execute MDP for  $K$  episodes, each consisting of  $H$  steps



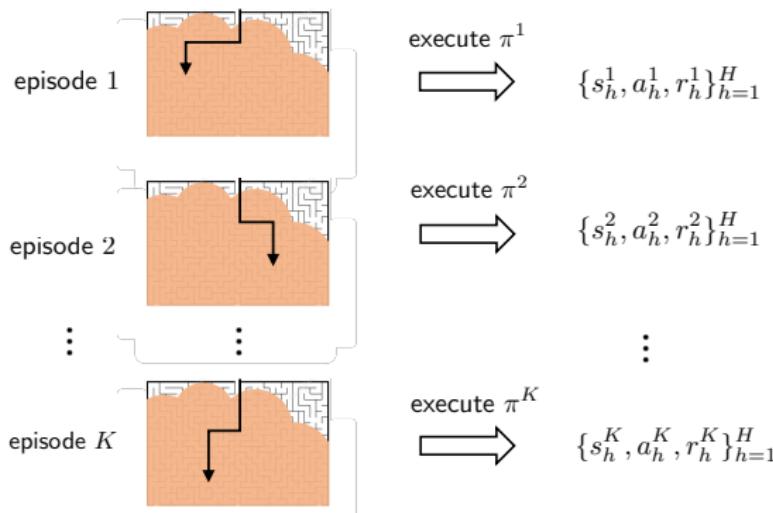
# Online episodic RL

Sequentially execute MDP for  $K$  episodes, each consisting of  $H$  steps



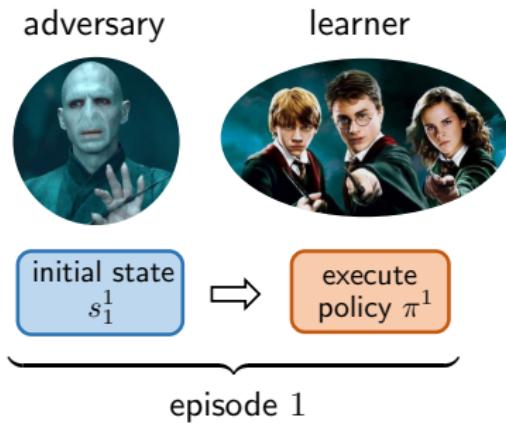
# Online episodic RL

Sequentially execute MDP for  $K$  episodes, each consisting of  $H$  steps  
— sample size:  $T = KH$

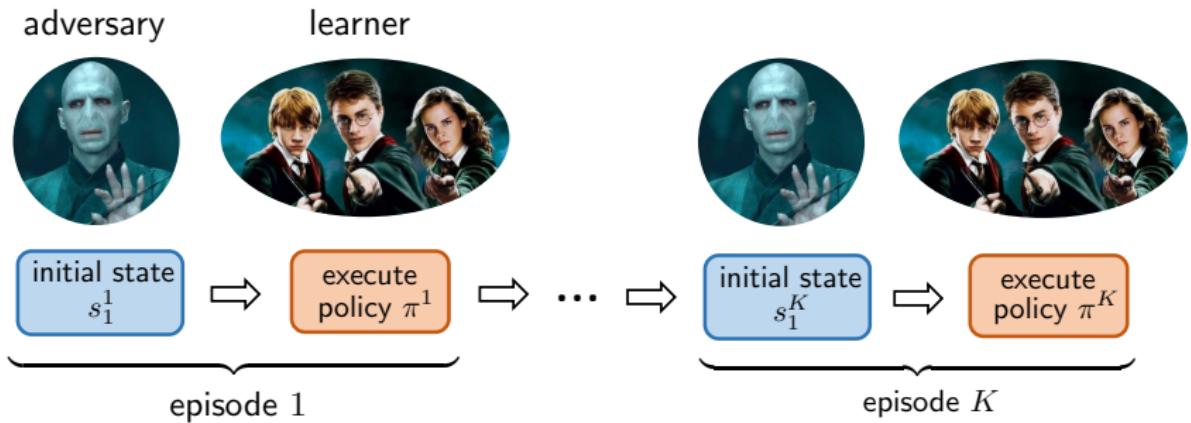


**exploration** (exploring unknowns) vs. **exploitation** (exploiting learned info)

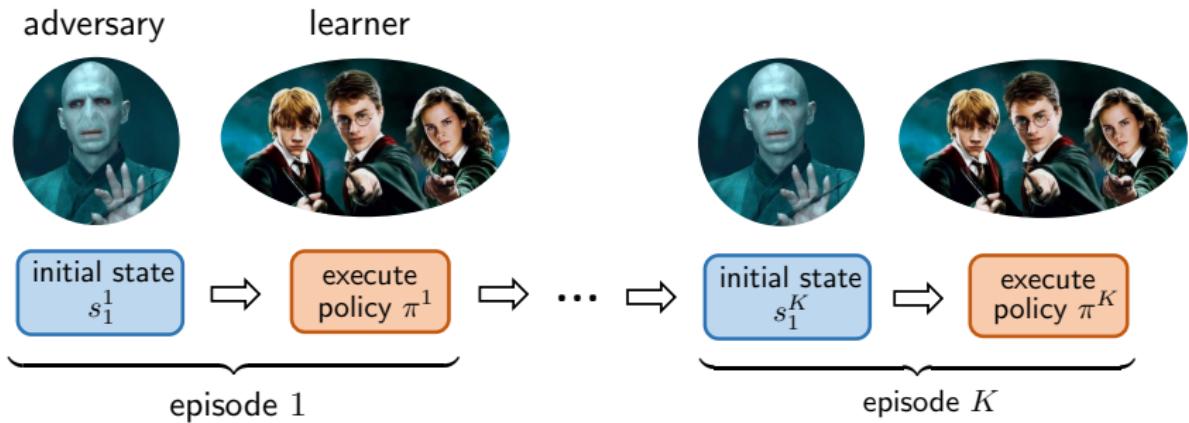
**Regret: gap between learned policy & optimal policy**



# Regret: gap between learned policy & optimal policy



# Regret: gap between learned policy & optimal policy



**Performance metric:** given initial states  $\{s_1^k\}_{k=1}^K$ , define

$$\text{Regret}(T) := \sum_{k=1}^K \left( V_1^\star(s_1^k) - V_1^{\pi^k}(s_1^k) \right)$$

## Existing algorithms

- UCB-VI: [Azar et al, 2017](#)
- UBEV: [Dann et al, 2017](#)
- UCB-Q-Hoeffding: [Jin et al, 2018](#)
- UCB-Q-Bernstein: [Jin et al, 2018](#)
- UCB2-Q-Bernstein: [Bai et al, 2019](#)
- EULER: [Zanette et al, 2019](#)
- UCB-Q-Advantage: [Zhang et al, 2020](#)
- MVP: [Zhang et al, 2020](#)
- UCB-M-Q: [Menard et al, 2021](#)
- Q-EarlySettled-Advantage: [Li et al, 2021](#)
- (modified) MVP: [Zhang et al, 2024](#)

## Lower bound

([Domingues et al, 2021](#))

$$\text{Regret}(T) \gtrsim \sqrt{H^2 SAT}$$

## Existing algorithms

- UCB-VI: [Azar et al, 2017](#)
- UBEV: [Dann et al, 2017](#)
- UCB-Q-Hoeffding: [Jin et al, 2018](#)
- UCB-Q-Bernstein: [Jin et al, 2018](#)
- UCB2-Q-Bernstein: [Bai et al, 2019](#)
- EULER: [Zanette et al, 2019](#)
- UCB-Q-Advantage: [Zhang et al, 2020](#)
- MVP: [Zhang et al, 2020](#)
- UCB-M-Q: [Menard et al, 2021](#)
- Q-EarlySettled-Advantage: [Li et al, 2021](#)
- (modified) MVP: [Zhang et al, 2024](#)

## Lower bound

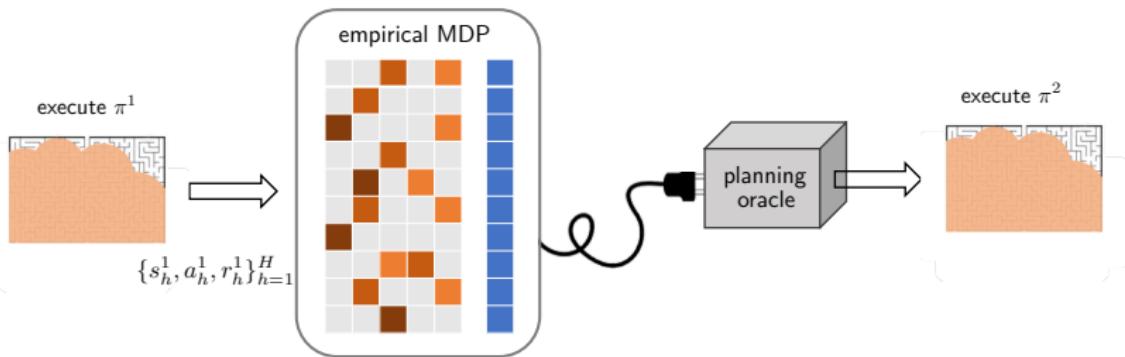
([Domingues et al, 2021](#))

$$\text{Regret}(T) \gtrsim \sqrt{H^2 SAT}$$

Which online RL algorithms achieve near-minimal regret?

*Model-based online RL with UCB exploration*

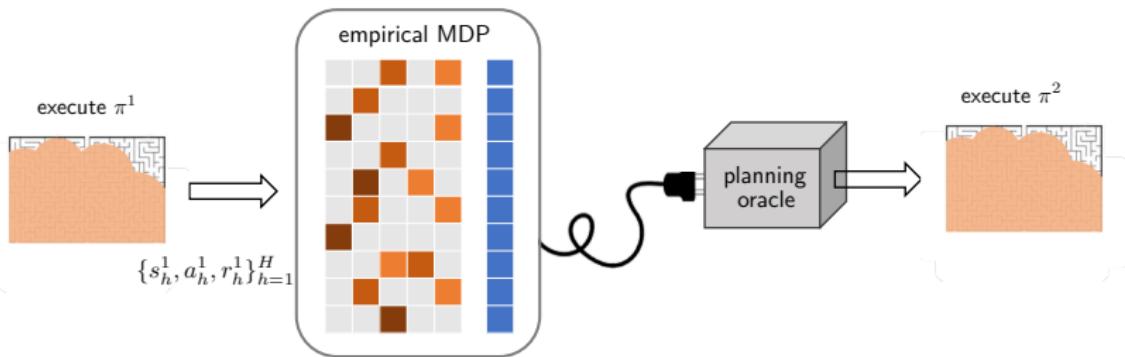
# Model-based approach for online RL



repeat:

- use collected data to estimate transition probabilities
- apply planning to the estimated model to derive a new policy for sampling in the next episode

# Model-based approach for online RL



repeat:

- use collected data to estimate transition probabilities
- apply planning to the estimated model to derive a new policy for sampling in the next episode

How to balance exploration and exploitation in this framework?



T. L. Lai

H. Robbins

## Optimism in the face of uncertainty:

- explores based on the best optimistic estimates associated with the actions!
- a common framework: utilize upper confidence bounds (UCB)  
accounts for estimates + uncertainty level



T. L. Lai

H. Robbins

### **Optimism in the face of uncertainty:**

- explores based on the best optimistic estimates associated with the actions!
- a common framework: utilize upper confidence bounds (UCB)  
accounts for estimates + uncertainty level

**Optimistic model-based approach:** incorporates **UCB** framework into model-based approach

## UCB-VI (Azar et al. '17)

---

For each episode:

1. Backtrack  $h = H, H - 1, \dots, 1$ : run **value iteration**

$$Q_h(s_h, a_h) \leftarrow r_h(s_h, a_h) + \underbrace{\hat{P}_{h, s_h, a_h}}_{\text{model estimate}} V_{h+1}$$

$$V_h(s_h) \leftarrow \max_{a \in \mathcal{A}} Q_h(s_h, a)$$

## UCB-VI (Azar et al. '17)

---

For each episode:

1. Backtrack  $h = H, H - 1, \dots, 1$ : run **optimistic value iteration**

$$Q_h(s_h, a_h) \leftarrow r_h(s_h, a_h) + \underbrace{\hat{P}_{h, s_h, a_h}}_{\text{model estimate}} V_{h+1} + \underbrace{b_h(s_h, a_h)}_{\text{bonus (upper confidence width)}}$$

$$V_h(s_h) \leftarrow \max_{a \in \mathcal{A}} Q_h(s_h, a)$$

## UCB-VI (Azar et al. '17)

---

For each episode:

1. Backtrack  $h = H, H - 1, \dots, 1$ : run **optimistic value iteration**

$$Q_h(s_h, a_h) \leftarrow r_h(s_h, a_h) + \underbrace{\hat{P}_{h, s_h, a_h}}_{\text{model estimate}} V_{h+1} + \underbrace{b_h(s_h, a_h)}_{\text{bonus (upper confidence width)}}$$

$$V_h(s_h) \leftarrow \max_{a \in \mathcal{A}} Q_h(s_h, a)$$

2. Forward  $h = 1, \dots, H$ : take actions according to **greedy policy**

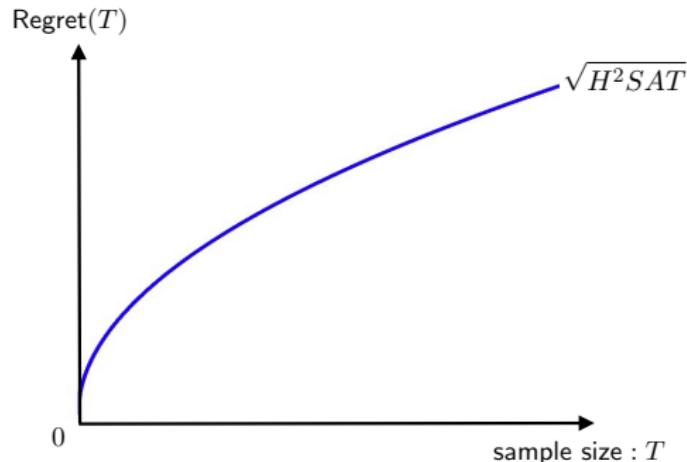
$$\pi_h(s) \leftarrow \operatorname{argmax}_{a \in \mathcal{A}} Q_h(s, a)$$

to sample a new episode  $\{s_h, a_h, r_h\}_{h=1}^H$

# UCB-VI is asymptotically regret-optimal

---

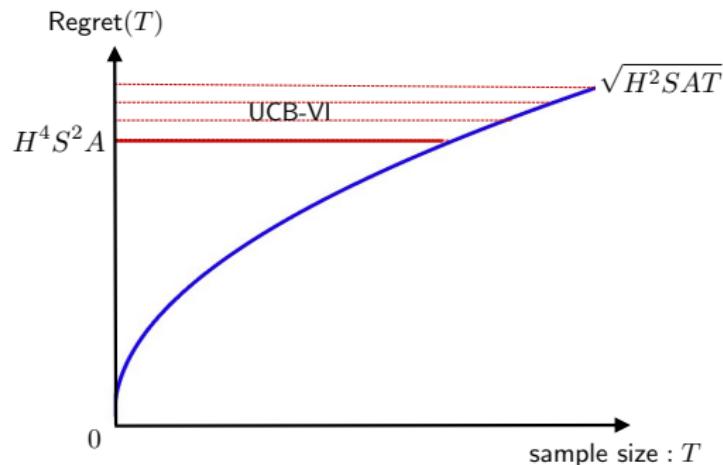
— Azar, Osband, Munos, 2017



# UCB-VI is asymptotically regret-optimal

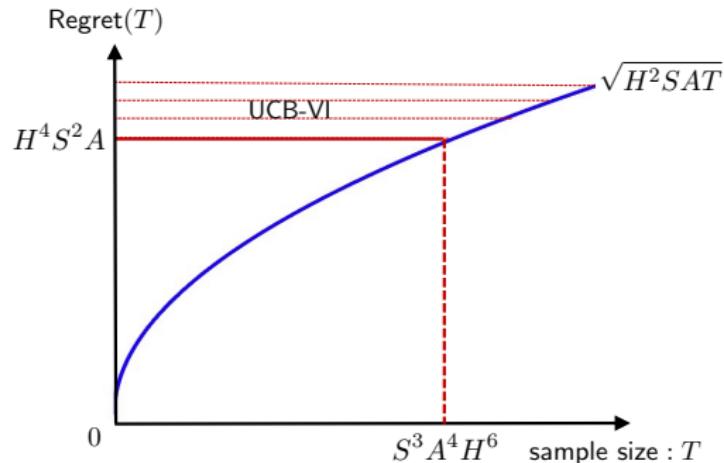
---

— Azar, Osband, Munos, 2017



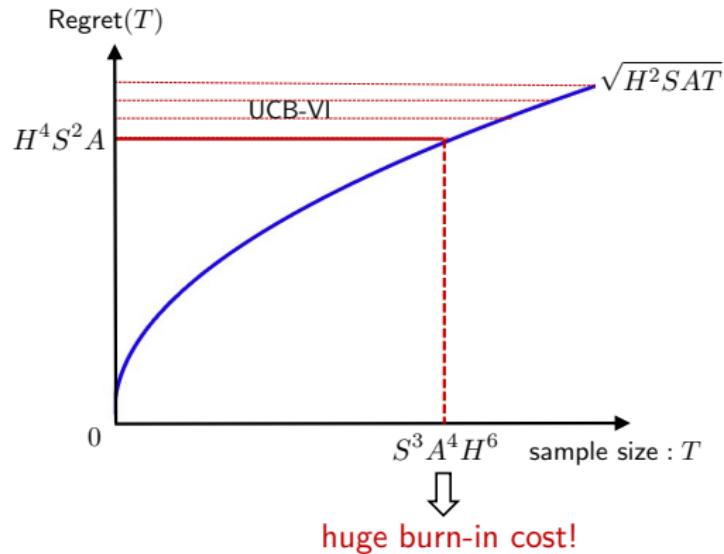
# UCB-VI is asymptotically regret-optimal

— Azar, Osband, Munos, 2017



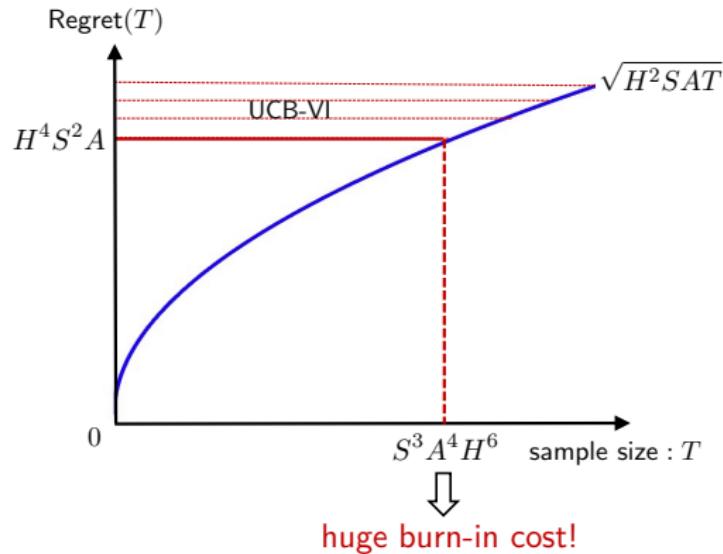
# UCB-VI is asymptotically regret-optimal

— Azar, Osband, Munos, 2017



# UCB-VI is asymptotically regret-optimal

— Azar, Osband, Munos, 2017



**Issues:** large burn-in cost

# Other asymptotically regret-optimal algorithms

---

Algorithm	Regret upper bound	Range of $K$ that attains optimal regret
UCBVI (Azar et al, 2017)	$\sqrt{SAH^2T} + S^2AH^3$	$[S^3AH^3, \infty)$
ORLC (Dann et al, 2019)	$\sqrt{SAH^2T} + S^2AH^4$	$[S^3AH^5, \infty)$
EULER (Zanette et al, 2019)	$\sqrt{SAH^2T} + S^{3/2}AH^3(\sqrt{S} + \sqrt{H})$	$[S^2AH^3(\sqrt{S} + \sqrt{H}), \infty)$
UCB-Adv (Zhang et al, 2020)	$\sqrt{SAH^2T} + S^2A^{3/2}H^{33/4}K^{1/4}$	$[S^6A^4H^{27}, \infty)$
MVP (Zhang et al, 2020)	$\sqrt{SAH^2T} + S^2AH^2$	$[S^3AH, \infty)$
UCB-M-Q (Menard et al, 2021)	$\sqrt{SAH^2T} + SAH^4$	$[SAH^5, \infty)$

# Other asymptotically regret-optimal algorithms

---

Algorithm	Regret upper bound	Range of $K$ that attains optimal regret
UCBVI (Azar et al, 2017)	$\sqrt{SAH^2T} + S^2AH^3$	$[S^3AH^3, \infty)$
ORLC (Dann et al, 2019)	$\sqrt{SAH^2T} + S^2AH^4$	$[S^3AH^5, \infty)$
EULER (Zanette et al, 2019)	$\sqrt{SAH^2T} + S^{3/2}AH^3(\sqrt{S} + \sqrt{H})$	$[S^2AH^3(\sqrt{S} + \sqrt{H}), \infty)$
UCB-Adv (Zhang et al, 2020)	$\sqrt{SAH^2T} + S^2A^{3/2}H^{33/4}K^{1/4}$	$[S^6A^4H^{27}, \infty)$
MVP (Zhang et al, 2020)	$\sqrt{SAH^2T} + S^2AH^2$	$[S^3AH, \infty)$
UCB-M-Q (Menard et al, 2021)	$\sqrt{SAH^2T} + SAH^4$	$[SAH^5, \infty)$

Can we find a regre-optimal algorithm with no burn-in cost?

# Monotonic Value Propagation

---

UCB-VI with **doubling update rules** and **variance-aware bonus**

- $(s, a, h)$  is updated only when visited the  $\{1, 3, 7, 15, \dots\}$ -th time

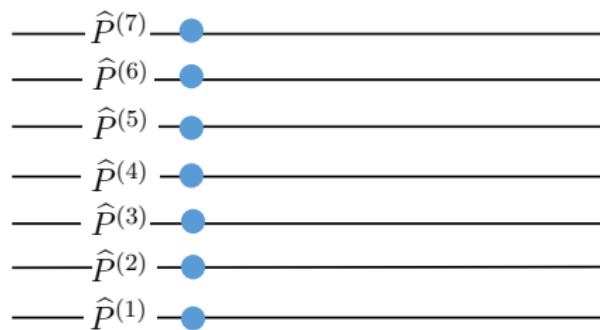
# Monotonic Value Propagation

---

UCB-VI with **doubling update rules** and **variance-aware bonus**

- $(s, a, h)$  is updated only when visited the  $\{1, 3, 7, 15, \dots\}$ -th time

UCB-VI

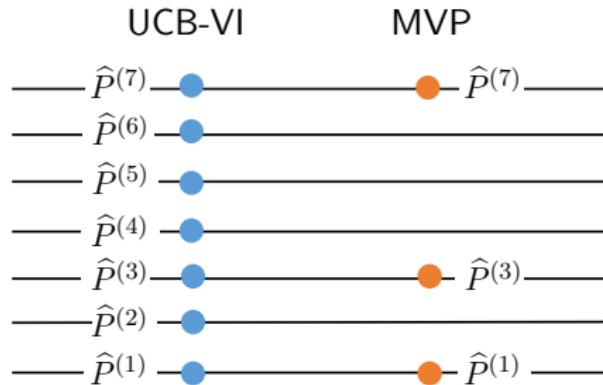


# Monotonic Value Propagation

---

UCB-VI with **doubling update rules** and **variance-aware bonus**

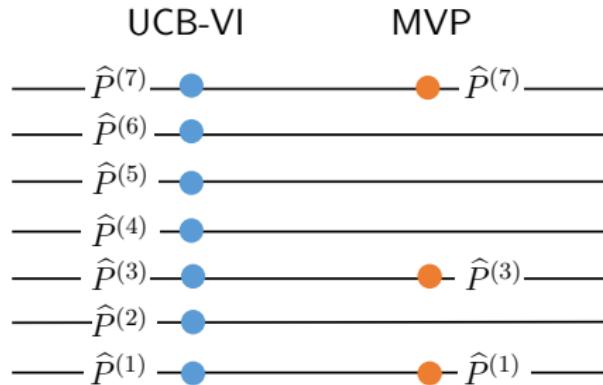
- $(s, a, h)$  is updated only when visited the  $\{1, 3, 7, 15, \dots\}$ -th time



# Monotonic Value Propagation

UCB-VI with **doubling update rules** and **variance-aware bonus**

- $(s, a, h)$  is updated only when visited the  $\{1, 3, 7, 15, \dots\}$ -th time

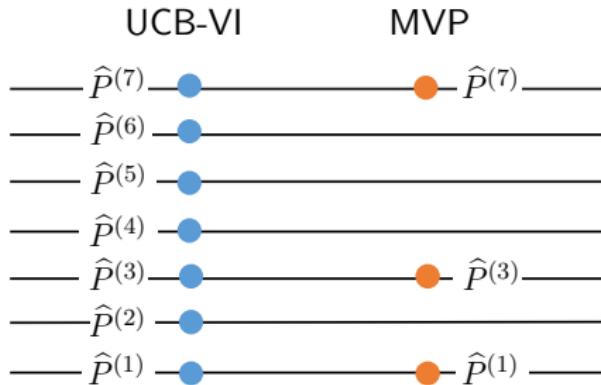


- visitation counts change much less frequently  
→ reduces covering number dramatically

# Monotonic Value Propagation

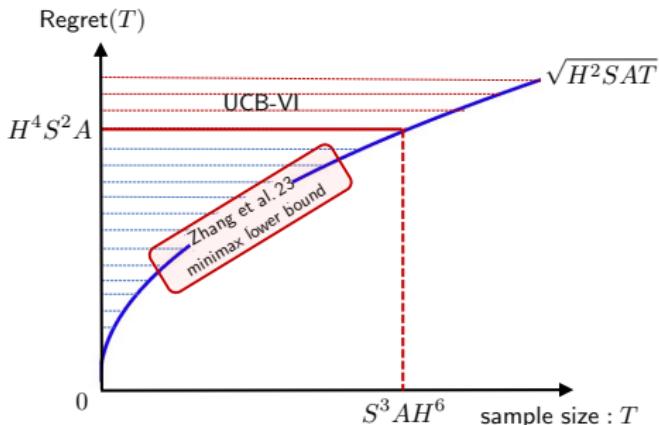
UCB-VI with **doubling update rules** and **variance-aware bonus**

- $(s, a, h)$  is updated only when visited the  $\{1, 3, 7, 15, \dots\}$ -th time



- visitation counts change much less frequently  
→ reduces covering number dramatically
- data-driven bonus terms (chosen based on empirical variances)

# Regret-optimal algorithm w/o burn-in cost

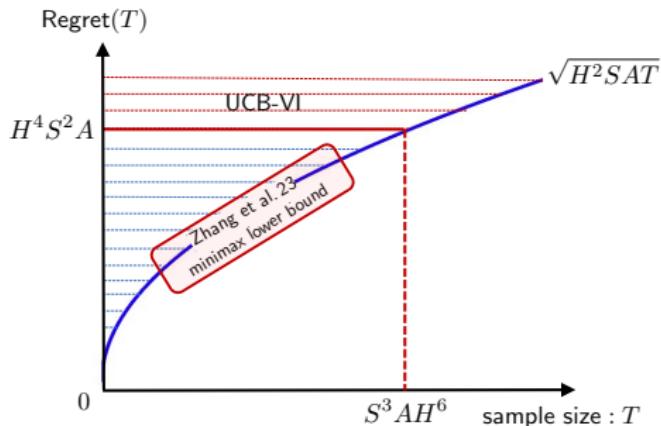


**Theorem (Zhang, Chen, Lee, Du '24)**

*The model-based algorithm Monotonic Value Propagation achieves*

$$\text{Regret}(T) \lesssim \tilde{O}(\sqrt{H^2 SAT})$$

# Regret-optimal algorithm w/o burn-in cost



## Theorem (Zhang, Chen, Lee, Du '24)

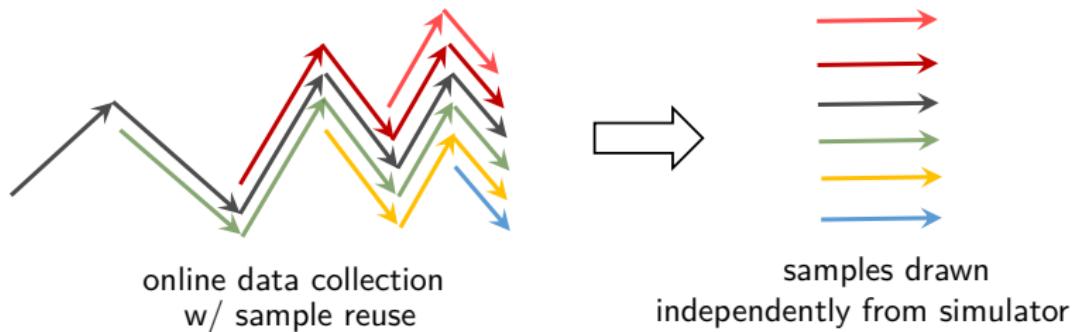
*The model-based algorithm Monotonic Value Propagation achieves*

$$\text{Regret}(T) \lesssim \tilde{O}(\sqrt{H^2 S A T})$$

- the only algorithm so far that is regret-optimal w/o burn-ins

# Key technical innovation

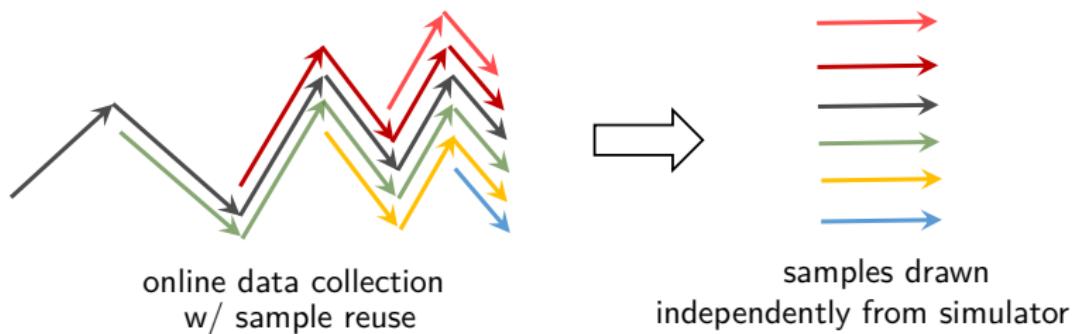
---



Decoupling complicated statistical dependency during online learning

# Key technical innovation

---

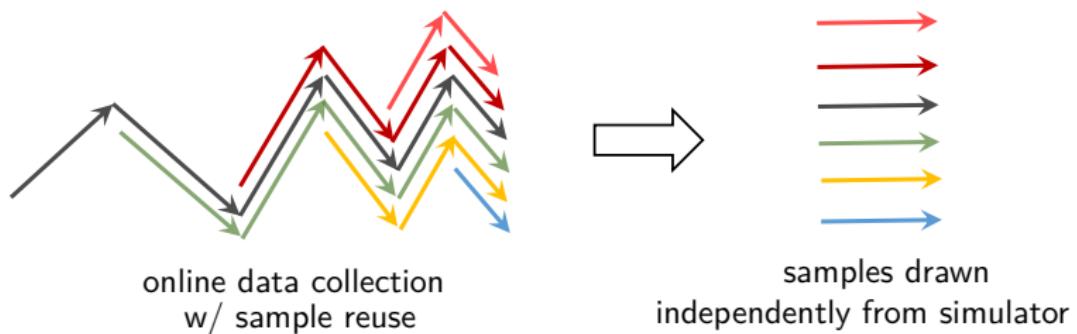


Decoupling complicated statistical dependency during online learning

- couples online data collection with i.i.d. sampling

# Key technical innovation

---



Decoupling complicated statistical dependency during online learning

- couples online data collection with i.i.d. sampling
- exploit *compressibility* of visitation counts
  - w/ the aid of doubling algorithmic trick

# How about memory complexity?

---

Algorithm	Regret upper bound	Range of $K$ that attains optimal regret	Memory complexity
UCBVI (Azar et al, 2017)	$\sqrt{SAH^2T} + S^2AH^3$	$[S^3AH^3, \infty)$	$S^2AH$
UCB-Adv (Zhang et al, 2020)	$\sqrt{SAH^2T} + S^2A^{3/2}H^{33/4}K^{1/4}$	$[S^6A^4H^{27}, \infty)$	$SAH$
MVP (Zhang et al, 2020)	$\sqrt{SAH^2T} + S^2AH^2$	$[S^3AH, \infty)$	$S^2AH$
UCB-M-Q (Menard et al. '21)	$\sqrt{SAH^2T} + SAH^4$	$[SAH^5, \infty)$	$S^2AH$
MVP (Zhang et al, 2024)	$\sqrt{SAH^2T}$	$[1, \infty)$	$S^2AH$

# How about memory complexity?

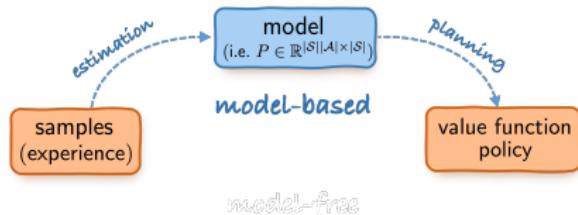
---

Algorithm	Regret upper bound	Range of $K$ that attains optimal regret	Memory complexity
UCBVI (Azar et al, 2017)	$\sqrt{SAH^2T} + S^2AH^3$	$[S^3AH^3, \infty)$	$S^2AH$
UCB-Adv (Zhang et al, 2020)	$\sqrt{SAH^2T} + S^2A^{3/2}H^{33/4}K^{1/4}$	$[S^6A^4H^{27}, \infty)$	$SAH$
MVP (Zhang et al, 2020)	$\sqrt{SAH^2T} + S^2AH^2$	$[S^3AH, \infty)$	$S^2AH$
UCB-M-Q (Menard et al. '21)	$\sqrt{SAH^2T} + SAH^4$	$[SAH^5, \infty)$	$S^2AH$
MVP (Zhang et al, 2024)	$\sqrt{SAH^2T}$	$[1, \infty)$	$S^2AH$

Can we find a regret-optimal algorithm with  
(1) low burn-in cost and (2) low memory complexity?

# Model-free RL is often more memory-efficient

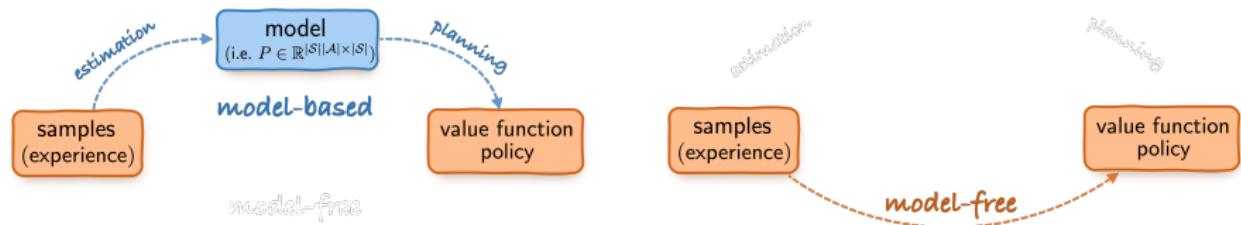
---



*store transition kernel estimates*

$$\rightarrow O(S^2 AH) \text{ memory}$$

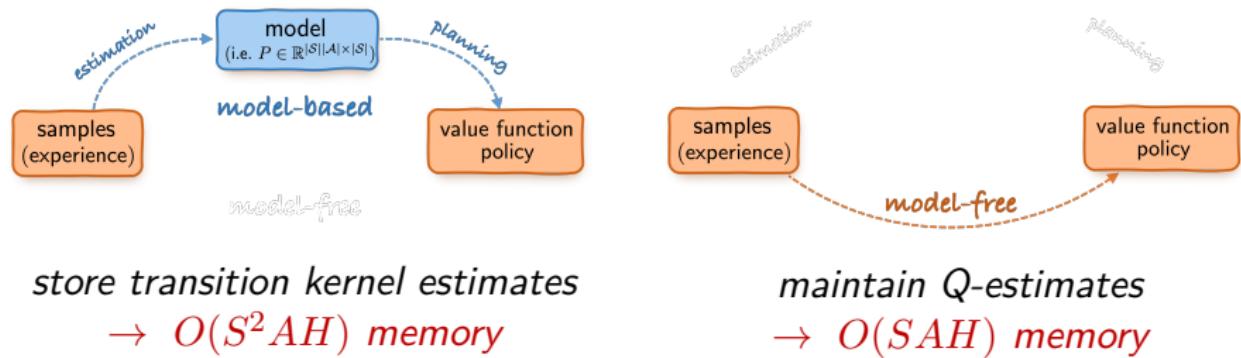
# Model-free RL is often more memory-efficient



*store transition kernel estimates*  
→  $O(S^2AH)$  memory

*maintain Q-estimates*  
→  $O(SAH)$  memory

# Model-free RL is often more memory-efficient

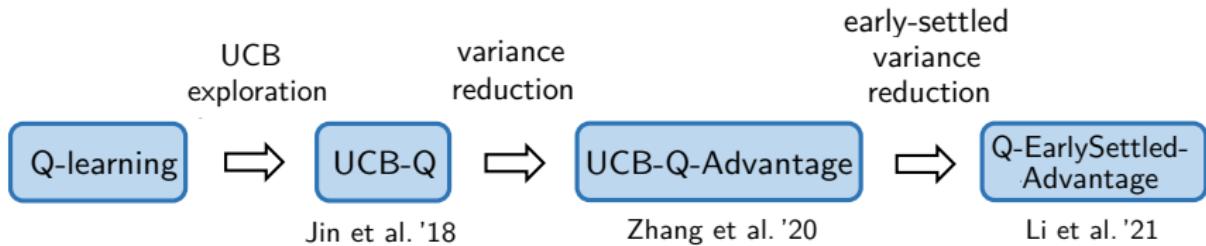


## Definition (Jin et al. '18)

An RL algorithm is **model-free** if its space complexity is  $o(S^2 AH)$

*Which model-free algorithms are sample-efficient for online RL?*

*Which model-free algorithms are sample-efficient for online RL?*



# Q-learning: a classical model-free algorithm

---



Chris Watkins



Peter Dayan

Stochastic approximation for solving Bellman equation

$$Q_h(s_h, a_h) \leftarrow (1 - \eta_k)Q_h(s_h, a_h) + \eta_k \mathcal{T}_k(Q_{h+1})(s_h, a_h)$$

# Q-learning: a classical model-free algorithm

---



Chris Watkins



Peter Dayan

Stochastic approximation for solving Bellman equation

$$Q_h(s_h, a_h) \leftarrow (1 - \eta_k)Q_h(s_h, a_h) + \eta_k \mathcal{T}_k(Q_{h+1})(s_h, a_h)$$

$$\mathcal{T}_k(Q_h)(s_h, a_h) = r(s_h, a_h) + \max_{a'} Q(s_{h+1}, a')$$

*using sample in k-th episode*

## Q-learning with UCB exploration (Jin et al., 2018)

---

$$Q_h(s_h, a_h) \leftarrow \underbrace{(1 - \eta_k)Q_h(s_h, a_h) + \eta_k \mathcal{T}_k(Q_{h+1})(s_h, a_h)}_{\text{classical Q-learning}} + \eta_k \underbrace{b_h(s_h, a_h)}_{\text{exploration bonus}}$$

# Q-learning with UCB exploration (Jin et al., 2018)

---

$$Q_h(s_h, a_h) \leftarrow \underbrace{(1 - \eta_k)Q_h(s_h, a_h) + \eta_k \mathcal{T}_k(Q_{h+1})(s_h, a_h)}_{\text{classical Q-learning}} + \eta_k \underbrace{b_h(s_h, a_h)}_{\text{exploration bonus}}$$

- $b_h(s, a)$ : upper confidence bound
  - *optimism in the face of uncertainty*
- inspired by UCB bandit algorithm (Lai, Robbins '85)

## Q-learning with UCB exploration (Jin et al., 2018)

$$Q_h(s_h, a_h) \leftarrow \underbrace{(1 - \eta_k)Q_h(s_h, a_h) + \eta_k \mathcal{T}_k(Q_{h+1})(s_h, a_h)}_{\text{classical Q-learning}} + \eta_k \underbrace{b_h(s_h, a_h)}_{\text{exploration bonus}}$$

- $b_h(s, a)$ : upper confidence bound
  - *optimism in the face of uncertainty*
- inspired by UCB bandit algorithm (Lai, Robbins '85)

Regret( $T$ )  $\lesssim \sqrt{H^3 SAT}$   $\implies$  sub-optimal by a factor of  $\sqrt{H}$

## Q-learning with UCB exploration (Jin et al., 2018)

$$Q_h(s_h, a_h) \leftarrow \underbrace{(1 - \eta_k)Q_h(s_h, a_h) + \eta_k \mathcal{T}_k(Q_{h+1})(s_h, a_h)}_{\text{classical Q-learning}} + \eta_k \underbrace{b_h(s_h, a_h)}_{\text{exploration bonus}}$$

- $b_h(s, a)$ : upper confidence bound
  - *optimism in the face of uncertainty*
- inspired by UCB bandit algorithm (Lai, Robbins '85)

Regret( $T$ )  $\lesssim \sqrt{H^3 SAT}$   $\implies$  sub-optimal by a factor of  $\sqrt{H}$

**Issue:** large variability in stochastic update rules

# **Q-learning with UCB and variance reduction**

---

— *Zhang et al. '20*

Incorporates **variance reduction** into UCB-Q:

# **Q-learning with UCB and variance reduction**

---

— *Zhang et al. '20*

Incorporates **reference-advantage decomposition** into UCB-Q:

# Q-learning with UCB and variance reduction

---

— Zhang et al. '20

Incorporates **reference-advantage decomposition** into UCB-Q:

$$\begin{aligned} Q_h(s_h, a_h) &\leftarrow (1 - \eta_k)Q_h(s_h, a_h) + \eta_k \underbrace{b_h(s_h, a_h)}_{\text{UCB bonus}} \\ &+ \eta_k \left( \underbrace{\mathcal{T}_k(Q_{h+1}) - \mathcal{T}_k(\bar{Q}_{h+1})}_{\text{advantage}} + \underbrace{\hat{\mathcal{T}}(\bar{Q}_{h+1})}_{\text{reference}} \right)(s_h, a_h) \end{aligned}$$

- Reference  $\bar{Q}_{h+1}$ , batch estimate  $\hat{\mathcal{T}}$ : help reduce variability

# Q-learning with UCB and variance reduction

---

— Zhang et al. '20

Incorporates **reference-advantage decomposition** into UCB-Q:

$$\begin{aligned} Q_h(s_h, a_h) &\leftarrow (1 - \eta_k)Q_h(s_h, a_h) + \eta_k \underbrace{b_h(s_h, a_h)}_{\text{UCB bonus}} \\ &+ \eta_k \left( \underbrace{\mathcal{T}_k(Q_{h+1}) - \mathcal{T}_k(\bar{Q}_{h+1})}_{\text{advantage}} + \underbrace{\hat{\mathcal{T}}(\bar{Q}_{h+1})}_{\text{reference}} \right)(s_h, a_h) \end{aligned}$$

- Reference  $\bar{Q}_{h+1}$ , batch estimate  $\hat{\mathcal{T}}$ : help reduce variability

UCB-Q-Advantage is asymptotically regret-optimal

# Q-learning with UCB and variance reduction

---

— Zhang et al. '20

Incorporates **reference-advantage decomposition** into UCB-Q:

$$\begin{aligned} Q_h(s_h, a_h) &\leftarrow (1 - \eta_k)Q_h(s_h, a_h) + \eta_k \underbrace{b_h(s_h, a_h)}_{\text{UCB bonus}} \\ &+ \eta_k \left( \underbrace{\mathcal{T}_k(Q_{h+1}) - \mathcal{T}_k(\bar{Q}_{h+1})}_{\text{advantage}} + \underbrace{\hat{\mathcal{T}}(\bar{Q}_{h+1})}_{\text{reference}} \right)(s_h, a_h) \end{aligned}$$

- Reference  $\bar{Q}_{h+1}$ , batch estimate  $\hat{\mathcal{T}}$ : help reduce variability

UCB-Q-Advantage is asymptotically regret-optimal

**Issue:** high burn-in cost  $O(S^6 A^4 H^{28})$

## Diagnosis of UCB-Q-Advantage

---

Variance reduction requires sufficiently good references  $\overline{Q}_h$

# Diagnosis of UCB-Q-Advantage

---

Variance reduction requires sufficiently good references  $\bar{Q}_h$



# Diagnosis of UCB-Q-Advantage

---

Variance reduction requires sufficiently good references  $\bar{Q}_h$



Updating references  $\bar{Q}_h$  and  $\bar{V}_h$  many times



# Diagnosis of UCB-Q-Advantage

---

Variance reduction requires sufficiently good references  $\bar{Q}_h$



Updating references  $\bar{Q}_h$  and  $\bar{V}_h$  many times



Large burn-in cost

# Diagnosis of UCB-Q-Advantage

---

Variance reduction requires sufficiently good references  $\bar{Q}_h$



Updating references  $\bar{Q}_h$  and  $\bar{V}_h$  many times



Large burn-in cost

**Key idea:** early settlement of the reference as soon as it reaches a reasonable quality (e.g.,  $\bar{V}_h \leq V_h^* + 1$ )

## Our algorithm: Q-EarlySettled-Advantage

---

**Theorem (Li, Shi, Chen, Gu, Chi '21)**

*With high prob., Q-EarlySettled-Advantage achieves (up to log factor)*

$$\text{Regret}(T) \lesssim \sqrt{H^2 SAT} + H^6 SA$$

*with a memory complexity of  $O(SAH)$*

## Our algorithm: Q-EarlySettled-Advantage

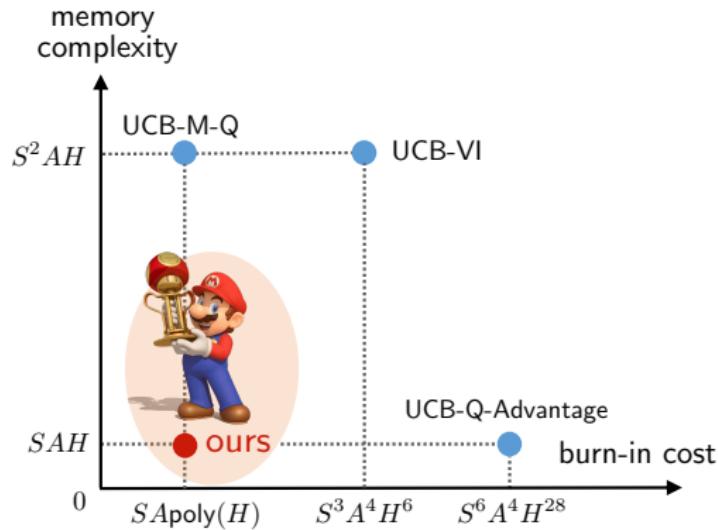
### Theorem (Li, Shi, Chen, Gu, Chi '21)

With high prob., Q-EarlySettled-Advantage achieves (up to log factor)

$$\text{Regret}(T) \lesssim \sqrt{H^2 SAT} + H^6 SA$$

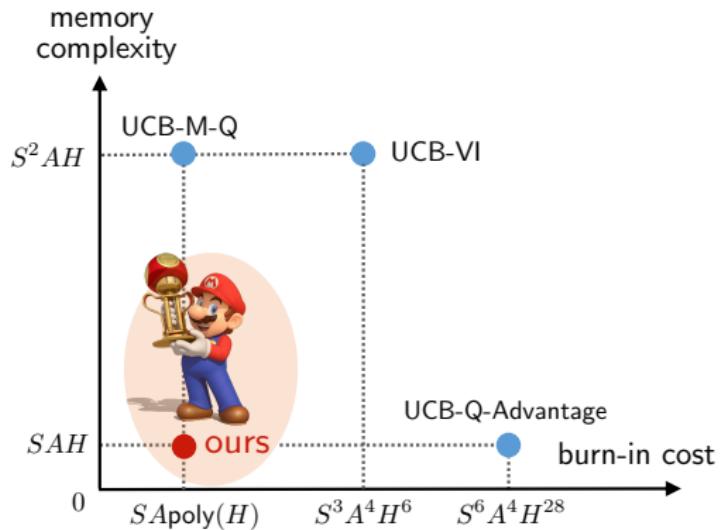
with a memory complexity of  $O(SAH)$

- regret-optimal with burn-in cost  $O(SA\text{poly}(H))$ 
  - optimal in  $SA$ , suboptimal in  $H$
- memory-efficient  $O(SAH)$
- computationally efficient: runtime  $O(T)$



Model-free algorithms can simultaneously achieve

- (1) regret optimality; (2) **low** burn-in cost; (3) memory efficiency



Model-free algorithms can simultaneously achieve

- (1) regret optimality; (2) **low** burn-in cost; (3) memory efficiency

## **Part 2**

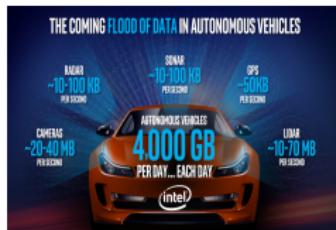
1. Online RL
2. Offline RL
3. Multi-agent RL
4. Robust RL

# Offline/batch RL

- Collecting new data might be costly, unsafe, unethical, or time-consuming



medical records



data of self-driving



clicking times of ads

# Offline/batch RL

- Collecting new data might be costly, unsafe, unethical, or time-consuming
- But we have already stored tons of historical data



medical records



data of self-driving



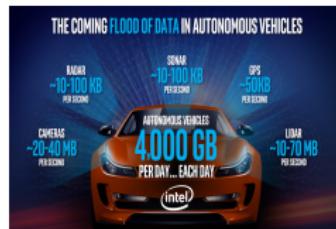
clicking times of ads

# Offline/batch RL

- Collecting new data might be costly, unsafe, unethical, or time-consuming
- But we have already stored tons of historical data



medical records



data of self-driving

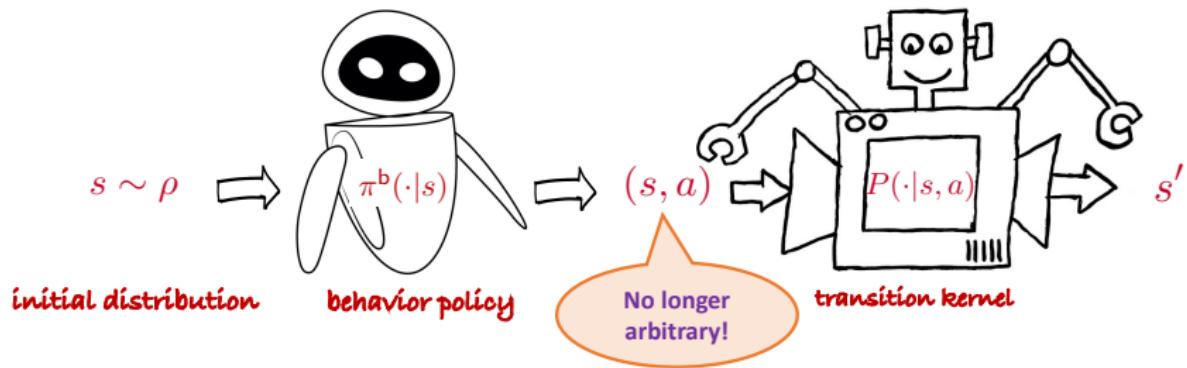


clicking times of ads

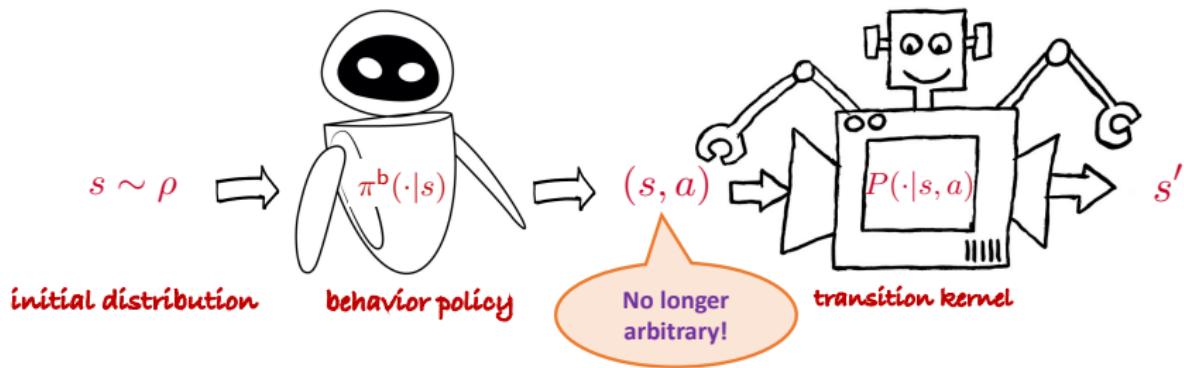
**Question:** can we learn based solely on historical data w/o active exploration?

# A mathematical model of offline data

---



# A mathematical model of offline data

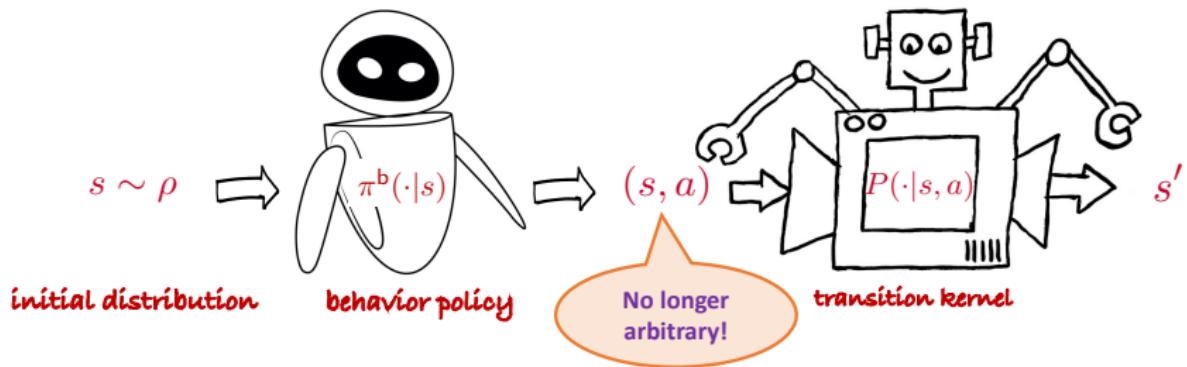


**historical dataset**  $\mathcal{D} = \{(s^{(i)}, a^{(i)}, s'^{(i)})\}$ :  $N$  independent copies of

$$s \sim \rho, \quad a \sim \pi^b(\cdot | s), \quad s' \sim P(\cdot | s, a)$$

- $\rho$ : initial state distribution;  $\pi^b$ : behavior policy

# A mathematical model of offline data



**Goal:** given a target accuracy level  $\varepsilon \in (0, H]$ , find  $\hat{\pi}$  s.t.

$$V^*(\rho) - V^{\hat{\pi}}(\rho) := \mathbb{E}_{s \sim \rho} [V^*(s)] - \mathbb{E}_{s \sim \rho} [V^{\hat{\pi}}(s)] \leq \varepsilon$$

— *in a sample-efficient manner*

# Challenges of offline RL

---

- **Distribution shift:**

distribution( $\mathcal{D}$ )  $\neq$  target distribution under optimal  $\pi^*$

# Challenges of offline RL

---

- **Distribution shift:**

$\text{distribution}(\mathcal{D}) \neq \text{target distribution under optimal } \pi^*$

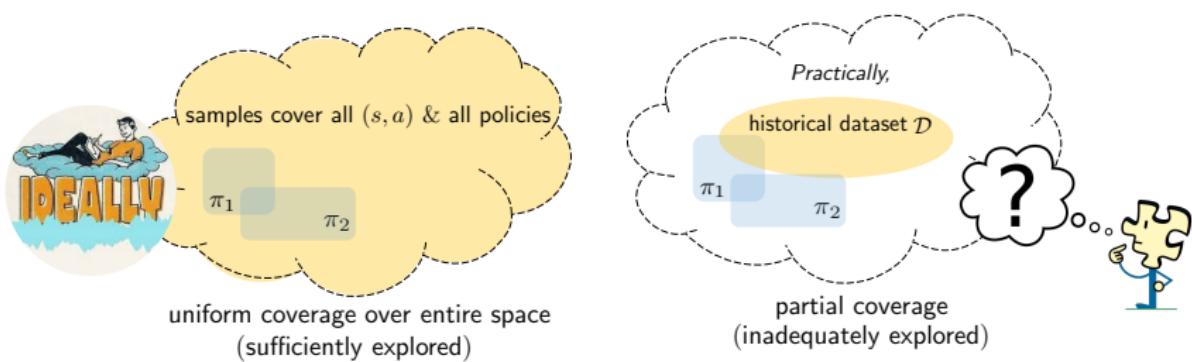


# Challenges of offline RL

- **Distribution shift:**

$\text{distribution}(\mathcal{D}) \neq \text{target distribution under optimal } \pi^*$

- **Partial coverage of state-action space:**



*How to quantify quality of historical dataset  $\mathcal{D}$  (induced by  $\pi^b$ )?*

How to quantify quality of historical dataset  $\mathcal{D}$  (induced by  $\pi^b$ )?

### Single-policy concentrability coefficient (Rashidinejad et al. '21)

$$C^* := \max_{s,a} \frac{d^{\pi^*}(s,a)}{d^{\pi^b}(s,a)} = \left\| \frac{\text{occupancy distribution of } \pi^*}{\text{occupancy distribution of } \pi^b} \right\|_\infty \geq 1$$

How to quantify quality of historical dataset  $\mathcal{D}$  (induced by  $\pi^b$ )?

### Single-policy concentrability coefficient (Rashidinejad et al. '21)

$$C^* := \max_{s,a} \frac{d^{\pi^*}(s,a)}{d^{\pi^b}(s,a)} = \left\| \frac{\text{occupancy distribution of } \pi^*}{\text{occupancy distribution of } \pi^b} \right\|_\infty \geq 1$$

- captures distributional shift

How to quantify quality of historical dataset  $\mathcal{D}$  (induced by  $\pi^b$ )?

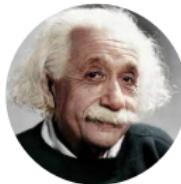
## Single-policy concentrability coefficient (Rashidinejad et al. '21)

$$C^* := \max_{s,a} \frac{d^{\pi^*}(s,a)}{d^{\pi^b}(s,a)} = \left\| \frac{\text{occupancy distribution of } \pi^*}{\text{occupancy distribution of } \pi^b} \right\|_\infty \geq 1$$

- captures distributional shift

$$C^* = O(1)$$

large  $C^*$



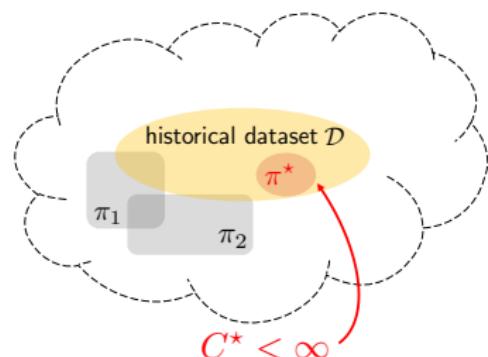
expert data

How to quantify quality of historical dataset  $\mathcal{D}$  (induced by  $\pi^b$ )?

## Single-policy concentrability coefficient (Rashidinejad et al. '21)

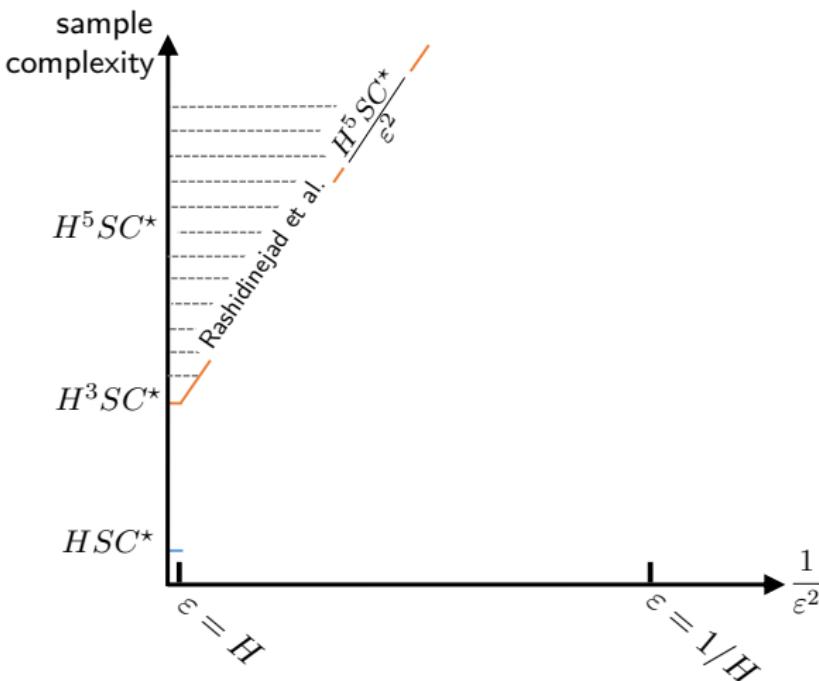
$$C^* := \max_{s,a} \frac{d^{\pi^*}(s,a)}{d^{\pi^b}(s,a)} = \left\| \frac{\text{occupancy distribution of } \pi^*}{\text{occupancy distribution of } \pi^b} \right\|_\infty \geq 1$$

- captures distributional shift
- allows for partial coverage
  - as long as it covers the part reachable by  $\pi^*$



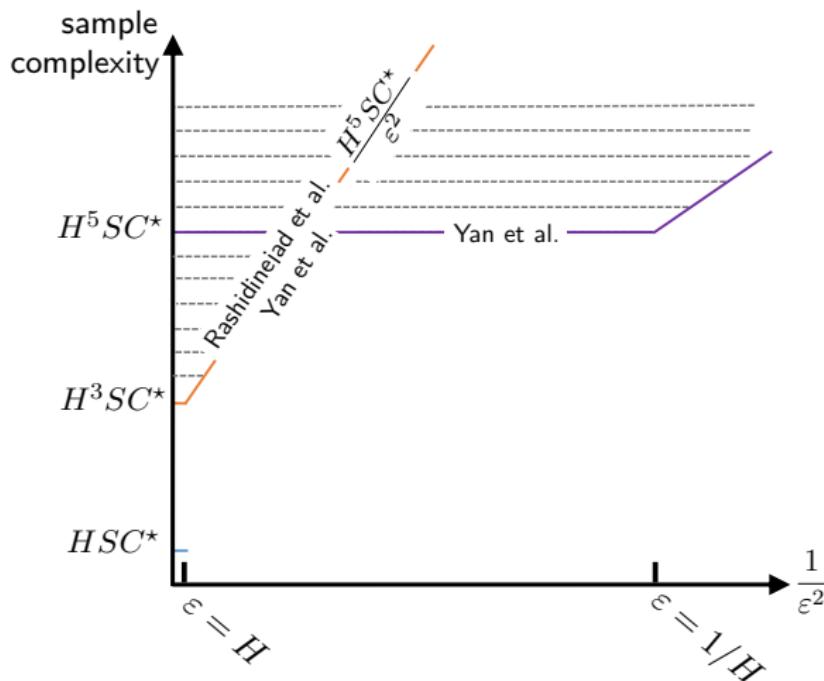
## Prior art: sample complexity bounds

---

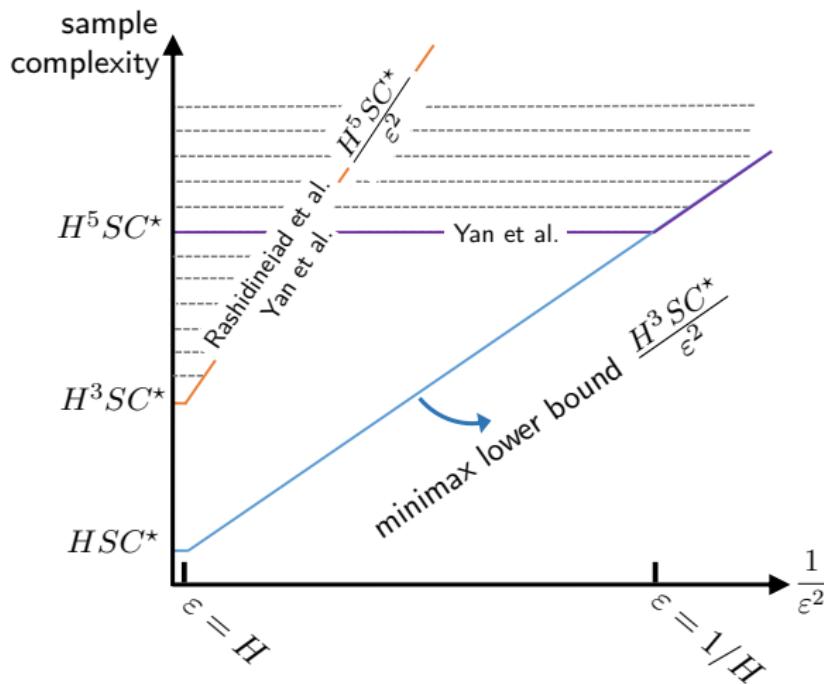


## Prior art: sample complexity bounds

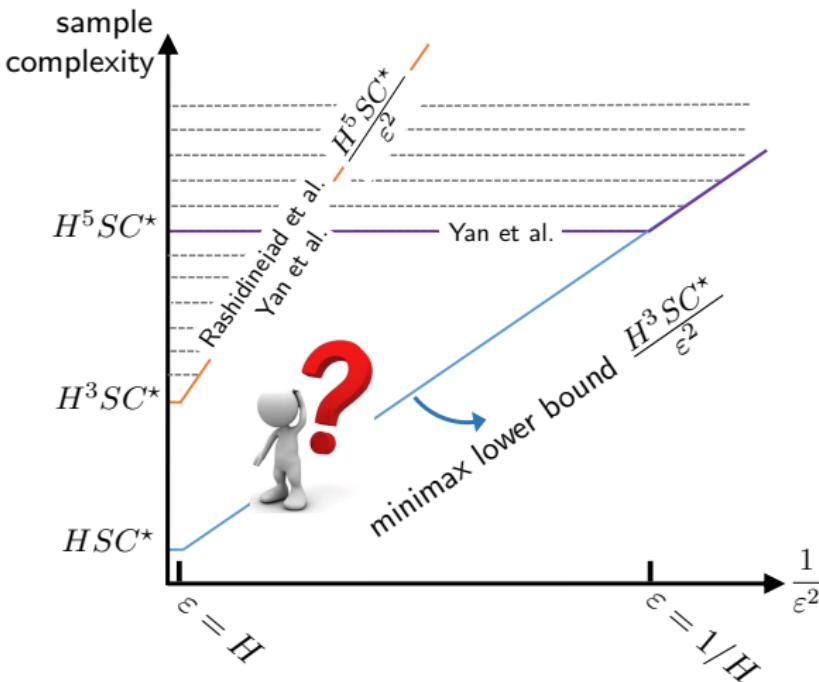
---



## Prior art: sample complexity bounds



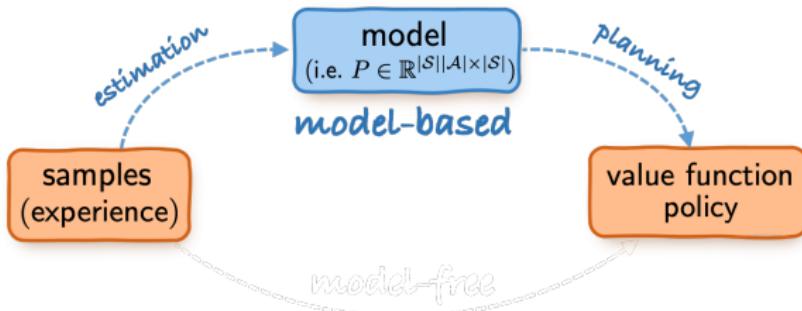
# Prior art: sample complexity bounds



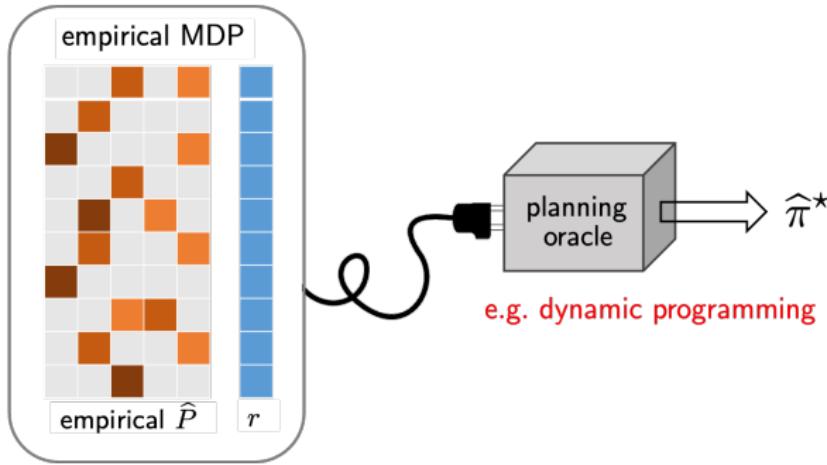
Can we close the gap between upper & lower bounds?

# Model-based (“plug-in”) approach?

---



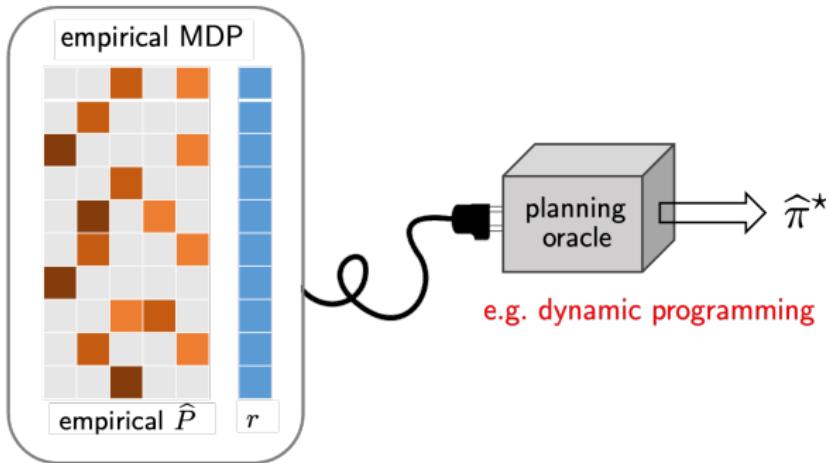
# Model-based (“plug-in”) approach?



1. construct empirical model  $\hat{P}$  :

$$\hat{P}(s' | s, a) = \underbrace{\frac{1}{N} \sum_{i=1}^N \mathbf{1}\{s'^{(i)} = s'\}}_{\text{empirical frequency}}$$

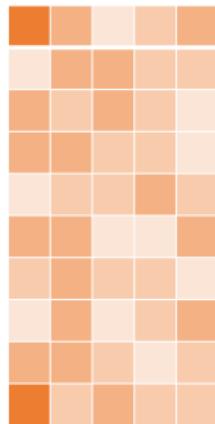
# Model-based (“plug-in”) approach?



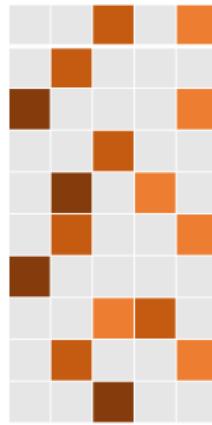
1. construct empirical model  $\hat{P}$
2. planning (e.g. value iteration) based on empirical MDP

## Issues & challenges in the sample-starved regime

---



truth:  $P \in \mathbb{R}^{SA \times S}$

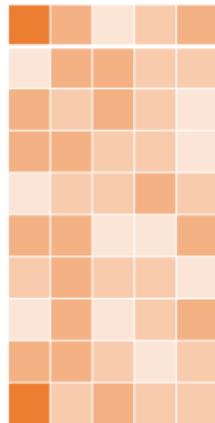


empirical  $\hat{P}$  (simulator)

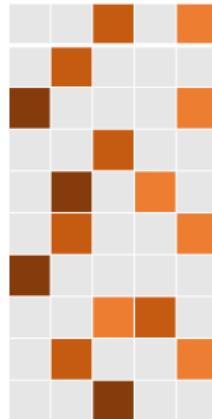
- can't recover  $P$  faithfully if sample size  $\ll S^2 A$

# Issues & challenges in the sample-starved regime

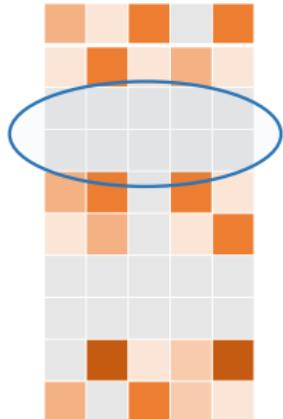
---



truth:  $P \in \mathbb{R}^{SA \times S}$



empirical  $\hat{P}$  (simulator)



empirical  $\hat{P}$  (offline)

- can't recover  $P$  faithfully if sample size  $\ll S^2 A$
- (possibly) insufficient coverage under offline data

# Key idea: pessimism in the face of uncertainty

---

— *Jin et al, 2020, Rashidinejad et al, 2021, Xie et al, 2021*



online

## upper confidence bounds

— promote exploration of under-explored  $(s, a)$

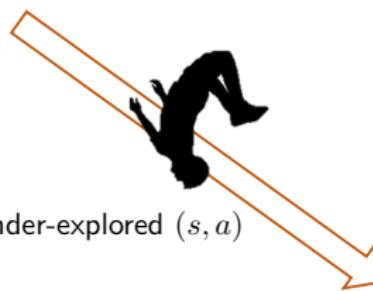
# Key idea: pessimism in the face of uncertainty

---

— Jin et al, 2020, Rashidinejad et al, 2021, Xie et al, 2021



online



**upper confidence bounds**

— promote exploration of under-explored  $(s, a)$



offline

**lower confidence bounds**

— stay cautious about under-explored  $(s, a)$

# Key idea: pessimism in the face of uncertainty

---

— Jin et al, 2020, Rashidinejad et al, 2021, Xie et al, 2021

1. build empirical model  $\hat{P}$
2. (**value iteration**) repeat: for all  $(s, a)$

$$\hat{Q}(s, a) \leftarrow \max \left\{ r(s, a) + \gamma \langle \hat{P}(\cdot | s, a), \hat{V} \rangle, 0 \right\}$$

where  $\hat{V}(s) = \max_a \hat{Q}(s, a)$

# Key idea: pessimism in the face of uncertainty

---

— Jin et al, 2020, Rashidinejad et al, 2021, Xie et al, 2021

Penalize those poorly visited  $(s, a) \dots$

1. build empirical model  $\hat{P}$
2. **(pessimistic value iteration)** repeat: for all  $(s, a)$

$$\hat{Q}(s, a) \leftarrow \max \left\{ r(s, a) + \gamma \langle \hat{P}(\cdot | s, a), \hat{V} \rangle - \underbrace{b(s, a; \hat{V})}_{\text{uncertainty penalty}}, 0 \right\}$$

where  $\hat{V}(s) = \max_a \hat{Q}(s, a)$

# Key idea: pessimism in the face of uncertainty

---

— Jin et al, 2020, Rashidinejad et al, 2021, Xie et al, 2021

Penalize those poorly visited  $(s, a) \dots$

1. build empirical model  $\hat{P}$
2. **(pessimistic value iteration)** repeat: for all  $(s, a)$

$$\hat{Q}(s, a) \leftarrow \max \left\{ r(s, a) + \gamma \langle \hat{P}(\cdot | s, a), \hat{V} \rangle - \underbrace{b(s, a; \hat{V})}_{\text{uncertainty penalty}}, 0 \right\}$$

compared w/ Rashidinejad et al, 2021

- sample-reuse across iterations
- Bernstein-style penalty

# Sample complexity of model-based offline RL

## Theorem (Li, Shi, Chen, Chi, Wei '24)

For any  $0 < \varepsilon \leq \frac{1}{1-\gamma}$ , the policy  $\hat{\pi}$  returned by VI-LCB using a Bernstein-style penalty term achieves

$$V^*(\rho) - V^{\hat{\pi}}(\rho) \leq \varepsilon$$

with high prob., with sample complexity at most

$$\tilde{O} \left( \frac{SC^*}{(1-\gamma)^3 \varepsilon^2} \right)$$

# Sample complexity of model-based offline RL

## Theorem (Li, Shi, Chen, Chi, Wei '24)

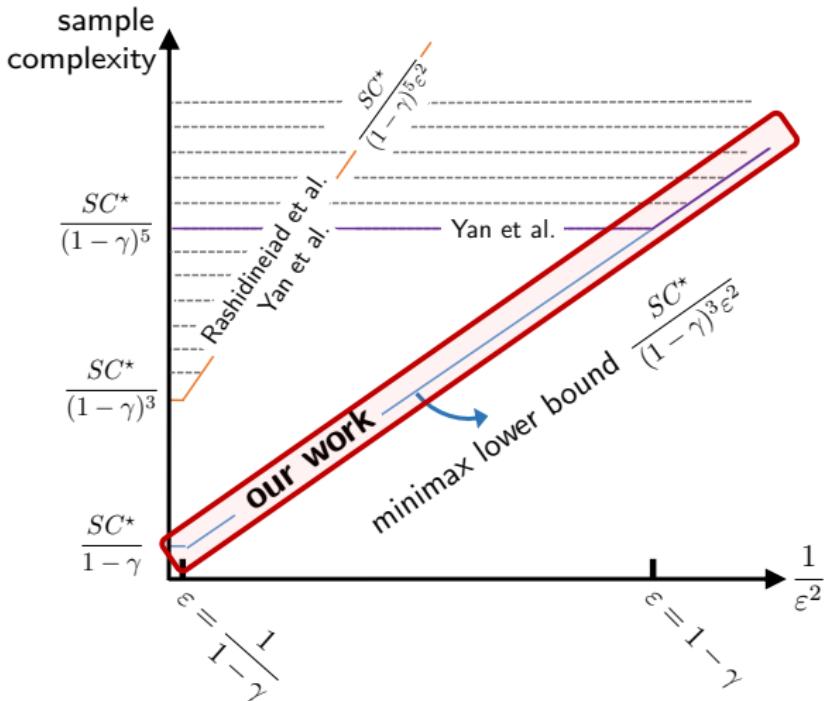
For any  $0 < \varepsilon \leq \frac{1}{1-\gamma}$ , the policy  $\hat{\pi}$  returned by VI-LCB using a Bernstein-style penalty term achieves

$$V^*(\rho) - V^{\hat{\pi}}(\rho) \leq \varepsilon$$

with high prob., with sample complexity at most

$$\tilde{O} \left( \frac{SC^*}{(1-\gamma)^3 \varepsilon^2} \right)$$

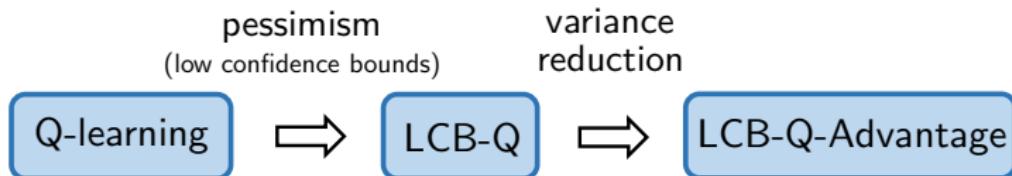
- depends on distribution shift (as reflected by  $C^*$ )
- achieves minimax optimality
- full  $\varepsilon$ -range (no burn-in cost)



Model-based offline RL is minimax optimal with no burn-in cost!

*Is it possible to design offline model-free algorithms  
with optimal sample efficiency?*

*Is it possible to design offline model-free algorithms  
with optimal sample efficiency?*



# LCB-Q: Q-learning with LCB penalty

---

— Shi et al, 2022, Yan et al, 2023

$$Q_{t+1}(s_t, a_t) \leftarrow \underbrace{(1 - \eta_t)Q_t(s_t, a_t) + \eta_t \mathcal{T}_t(Q_t)(s_t, a_t)}_{\text{classical Q-learning}} - \eta_t \underbrace{b_t(s_t, a_t)}_{\text{LCB penalty}}$$

# LCB-Q: Q-learning with LCB penalty

---

— Shi et al, 2022, Yan et al, 2023

$$Q_{t+1}(s_t, a_t) \leftarrow \underbrace{(1 - \eta_t)Q_t(s_t, a_t) + \eta_t \mathcal{T}_t(Q_t)(s_t, a_t)}_{\text{classical Q-learning}} - \eta_t \underbrace{b_t(s_t, a_t)}_{\text{LCB penalty}}$$

- $b_t(s, a)$ : Hoeffding-style confidence bound
- pessimism in the face of uncertainty

# LCB-Q: Q-learning with LCB penalty

---

— Shi et al, 2022, Yan et al, 2023

$$Q_{t+1}(s_t, a_t) \leftarrow \underbrace{(1 - \eta_t)Q_t(s_t, a_t) + \eta_t \mathcal{T}_t(Q_t)(s_t, a_t)}_{\text{classical Q-learning}} - \eta_t \underbrace{b_t(s_t, a_t)}_{\text{LCB penalty}}$$

- $b_t(s, a)$ : Hoeffding-style confidence bound
- pessimism in the face of uncertainty

sample size:  $\tilde{O}\left(\frac{SC^*}{(1-\gamma)^5 \varepsilon^2}\right) \implies$  sub-optimal by a factor of  $\frac{1}{(1-\gamma)^2}$

**Issue:** large variability in stochastic update rules

# Q-learning with LCB and variance reduction

---

— Shi et al, 2022, Yan et al, 2023

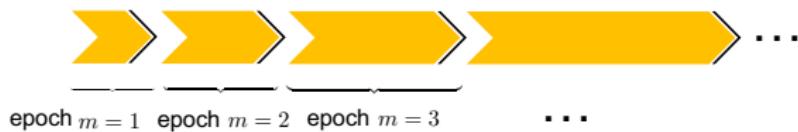
$$\begin{aligned} Q_{t+1}(s_t, a_t) &\leftarrow (1 - \eta_t)Q_t(s_t, a_t) - \eta_t \underbrace{b_t(s_t, a_t)}_{\text{LCB penalty}} \\ &+ \eta_t \left( \underbrace{\mathcal{T}_t(Q_t) - \mathcal{T}_t(\bar{Q})}_{\text{advantage}} + \underbrace{\hat{\mathcal{T}}(\bar{Q})}_{\text{reference}} \right)(s_t, a_t) \end{aligned}$$

# Q-learning with LCB and variance reduction

— Shi et al, 2022, Yan et al, 2023

$$\begin{aligned} Q_{t+1}(s_t, a_t) \leftarrow & (1 - \eta_t) Q_t(s_t, a_t) - \eta_t \underbrace{b_t(s_t, a_t)}_{\text{LCB penalty}} \\ & + \eta_t \left( \underbrace{\mathcal{T}_t(Q_t) - \mathcal{T}_t(\bar{Q})}_{\text{advantage}} + \underbrace{\hat{\mathcal{T}}(\bar{Q})}_{\text{reference}} \right)(s_t, a_t) \end{aligned}$$

- incorporates **variance reduction** into LCB-Q

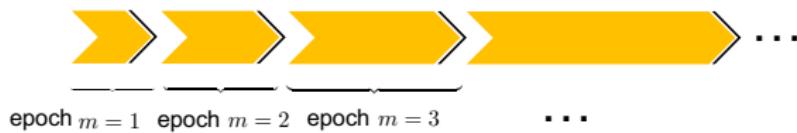


# Q-learning with LCB and variance reduction

— Shi et al, 2022, Yan et al, 2023

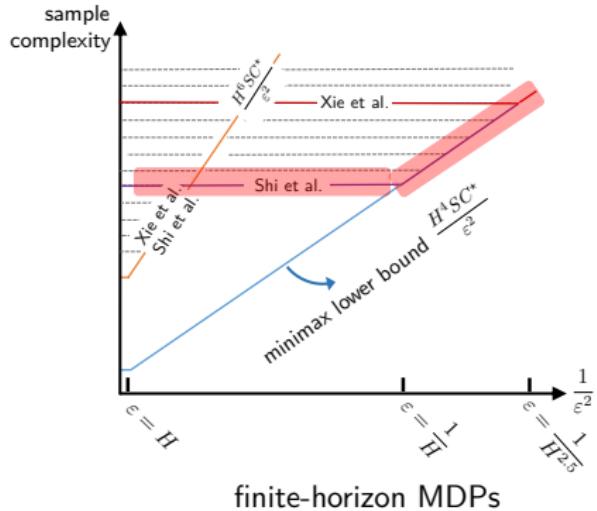
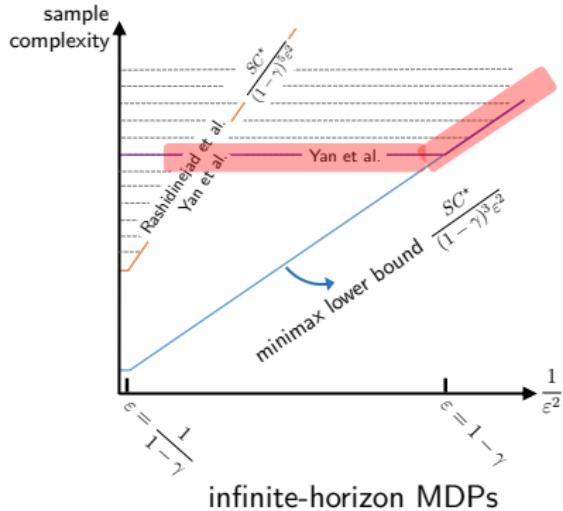
$$\begin{aligned} Q_{t+1}(s_t, a_t) \leftarrow & (1 - \eta_t) Q_t(s_t, a_t) - \eta_t \underbrace{b_t(s_t, a_t)}_{\text{LCB penalty}} \\ & + \eta_t \left( \underbrace{\mathcal{T}_t(Q_t) - \mathcal{T}_t(\bar{Q})}_{\text{advantage}} + \underbrace{\hat{\mathcal{T}}(\bar{Q})}_{\text{reference}} \right)(s_t, a_t) \end{aligned}$$

- incorporates **variance reduction** into LCB-Q



**Theorem (Yan, Li, Chen, Fan '23, Shi, Li, Wei, Chen, Chi '22)**

For  $\varepsilon \in (0, 1 - \gamma]$ , LCB-Q-Advantage achieves  $V^*(\rho) - V^\pi(\rho) \leq \varepsilon$  with optimal sample complexity  $\tilde{O}\left(\frac{SC^*}{(1-\gamma)^3 \varepsilon^2}\right)$



Model-free offline RL attains sample optimality too!

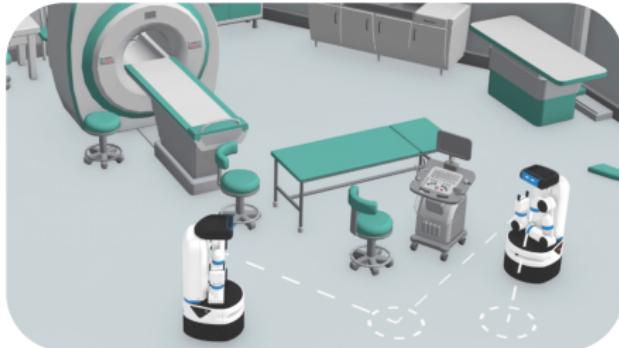
— with some burn-in cost though ...

## **Part 2**

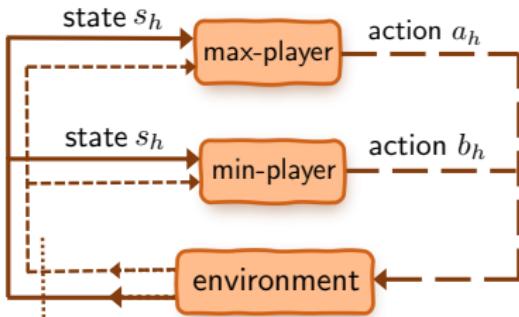
1. Online RL
2. Offline RL
3. **Multi-agent RL**
4. Robust RL

# Multi-agent reinforcement learning (MARL)

---

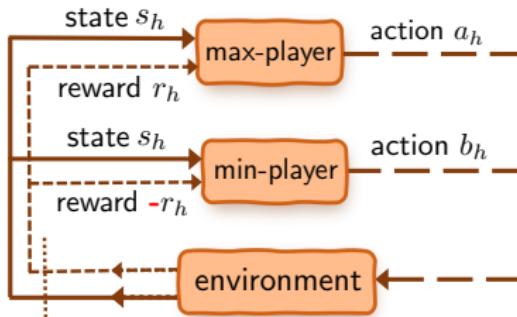


# Two-player zero-sum Markov games (finite-horizon)



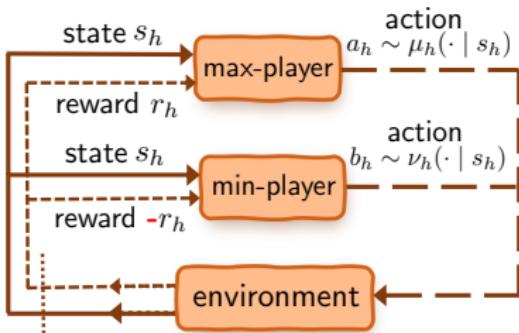
- $\mathcal{S} = [S]$ : state space
- $H$ : horizon
- $\mathcal{A} = [A]$ : action space of max-player
- $\mathcal{B} = [B]$ : action space of min-player

# Two-player zero-sum Markov games (finite-horizon)



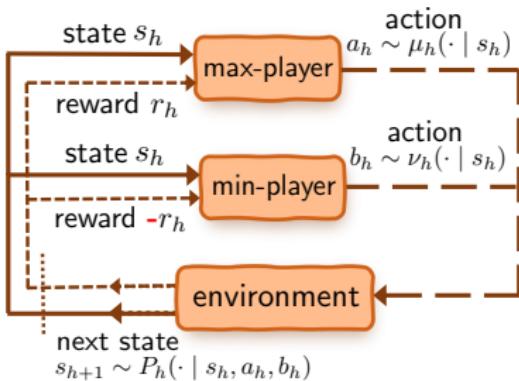
- $\mathcal{S} = [S]$ : state space
- $H$ : horizon
- immediate reward: max-player  $r(s, a, b) \in [0, 1]$   
min-player  $-r(s, a, b)$
- $\mathcal{A} = [A]$ : action space of max-player
- $\mathcal{B} = [B]$ : action space of min-player

# Two-player zero-sum Markov games (finite-horizon)



- $\mathcal{S} = [S]$ : state space
- $H$ : horizon
- immediate reward: max-player  $r(s, a, b) \in [0, 1]$   
min-player  $-r(s, a, b)$
- $\mu : \mathcal{S} \times [H] \rightarrow \Delta(\mathcal{A})$ : policy of max-player
- $\nu : \mathcal{S} \times [H] \rightarrow \Delta(\mathcal{B})$ : policy of min-player
- $\mathcal{A} = [A]$ : action space of max-player
- $\mathcal{B} = [B]$ : action space of min-player

# Two-player zero-sum Markov games (finite-horizon)



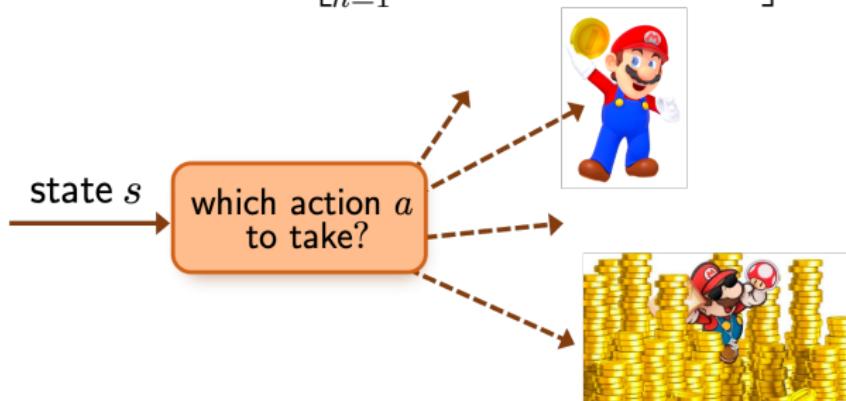
- $\mathcal{S} = [S]$ : state space
- $H$ : horizon
- immediate reward: max-player  $r(s, a, b) \in [0, 1]$   
min-player  $-r(s, a, b)$
- $\mu : \mathcal{S} \times [H] \rightarrow \Delta(\mathcal{A})$ : policy of max-player
- $\nu : \mathcal{S} \times [H] \rightarrow \Delta(\mathcal{B})$ : policy of min-player
- $P_h(\cdot | s, a, b)$ : **unknown** transition probabilities

**Value function** under *independent* policies  $(\mu, \nu)$  (no coordination)

$$V^{\mu, \nu}(s) := \mathbb{E} \left[ \sum_{h=1}^H r_h(s_h, a_h, b_h) \mid s_1 = s \right]$$

## **Value function** under *independent* policies $(\mu, \nu)$ (no coordination)

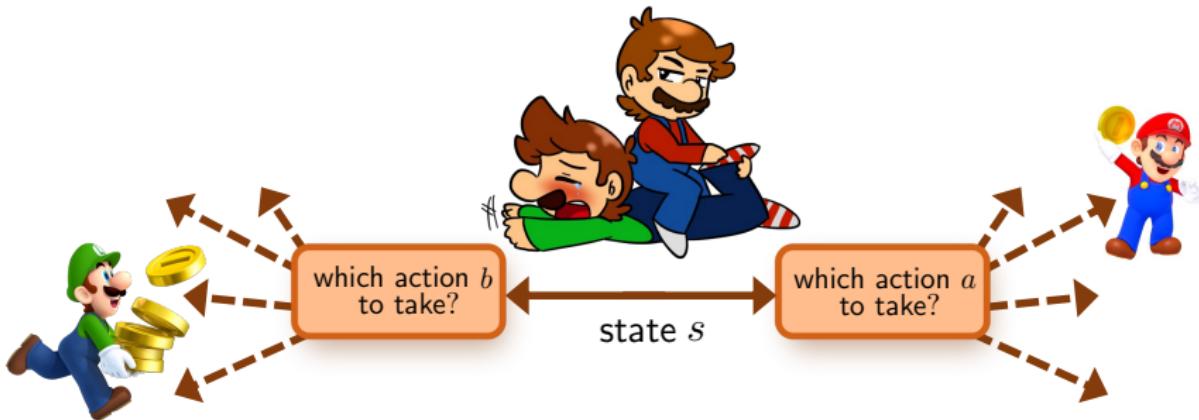
$$V^{\mu, \nu}(s) := \mathbb{E} \left[ \sum_{h=1}^H r_h(s_h, a_h, b_h) \mid s_1 = s \right]$$



- Each agent seeks **optimal policy** maximizing her own value

## Value function under *independent* policies $(\mu, \nu)$ (no coordination)

$$V^{\mu, \nu}(s) := \mathbb{E} \left[ \sum_{h=1}^H r_h(s_h, a_h, b_h) \mid s_1 = s \right]$$



- Each agent seeks **optimal policy** maximizing her own value
- But two agents have conflicting goals ...

# Compromise: Nash equilibrium (NE)

---



*John von Neumann*

*John Nash*

An NE policy pair  $(\mu^*, \nu^*)$  obeys

$$\max_{\mu} V^{\mu, \nu^*} = V^{\mu^*, \nu^*} = \min_{\nu} V^{\mu^*, \nu}$$

# Compromise: Nash equilibrium (NE)

---



*John von Neumann*

*John Nash*

An NE policy pair  $(\mu^*, \nu^*)$  obeys

$$\max_{\mu} V^{\mu, \nu^*} = V^{\mu^*, \nu^*} = \min_{\nu} V^{\mu^*, \nu}$$

- no unilateral deviation is beneficial

# Compromise: Nash equilibrium (NE)

---



*John von Neumann*

*John Nash*

An NE policy pair  $(\mu^*, \nu^*)$  obeys

$$\max_{\mu} V^{\mu, \nu^*} = V^{\mu^*, \nu^*} = \min_{\nu} V^{\mu^*, \nu}$$

- no unilateral deviation is beneficial
- no coordination between two agents (they act *independently*)

# Compromise: Nash equilibrium (NE)

---



*John von Neumann*

*John Nash*

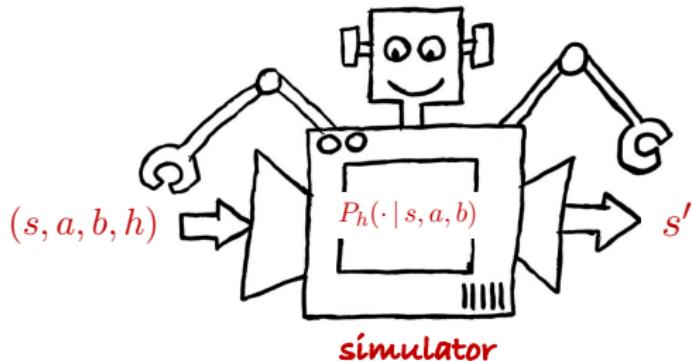
An  $\varepsilon$ -NE policy pair  $(\hat{\mu}, \hat{\nu})$  obeys

$$\max_{\mu} V^{\mu, \hat{\nu}} - \varepsilon \leq V^{\hat{\mu}, \hat{\nu}} \leq \min_{\nu} V^{\hat{\mu}, \nu} + \varepsilon$$

- no unilateral deviation is beneficial
- no coordination between two agents (they act *independently*)

# Learning NEs with a simulator

---

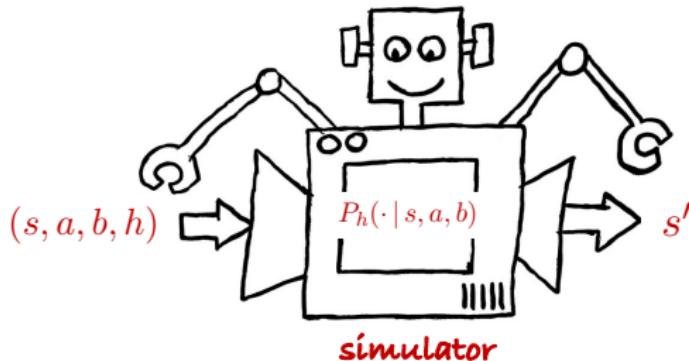


**input:** any  $(s, a, b, h)$

**output:** an independent sample  $s' \sim P_h(\cdot | s, a, b)$

# Learning NEs with a simulator

---



**input:** any  $(s, a, b, h)$

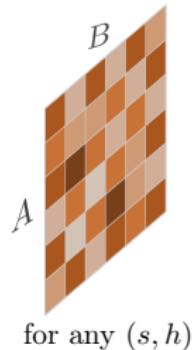
**output:** an independent sample  $s' \sim P_h(\cdot | s, a, b)$

**Question:** how many samples are sufficient to learn an  $\varepsilon$ -Nash policy pair?

# Model-based approach (non-adaptive sampling)

---

— *Zhang, Kakade, Başar, Yang '20*

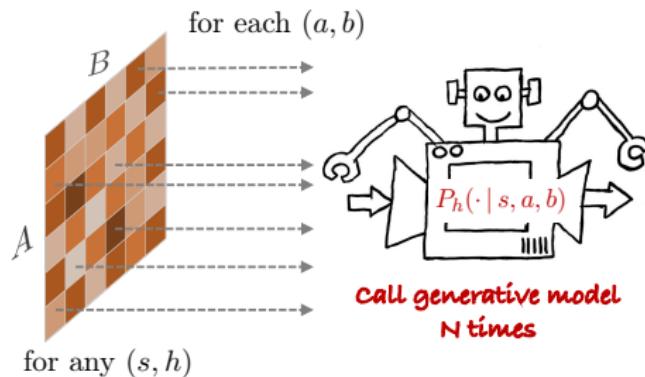


for any  $(s, h)$

1. for each  $(s, a, b, h)$ , call simulator  $N$  times

# Model-based approach (non-adaptive sampling)

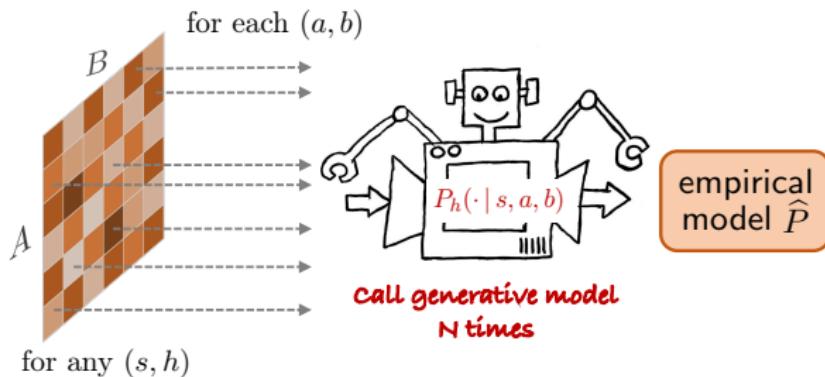
— Zhang, Kakade, Başar, Yang '20



1. for each  $(s, a, b, h)$ , call simulator  $N$  times

# Model-based approach (non-adaptive sampling)

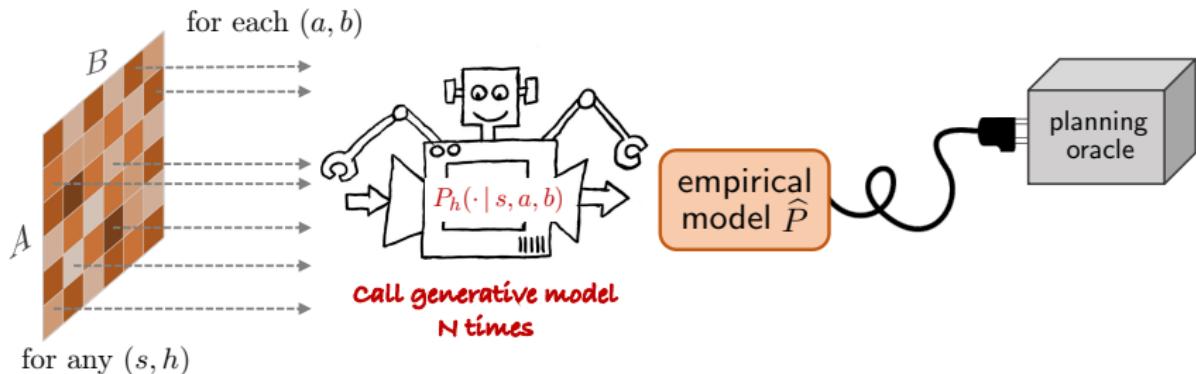
— Zhang, Kakade, Başar, Yang '20



1. for each  $(s, a, b, h)$ , call simulator  $N$  times
2. build empirical model  $\hat{P}$

# Model-based approach (non-adaptive sampling)

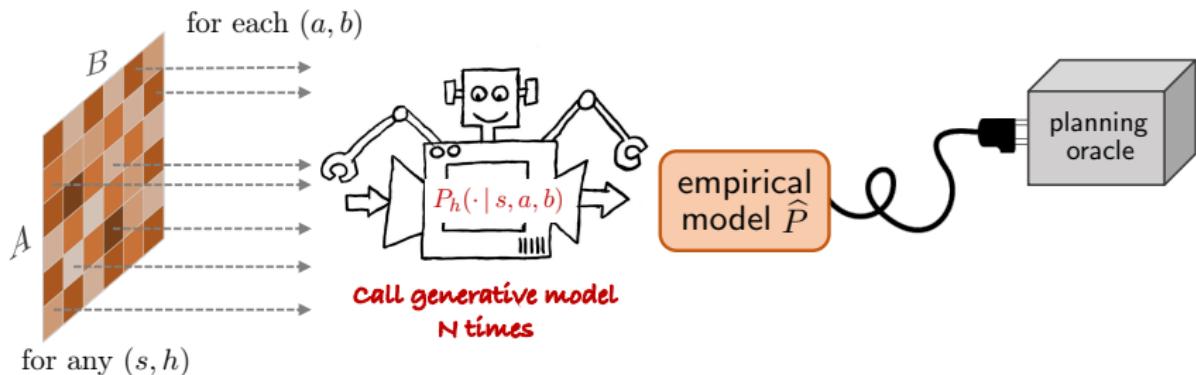
— Zhang, Kakade, Başar, Yang '20



1. for each  $(s, a, b, h)$ , call simulator  $N$  times
2. build empirical model  $\hat{P}$ , and run “plug-in” methods

# Model-based approach (non-adaptive sampling)

— Zhang, Kakade, Başar, Yang '20

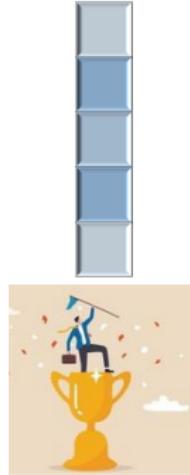


1. for each  $(s, a, b, h)$ , call simulator  $N$  times
2. build empirical model  $\hat{P}$ , and run “plug-in” methods

sample complexity:  $\frac{H^4 S A B}{\varepsilon^2}$

# Curse of multiple agents

---

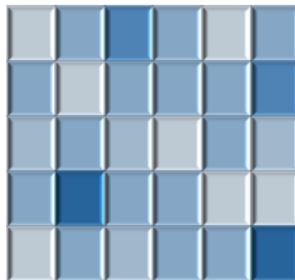


1 player:  $A$

Let's look at the **size** of joint action space ...

# Curse of multiple agents

---



1 player:  $A$



2 players:  $AB$

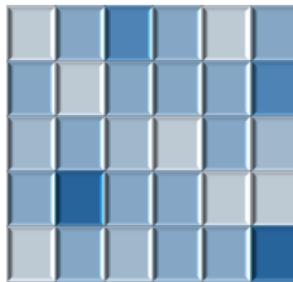
Let's look at the **size** of joint action space ...

# Curse of multiple agents

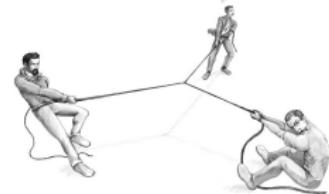
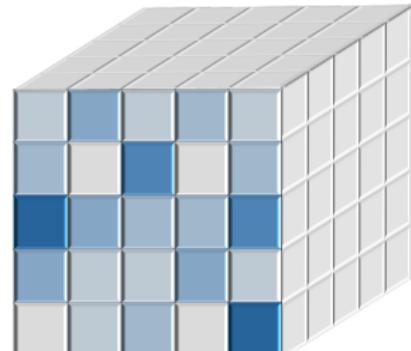
---



1 player:  $A$



2 players:  $AB$



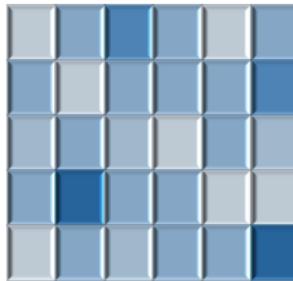
$m$  players:  $A_1 A_2 \cdots A_m$

Let's look at the **size** of joint action space ...

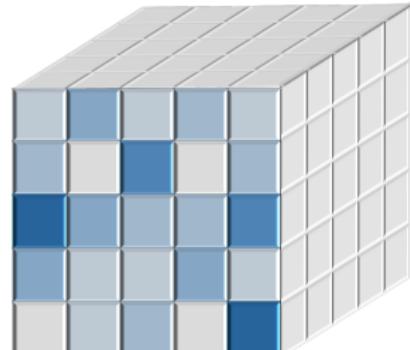
# Curse of multiple agents



1 player:  $A$



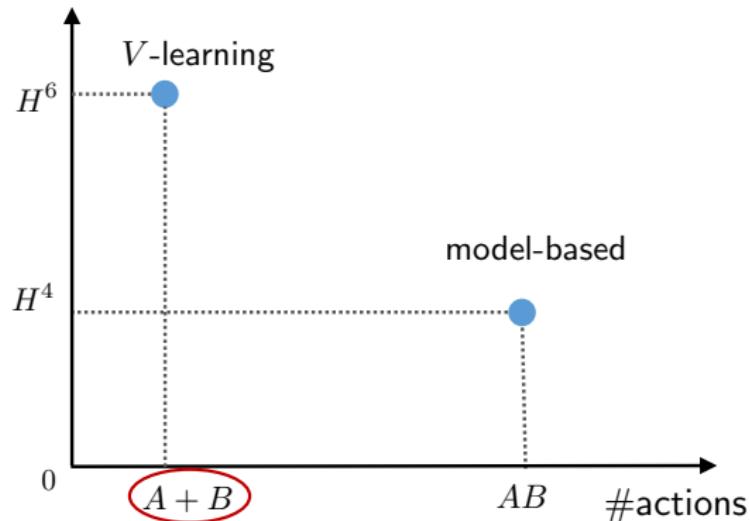
2 players:  $AB$



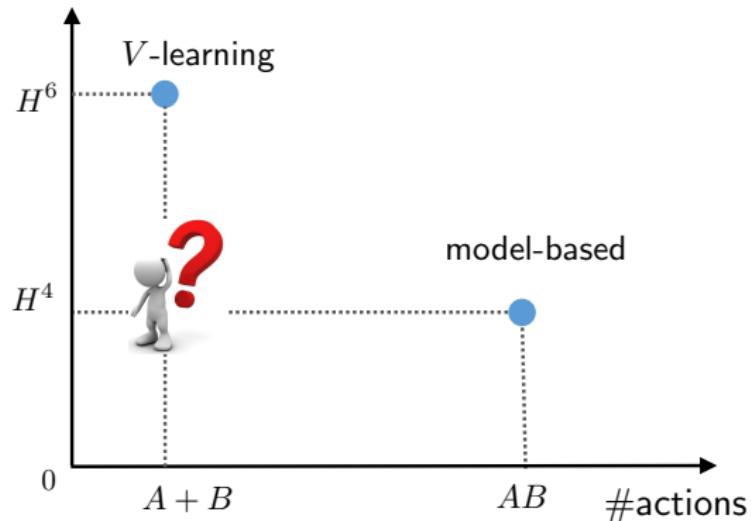
$m$  players:  $A_1 A_2 \cdots A_m$

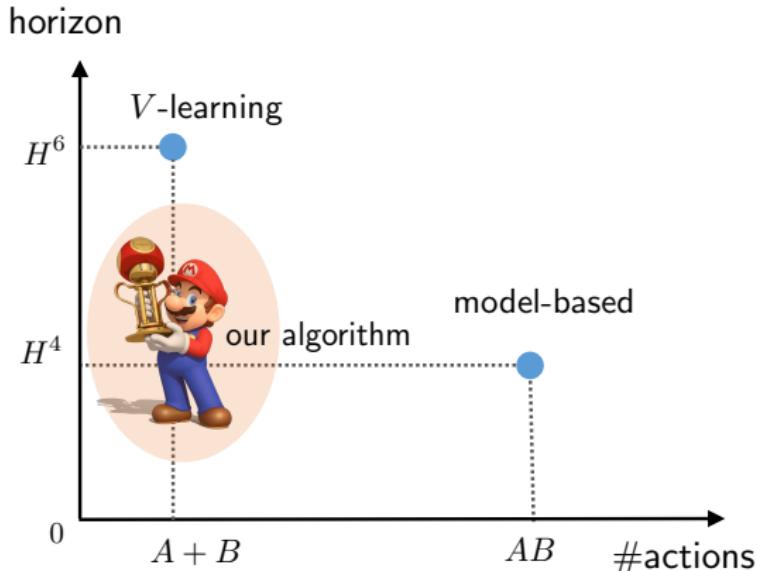
# joint actions **blows up geometrically in # players!**

horizon



horizon





### Theorem (Li, Chi, Wei, Chen '22)

For any  $0 < \varepsilon \leq H$ , one can design an algorithm that finds an  $\varepsilon$ -Nash policy pair  $(\hat{\mu}, \hat{\nu})$  with high prob., with sample complexity at most

$$\tilde{O}\left(\frac{H^4 S(A+B)}{\varepsilon^2}\right) \quad (\text{minimax-optimal } \forall \varepsilon)$$

## **Part 2**

1. Online RL
2. Offline RL
3. Multi-agent RL
4. Robust RL

# Safety and robustness in RL

---

(Zhou et al., 2021; Panaganti and Kalathil, 2022; Yang et al., 2022;)



Training environment



Test environment

$\neq$

# Safety and robustness in RL

(Zhou et al., 2021; Panaganti and Kalathil, 2022; Yang et al., 2022;)



Training environment



Test environment

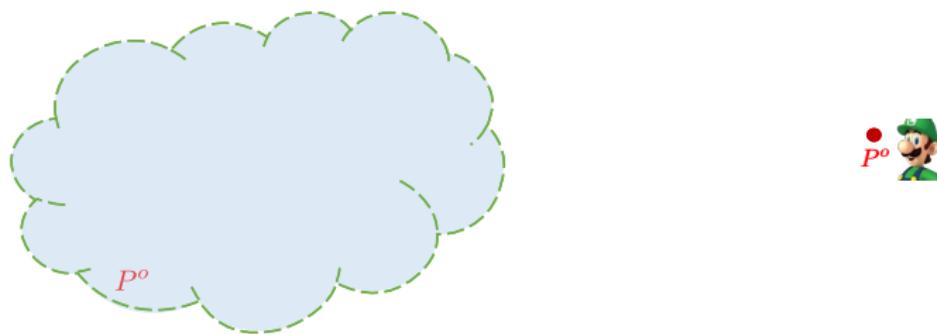
**Sim2Real Gap:** Can we learn optimal policies that are robust  
to model perturbations?

# Modeling environment uncertainty

---

Uncertainty set of the nominal transition kernel  $P^o$ :

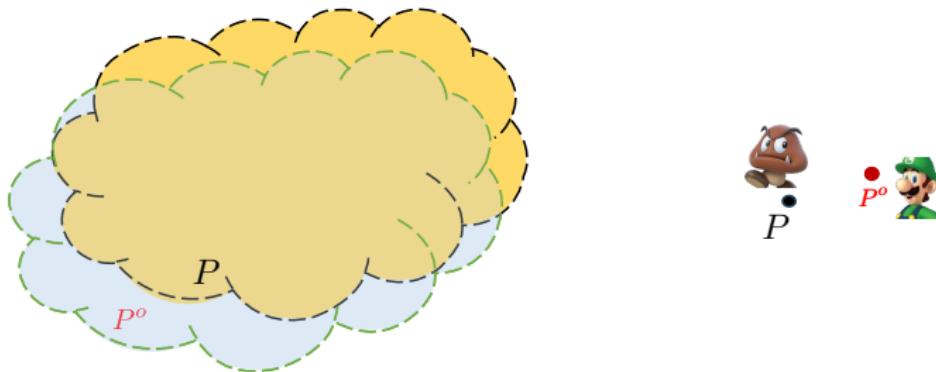
$$\mathcal{U}^\sigma(P^o) = \{P : \rho(P, P^o) \leq \sigma\}$$



# Modeling environment uncertainty

Uncertainty set of the nominal transition kernel  $P^o$ :

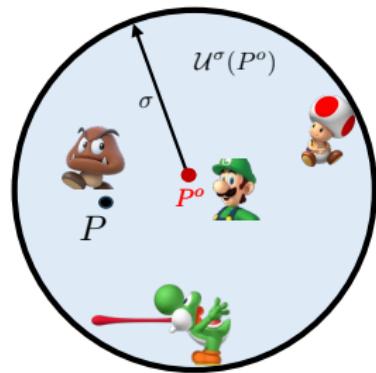
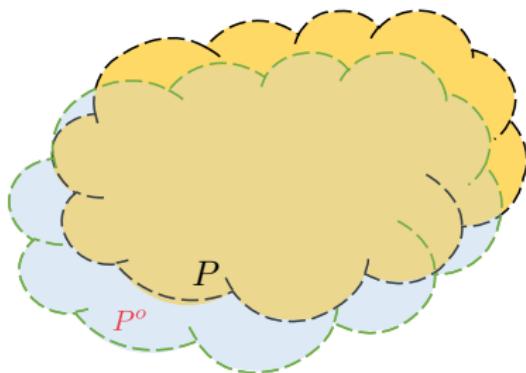
$$\mathcal{U}^\sigma(P^o) = \{P : \rho(P, P^o) \leq \sigma\}$$



# Modeling environment uncertainty

Uncertainty set of the nominal transition kernel  $P^o$ :

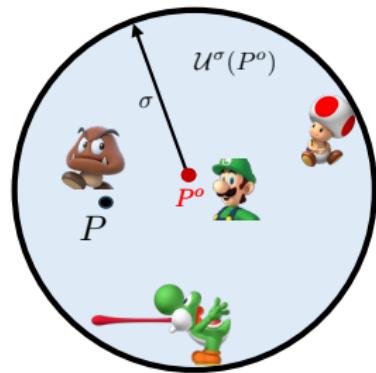
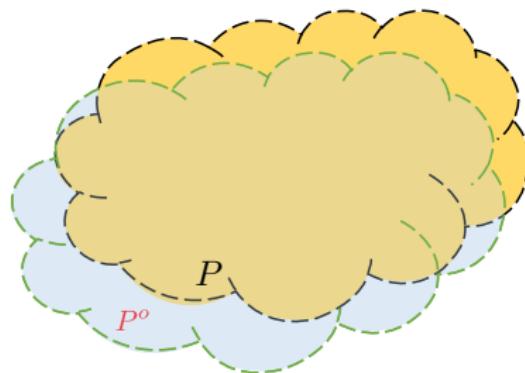
$$\mathcal{U}^\sigma(P^o) = \{P : \rho(P, P^o) \leq \sigma\}$$



# Modeling environment uncertainty

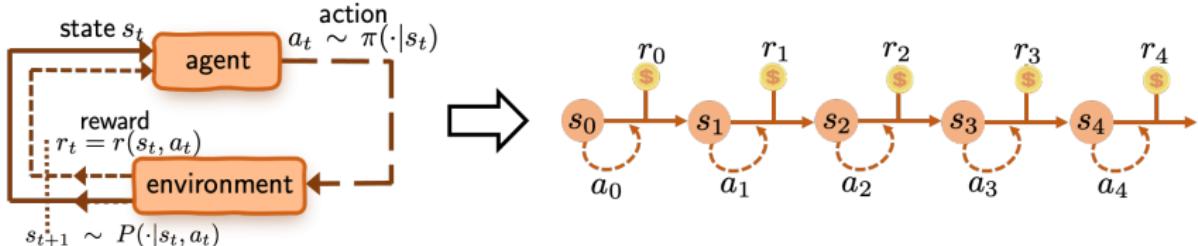
Uncertainty set of the nominal transition kernel  $P^o$ :

$$\mathcal{U}^\sigma(P^o) = \{P : \rho(P, P^o) \leq \sigma\}$$



- Examples of  $\rho$ : f-divergence (TV,  $\chi^2$ , KL...)

# Robust value/Q function



**Robust value/Q function** of policy  $\pi$ :

$$\forall s \in \mathcal{S} : \quad V^{\pi, \sigma}(s) := \inf_{P \in \mathcal{U}^\sigma(P^o)} \mathbb{E}_{\pi, P} \left[ \sum_{t=0}^{\infty} \gamma^t r_t \mid s_0 = s \right]$$

$$\forall (s, a) \in \mathcal{S} \times \mathcal{A} : \quad Q^{\pi, \sigma}(s, a) := \inf_{P \in \mathcal{U}^\sigma(P^o)} \mathbb{E}_{\pi, P} \left[ \sum_{t=0}^{\infty} \gamma^t r_t \mid s_0 = s, a_0 = a \right]$$

Measures the **worst-case** performance of the policy in the uncertainty set.

# Distributionally robust MDP

---

## Robust MDP

*Find the policy  $\pi^*$  that maximizes  $V^{\pi, \sigma}$*

(Iyengar. '05, Niliim and El Ghaoui. '05)

# Distributionally robust MDP

## Robust MDP

*Find the policy  $\pi^*$  that maximizes  $V^{\pi,\sigma}$*

(Iyengar. '05, Niliim and El Ghaoui. '05)

**Robust Bellman's optimality equation:** the optimal robust policy  $\pi^*$  and optimal robust value  $V^{*,\sigma} := V^{\pi^*,\sigma}$  satisfy

$$Q^{*,\sigma}(s, a) = r(s, a) + \gamma \inf_{P_{s,a} \in \mathcal{U}^\sigma(P_{s,a}^o)} \langle P_{s,a}, V^{*,\sigma} \rangle,$$

$$V^{*,\sigma}(s) = \max_a Q^{*,\sigma}(s, a)$$

# Distributionally robust MDP

## Robust MDP

*Find the policy  $\pi^*$  that maximizes  $V^{\pi,\sigma}$*

(Iyengar. '05, Niliim and El Ghaoui. '05)

**Robust Bellman's optimality equation:** the optimal robust policy  $\pi^*$  and optimal robust value  $V^{*,\sigma} := V^{\pi^*,\sigma}$  satisfy

$$Q^{*,\sigma}(s, a) = r(s, a) + \gamma \inf_{P_{s,a} \in \mathcal{U}^\sigma(P_{s,a}^o)} \langle P_{s,a}, V^{*,\sigma} \rangle,$$

$$V^{*,\sigma}(s) = \max_a Q^{*,\sigma}(s, a)$$

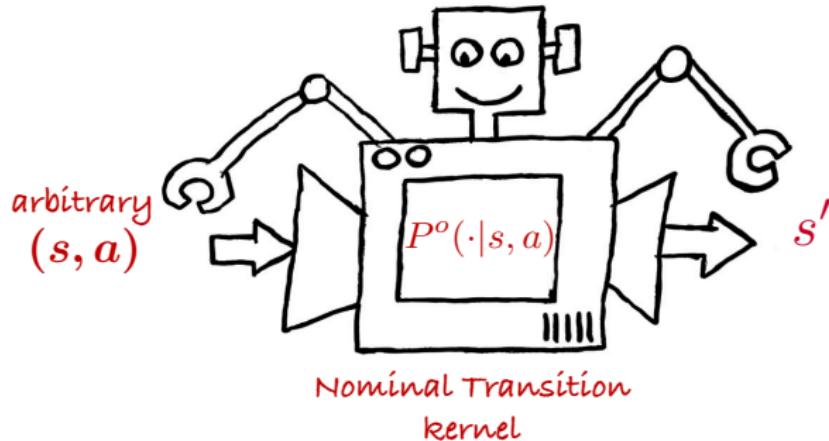
**Distributionally robust value iteration (DRVl):**

$$Q(s, a) \leftarrow r(s, a) + \gamma \inf_{P_{s,a} \in \mathcal{U}^\sigma(P_{s,a}^o)} \langle P_{s,a}, V \rangle,$$

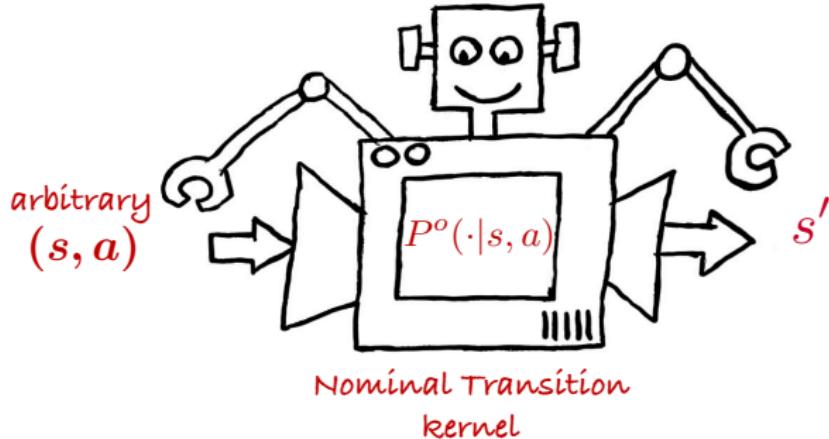
where  $V(s) = \max_a Q(s, a)$ .

# Learning distributionally robust MDPs

---



# Learning distributionally robust MDPs



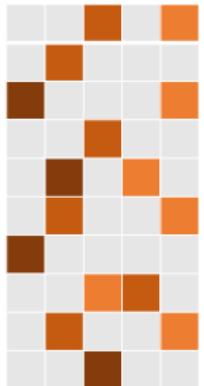
**Goal of robust RL:** given  $\mathcal{D} := \{(s_i, a_i, s'_i)\}_{i=1}^N$  from the *nominal* environment  $P^0$ , find an  $\varepsilon$ -optimal robust policy  $\hat{\pi}$  obeying

$$V^{*,\sigma} - V^{\hat{\pi},\sigma} \leq \varepsilon$$

— in a sample-efficient manner

# A curious question

---



empirical MDP



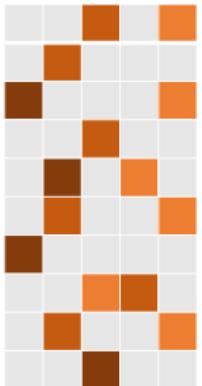
Learn the optimal policy of  
the nominal MDP?

Learn the **robust** policy  
around the nominal MDP?

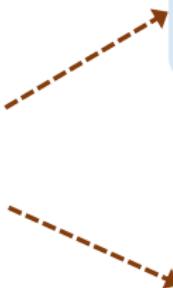


# A curious question

---



Learn the optimal policy of  
the nominal MDP?

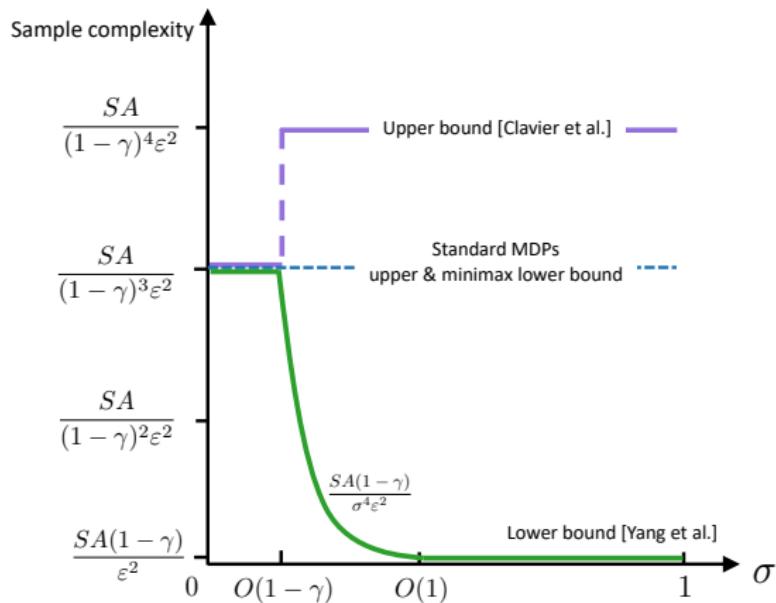


Learn the **robust** policy  
around the nominal MDP?



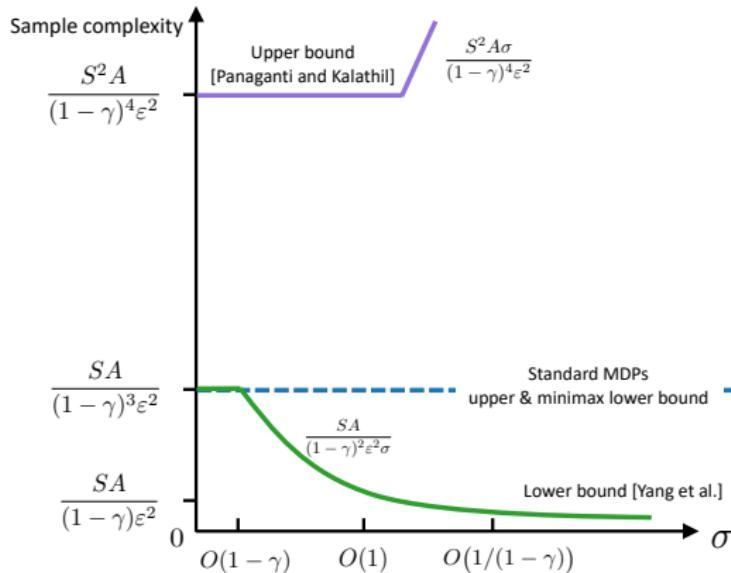
**Robustness-statistical trade-off?** Is there a statistical premium  
that one needs to pay in quest of additional robustness?

# Prior art: TV uncertainty



- Large gaps between existing upper and lower bounds
- Unclear benchmarking with standard MDP

# Prior art: $\chi^2$ uncertainty



- Large gaps between existing upper and lower bounds
- Unclear benchmarking with standard MDP

# Our theorem under TV uncertainty

## Theorem (Shi et al., 2023)

Assume the uncertainty set is measured via the TV distance with radius  $\sigma \in [0, 1]$ . For sufficiently small  $\varepsilon > 0$ , DRVI outputs a policy  $\hat{\pi}$  that satisfies  $V^{*,\sigma} - V^{\hat{\pi},\sigma} \leq \varepsilon$  with sample complexity at most

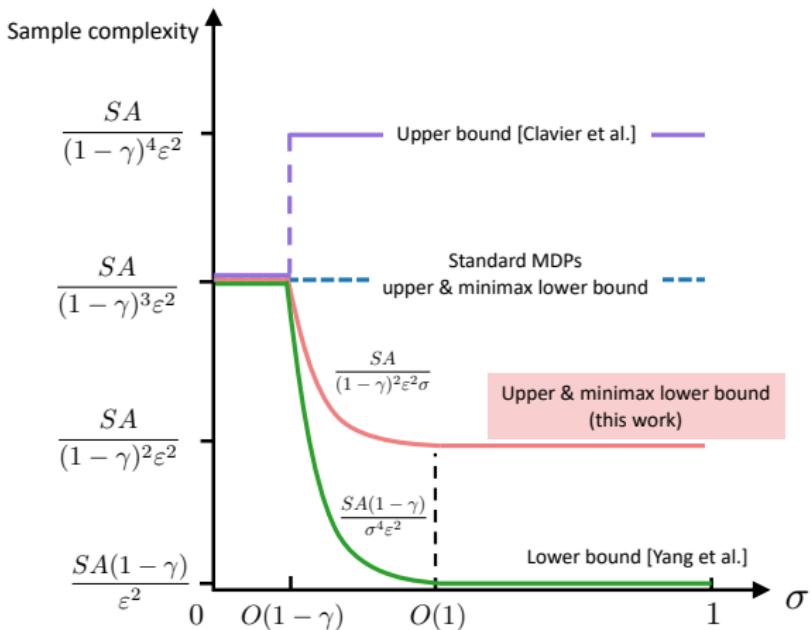
$$\tilde{O}\left(\frac{SA}{(1-\gamma)^2 \max\{1-\gamma, \sigma\} \varepsilon^2}\right)$$

ignoring logarithmic factors. In addition, no algorithm can succeed if the sample size is below

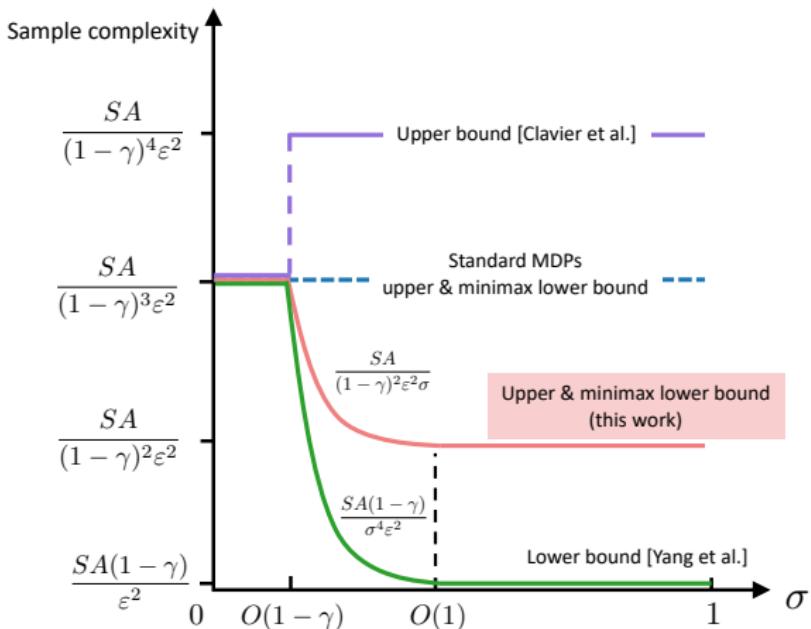
$$\tilde{\Omega}\left(\frac{SA}{(1-\gamma)^2 \max\{1-\gamma, \sigma\} \varepsilon^2}\right).$$

- Establish the minimax optimality of DRVI for RMDP under the TV uncertainty set over the full range of  $\sigma$ .

# When the uncertainty set is TV



# When the uncertainty set is TV



RMDPs are **easier** to learn than standard MDPs.

## Our theorem under $\chi^2$ uncertainty

### Theorem (Upper bound, Shi et al., 2023)

Assume the uncertainty set is measured via the  $\chi^2$  divergence with radius  $\sigma \in [0, \infty)$ . For sufficiently small  $\varepsilon > 0$ , DRVI outputs a policy  $\hat{\pi}$  that satisfies  $V^{*,\sigma} - V^{\hat{\pi},\sigma} \leq \varepsilon$  with sample complexity at most

$$\tilde{O}\left(\frac{SA(1+\sigma)}{(1-\gamma)^4\varepsilon^2}\right)$$

ignoring logarithmic factors.

# Our theorem under $\chi^2$ uncertainty

## Theorem (Upper bound, Shi et al., 2023)

Assume the uncertainty set is measured via the  $\chi^2$  divergence with radius  $\sigma \in [0, \infty)$ . For sufficiently small  $\varepsilon > 0$ , DRVI outputs a policy  $\hat{\pi}$  that satisfies  $V^{*,\sigma} - V^{\hat{\pi},\sigma} \leq \varepsilon$  with sample complexity at most

$$\tilde{O}\left(\frac{SA(1+\sigma)}{(1-\gamma)^4\varepsilon^2}\right)$$

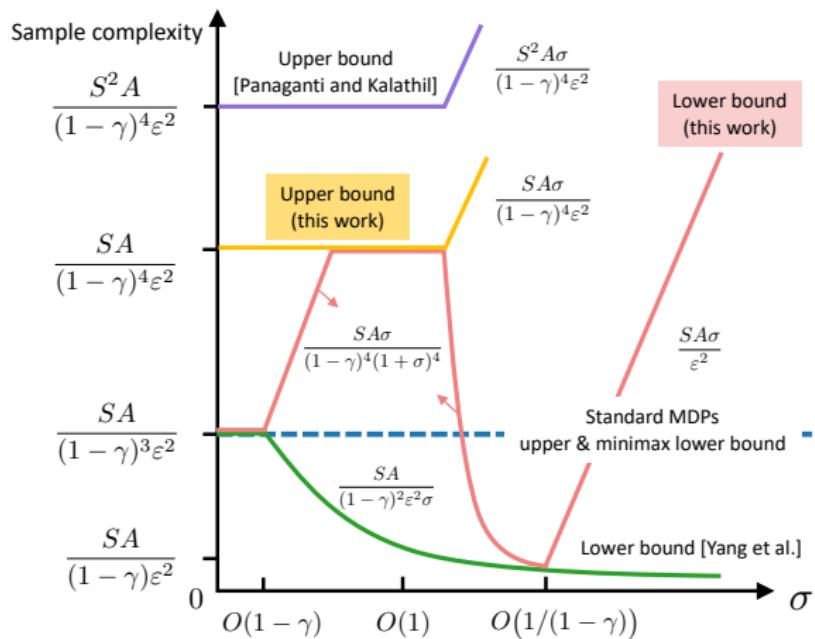
ignoring logarithmic factors.

## Theorem (Lower bound, Shi et al., 2023)

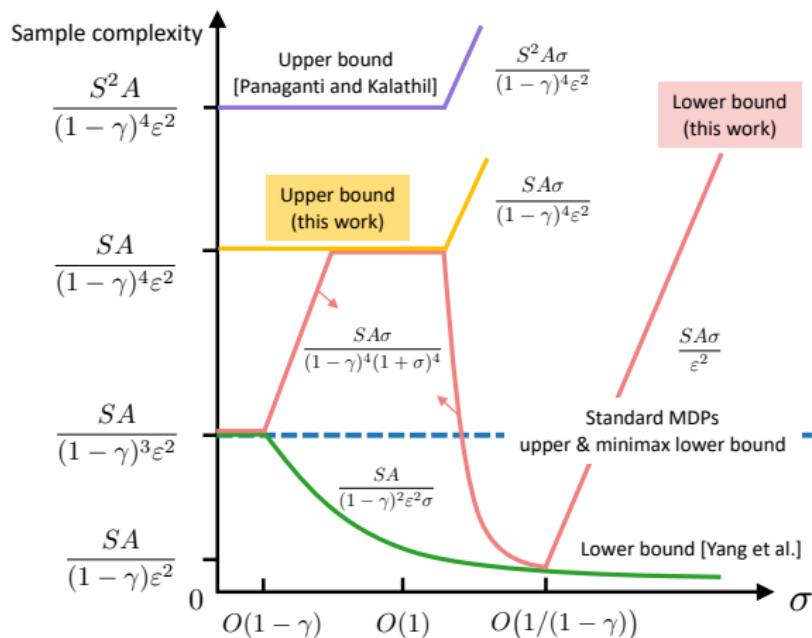
In addition, no algorithm succeeds when the sample size is below

$$\begin{cases} \tilde{\Omega}\left(\frac{SA}{(1-\gamma)^3\varepsilon^2}\right) & \text{if } \sigma \lesssim 1 - \gamma \\ \tilde{\Omega}\left(\frac{\sigma SA}{\min\{1, (1-\gamma)^4(1+\sigma)^4\}\varepsilon^2}\right) & \text{otherwise} \end{cases}$$

# When the uncertainty set is $\chi^2$ divergence



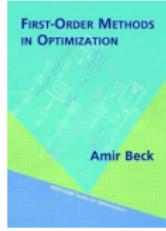
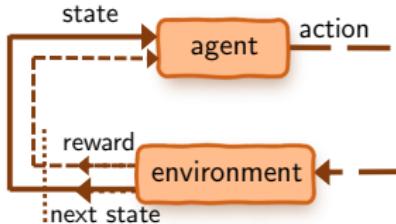
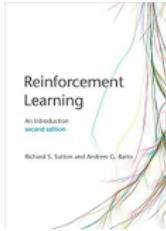
# When the uncertainty set is $\chi^2$ divergence



RMDPs can be **harder** to learn than standard MDPs.

## *Concluding Remarks*

# Concluding remarks



Understanding non-asymptotic performances of RL algorithms is a fruitful playground!

## Promising directions:

- function approximation
- multi-agent/federated RL
- hybrid RL
- many more...

# Beyond the tabular setting

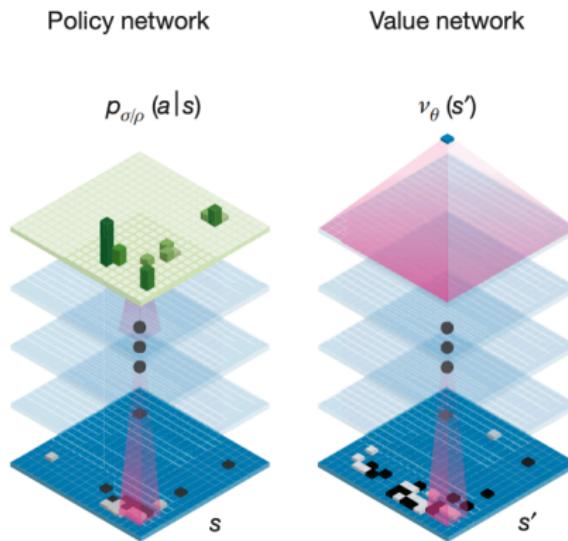


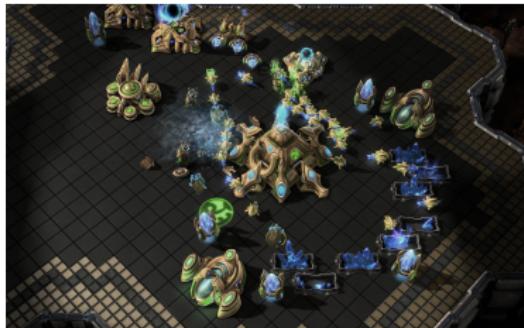
Figure credit: (Silver et al., 2016)

- function approximation for dimensionality reduction
- Provably efficient RL algorithms under minimal assumptions

(Osband and Van Roy, 2014; Dai et al., 2018; Du et al., 2019; Jin et al., 2020)

# Multi-agent RL

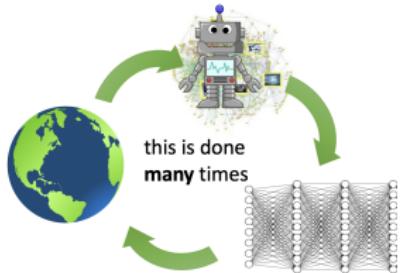
---



- **Competitive setting:** finding Nash equilibria for Markov games
- **Collaborative setting:** multiple agents jointly optimize the policy to maximize the total reward

(Zhang, Yang, and Basar, 2021; Cen, Wei, and Chi, 2021)

# Hybrid RL

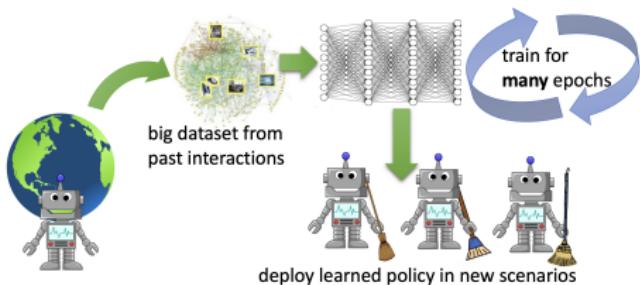


## *Online RL*

- interact with environment
- actively collect new data

## *Offline/Batch RL*

- no interaction
- data is given

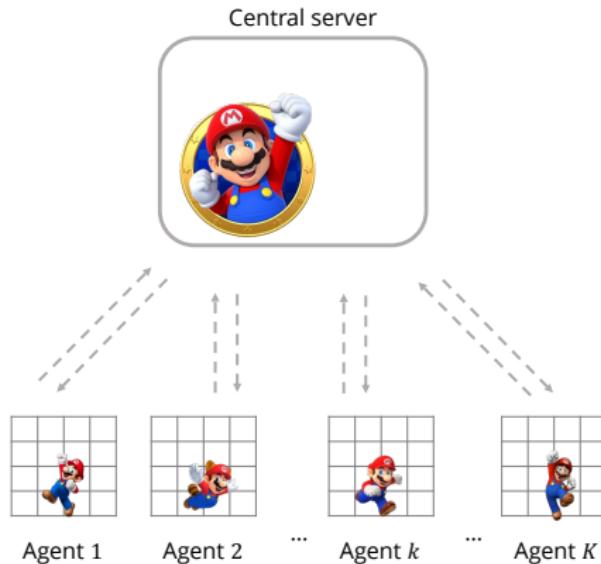


**Can we achieve the best of both worlds?**

(Wagenmaker and Pacchiano, 2022; Song et al., 2022; Li et al., 2023)

# RL meets federated learning

Federated reinforcement learning enables multiple agents to collaboratively learn a global model without sharing datasets.



**Can we achieve linear speedup via federated learning?**

(Khodadadian et al., 2022; Woo et al., 2023)

## Reference: online RL I

---

- “*Asymptotically efficient adaptive allocation rules,*” T. L. Lai, H. Robbins, *Advances in applied mathematics*, vol. 6, no. 1, 1985
- “*Finite-time analysis of the multiarmed bandit problem,*” P. Auer, N. Cesa-Bianchi, P. Fischer, *Machine learning*, vol. 47, pp. 235-256, 2002
- “*Minimax regret bounds for reinforcement learning,*” M. G. Azar, I. Osband, R. Munos, *ICML*, 2017
- “*Is Q-learning provably efficient?*” C. Jin, Z. Allen-Zhu, S. Bubeck, and M. Jordan, *NeurIPS*, 2018
- “*Provably efficient Q-learning with low switching cost,*” Y. Bai, T. Xie, N. Jiang, Y. X. Wang, *NeurIPS*, 2019
- “*Episodic reinforcement learning in finite MDPs: Minimax lower bounds revisited*” O. D. Domingues, P. Menard, E. Kaufmann, M. Valko, *Algorithmic Learning Theory*, 2021
- “*Almost optimal model-free reinforcement learning via reference-advantage decomposition,*” Z. Zhang, Y. Zhou, X. Ji, *NeurIPS*, 2020

## Reference: online RL II

---

- “*Is reinforcement learning more difficult than bandits? a near-optimal algorithm escaping the curse of horizon,*” Z. Zhang, X. Ji, and S. Du, *COLT*, 2021
- “*Breaking the sample complexity barrier to regret-optimal model-free reinforcement learning,*” G. Li, L. Shi, Y. Chen, Y. Gu, Y. Chi, *NeurIPS*, 2021
- “*Regret-optimal model-free reinforcement learning for discounted MDPs with short burn-in time,*” X. Ji, G. Li, *NeurIPS*, 2023
- “*Reward-free exploration for reinforcement learning,*” C. Jin, A. Krishnamurthy, M. Simchowitz, T. Yu, *ICML*, 2020
- “*Minimax-optimal reward-agnostic exploration in reinforcement learning,*” G. Li, Y. Yan, Y. Chen, J. Fan, *COLT*, 2024
- “*Settling the sample complexity of online reinforcement learning,*” Z. Zhang, Y. Chen, J. D. Lee, S. S. Du, *COLT*, 2024

## Reference: offline RL I

---

- “*Bridging offline reinforcement learning and imitation learning: A tale of pessimism,*” P. Rashidinejad, B. Zhu, C. Ma, J. Jiao, S. Russell, *NeurIPS*, 2021
- “*Is pessimism provably efficient for offline RL?*” Y. Jin, Z. Yang, Z. Wang, *ICML*, 2021
- “*Settling the sample complexity of model-based offline reinforcement learning,*” G. Li, L. Shi, Y. Chen, Y. Chi, Y. Wei, *Annals of Statistics*, vol. 52, no. 1, pp. 233-260, 2024
- “*Pessimistic Q-learning for offline reinforcement learning: Towards optimal sample complexity,*” L. Shi, G. Li, Y. Wei, Y. Chen, Y. Chi, *ICML*, 2022
- “*The efficacy of pessimism in asynchronous Q-learning,*” Y. Yan, G. Li, Y. Chen, J. Fan, *IEEE Transactions on Information Theory*, 2023
- “*Policy finetuning: Bridging sample-efficient offline and online reinforcement learning*” T. Xie, N. Jiang, H. Wang, C. Xiong, Y. Bai, *NeurIPS*, 2021

# Reference: multi-agent RL I

---

- “*Stochastic games*,” L. S. Shapley, *Proceedings of the national academy of sciences*, 1953
- “*Twenty lectures on algorithmic game theory*,” T. Roughgarden, 2016
- “*Model-based multi-agent RL in zero-sum Markov games with near-optimal sample complexity*,” K. Zhang, S. Kakade, T. Basar, L. Yang, *NeurIPS*, 2020
- “*When can we learn general-sum Markov games with a large number of players sample-efficiently?*” Z. Song, S. Mei, Y. Bai, *ICLR*, 2021
- “*V-learning—A simple, efficient, decentralized algorithm for multiagent RL*,” C. Jin, Q. Liu, Y. Wang, T. Yu, 2021
- “*Minimax-optimal multi-agent RL in Markov games with a generative model*,” G. Li, Y. Chi, Y. Wei, Y. Chen, *NeurIPS*, 2022
- “*When are offline two-player zero-sum Markov games solvable?*” Q. Cui, S. S. Du, *NeurIPS*, 2022
- “*Model-based reinforcement learning for offline zero-sum Markov games*,” Y. Yan, G. Li, Y. Chen, J. Fan, *Operations Research*, 2024

# Reference: robust RL I

---

- “*Robust dynamic programming,*” G. Iyengar, *Mathematics of Operations Research*, 2005
- “*The curious price of distributional robustness in reinforcement learning with a generative model.,*” L. Shi, G. Li, Y. Wei, Y. Chen, M. Geist, Y. Chi, *NeurIPS*, 2023
- “*Distributionally robust model-based offline reinforcement learning with near-optimal sample complexity,*” L. Shi, Y. Chi, 2022
- “*On the foundation of distributionally robust reinforcement learning,*” S. Wang, N. Si, J. Blanchet, and Z. Zhou, 2023
- “*Sample complexity of robust reinforcement learning with a generative model,*” K. Panaganti, D. Kalathil, *AISTATS*, 2022
- “*Sample-Efficient Robust Multi-Agent Reinforcement Learning in the Face of Environmental Uncertainty,*” L. Shi, E. Mazumdar, Y. Chi, and A. Wierman, *ICML*, 2024

# Thanks!

---



<https://users.ece.cmu.edu/~yuejiec/>