Demystifying the efficiency of reinforcement learning: A few recent stories

Yuxin Chen

EE, Princeton University
Reinforcement learning (RL)
RL challenges

In RL, an agent learns by interacting with an environment

- unknown or changing environments
- delayed rewards or feedback
- enormous state and action space
- nonconvexity
Sample efficiency

Collecting data samples might be expensive or time-consuming
Sample efficiency

Collecting data samples might be expensive or time-consuming

clinical trials

online ads

Calls for design of sample-efficient RL algorithms!
Computational efficiency

Running RL algorithms might take a long time . . .

- enormous state-action space
- nonconvexity
Computational efficiency

Running RL algorithms might take a long time . . .

- enormous state-action space
- nonconvexity

Calls for computationally efficient RL algorithms!
This talk: three recent stories

(large-scale) optimization (high-dimensional) statistics

Demystify sample- and computational efficiency of RL algorithms
This talk: three recent stories

(large-scale) optimization

(high-dimensional) statistics

Demystify sample- and computational efficiency of RL algorithms

1. model-based RL
2. policy-based RL
3. value-based RL
This talk: three recent stories

(large-scale) optimization (high-dimensional) statistics

Demystify sample- and computational efficiency of RL algorithms

1. **model-based RL**: breaking a sample size barrier

2. **policy-based RL**: natural policy gradient (NPG) methods

3. **value-based RL**: Q-learning over Markovian samples
Background: Markov decision processes
Markov decision process (MDP)

- $S$: state space
- $A$: action space

$\mathbb{r}(s, a) \in [0, 1]$: immediate reward
Markov decision process (MDP)

- $S$: state space
- $A$: action space
- $r(s, a) \in [0, 1]$: immediate reward
Markov decision process (MDP)

- $S$: state space
- $A$: action space
- $r(s, a) \in [0, 1]$: immediate reward
- $\pi(\cdot | s)$: policy (or action selection rule)
**Markov decision process (MDP)**

- $S$: state space
- $A$: action space
- $r(s, a) \in [0, 1]$: immediate reward
- $\pi(\cdot | s)$: policy (or action selection rule)
- $P(\cdot | s, a)$: unknown transition probabilities
Value function

Value of policy $\pi$: long-term discounted reward

$$\forall s \in S : \quad V^{\pi}(s) := \mathbb{E}\left[ \sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \mid s_0 = s \right]$$
Value of policy \( \pi \): long-term discounted reward

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

- \((a_0, s_1, a_1, s_2, a_2, \cdots)\): generated under policy \( \pi \)
Value function

Value of policy $\pi$: long-term discounted reward

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

- $(a_0, s_1, a_1, s_2, a_2, \cdots)$: generated under policy $\pi$
- $\gamma \in [0, 1)$: discount factor
  - take $\gamma \to 1$ to approximate long-horizon MDPs
Q-function of policy \( \pi \)

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

- \((a_0, s_1, a_1, s_2, a_2, \cdots)\): generated under policy \( \pi \)
• **Optimal policy** $\pi^*$: maximizing the value function
Optimal policy and optimal values

- **Optimal policy** $\pi^*$: maximizing the value function
- **Optimal values**: $V^* \equiv V^{\pi^*}$
Story 1: breaking the sample size barrier via \textit{model-based RL} under a generative model

Gen Li
Tsinghua EE

Yuting Wei
CMU Stats

Yuejie Chi
CMU ECE

Yuantao Gu
Tsinghua EE
When the model is known . . .

Two approaches

Model-based approach (“plug-in”)
1. Build empirical estimate \( P \) for \( P \)
2. Planning based on empirical \( P \)

Model-free approach (e.g. Q-learning, SARSA)
— learning without explicitly constructing a model

\( \pi^* \)

Planning: computing the optimal policy \( \pi^* \) given MDP specification
When the model is unknown . . .

Need to learn optimal policy from samples w/o model specification
For each state-action pair \((s, a)\), collect \(N\) samples \(\{(s, a, s'_{(i)})\}_{1 \leq i \leq N}\).
Question: how many samples are sufficient to learn an $\epsilon$-optimal policy?
**Question:** how many samples are sufficient to learn an $\varepsilon$-optimal policy?

\[
\forall s: \hat{V}_\pi(s) \geq V^*(s) - \varepsilon
\]
An incomplete list of prior art

- Kearns & Singh ’99
- Kakade ’03
- Kearns, Mansour & Ng ’02
- Azar, Munos & Kappen ’12
- Azar, Munos, Ghavamzadeh & Kappen ’13
- Sidford, Wang, Wu, Yang & Ye ’18
- Sidford, Wang, Wu & Ye ’18
- Wang ’17
- Agarwal, Kakade & Yang ’19
- Wainwright ’19a
- Wainwright ’19b
- Pananjady & Wainwright ’20
- Yang & Wang ’19
- Khamaru, Pananjady, Ruan, Wainwright & Jordan ’20
- Mou, Li, Wainwright, Bartlett & Jordan ’20
- ...
## An even shorter list of prior art

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Sample Size Range</th>
<th>Sample Complexity</th>
<th>$\varepsilon$-Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>Empirical QVI</td>
<td>$\left[ \frac{</td>
<td>S</td>
<td>^2</td>
</tr>
<tr>
<td>Azar et al. '13</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sublinear randomized VI</td>
<td>$\left[ \frac{</td>
<td>S</td>
<td></td>
</tr>
<tr>
<td>Sidford et al. '18a</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Variance-reduced QVI</td>
<td>$\left[ \frac{</td>
<td>S</td>
<td></td>
</tr>
<tr>
<td>Sidford et al. '18b</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Empirical MDP + planning</td>
<td>$\left[ \frac{</td>
<td>S</td>
<td></td>
</tr>
<tr>
<td>Agarwal et al. '19</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

— see also Wainwright '19 (for estimating optimal values)
All prior theory requires sample size barrier.
All prior theory requires sample size

\[
\frac{|S||A|}{(1 - \gamma)^3}
\]

\[
\frac{|S||A|}{(1 - \gamma)^2}
\]

\[
\frac{|S||A|}{1 - \gamma}
\]

\[
\frac{1}{\varepsilon^2}
\]

sample complexity
All prior theory requires sample size \( > \frac{|S||A|}{(1 - \gamma)^2} \) sample size barrier.
Is it possible to close the gap?
Two approaches

Model-based approach ("plug-in")

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

Model-based approach ("plug-in")
1. build an empirical estimate $\hat{P}$ for $P$
2. planning based on empirical $\hat{P}$

Model-free approach
— learning w/o constructing model explicitly
Two approaches

Model-based approach ("plug-in")
1. build empirical estimate $\hat{P}$ for $P$
2. planning based on empirical $\hat{P}$

Model-free approach
— learning w/o constructing model explicitly
**Model estimation**

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

Diagram:
- A robot labeled "generative model" with input $(s, a)$ and output $P(\cdot | s, a)$ and $s'$.
Model estimation

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

**Empirical estimates:** estimate \(\hat{P}(s'|s, a)\) by \(\frac{1}{N} \sum_{i=1}^{N} 1\{s'_i = s'\}\)

\(\text{generative model}\)
Model-based (plug-in) estimator

— Azar et al. '13, Agarwal et al. '19, Pananjady et al. '20

Planning based on the empirical MDP with slightly perturbed rewards
Our method: plug-in estimator + perturbation

— Li, Wei, Chi, Gu, Chen ’20

original MDP: \((P, r)\)
empirical MDP: \((bP, r)\)
perturbed empirical MDP: \((bP, r_p)\)

planning oracle

e.g. policy iteration

MDP specification
rewards perturb

Run planning algorithms based on the \textit{empirical} MDP
Challenges in the sample-starved regime

truth: \( P \in \mathbb{R}^{|S| \times |A| \times |S|} \)

empirical estimate: \( \hat{P} \)

- Can’t recover \( P \) faithfully if sample size \( \ll |S|^2|A|! \)
Challenges in the sample-starved regime

truth: \( P \in \mathbb{R}^{|S| \times |A|} \)

Can’t recover \( P \) faithfully if sample size \( \ll |S|^2|A| \)!

Can we trust our policy estimate when reliable model estimation is infeasible?
Main result

**Theorem 1 (Li, Wei, Chi, Gu, Chen ’20)**

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

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

with sample complexity at most

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

**Theorem 1 (Li, Wei, Chi, Gu, Chen ’20)**

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

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

with sample complexity at most

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

- $\hat{\pi}_p^*$: obtained by empirical QVI or PI within $\tilde{O}\left(\frac{1}{1-\gamma}\right)$ iterations
Main result

**Theorem 1 (Li, Wei, Chi, Gu, Chen ’20)**

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

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

with sample complexity at most

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

- $\hat{\pi}^*_p$: obtained by empirical QVI or PI within $\tilde{O}\left(\frac{1}{1 - \gamma}\right)$ iterations
- **Minimax lower bound**: $\tilde{\Omega}\left(\frac{|S||A|}{(1 - \gamma)^3\varepsilon^2}\right)$ (Azar et al. ’13)
Analysis
Notation and Bellman equation

- $V^\pi$: true value function under policy $\pi$
  - Bellman equation: $V^\pi = (I - P_\pi)^{-1}r$

- $\hat{V}^\pi$: estimate of value function under policy $\pi$
  - Bellman equation: $\hat{V}^\pi = (I - \hat{P}_\pi)^{-1}r$

- $\pi^\star$: optimal policy w.r.t. true value function
- $\hat{\pi}^\star$: optimal policy w.r.t. empirical value function

- $V^\star := V^{\pi^\star}$: optimal values under true models
- $\hat{V}^\star := \hat{V}^{\hat{\pi}^\star}$: optimal values under empirical models
Notation and Bellman equation

- $V^\pi$: true value function under policy $\pi$
  - Bellman equation: $V^\pi = (I - P_\pi)^{-1}r$

- $\hat{V}^\pi$: estimate of value function under policy $\pi$
  - Bellman equation: $\hat{V}^\pi = (I - \hat{P}_\pi)^{-1}r$
Notation and Bellman equation

- $V^\pi$: true value function under policy $\pi$
  - Bellman equation: $V^\pi = (I - P_\pi)^{-1}r$

- $\hat{V}^\pi$: estimate of value function under policy $\pi$
  - Bellman equation: $\hat{V}^\pi = (I - \hat{P}_\pi)^{-1}r$

- $\pi^*$: optimal policy w.r.t. true value function

- $\hat{\pi}^*$: optimal policy w.r.t. empirical value function
Notation and Bellman equation

• $V^\pi$: true value function under policy $\pi$
  o Bellman equation: $V^\pi = (I - P_\pi)^{-1}r$

• $\hat{V}^\pi$: estimate of value function under policy $\pi$
  o Bellman equation: $\hat{V}^\pi = (I - \hat{P}_\pi)^{-1}r$

• $\pi^\star$: optimal policy w.r.t. true value function

• $\hat{\pi}^\star$: optimal policy w.r.t. empirical value function

• $V^\star := V^{\pi^\star}$: optimal values under true models

• $\hat{V}^\star := \hat{V}^{\hat{\pi}^\star}$: optimal values under empirical models
Proof ideas

Elementary decomposition:

\[ V^* - \hat{V}^{\pi*} = (V^* - \hat{V}^{\pi*}) + (\hat{V}^{\pi*} - \hat{V}^{\pi*}) + (\hat{V}^{\pi*} - V^{\pi*}) \]
Proof ideas

Elementary decomposition:

\[
V^* - V_{\hat{\pi}}^* = (V^* - \hat{V}_\pi^*) + (\hat{V}_\pi^* - \hat{V}_{\hat{\pi}}^*) + (\hat{V}_{\hat{\pi}}^* - V_{\pi}^*) \\
\leq (V_{\pi}^* - \hat{V}_{\pi}^*) + 0 + (\hat{V}_{\hat{\pi}}^* - V_{\pi}^*)
\]

- **Step 1:** control \( V_{\pi}^* - \hat{V}_{\pi}^* \) for a fixed \( \pi \)  
  ([Bernstein inequality] + [high-order decomposition])
Proof ideas

Elementary decomposition:

\[ V^* - \hat{V}^\pi^* = (V^* - \hat{V}^\pi^*) + (\hat{V}^\pi^* - \hat{V}^\hat{\pi}^*) + (\hat{V}^\hat{\pi}^* - V^\hat{\pi}^*) \]
\[ \leq (V^\pi^* - \hat{V}^\pi^*) + 0 + (\hat{V}^\hat{\pi}^* - V^\hat{\pi}^*) \]

- **Step 1:** control \( V^\pi - \hat{V}^\pi \) for a fixed \( \pi \)
  
  (Bernstein inequality + high-order decomposition)

- **Step 2:** extend it to control \( \hat{V}^\hat{\pi}^* - V^\hat{\pi}^* \) (\( \hat{\pi}^* \) depends on samples)
  
  (decouple statistical dependency)
Step 1: improved theory for policy evaluation

Model-based policy evaluation:

— given a fixed policy \( \pi \), estimate \( V^\pi \) via the plug-in estimate \( \hat{V}^\pi \)
Step 1: improved theory for policy evaluation

Model-based policy evaluation:
— given a fixed policy $\pi$, estimate $V^\pi$ via the plug-in estimate $\hat{V}^\pi$

- A sample size barrier $\frac{|S|}{(1-\gamma)^2}$ already appeared in prior work
  (Agarwal et al. ’19, Pananjady & Wainwright ’19, Khamaru et al. ’20)
Step 1: improved theory for policy evaluation

Model-based policy evaluation:
— given a fixed policy $\pi$, estimate $V^\pi$ via the plug-in estimate $\hat{V}^\pi$

Theorem 2 (Li, Wei, Chi, Gu, Chen’20)

Fix any policy $\pi$. For $0 < \varepsilon \leq \frac{1}{1-\gamma}$, the plug-in estimator $\hat{V}^\pi$ obeys

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

with sample complexity at most

$$\tilde{O}\left(\frac{|S|}{(1-\gamma)^3\varepsilon^2}\right)$$
Step 1: improved theory for policy evaluation

Model-based policy evaluation:

— given a fixed policy $\pi$, estimate $V^\pi$ via the plug-in estimate $\hat{V}^\pi$

Theorem 2 (Li, Wei, Chi, Gu, Chen’20)

Fix any policy $\pi$. For $0 < \varepsilon \leq \frac{1}{1-\gamma}$, the plug-in estimator $\hat{V}^\pi$ obeys

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

with sample complexity at most

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

- Minimax optimal for all $\varepsilon$ (Azar et al.’13, Pananjady & Wainwright ’19)
Key idea 1: a peeling argument

Agarwal, Kakade, Yang 19: first-order expansion

\[ \hat{V}^\pi - V^\pi = \gamma (I - \gamma P_\pi)^{-1} (\hat{P}_\pi - P_\pi) \hat{V}^\pi \] (\star)

Ours: higher-order expansion \rightarrow tighter control

\[ \hat{V}^\pi - V^\pi = \gamma (I - \gamma P_\pi)^{-1} (\hat{P}_\pi - P_\pi) \hat{V}^\pi + \]
Key idea 1: a peeling argument

Agarwal, Kakade, Yang 19: first-order expansion

\[ \hat{V}^\pi - V^\pi = \gamma(I - \gamma P_\pi)^{-1}(\hat{P}_\pi - P_\pi)\hat{V}^\pi \]  

Ours: higher-order expansion \rightarrow tighter control

\[ \hat{V}^\pi - V^\pi = \gamma(I - \gamma P_\pi)^{-1}(\hat{P}_\pi - P_\pi)V^\pi + \] 
\[ + \gamma(I - \gamma P_\pi)^{-1}(\hat{P}_\pi - P_\pi)(\hat{V}^\pi - V^\pi) \]
Key idea 1: a peeling argument

Agarwal, Kakade, Yang 19: first-order expansion

\[ \hat{V}^\pi - V^\pi = \gamma (I - \gamma P_\pi)^{-1} (\hat{P}_\pi - P_\pi) \hat{V}^\pi \]  

(\star)

Ours: higher-order expansion \rightarrow tighter control

\[ \hat{V}^\pi - V^\pi = \gamma (I - \gamma P_\pi)^{-1} (\hat{P}_\pi - P_\pi) V^\pi + \]

\[ + \gamma^2 \left((I - \gamma P_\pi)^{-1} (\hat{P}_\pi - P_\pi)\right)^2 V^\pi \]

\[ + \gamma^3 \left((I - \gamma P_\pi)^{-1} (\hat{P}_\pi - P_\pi)\right)^3 V^\pi \]

+ ...
Step 2: controlling $\hat{V}^{\pi^*} - \hat{V}^{\pi^*}$

A natural idea: apply our policy evaluation theory + union bound
Step 2: controlling $\hat{V}^{\pi^*} - \hat{V}^{\pi^*}$

A natural idea: apply our policy evaluation theory + union bound

- highly suboptimal! (there are exponentially many policies)
Key idea 2: 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 2: leave-one-out analysis

- Stein ’72
- El Karoui, Bean, Bickel, Lim, Yu ’13
- El Karoui ’15
- Javanmard, Montanari ’15
- Zhong, Boumal ’17
- Lei, Bickel, El Karoui ’17
- Sur, Chen, Candès ’17
- Abbe, Fan, Wang, Zhong ’17
- Chen, Fan, Ma, Wang ’17
- Ma, Wang, Chi, Chen ’17
- Chen, Chi, Fan, Ma ’18
- Ding, Chen ’18
- Dong, Shi ’18
- Chen, Chi, Fan, Ma, Yan ’19
- Chen, Fan, Ma, Yan ’19
- Cai, Li, Poor, Chen ’19
- Agarwal, Kakade, Yang ’19
- Pananjady, Wainwright ’19
- Ling ’20
Key idea 2: 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 2: 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
Key idea 2: 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
3. $\hat{\pi}^*$ can be determined by this $\epsilon$-net under separation condition

$$\forall s \in S, \quad \hat{Q}^*(s, \hat{\pi}^*(s)) - \max_{a: a \neq \hat{\pi}^*(s)} \hat{Q}^*(s, a) > 0$$
Key idea 2: leave-one-out analysis

Our decoupling argument vs. Agarwal, Kakade, Yang ’19

- *Agarwal et al. ’19*: dependency btw value $\hat{V}$ & samples
- *Ours*: dependency btw policy $\hat{\pi}$ & samples
Key idea 3: tie-breaking via perturbation

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

\[
\forall s \in S, \quad \hat{Q}^*(s, \hat{\pi}^*(s)) - \max_{a: a \neq \hat{\pi}^*(s)} \hat{Q}^*(s, a) > 0
\]
Key idea 3: tie-breaking via perturbation

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

\[ \forall s \in \mathcal{S}, \quad \hat{Q}^*(s, \hat{\pi}^*(s)) - \max_{a: a \neq \hat{\pi}^*(s)} \hat{Q}^*(s, a) > 0 \]

- **Solution:** *slightly perturb rewards* \( r \implies \hat{\pi}_p^* 
  
  - ensures \( \hat{\pi}_p^* \) can be differentiated from others
  
  - \( V_{\pi_p}^* \approx V_{\pi}^* \)
Key idea 3: tie-breaking via perturbation

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

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

- **Solution:** *slightly* perturb rewards $r \implies \hat{\pi}^*_p$
  - ensures $\hat{\pi}^*_p$ can be differentiated from others
  - $V_{\hat{\pi}^*} \approx V_{\pi^*}$
Model-based RL is minimax optimal and does not suffer from a sample size barrier!
Summary

Model-based RL is minimax optimal and does not suffer from a sample size barrier!

future directions

- finite-horizon episodic MDPs
- Markov games
Story 2: fast global convergence of entropy-regularized natural policy gradient (NPG) methods

Shicong Cen
CMU ECE

Chen Cheng
Stanford Stats

Yuting Wei
CMU Stats

Yuejie Chi
CMU ECE
Policy optimization: a major contributor to these successes
Policy gradient (PG) methods

Given initial state distribution $s \sim \rho$:

$$\text{maximize}_\pi \quad V^\pi(\rho) := \mathbb{E}_{s \sim \rho} [V^\pi(s)]$$

PG method (Sutton et al. '00)

$$\theta(t+1) = \theta(t) + \eta \nabla_\theta V^\pi(\rho), \quad t = 0, 1, \cdots$$

• $\eta$: learning rate
Policy gradient (PG) methods

Given initial state distribution $s \sim \rho$:

$$\text{maximize}_{\pi} \quad V^\pi(\rho) := \mathbb{E}_{s \sim \rho} [V^\pi(s)]$$

softmax parameterization:

$$\pi_\theta(a|s) = \frac{\exp(\theta(s, a))}{\sum_a \exp(\theta(s, a))}$$

PG method (Sutton et al. '00)

$$\theta(t+1) = \theta(t) + \eta \nabla_\theta V^\pi_\theta(\rho), \quad t = 0, 1, \ldots$$

$\eta$: learning rate
Policy gradient (PG) methods

Given initial state distribution $s \sim \rho$:

$$\text{maximize}_\pi \quad V^\pi(\rho) := \mathbb{E}_{s \sim \rho} [V^\pi(s)]$$

softmax parameterization:

$$\pi_\theta(a|s) = \frac{\exp(\theta(s, a))}{\sum_a \exp(\theta(s, a))}$$

$$\text{maximize}_\theta \quad V^{\pi_\theta}(\rho) := \mathbb{E}_{s \sim \rho} [V^{\pi_\theta}(s)]$$

PG method (Sutton et al. '00)

$$\theta(t+1) = \theta(t) + \eta \nabla_{\theta} V_\pi^{\pi_\theta}(\rho), \quad t = 0, 1, \ldots$$

$\eta$: learning rate
Policy gradient (PG) methods

Given initial state distribution \( s \sim \rho \):

\[
\text{maximize}_\pi \quad V^\pi(\rho) := \mathbb{E}_{s \sim \rho}[V^\pi(s)]
\]

softmax parameterization:

\[
\pi_\theta(a|s) = \frac{\exp(\theta(s, a))}{\sum_a \exp(\theta(s, a))}
\]

\[
\text{maximize}_\theta \quad V^{\pi_\theta}(\rho) := \mathbb{E}_{s \sim \rho}[V^{\pi_\theta}(s)]
\]

PG method (Sutton et al. '00)

\[
\theta^{(t+1)} = \theta^{(t)} + \eta \nabla_\theta V^{\pi_\theta^{(t)}}(\rho), \quad t = 0, 1, \ldots
\]

- \( \eta \): learning rate
Booster 1: natural policy gradient (NPG)

precondition gradients to improve search directions ...

\[ \theta(t+1) = \theta(t) + \eta (\mathcal{F}_\rho^\theta)^\dagger \nabla_\theta V^{\pi_\theta}_\theta (\rho), \quad t = 0, 1, \ldots \]

- \( \mathcal{F}_\rho^\theta := \mathbb{E} \left[ (\nabla_\theta \log \pi_\theta(a|s)) (\nabla_\theta \log \pi_\theta(a|s))^\top \right] \): Fisher info matrix
Booster 2: entropy regularization

accelerate convergence by regularizing objective function

\[ V_\tau^\pi(s_0) := \mathbb{E} \left[ \sum_{t=0}^{\infty} \gamma^t (r_t - \tau \log \pi(a_t|s_t)) \bigg| s_0 \right] \]

\[ = V_0^\pi(s) + \frac{\tau}{1 - \gamma} \mathbb{E}_{s \sim d_s^\pi} \left[ - \sum_a \pi(a|s) \log \pi(a|s) \bigg| s_0 \right] \]

- \( \tau \): regularization parameter
- \( d_s^\pi \): discounted state visitation distribution
Booster 2: entropy regularization

accelerate convergence by regularizing objective function

\[
V_\tau^\pi (s_0) := \mathbb{E} \left[ \sum_{t=0}^{\infty} \gamma^t (r_t - \tau \log \pi (a_t | s_t)) \right] | s_0
\]

\[
= V^\pi (s) + \frac{\tau}{1 - \gamma} \mathbb{E}_{s \sim d^\pi_s} \left[ - \sum_a \pi(a | s) \log \pi(a | s) \right] | s_0
\]

- \( \tau \): regularization parameter
- \( d^\pi_s \): discounted state visitation distribution

entropy-regularized value maximization

\[
\text{maximize}_\theta \quad V_\tau^\pi_\theta (\rho) := \mathbb{E}_{s \sim \rho} [V_\tau^\pi_\theta (s)] \quad \text{("soft" value function)}
\]
Entropy-regularized natural gradient helps!

A toy bandit example: 3 arms with rewards 1, 0.9 and 0.1
Challenge: non-concavity
Challenge: non-concavity

Recent advances

- PG for control (Fazel et al., 2018; Bhandari and Russo, 2019)
- PG for tabular MDPs (Agarwal et al. 19, Bhandari and Russo ’19, Mei et al ’20)
- unregularized NPG for tabular MDPs (Agarwal et al. ’19, Bhandari and Russo ’20)
- ...
This work: understanding entropy-regularized NPG methods in tabular settings
Entropy-regularized NPG in tabular settings

An alternative expression in policy space (tabular setting)

\[
\pi^{(t+1)}(a|s) \propto \pi^{(t)}(a|s)^{1 - \frac{\eta \tau}{1 - \gamma}} \exp \left( \frac{\eta Q^{(t)}(s, a)}{1 - \gamma} \right), \quad t = 0, 1, \ldots
\]

- \( Q^{(t)}_\tau \): soft Q-function of \( \pi^{(t)} \); \( 0 < \eta \leq \frac{1 - \gamma}{\tau} \): learning rate

- invariant to the choice of initial state distribution \( \rho \)
Linear convergence with exact gradients

optimal policy: $\pi_\tau^*$; optimal “soft” $Q$ function: $Q_\tau^* := Q_{\pi_\tau^*}^\tau$

Exact oracle: perfect gradient evaluation

**Theorem 3 (Cen, Cheng, Chen, Wei, Chi ’20)**

For any $0 < \eta \leq (1 - \gamma)/\tau$, entropy-regularized NPG achieves

$$\|Q_\tau^* - Q_\tau^{(t+1)}\|_\infty \leq C_1 \gamma (1 - \eta \tau)^t, \quad t = 0, 1, \cdots$$

• $C_1 = \|Q_\tau^* - Q_\tau^{(0)}\|_\infty + 2\tau \left(1 - \frac{\eta \tau}{1 - \gamma}\right) \|\log \pi_\tau^* - \log \pi^{(0)}\|_\infty$
Implications: iteration complexity

number of iterations needed to reach \( \| Q^\tau - Q^{(t+1)}_{\tau} \|_\infty \leq \varepsilon \) is at most

- **General learning rates** \((0 < \eta < \frac{1-\gamma}{\tau})\):
  \[
  \frac{1}{\eta \tau} \log \left( \frac{C_1 \gamma}{\varepsilon} \right)
  \]

- **Soft policy iteration** \((\eta = \frac{1-\gamma}{\tau})\):
  \[
  \frac{1}{1-\gamma} \log \left( \frac{\| Q^\tau - Q^{(0)}_{\tau} \|_\infty \gamma}{\varepsilon} \right)
  \]
Implications: iteration complexity

number of iterations needed to reach $\|Q^*_\tau - Q^{(t+1)}_{\tau}\|_\infty \leq \varepsilon$ is at most

- **General learning rates** ($0 < \eta < \frac{1-\gamma}{\tau}$):

  $$\frac{1}{\eta^\gamma} \log \left( \frac{C_1\gamma}{\varepsilon} \right)$$

- **Soft policy iteration** ($\eta = \frac{1-\gamma}{\tau}$):

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

Nearly **dimension-free** global linear convergence!
Regularized NPG vs. unregularized NPG

**regularized NPG**

\[ \tau = 0.001 \]

<table>
<thead>
<tr>
<th>( \eta )</th>
<th>( Q^* - Q(t) )</th>
<th>#iterations</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.01</td>
<td>10^{-12}</td>
<td>0</td>
</tr>
<tr>
<td>0.1</td>
<td>10^{-6}</td>
<td>1000</td>
</tr>
<tr>
<td>1</td>
<td>10^{-8}</td>
<td>2000</td>
</tr>
</tbody>
</table>

**unregularized NPG**

\[ \tau = 0 \]

<table>
<thead>
<tr>
<th>( \eta )</th>
<th>( Q^* - Q(t) )</th>
<th>#iterations</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.01</td>
<td>10^{-12}</td>
<td>0</td>
</tr>
<tr>
<td>0.1</td>
<td>10^{-8}</td>
<td>1000</td>
</tr>
<tr>
<td>1</td>
<td>10^{-10}</td>
<td>2000</td>
</tr>
</tbody>
</table>

**linear rate:** \( \frac{1}{\eta \tau} \log \left( \frac{1}{\epsilon} \right) \)

**sublinear rate:** \( \frac{1}{\min\{\eta, (1-\gamma)^2\}} \epsilon \)

*Ours* (Agarwal et al. ’19)

Entropy regularization enables faster convergence!
Regularized NPG vs. unregularized NPG

regularized NPG
\( \tau = 0.001 \)

unregularized NPG
\( \tau = 0 \)

linear rate: \( \frac{1}{\eta \tau} \log \left( \frac{1}{\varepsilon} \right) \)

sublinear rate: \( \frac{1}{\min\{\eta, (1-\gamma)^2\} \varepsilon} \)

Ours

(Agarwal et al.'19)

Entropy regularization enables faster convergence!
Returning to the original MDP?

How to employ entropy-regularized NPG to find an $\varepsilon$-optimal policy for the original (unregularized) MDP?

- suffices to find an $\frac{\varepsilon}{2}$-optimal policy of regularized MDP with regularization parameter $\tau = \frac{(1-\gamma)\varepsilon}{4\log|A|}$

- iteration complexity is the same as before (up to log factor)
Entropy-regularized NPG with inexact gradients

**Inexact oracle:** inexact evaluation of $Q_{\tau}^{(t)}$, which returns $\hat{Q}_{\tau}^{(t)}$ s.t.

$$\|\hat{Q}_{\tau}^{(t)} - Q_{\tau}^{(t)}\|_{\infty} \leq \delta,$$

e.g. using sample-based estimators
Entropy-regularized NPG with inexact gradients

Inexact oracle: inexact evaluation of $Q^{(t)}_\tau$, which returns $\hat{Q}^{(t)}_\tau$ s.t.

$$\|\hat{Q}^{(t)}_\tau - Q^{(t)}_\tau\|_\infty \leq \delta,$$

e.g. using sample-based estimators

Inexact entropy-regularized NPG:

$$\pi^{(t+1)}(a|s) \propto (\pi^{(t)}(a|s))^{1-\frac{\eta}{1-\gamma}} \exp\left(\frac{\eta \hat{Q}^{(t)}_\tau(s, a)}{1 - \gamma}\right)$$
Entropy-regularized NPG with inexact gradients

**Inexact oracle:** inexact evaluation of $Q^{(t)}_\tau$, which returns $\hat{Q}^{(t)}\tau$ s.t.

$$\|\hat{Q}^{(t)}_\tau - Q^{(t)}_\tau\|_\infty \leq \delta,$$

e.g. using sample-based estimators

**Inexact entropy-regularized NPG:**

$$\pi^{(t+1)}(a|s) \propto (\pi^{(t)}(a|s))^{1 - \frac{\eta\tau}{1 - \gamma}} \exp\left(\frac{\eta\hat{Q}^{(t)}_\tau(s, a)}{1 - \gamma}\right)$$

**Question:** stability vis-à-vis inexact gradient evaluation?
Linear convergence with inexact gradients

\[ \| \hat{Q}_\tau^{(t)} - Q_\tau^{(t)} \|_\infty \leq \delta \]

**Theorem 4 (Cen, Cheng, Chen, Wei, Chi ’20)**

For any stepsize \( 0 < \eta \leq (1 - \gamma)/\tau \), entropy-regularized NPG attains

\[ \| Q_\tau^* - Q_\tau^{(t+1)} \|_\infty \leq \gamma (1 - \eta \tau)^t C_1 + C_2 \]

- \( C_1 = \| Q_\tau^* - Q_\tau^{(0)} \|_\infty + 2\tau \left( 1 - \frac{\eta \tau}{1 - \gamma} \right) \| \log \pi_\tau^* - \log \pi^{(0)}_\tau \|_\infty \)
- \( C_2 = \frac{2\gamma (1 + \frac{\gamma}{\eta \tau})}{1 - \gamma} \delta : \text{error floor} \)

- converges linearly at the same rate until an error floor is hit
A little analysis when $\eta = \frac{1-\gamma}{\tau}$
A key lemma: monotonic performance improvement

\[ V^{(t+1)}(\rho) - V^{(t)}(\rho) = \mathbb{E}_{s \sim d^{(t+1)}_\rho} \left[ \left( \frac{1}{\eta} - \frac{\tau}{1 - \gamma} \right) \text{KL} \left( \pi^{(t+1)}(\cdot|s) \| \pi^{(t)}(\cdot|s) \right) \right] \]

\[ + \frac{1}{\eta} \text{KL} \left( \pi^{(t)}(\cdot|s) \| \pi^{(t+1)}(\cdot|s) \right) \]
A key lemma: monotonic performance improvement

\[ V^{(t+1)}_\tau(\rho) - V^{(t)}_\tau(\rho) = \mathbb{E}_{s \sim d^{(t+1)}_\rho} \left[ \left( \frac{1}{\eta} - \frac{\tau}{1 - \gamma} \right) \text{KL} \left( \pi^{(t+1)}(\cdot|s) \middle\| \pi^{(t)}(\cdot|s) \right) \right] \\
+ \frac{1}{\eta} \text{KL} \left( \pi^{(t)}(\cdot|s) \middle\| \pi^{(t+1)}(\cdot|s) \right) \geq 0 \quad \text{(if } 0 < \eta \leq \frac{1 - \gamma}{\tau} \text{)} \]
"Soft" Bellman operator

\[ T_\tau(Q)(s, a) := r(s, a) \]

\[ + \gamma \mathbb{E}_{s' \sim P(\cdot | s, a)} \left[ \max_{\pi(\cdot | s')} \mathbb{E}_{a' \sim \pi(\cdot | s')} \left[ Q(s', a') - \tau \log \pi(a' | s') \right] \right] \]

immediate reward

next state's value

regularizer

\[ \text{Soft Bellman equation: } Q^\star \tau \text{ is the unique solution to } T_\tau(Q) = Q \]
“Soft” Bellman operator

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

Soft Bellman equation: \( Q^*_\tau \) is the unique solution to

\[
\mathcal{T}_\tau(Q) = Q
\]

\( \gamma \)-contraction of soft Bellman operator:

\[
\| \mathcal{T}_\tau(Q_1) - \mathcal{T}_\tau(Q_2) \|_\infty \leq \gamma \| Q_1 - Q_2 \|_\infty
\]
policy iteration

\[ \pi^{(0)} \xrightarrow{\text{evaluate}} Q_\pi^{(0)} \]
\[ \pi^{(1)} \xrightarrow{\text{greedy}} Q_\pi^{(1)} \]
\[ \pi^{(2)} \xrightarrow{\text{greedy}} Q_\pi^{(2)} \]
\[ \vdots \]
\[ Q^* \]
\[ \pi^* \]

Bellman operator
policy iteration

\[ \pi^{(0)}, \pi^{(1)}, \pi^{(2)}, \ldots \]

\[ \begin{align*}
\pi^{(0)} & \xrightarrow{\text{evaluate}} Q^{\pi^{(0)}} \\
\pi^{(1)} & \xrightarrow{\text{evaluate}} Q^{\pi^{(1)}} \\
\pi^{(2)} & \xrightarrow{\text{greedy}} Q^{\pi^{(1)}} \\
\vdots & \\
\pi^{*} & \xrightarrow{} Q^{*}
\end{align*} \]

Bellman operator

soft policy iteration \( (\eta = \frac{1-\gamma}{\tau}) \)

\[ \pi^{(0)}, \pi^{(1)}, \pi^{(2)}, \ldots \]

\[ \begin{align*}
\pi^{(0)} & \xrightarrow{\text{evaluate}} Q^{\pi^{(0)}}^\tau \\
\pi^{(1)} & \xrightarrow{\text{evaluate}} Q^{\pi^{(1)}}^\tau \\
\pi^{(2)} & \xrightarrow{\text{soft greedy}} Q^{\pi^{(1)}}^\tau \\
\vdots & \\
\pi^{\tau} & \xrightarrow{} Q^{\pi^\tau}
\end{align*} \]

soft Bellman operator
Global linear convergence of entropy-regularized NPG methods for tabular discounted MDPs

future directions:

- function approximation
- sample complexities
- soft actor-critic algorithms
Story 3: sample complexity of 
(asynchronous) Q-learning on Markovian samples

Gen Li
Tsinghua EE

Yuting Wei
CMU Stats

Yuejie Chi
CMU ECE

Yuantao Gu
Tsinghua EE
Model-based vs. model-free RL

Model-based approach ("plug-in")
1. build an empirical estimate $\hat{P}$ for $P$
2. planning based on empirical $\hat{P}$

Model-free approach
— learning w/o modeling & estimating environment explicitly
A classical example: **Q-learning** on Markovian samples
Markovian samples and behavior policy

\textbf{Observed:} \( \{s_t, a_t, r_t\}_{t \geq 0} \) generated by behavior policy \( \pi_b \)

\textbf{Markovian trajectory}

\textbf{Goal:} learn optimal value \( V^* \) and \( Q^* \) based on sample trajectory
Markovian samples and behavior policy

Observed:

\[
\{s_t, a_t, r_t\} \quad t \geq 0
\]

Markovian trajectory generated by behavior policy \(\pi_b\).

Goal: learn optimal value \(V^*\) and \(Q^*\) based on sample trajectory.

Key quantities of sample trajectory:

- Minimum state-action occupancy probability:
  \[
  \mu_{\text{min}} := \min_{s, a} \mu_{\pi_b}(s, a)
  \]
- Mixing time: \(t_{\text{mix}}\)
Q-learning: a classical model-free algorithm

Stochastic approximation for solving **Bellman equation** \( Q = \mathcal{T}(Q) \)

Robbins & Monro '51
Aside: Bellman optimality principle

Bellman operator

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

- one-step look-ahead
Aside: Bellman optimality principle

Bellman operator

\[ T(Q)(s, a) := r(s, a) + \gamma \sum_{s' \sim P(\cdot | s, a)} \mathbb{E} \left[ \max_{a' \in A} Q(s', a') \right] \]

- one-step look-ahead

Bellman equation: \( Q^* \) is unique solution to

\[ T(Q^*) = Q^* \]

Richard Bellman
Q-learning: a classical model-free algorithm

Chris Watkins  Peter Dayan

Stochastic approximation for solving Bellman equation \( Q = T(Q) \)

\[
Q_{t+1}(s_t, a_t) = (1 - \eta_t)Q_t(s_t, a_t) + \eta_t T_t(Q_t)(s_t, a_t), \quad t \geq 0
\]

*only update \((s_t, a_t)\)-th entry*
Q-learning: a classical model-free algorithm

Chris Watkins  Peter Dayan

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

\[
Q_{t+1}(s_t, a_t) = (1 - \eta_t)Q_t(s_t, a_t) + \eta_t \mathcal{T}_t(Q_t)(s_t, a_t), \quad t \geq 0
\]

only update \((s_t, a_t)\)-th entry

\[
\mathcal{T}_t(Q)(s_t, a_t) = r(s_t, a_t) + \gamma \max_{a'} Q(s_{t+1}, a')
\]

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

- **asynchronous**: only a single entry is updated each iteration

\[
\begin{align*}
\text{observed:} & \quad s_0 \rightarrow s_1 \rightarrow s_2 \rightarrow s_3 \rightarrow s_4 \rightarrow s_5 \\
& \quad a_0 \rightarrow a_1 \rightarrow a_2 \rightarrow a_3 \rightarrow a_4 \rightarrow a_5
\end{align*}
\]
Q-learning on Markovian samples

- asynchronous: only a single entry is updated each iteration
  - resembles Markov-chain coordinate descent
Q-learning on Markovian samples

- **asynchronous**: only a single entry is updated each iteration
  - resembles Markov-chain coordinate descent

- **off-policy**: target policy $\pi^* \neq$ behavior policy $\pi_b$
A highly incomplete list of prior work

- Watkins, Dayan ’92
- Tsitsiklis ’94
- Jaakkola, Jordan, Singh ’94
- Szepesvári ’98
- Kearns, Singh ’99
- Borkar, Meyn ’00
- Even-Dar, Mansour ’03
- Beck, Srikant ’12
- Chi, Zhu, Bubeck, Jordan ’18
- Shah, Xie ’18
- Lee, He ’18
- Wainwright ’19
- Chen, Zhang, Doan, Maguluri, Clarke ’19
- Yang, Wang ’19
- Du, Lee, Mahajan, Wang ’20
- Chen, Maguluri, Shakkottai, Shanmugam ’20
- Qu, Wierman ’20
- Devraj, Meyn ’20
- Weng, Gupta, He, Ying, Srikant ’20
- ...

67/79
What is sample complexity of (async) Q-learning?
**Prior art: async Q-learning**

**Question:** how many samples are needed to ensure $\| \hat{Q} - Q^* \|_\infty \leq \varepsilon$?

<table>
<thead>
<tr>
<th>paper</th>
<th>sample complexity</th>
<th>learning rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Even-Dar &amp; Mansour ’03</td>
<td>$\frac{(t_{\text{cover}})\frac{1-\gamma}{(1-\gamma)^4\varepsilon^2}}{t}$</td>
<td>linear: $\frac{1}{t}$</td>
</tr>
<tr>
<td>Even-Dar &amp; Mansour ’03</td>
<td>$\left( \frac{t_{\text{cover}}^{1+3\omega}}{(1-\gamma)^4\varepsilon^2} \right)^{\frac{1}{\omega}} + \left( \frac{t_{\text{cover}}}{1-\gamma} \right)^{\frac{1}{1-\omega}}$</td>
<td>poly: $\frac{1}{t^\omega}$, $\omega \in \left(\frac{1}{2}, 1\right)$</td>
</tr>
<tr>
<td>Beck &amp; Srikant ’12</td>
<td>$\frac{t_{\text{cover}}^3</td>
<td>S</td>
</tr>
<tr>
<td>Qu &amp; Wierman ’20</td>
<td>$\frac{t_{\text{mix}}}{\mu_{\text{min}}^2(1-\gamma)^5\varepsilon^2}$</td>
<td>rescaled linear</td>
</tr>
</tbody>
</table>
Prior art: async Q-learning

**Question:** how many samples are needed to ensure $\|\hat{Q} - Q^*\|_\infty \leq \varepsilon$?

![Graph showing sample complexity and function growth](image)

- Even-Dar & Mansour '03
- Beck & Srikant '12
- Qu & Wierman '20

If we take $\mu_{\text{min}} \asymp \frac{1}{|S||A|}$, $t_{\text{cover}} \asymp \frac{t_{\text{mix}}}{\mu_{\text{min}}}$
Question: how many samples are needed to ensure $\|\hat{Q} - Q^*\|_\infty \leq \varepsilon$?

All prior results require sample size of at least $t_{\text{mix}}|S|^2|A|^2$!
Question: how many samples are needed to ensure $\|\hat{Q} - Q^*\|_\infty \leq \varepsilon$?
Main result: $\ell_\infty$-based sample complexity

Theorem 5 (Li, Wei, Chi, Gu, Chen ’20)

For any $0 < \varepsilon \leq \frac{1}{1-\gamma}$, sample complexity of async Q-learning to yield $\|\hat{Q} - Q^*\|_\infty \leq \varepsilon$ is at most (up to some log factor)

$$\frac{1}{\mu_{\min}(1-\gamma)^5\varepsilon^2} + \frac{t_{\text{mix}}}{\mu_{\min}(1-\gamma)}$$
Main result: $\ell_\infty$-based sample complexity

Theorem 5 (Li, Wei, Chi, Gu, Chen ’20)

For any $0 < \varepsilon \leq \frac{1}{1-\gamma}$, sample complexity of async Q-learning to yield $\|\hat{Q} - Q^*\|_\infty \leq \varepsilon$ is at most (up to some log factor)

$$\frac{1}{\mu_{\min}(1-\gamma)^5\varepsilon^2} + \frac{t_{\text{mix}}}{\mu_{\min}(1-\gamma)}$$

• Improves upon prior art by at least $|S||A|$

— prior art: $\frac{t_{\text{mix}}}{\mu_{\min}^2(1-\gamma)^5\varepsilon^2}$ (Qu & Wierman ’20)
Effect of mixing time on sample complexity

\[ \frac{1}{\mu_{\text{min}}(1 - \gamma)^5 \varepsilon^2} + \frac{t_{\text{mix}}}{\mu_{\text{min}}(1 - \gamma)} \]

- reflects cost taken to reach steady state
- one-time expense (almost independent of $\varepsilon$)
  — it becomes amortized as algorithm runs
Effect of mixing time on sample complexity

\[ \frac{1}{\mu_{\text{min}} (1 - \gamma)^5 \varepsilon^2} + \frac{t_{\text{mix}}}{\mu_{\text{min}} (1 - \gamma)} \]

- reflects cost taken to reach steady state
- one-time expense (almost independent of \( \varepsilon \))
  --- it becomes amortized as algorithm runs

--- prior art: \( \frac{t_{\text{mix}}}{\mu_{\text{min}}^2 (1 - \gamma)^5 \varepsilon^2} \) (Qu & Wierman '20)
Learning rates

Our choice: constant stepsize \( \eta_t \equiv \min \left\{ \frac{(1-\gamma)^4 \varepsilon^2}{\gamma^2}, \frac{1}{t_{mix}} \right\} \)

- Qu & Wierman ‘20: rescaled linear \( \eta_t = \frac{1}{t + \max\left\{ \frac{1}{\mu_{\min}(1-\gamma)}, t_{mix} \right\}} \)

- Beck & Srikant ‘12: constant \( \eta_t \equiv \frac{(1 - \gamma)^4 \varepsilon^2}{|S||A|^{2\text{cover}} t^2} \) too conservative

- Even-Dar & Mansour ’03: polynomial \( \eta_t = t^{-\omega} \) (\( \omega \in (\frac{1}{2}, 1] \))
Minimax lower bound

minimax lower bound
(Azar et al. ’13)

\[
\frac{1}{\mu_{\min}(1 - \gamma)^3 \varepsilon^2}
\]

asyn Q-learning
(ignoring dependency on \(t_{\text{mix}}\))

\[
\frac{1}{\mu_{\min}(1 - \gamma)^5 \varepsilon^2}
\]
Minimax lower bound

minimax lower bound
(Azar et al.'13)

\[
\frac{1}{\mu_{\text{min}}(1 - \gamma)^3 \varepsilon^2}
\]

asyn Q-learning
(ignoring dependency on \( t_{\text{mix}} \))

\[
\frac{1}{\mu_{\text{min}}(1 - \gamma)^5 \varepsilon^2}
\]

Can we improve dependency on \textbf{discount complexity} \( \frac{1}{1 - \gamma} \)?
One strategy: variance reduction

— inspired by Johnson & Zhang ’13, Wainwright ’19

Variance-reduced Q-learning updates

\[ Q_t(s_t, a_t) = (1 - \eta)Q_{t-1}(s_t, a_t) + \eta \left( \mathcal{T}_t(Q_{t-1}) - \mathcal{T}_t(\overline{Q}) + \tilde{T}(\overline{Q}) \right)(s_t, a_t) \]

use \( \overline{Q} \) to help reduce variability

- \( \overline{Q} \): some reference Q-estimate
- \( \tilde{T} \): empirical Bellman operator (using a batch of samples)
Variance-reduced Q-learning

— inspired by Johnson & Zhang ’13, Sidford et al. ’18, Wainwright ’19

for each epoch

1. update $\overline{Q}$ and $\tilde{T}(\overline{Q})$
2. run variance-reduced Q-learning updates
Main result: $\ell_\infty$-based sample complexity

Theorem 6 (Li, Wei, Chi, Gu, Chen ’20)

For any $0 < \varepsilon \leq 1$, sample complexity for (async) variance-reduced Q-learning to yield $\|\hat{Q} - Q^*\|_\infty \leq \varepsilon$ is at most on the order of

$$\frac{1}{\mu_{\min}(1 - \gamma)^3 \varepsilon^2} + \frac{t_{\text{mix}}}{\mu_{\min}(1 - \gamma)}$$

- more aggressive learning rates: $\eta_t \equiv \min \left\{ \frac{(1 - \gamma)^4 (1 - \gamma)^2}{\gamma^2}, \frac{1}{t_{\text{mix}}} \right\}$
Main result: $\ell_\infty$-based sample complexity

Theorem 6 (Li, Wei, Chi, Gu, Chen ’20)

For any $0 < \varepsilon \leq 1$, sample complexity for (async) variance-reduced $Q$-learning to yield $\|\hat{Q} - Q^*\|_\infty \leq \varepsilon$ is at most on the order of

$$\frac{1}{\mu_{\min}(1 - \gamma)^3\varepsilon^2} + \frac{t_{\text{mix}}}{\mu_{\min}(1 - \gamma)}$$

- more aggressive learning rates: $\eta_t \equiv \min \left\{ \frac{(1 - \gamma)^4(1 - \gamma)^2}{\gamma^2}, \frac{1}{t_{\text{mix}}} \right\}$
- minimax-optimal for $0 < \varepsilon \leq 1$
Summary

Sharpens finite-sample understanding of Q-learning on Markovian data
Summary

Sharpens finite-sample understanding of Q-learning on Markovian data

future directions

• function approximation
• on-policy algorithms like SARSA
• general Markov-chain-based optimization algorithms
Concluding remarks

Understanding RL requires modern statistics and optimization
Concluding remarks

Understanding RL requires modern statistics and optimization

future directions

• beyond tabular settings
• finite-horizon episodic MDPs
• multi-agent RL (e.g. Markov games)
• ...
Papers:

“Breaking the sample size barrier in model-based reinforcement learning with a generative model,” G. Li, Y. Wei, Y. Chi, Y. Gu, Y. Chen, NeurIPS, 2020

“Sample complexity of asynchronous Q-learning: Sharper analysis and variance reduction,” G. Li, Y. Wei, Y. Chi, Y. Gu, Y. Chen, NeurIPS 2020