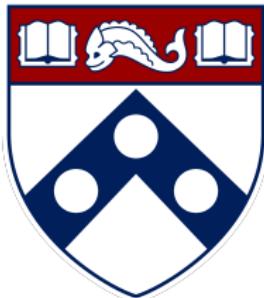


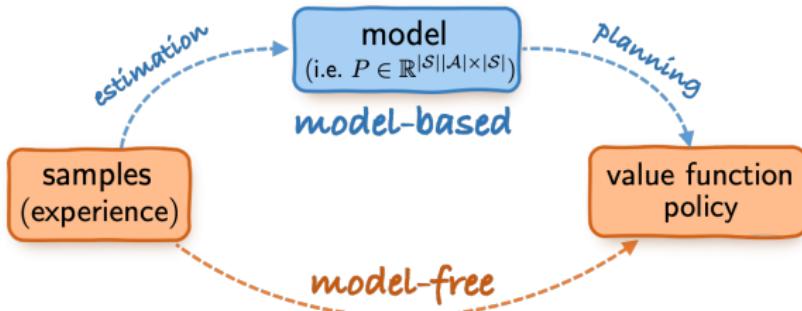
Reinforcement learning (Part 2): Model-free RL



Yuxin Chen

Wharton Statistics & Data Science, Spring 2022

Model-based vs. model-free RL

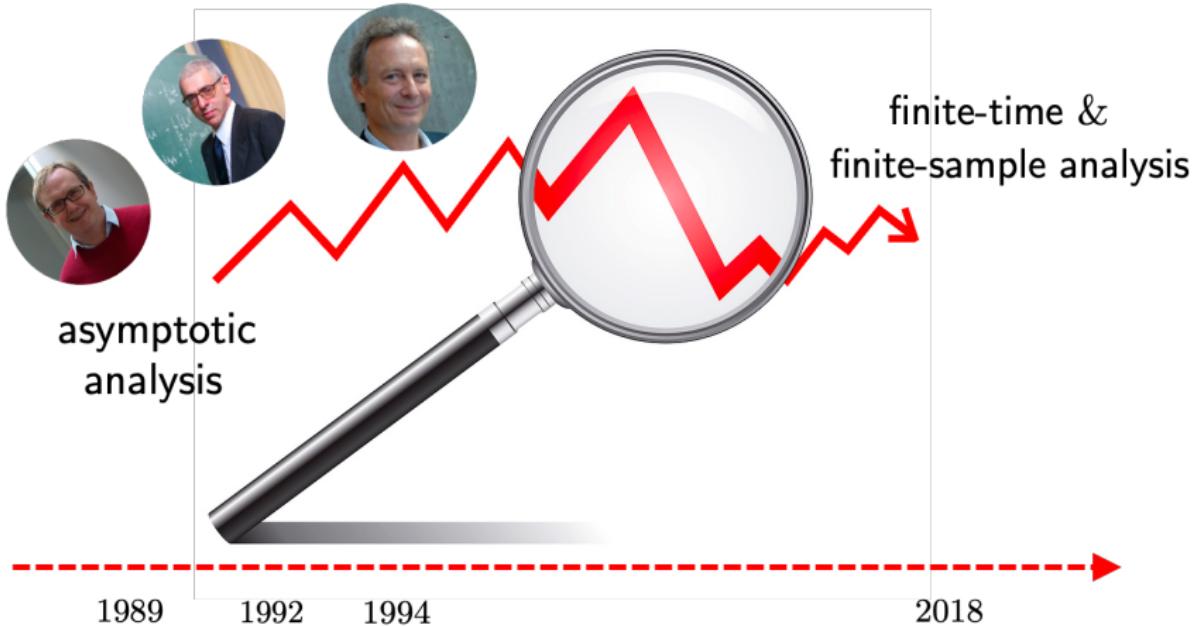


Model-based approach (“plug-in”)

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

Model-free approach

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



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

Model-free RL

1. Basics of Q-learning
2. Synchronous Q-learning and variance reduction (simulator)
3. Asynchronous Q-learning (Markovian data)
4. Q-learning with lower confidence bounds (offline RL)
5. Q-learning with upper confidence bounds (online RL)

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

A detour: stochastic approximation

- **Goal:** solve

$$G(x) = \mathbb{E}[g(x; \xi)] = 0$$

- ξ : randomness in problem

- **What we can query:** for any given input x , we receive a *random* sample $g(x; \xi)$ obeying $\mathbb{E}[g(x; \xi)] = G(x)$

Stochastic approximation (Robbins, Monro '51)



Herbert Robbins



Sutton Monro

stochastic approximation

$$\mathbf{x}^{t+1} = \mathbf{x}^t - \eta_t \mathbf{g}(\mathbf{x}^t; \boldsymbol{\xi}^t) \quad (1)$$

where $\mathbf{g}(\mathbf{x}^t; \boldsymbol{\xi}^t)$ is *unbiased* estimate of $\mathbf{G}(\mathbf{x}^t)$, i.e.

$$\mathbb{E}[\mathbf{g}(\mathbf{x}^t; \boldsymbol{\xi}^t)] = \mathbf{G}(\mathbf{x}^t)$$

Stochastic approximation (Robbins, Monro '51)



Herbert Robbins



Sutton Monro

stochastic approximation

$$\boldsymbol{x}^{t+1} = \boldsymbol{x}^t - \eta_t \mathbf{g}(\boldsymbol{x}^t; \boldsymbol{\xi}^t) \quad (1)$$

a stochastic algorithm for finding roots of $\mathbf{G}(\boldsymbol{x}) := \mathbb{E}[\mathbf{g}(\boldsymbol{x}; \boldsymbol{\xi})]$

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

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

Q-learning: a stochastic approximation algorithm



Chris Watkins



Peter Dayan

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

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

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

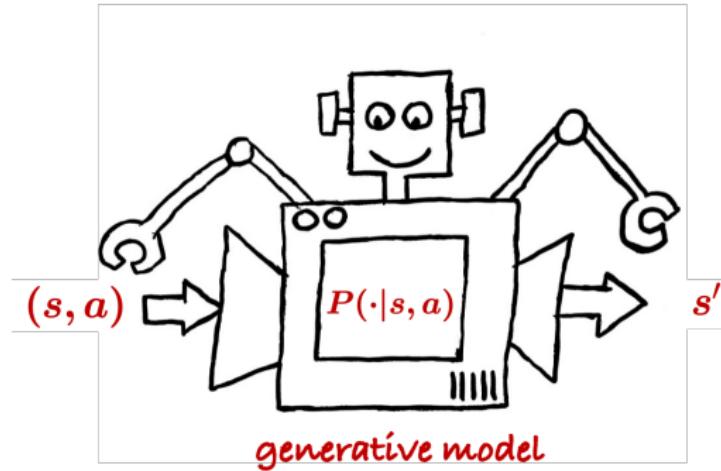
$$\mathcal{T}(Q)(s, a) = r(s, a) + \gamma \mathbb{E}_{s' \sim P(\cdot | s, a)} \left[\max_{a'} Q(s', a') \right]$$

Model-free RL

1. Basics of Q-learning
2. **Synchronous Q-learning and variance reduction (simulator)**
3. Asynchronous Q-learning (Markovian data)
4. Q-learning with lower confidence bounds (offline RL)
5. Q-learning with upper confidence bounds (online RL)

A generative model / simulator

— Kearns, Singh '99



In each iteration, collect an independent sample (s, a, s') for each (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

Sample complexity of synchronous Q-learning

Theorem 1 (Li, Cai, Chen, Gu, Wei, Chi '21)

For any $0 < \varepsilon \leq 1$, synchronous Q-learning yields $\|\hat{Q} - Q^*\|_\infty \leq \varepsilon$ with high prob., with sample complexity (i.e., $T|\mathcal{S}||\mathcal{A}|$) at most

$$\tilde{O}\left(\frac{|\mathcal{S}||\mathcal{A}|}{(1-\gamma)^4\varepsilon^2}\right)$$

other papers	sample complexity
Even-Dar & Mansour '03	$2^{\frac{1}{1-\gamma}} \frac{ \mathcal{S} \mathcal{A} }{(1-\gamma)^4\varepsilon^2}$
Beck & Srikant '12	$\frac{ \mathcal{S} ^2 \mathcal{A} ^2}{(1-\gamma)^5\varepsilon^2}$
Wainwright '19	$\frac{ \mathcal{S} \mathcal{A} }{(1-\gamma)^5\varepsilon^2}$
Chen et al. '20	$\frac{ \mathcal{S} \mathcal{A} }{(1-\gamma)^5\varepsilon^2}$

Sample complexity of synchronous Q-learning

Theorem 1 (Li, Cai, Chen, Gu, Wei, Chi '21)

For any $0 < \varepsilon \leq 1$, synchronous Q-learning yields $\|\hat{Q} - Q^*\|_\infty \leq \varepsilon$ with high prob., with sample complexity (i.e., $T|\mathcal{S}||\mathcal{A}|$) at most

$$\tilde{O}\left(\frac{|\mathcal{S}||\mathcal{A}|}{(1-\gamma)^4\varepsilon^2}\right)$$

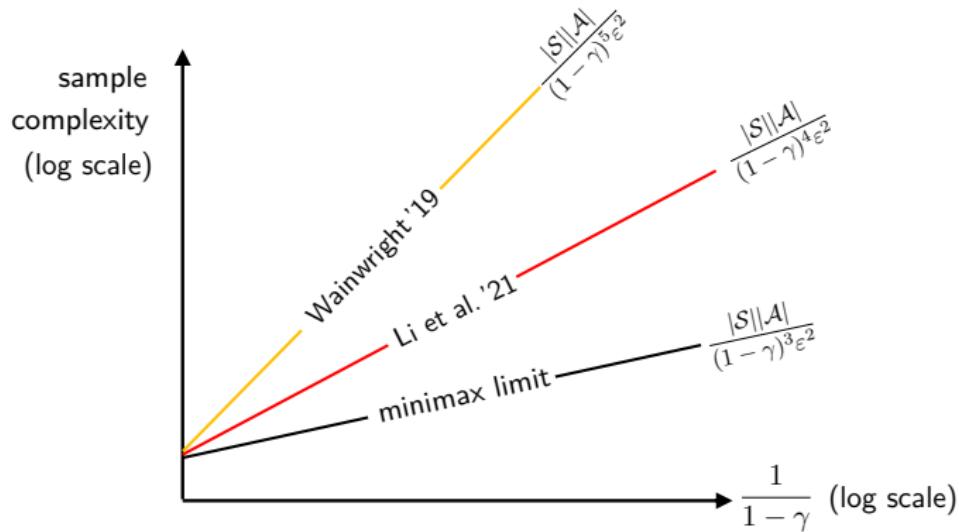
- Covers both *constant* and *rescaled linear* learning rates:

$$\eta_t \equiv \frac{1}{1 + \frac{c_1(1-\gamma)T}{\log^2 T}}$$

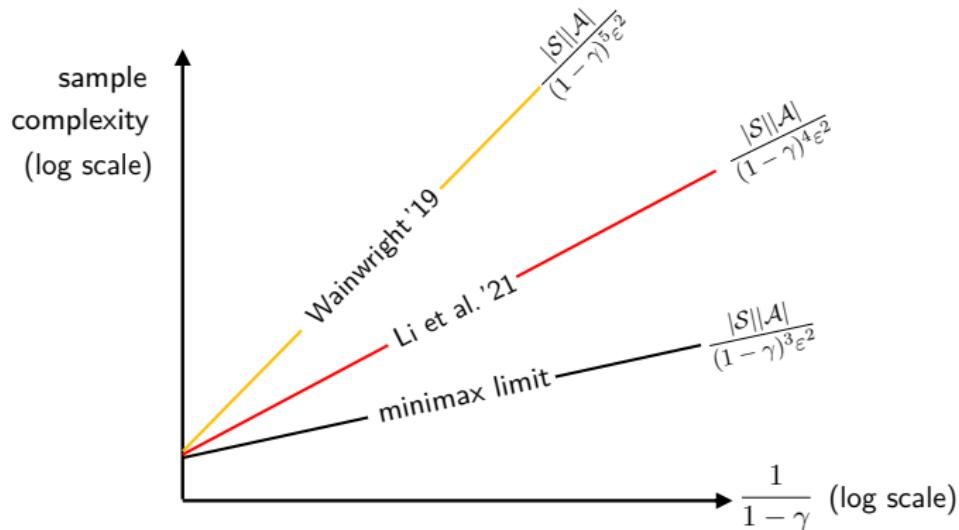
or $\eta_t = \frac{1}{1 + \frac{c_2(1-\gamma)t}{\log^2 T}}$

other papers	sample complexity
Even-Dar & Mansour '03	$2^{\frac{1}{1-\gamma}} \frac{ \mathcal{S} \mathcal{A} }{(1-\gamma)^4\varepsilon^2}$
Beck & Srikant '12	$\frac{ \mathcal{S} ^2 \mathcal{A} ^2}{(1-\gamma)^5\varepsilon^2}$
Wainwright '19	$\frac{ \mathcal{S} \mathcal{A} }{(1-\gamma)^5\varepsilon^2}$
Chen et al. '20	$\frac{ \mathcal{S} \mathcal{A} }{(1-\gamma)^5\varepsilon^2}$

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



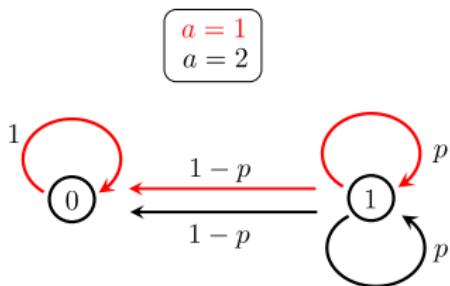
All this requires sample size at least $\frac{|S||\mathcal{A}|}{(1-\gamma)^4 \varepsilon^2} \dots$



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

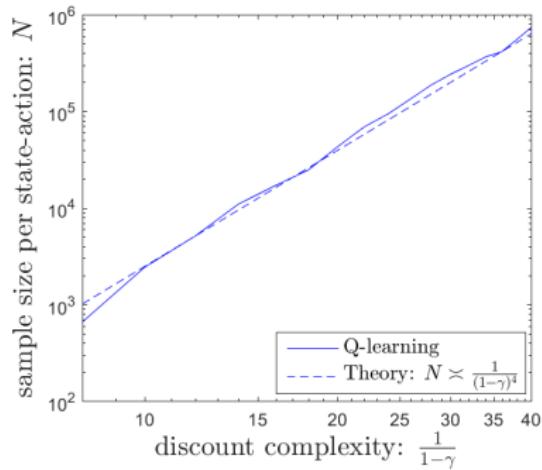
A numerical example: $\frac{|S||A|}{(1-\gamma)^4 \varepsilon^2}$ samples seem necessary . . .

— 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 2 (Li, Cai, Chen, Gu, Wei, Chi, 2021)

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

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

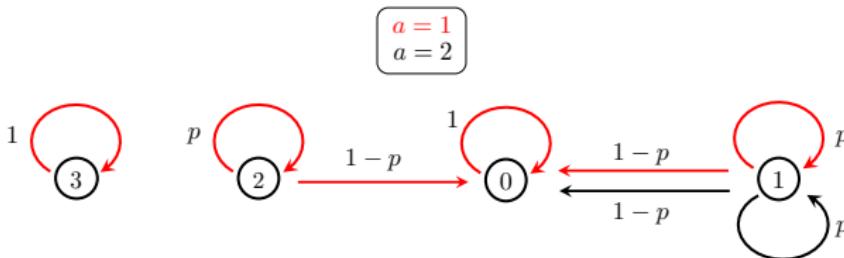
Q-learning is NOT minimax optimal

Theorem 2 (Li, Cai, Chen, Gu, Wei, Chi, 2021)

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

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

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

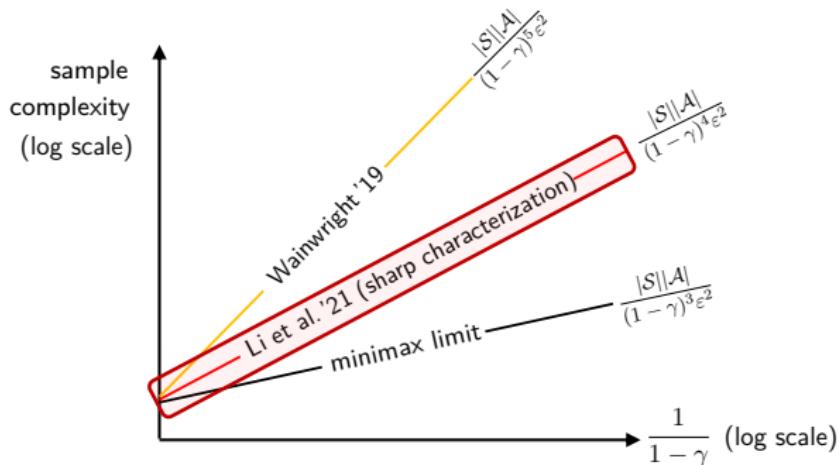


Q-learning is NOT minimax optimal

Theorem 2 (Li, Cai, Chen, Gu, Wei, Chi, 2021)

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

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



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)

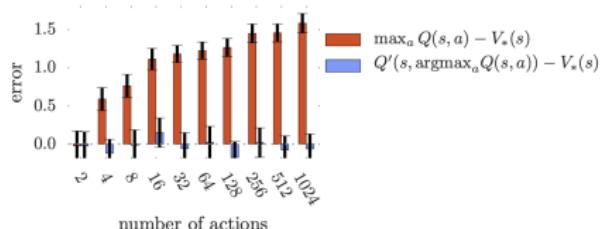


Figure 1: The orange bars show the bias in a single Q-learning update when the action values are $Q(s, a) = V_*(s) + \epsilon_a$ and the errors $\{\epsilon_a\}_{a=1}^m$ are independent standard normal random variables. The second set of action values Q' , used for the blue bars, was generated identically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

Improving sample complexity via variance reduction

A detour: finite-sum optimization

$$\text{minimize}_{\boldsymbol{x} \in \mathbb{R}^d} \quad F(\boldsymbol{x}) := \frac{1}{n} \sum_{i=1}^n f_i(\boldsymbol{x})$$

- $F(\cdot)$: μ -strongly convex
- f_i : convex and L -smooth (i.e., ∇f_i is L -Lipschitz)
- $\kappa := L/\mu$: condition number

Recall: SGD theory with fixed stepsizes

$$\mathbf{x}^{t+1} = \mathbf{x}^t - \eta_t \mathbf{g}^t$$

- \mathbf{g}^t : an unbiased stochastic estimate of $F(\mathbf{x}^t)$
- $\mathbb{E}[\|\mathbf{g}^t\|_2^2] \leq \sigma_g^2 + c_g \|\nabla F(\mathbf{x}^t)\|_2^2$

This SGD-type algorithm with $\eta_t \equiv \eta$ obeys

$$\mathbb{E}[F(\mathbf{x}^t) - F(\mathbf{x}^*)] \leq \frac{\eta L \sigma_g^2}{2\mu} + (1 - \eta\mu)^t (F(\mathbf{x}^0) - F(\mathbf{x}^*))$$

Recall: SGD theory with fixed stepsizes

$$\mathbb{E}[F(\mathbf{x}^t) - F(\mathbf{x}^*)] \leq \frac{\eta L \sigma_g^2}{2\mu} + (1 - \eta\mu)^t (F(\mathbf{x}^0) - F(\mathbf{x}^*))$$

- vanilla SGD: $\mathbf{g}^t = \nabla f_{i_t}(\mathbf{x}^t)$
 - **issue:** σ_g^2 is non-negligible even when $\mathbf{x}^t = \mathbf{x}^*$
- **question:** it is possible to design \mathbf{g}^t with reduced variability σ_g^2 ?

A simple idea

Imagine we take some \mathbf{v}^t with $\mathbb{E}[\mathbf{v}^t] = \mathbf{0}$ and set

$$\mathbf{g}^t = \nabla f_{i_t}(\mathbf{x}^t) - \mathbf{v}^t$$

— so \mathbf{g}^t is still an unbiased estimate of $\nabla F(\mathbf{x}^t)$

A simple idea

Imagine we take some \mathbf{v}^t with $\mathbb{E}[\mathbf{v}^t] = \mathbf{0}$ and set

$$\mathbf{g}^t = \nabla f_{i_t}(\mathbf{x}^t) - \mathbf{v}^t$$

— so \mathbf{g}^t is still an unbiased estimate of $\nabla F(\mathbf{x}^t)$

question: how to reduce variability (i.e. $\mathbb{E}[\|\mathbf{g}^t\|_2^2] < \mathbb{E}[\|\nabla f_{i_t}(\mathbf{x}^t)\|_2^2]$)?

A simple idea

Imagine we take some \mathbf{v}^t with $\mathbb{E}[\mathbf{v}^t] = \mathbf{0}$ and set

$$\mathbf{g}^t = \nabla f_{i_t}(\mathbf{x}^t) - \mathbf{v}^t$$

— so \mathbf{g}^t is still an unbiased estimate of $\nabla F(\mathbf{x}^t)$

question: how to reduce variability (i.e. $\mathbb{E}[\|\mathbf{g}^t\|_2^2] < \mathbb{E}[\|\nabla f_{i_t}(\mathbf{x}^t)\|_2^2]$)?

answer: find some zero-mean \mathbf{v}^t that is positively correlated with $\nabla f_{i_t}(\mathbf{x}^t)$ (i.e. $\langle \mathbf{v}^t, \nabla f_{i_t}(\mathbf{x}^t) \rangle > 0$) ([why?](#))

Reducing variance via gradient aggregation

If the current iterate is not too far away from previous iterates, then historical gradient info might be useful in producing such a v^t to reduce variance

main idea of variance reduction: aggregate previous gradient info to help improve the convergence rate

Stochastic variance reduced gradient (SVRG)

— Johnson, Zhang '13

key idea: if we have access to a history point \mathbf{x}^{old} and $\nabla F(\mathbf{x}^{\text{old}})$, then

$$\underbrace{\nabla f_{i_t}(\mathbf{x}^t) - \nabla f_{i_t}(\mathbf{x}^{\text{old}})}_{\rightarrow 0 \text{ if } \mathbf{x}^t \approx \mathbf{x}^{\text{old}}} + \underbrace{\nabla F(\mathbf{x}^{\text{old}})}_{\rightarrow 0 \text{ if } \mathbf{x}^{\text{old}} \approx \mathbf{x}^*} \quad \text{with } i_t \sim \text{Unif}(1, \dots, n)$$

- is an unbiased estimate of $\nabla F(\mathbf{x}^t)$
- $\underbrace{\nabla f_{i_t}(\mathbf{x}^t) - \nabla f_{i_t}(\mathbf{x}^{\text{old}})}_{\text{variability is reduced!}}$ if $\mathbf{x}^t \approx \mathbf{x}^{\text{old}} \approx \mathbf{x}^*$

Stochastic variance reduced gradient (SVRG)

- operate in epochs
- in the s^{th} epoch
 - **very beginning:** take a snapshot $\mathbf{x}_s^{\text{old}}$ of the current iterate, and compute the **batch gradient** $\nabla F(\mathbf{x}_s^{\text{old}})$
 - **inner loop:** use the snapshot point to help reduce variance

$$\mathbf{x}_s^{t+1} = \mathbf{x}_s^t - \eta \left\{ \nabla f_{i_t}(\mathbf{x}_s^t) - \nabla f_{i_t}(\mathbf{x}_s^{\text{old}}) + \nabla F(\mathbf{x}_s^{\text{old}}) \right\}$$

a hybrid approach: batch gradient is computed only once per epoch

Remark

- constant stepsize η
- each epoch contains $2m + n$ gradient computations
 - the batch gradient is computed only once every m iterations
 - the average per-iteration cost of SVRG is comparable to that of SGD if $m \gtrsim n$
- linear convergence

Remark

- constant stepsize η
- each epoch contains $2m + n$ gradient computations
 - the batch gradient is computed only once every m iterations
 - the average per-iteration cost of SVRG is comparable to that of SGD if $m \gtrsim n$
- linear convergence
- **total computational cost:**

$$\underbrace{(m+n)}_{\text{number of grad computation per epoch}} \log \frac{1}{\varepsilon} \asymp \underbrace{(n+\kappa) \log \frac{1}{\varepsilon}}_{\text{if } m \asymp \max\{n, \kappa\}}$$

Back to Q-learning . . .

— inspired by Johnson & Zhang '13

Variance-reduced Q-learning updates (Wainwright '19)

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

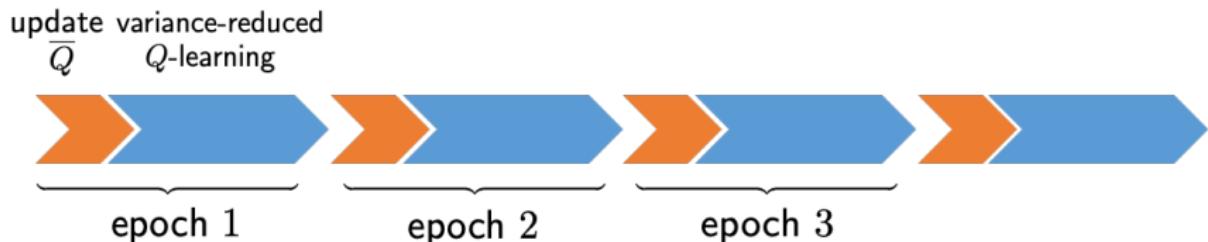
- \bar{Q} : some reference Q-estimate
- $\tilde{\mathcal{T}}$: empirical Bellman operator (using a batch of samples)

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

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

An epoch-based stochastic algorithm

— inspired by Johnson & Zhang '13



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 3 (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

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

- allows for more aggressive learning rates

Sample complexity of variance-reduced Q-learning

Theorem 3 (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

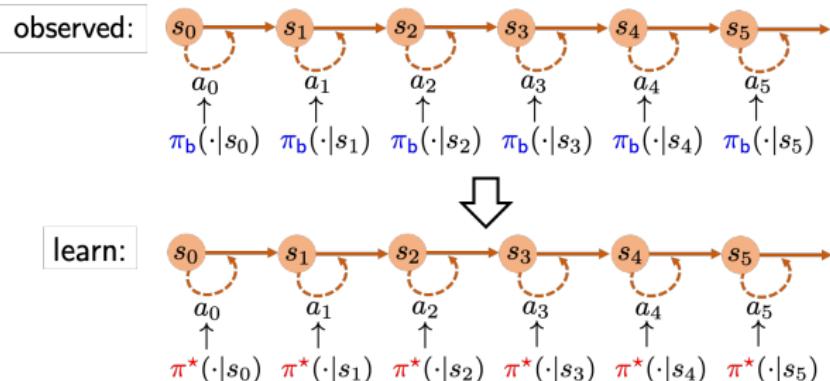
$$\tilde{O}\left(\frac{|\mathcal{S}||\mathcal{A}|}{(1-\gamma)^3\varepsilon^2}\right)$$

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

Model-free RL

1. Basics of Q-learning
2. Synchronous Q-learning and variance reduction (simulator)
3. **Asynchronous Q-learning (Markovian data)**
4. Q-learning with lower confidence bounds (offline RL)
5. Q-learning with upper confidence bounds (online RL)

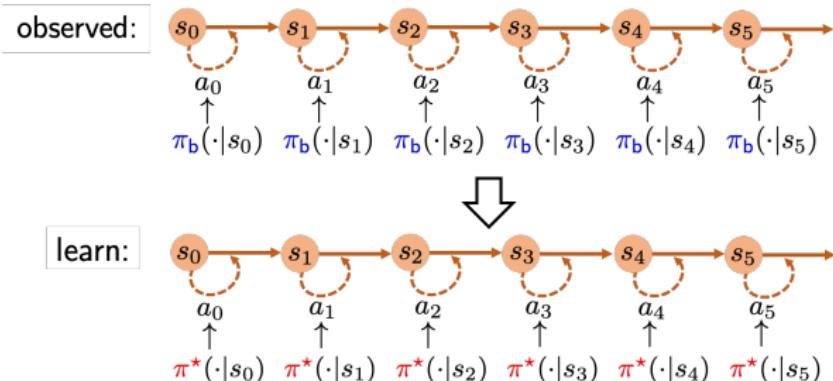
Markovian samples and behavior policy



Observed: $\underbrace{\{s_t, a_t, r_t\}_{t \geq 0}}_{\text{stationary Markovian trajectory}}$ generated by **behavior policy** π_b

Goal: learn optimal value V^* and Q^* based on sample trajectory

Markovian samples and behavior policy



Key quantities of sample trajectory

- minimum state-action occupancy probability

$$\mu_{\min} := \min \underbrace{\mu_{\pi_b}(s, a)}_{\text{stationary distribution}}$$

- mixing time: t_{mix}

Q-learning on Markovian samples



Chris Watkins



Peter Dayan

$$\underbrace{Q_{t+1}(s_t, a_t) = (1 - \eta_t)Q_t(s_t, a_t) + \eta_t \mathcal{T}_t(Q_t)(s_t, a_t),}_{\text{only update } (s_t, a_t)\text{-th entry}} \quad t \geq 0$$

Q-learning on Markovian samples



Chris Watkins

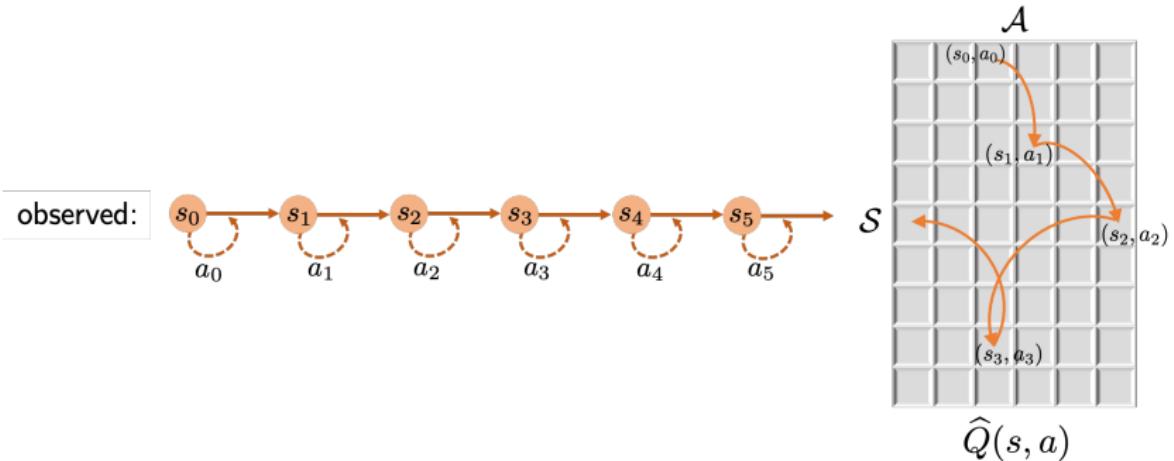


Peter Dayan

$$Q_{t+1}(s_t, a_t) = \underbrace{(1 - \eta_t)Q_t(s_t, a_t) + \eta_t \mathcal{T}_t(Q_t)(s_t, a_t)}_{\text{only update } (s_t, a_t)\text{-th entry}}, \quad t \geq 0$$

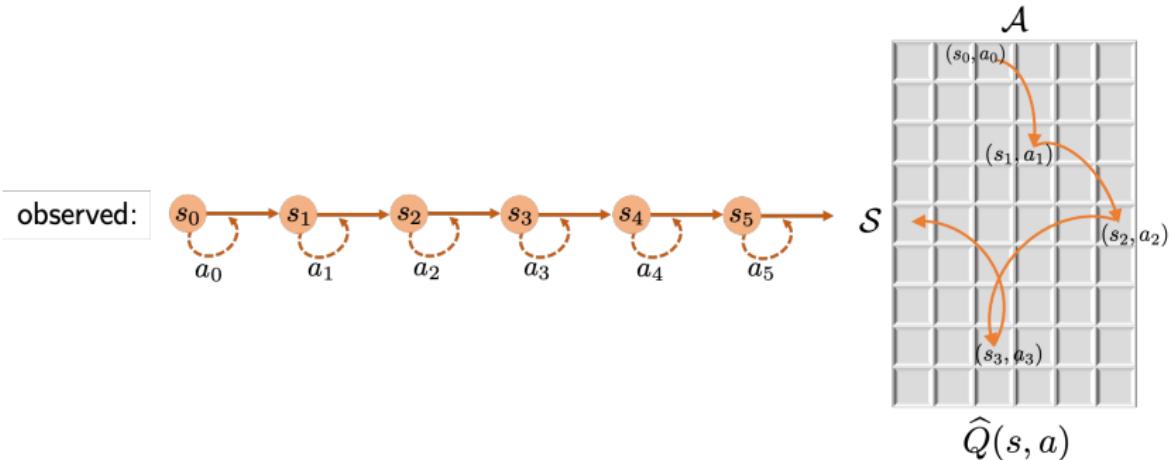
$$\mathcal{T}_t(Q)(s_t, a_t) = r(s_t, a_t) + \gamma \max_{a'} Q(s_{t+1}, a')$$

Q-learning on Markovian samples



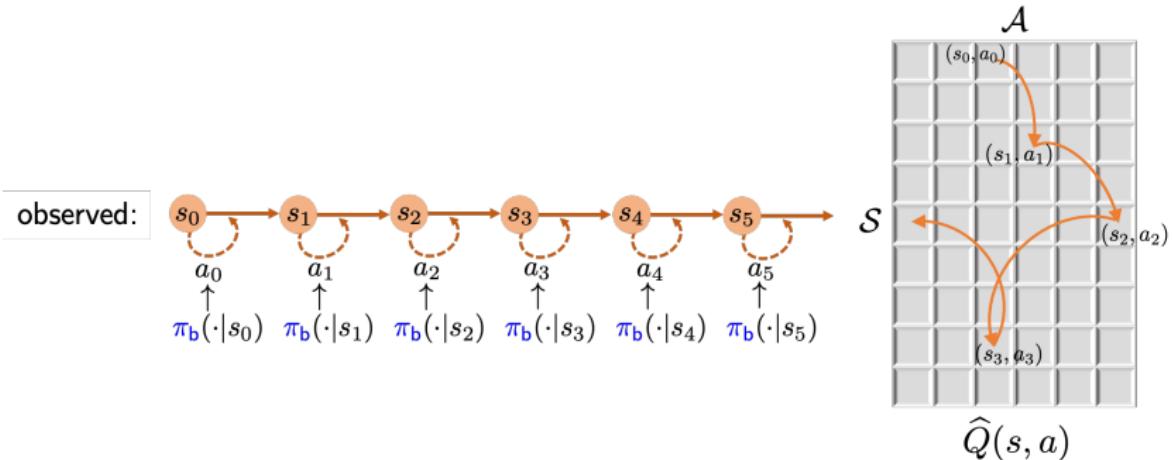
- **asynchronous:** only a single entry is updated each iteration

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 π_b

A highly incomplete list of works

- Watkins, Dayan '92
- Tsitsiklis '94
- Jaakkola, Jordan, Singh '94
- Szepesvári '98
- Borkar, Meyn '00
- Even-Dar, Mansour '03
- Beck, Srikant '12
- Chi, Zhu, Bubeck, Jordan '18
- Lee, He '18
- Chen, Zhang, Doan, Maguluri, Clarke '19
- Du, Lee, Mahajan, Wang '20
- Chen, Maguluri, Shakkottai, Shanmugam '20
- Qu, Wierman '20
- Devraj, Meyn '20
- Weng, Gupta, He, Ying, Srikant '20
- Li, Wei, Chi, Gu, Chen '20
- Li, Cai, Chen, Gu, Wei, Chi '21
- Chen, Maguluri, Shakkottai, Shanmugam '21
- ...

Sample complexity of asynchronous Q-learning

Theorem 4 (Li, Cai, Chen, Gu, Wei, Chi '21)

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 log factor)

$$\frac{1}{\mu_{\min}(1-\gamma)^4 \varepsilon^2} + \frac{t_{\text{mix}}}{\mu_{\min}(1-\gamma)}$$

Sample complexity of asynchronous Q-learning

Theorem 4 (Li, Cai, Chen, Gu, Wei, Chi '21)

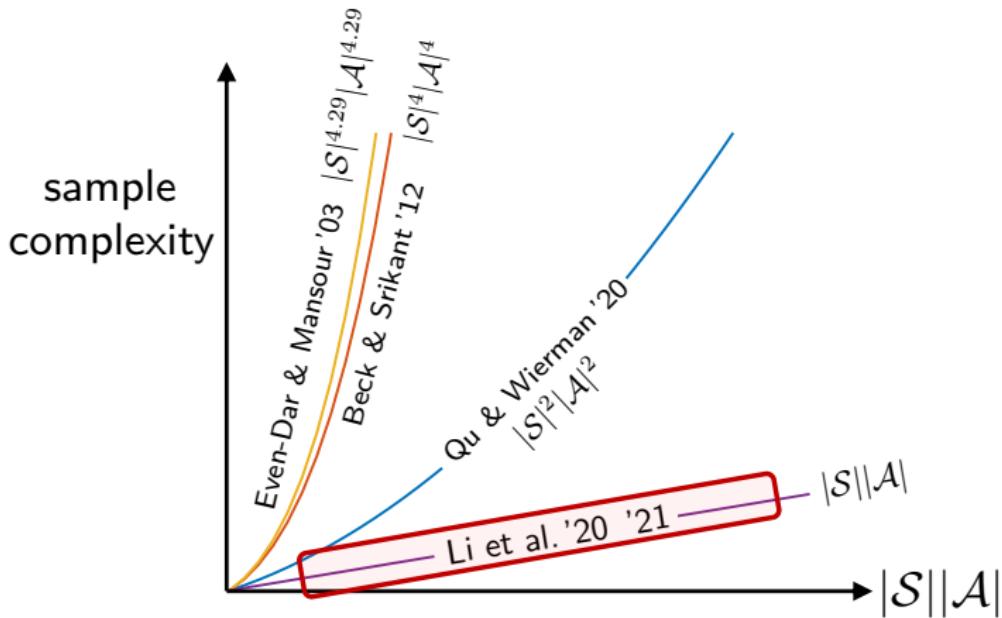
For any $0 < \varepsilon \leq \frac{1}{1-\gamma}$, sample complexity of async Q-learning to yield $\|\hat{Q} - Q^*\|_\infty \leq \varepsilon$ is at most (up to log factor)

$$\frac{1}{\mu_{\min}(1-\gamma)^4 \varepsilon^2} + \frac{t_{\text{mix}}}{\mu_{\min}(1-\gamma)}$$

- learning rates:
constant & rescaled linear

other papers	sample complexity
Even-Dar et al. '03	$\frac{(t_{\text{cover}})^{\frac{1}{1-\gamma}}}{(1-\gamma)^4 \varepsilon^2}$
Even-Dar et al. '03	$\left(\frac{t_{\text{cover}}^{1+3\omega}}{(1-\gamma)^4 \varepsilon^2}\right)^{\frac{1}{\omega}} + \left(\frac{t_{\text{cover}}}{1-\gamma}\right)^{\frac{1}{1-\omega}}, \omega \in (\frac{1}{2}, 1)$
Beck & Srikant '12	$\frac{t_{\text{cover}}^3 \mathcal{S} \mathcal{A} }{(1-\gamma)^5 \varepsilon^2}$
Qu & Wierman '20	$\frac{t_{\text{mix}}}{\mu_{\min}^2 (1-\gamma)^5 \varepsilon^2}$
Li et al. '20	$\frac{1}{\mu_{\min} (1-\gamma)^5 \varepsilon^2} + \frac{t_{\text{mix}}}{\mu_{\min} (1-\gamma)}$
Chen et al. '21	$\frac{1}{\mu_{\min}^3 (1-\gamma)^5 \varepsilon^2} + \text{other-term}(t_{\text{mix}})$

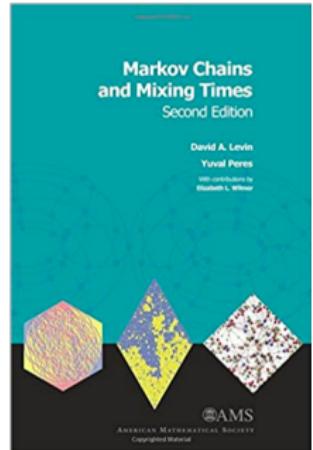
Linear dependency on $1/\mu_{\min}$



if we take $\mu_{\min} \asymp \frac{1}{|S||A|}$, $t_{\text{cover}} \asymp \frac{t_{\text{mix}}}{\mu_{\min}}$

Effect of mixing time on sample complexity

$$\frac{1}{\mu_{\min}(1-\gamma)^4 \varepsilon^2} + \frac{t_{\text{mix}}}{\mu_{\min}(1-\gamma)}$$



- reflects cost taken to reach steady state
- one-time expense (almost independent of ε)
 - it becomes amortized as algorithm runs
- can be improved with the aid of variance reduction (Li et al. '20)
 - prior art: $\frac{t_{\text{mix}}}{\mu_{\min}^2(1-\gamma)^5 \varepsilon^2}$ (Qu & Wierman '20)

Model-free RL

1. Basics of Q-learning
2. Synchronous Q-learning and variance reduction (simulator)
3. Asynchronous Q-learning (Markovian data)
4. **Q-learning with lower confidence bounds (offline RL)**
5. Q-learning with upper confidence bounds (online RL)

Recap: offline RL / batch RL

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

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

for some state distribution ρ^b and behavior policy π^b

Recap: offline RL / batch RL

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

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

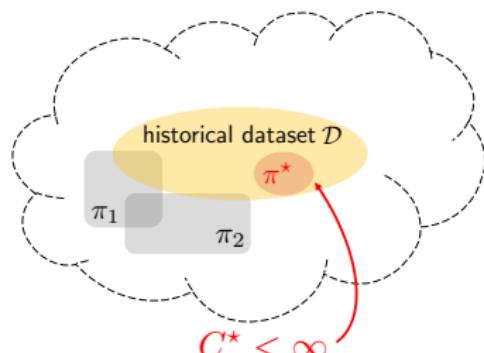
for some state distribution ρ^b and behavior policy π^b

Single-policy concentrability

$$C^* := \max_{s,a} \frac{d^{\pi^*}(s,a)}{d^{\pi^b}(s,a)} \geq 1$$

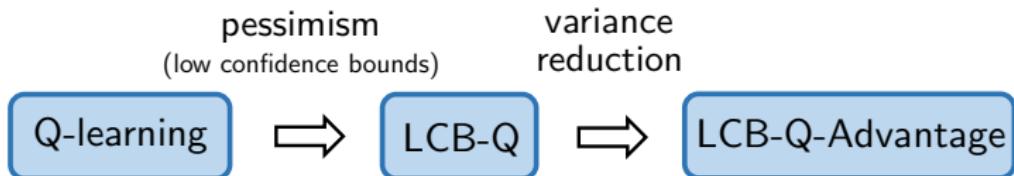
where d^π : occupancy distribution under π

- captures **distributional shift**
- allows for partial coverage



*How to design offline model-free algorithms
with optimal sample efficiency?*

*How to design offline model-free algorithms
with optimal sample efficiency?*



LCB-Q: Q-learning with LCB penalty

— *Shi et al. '22, Yan et al. '22*

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

LCB-Q: Q-learning with LCB penalty

— Shi et al. '22, Yan et al. '22

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

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

LCB-Q: Q-learning with LCB penalty

— Shi et al. '22, Yan et al. '22

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

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

sample size: $O\left(\frac{SC^*}{(1-\gamma)^5 \varepsilon^2}\right) \implies$ 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. '22, Yan et al. '22

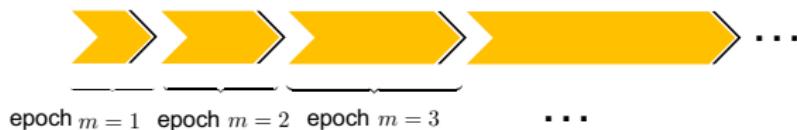
$$\begin{aligned} Q_{t+1}(s_t, a_t) \leftarrow & (1 - \eta_t) Q_t(s_t, a_t) - \eta_t \underbrace{b_t(s_t, a_t)}_{\text{LCB penalty}} \\ & + \eta_t \left(\underbrace{\mathcal{T}_t(Q_t) - \mathcal{T}_t(\bar{Q})}_{\text{advantage}} + \underbrace{\hat{\mathcal{T}}(\bar{Q})}_{\text{reference}} \right)(s_t,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

Q-learning with LCB and variance reduction

— Shi et al. '22, Yan et al. '22

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

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

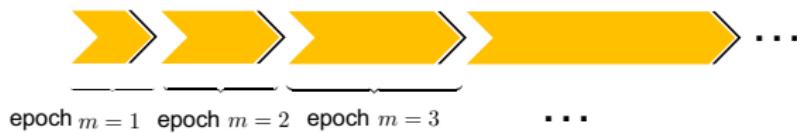


Q-learning with LCB and variance reduction

— Shi et al. '22, Yan et al. '22

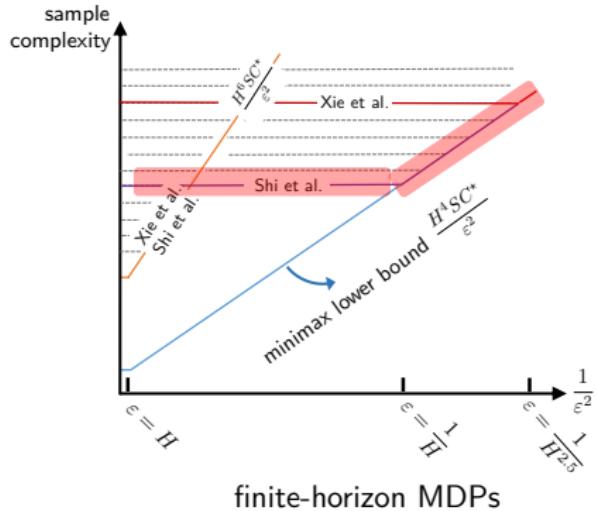
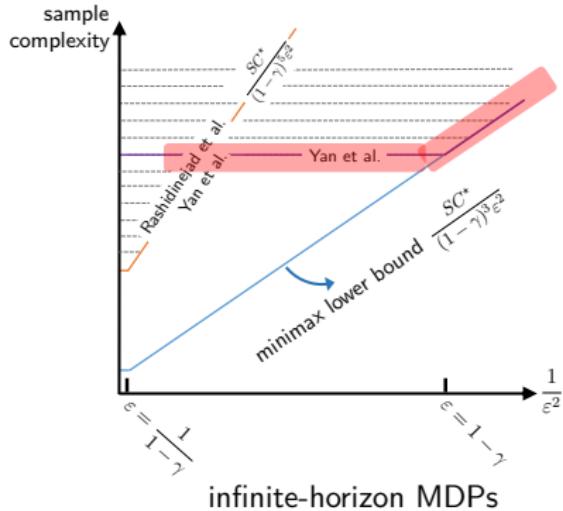
$$\begin{aligned} Q_{t+1}(s_t, a_t) \leftarrow & (1 - \eta_t) Q_t(s_t, a_t) - \eta_t \underbrace{b_t(s_t, a_t)}_{\text{LCB penalty}} \\ & + \eta_t \left(\underbrace{\mathcal{T}_t(Q_t) - \mathcal{T}_t(\bar{Q})}_{\text{advantage}} + \underbrace{\hat{\mathcal{T}}(\bar{Q})}_{\text{reference}} \right)(s_t,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

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



Theorem 5 (Yan, Li, Chen, Fan '22, 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 $\tilde{O}\left(\frac{SC^*}{(1-\gamma)^3 \varepsilon^2}\right)$



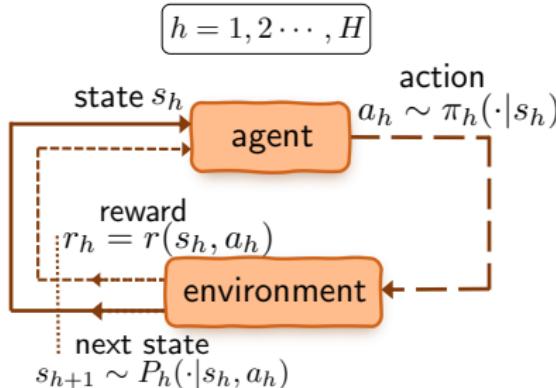
Model-free offline RL attains sample optimality too!

— with some burn-in cost though ...

Model-free RL

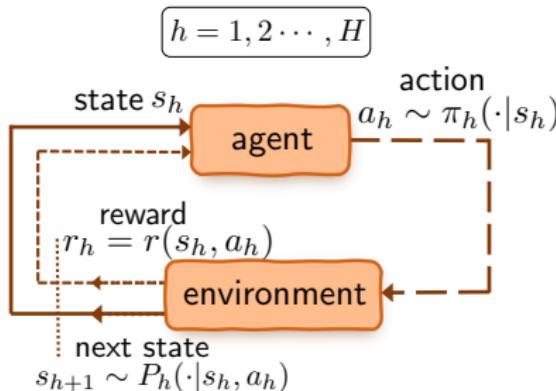
1. Basics of Q-learning
2. Synchronous Q-learning and variance reduction (simulator)
3. Asynchronous Q-learning (Markovian data)
4. Q-learning with lower confidence bounds (offline RL)
5. **Q-learning with upper confidence bounds (online RL)**

Finite-horizon MDPs



- 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



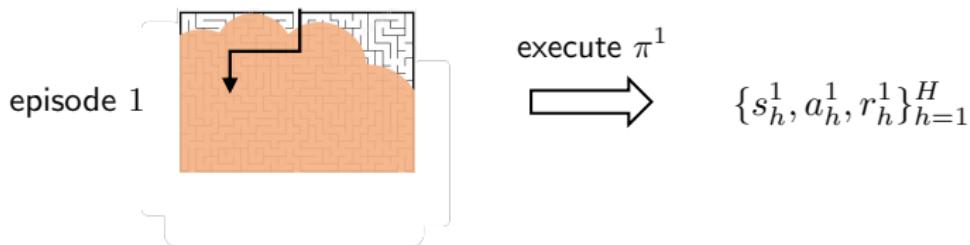
value function: $V_h^\pi(s) := \mathbb{E} \left[\sum_{t=h}^H r_h(s_h, a_h) \mid s_h = s \right]$

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



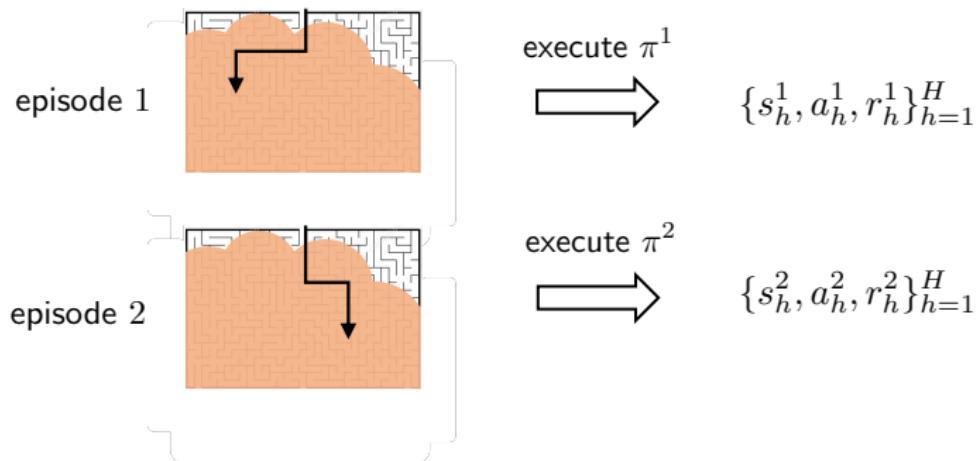
Online RL: interacting with real environments

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



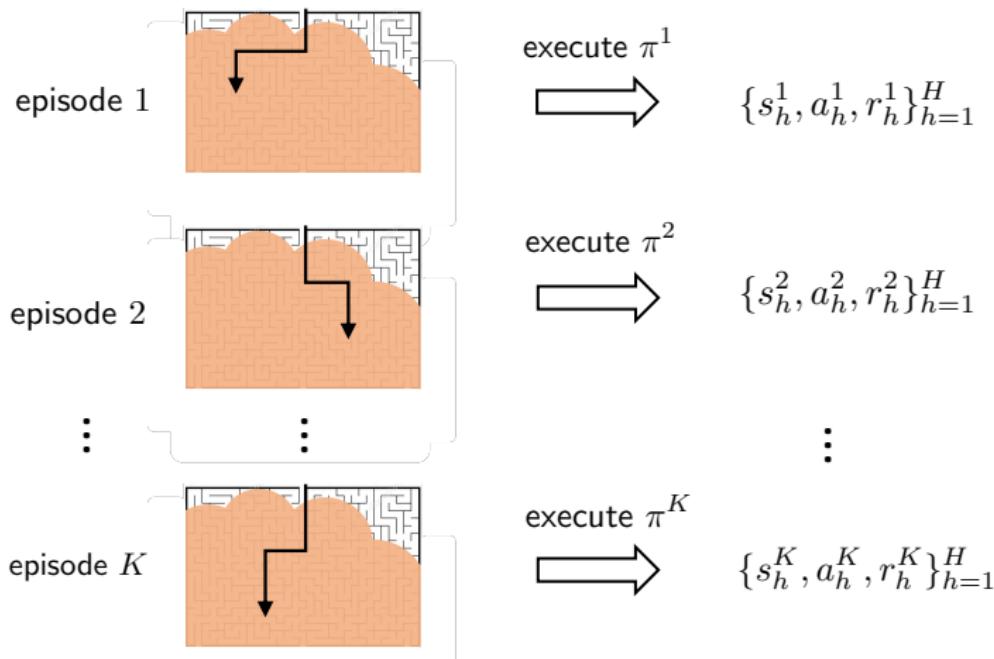
Online RL: interacting with real environments

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



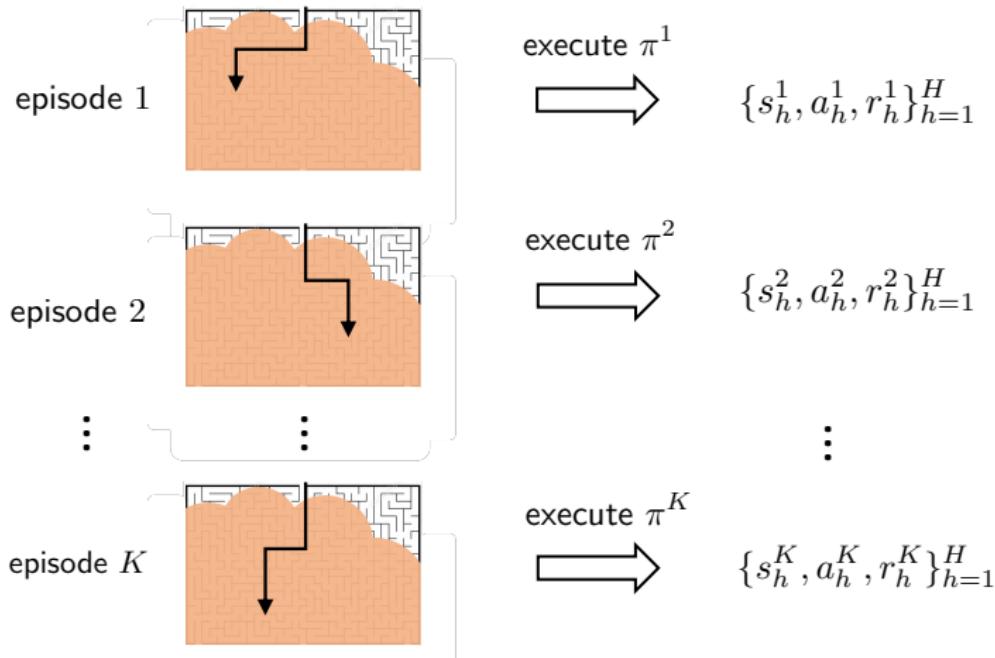
Online RL: interacting with real environments

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

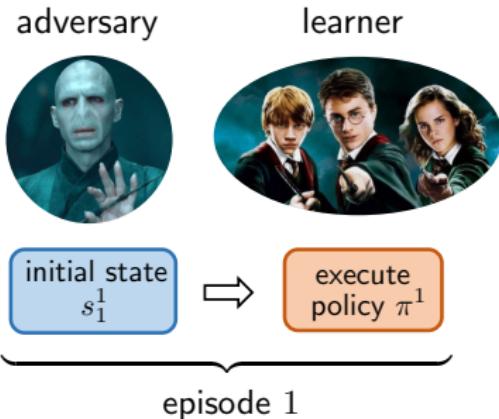


Online RL: interacting with real environments

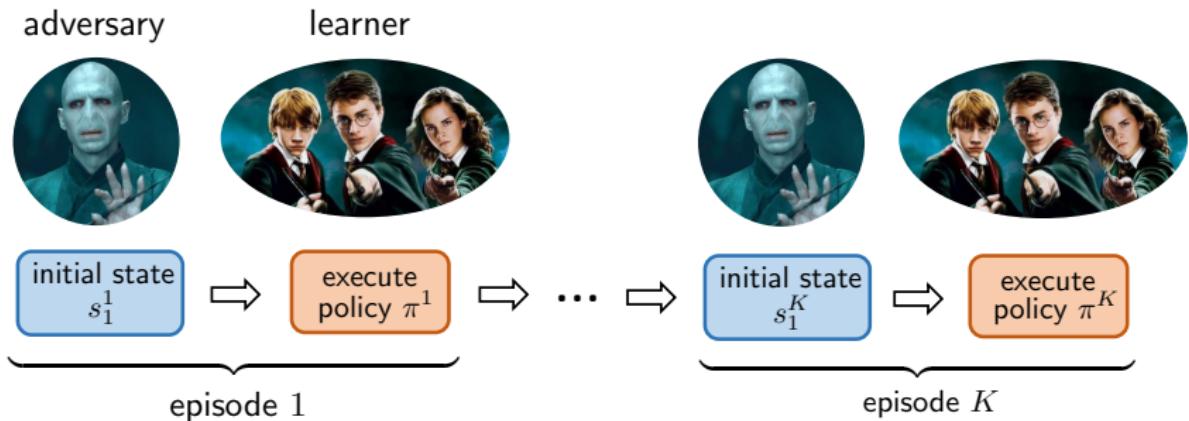
Sequentially execute MDP for K episodes, each consisting of H steps
— sample size: $T = KH$



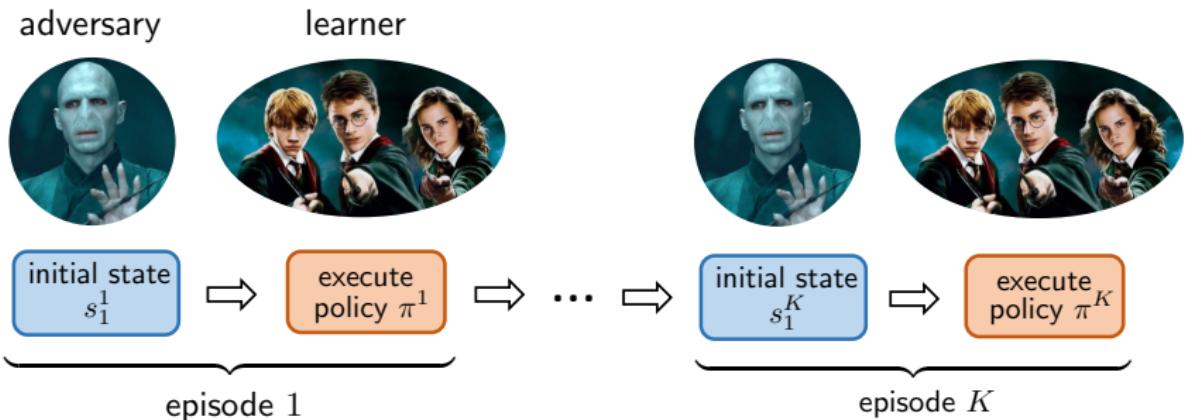
Regret: gap between learned policy & optimal policy



Regret: gap between learned policy & optimal policy



Regret: gap between learned policy & optimal policy



Performance metric: given initial states $\underbrace{\{s_1^k\}_{k=1}^K}_{\text{chosen by nature/adversary}}$, define

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

Lower bound

(Domingues et al. '21)

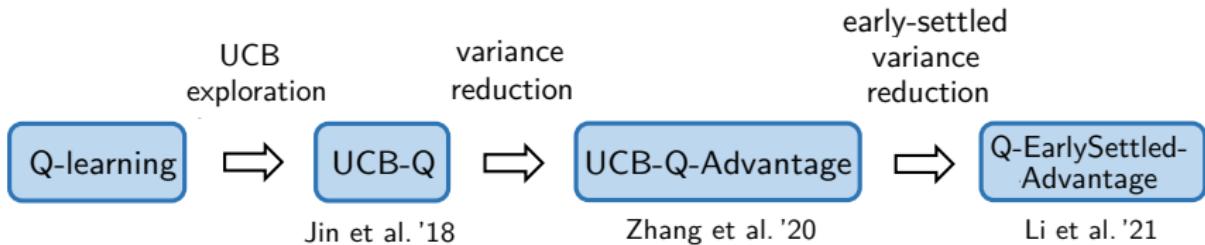
$$\text{Regret}(T) \gtrsim \sqrt{H^2 SAT}$$

Existing algorithms

- UCB-VI: Azar et al. '17
- UBEV: Dann et al. '17
- UCB-Q-Hoeffding: Jin et al. '18
- **UCB-Q-Bernstein: Jin et al. '18**
- UCB2-Q-Bernstein: Bai et al. '19
- EULER: Zanette et al. '19
- **UCB-Q-Advantage: Zhang et al. '20**
- UCB-M-Q: Menard et al. '21
- **Q-EarlySettled-Advantage: Li et al. '21**

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

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



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; encourage exploration
 - *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; encourage exploration
 - *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; encourage exploration
 - *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 **variance reduction** into UCB-Q:

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

- employ variance reduction to help accelerate convergence

Q-learning with UCB and variance reduction

— Zhang et al. '20

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

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

- employ variance reduction to help accelerate convergence

UCB-Q-Advantage is asymptotically regret-optimal

Q-learning with UCB and variance reduction

— Zhang et al. '20

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

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

- employ variance reduction to help accelerate convergence

UCB-Q-Advantage is asymptotically regret-optimal

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

UCB-Q with variance reduction and early settlement

One additional key idea: early settlement of the reference as soon as it reaches a reasonable quality

UCB-Q with variance reduction and early settlement

One additional key idea: early settlement of the reference as soon as it reaches a reasonable quality

Theorem 6 (Li, Shi, Chen, Gu, Chi '21)

With high prob., Q-EarlySettled-Advantage achieves

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

UCB-Q with variance reduction and early settlement

One additional key idea: early settlement of the reference as soon as it reaches a reasonable quality

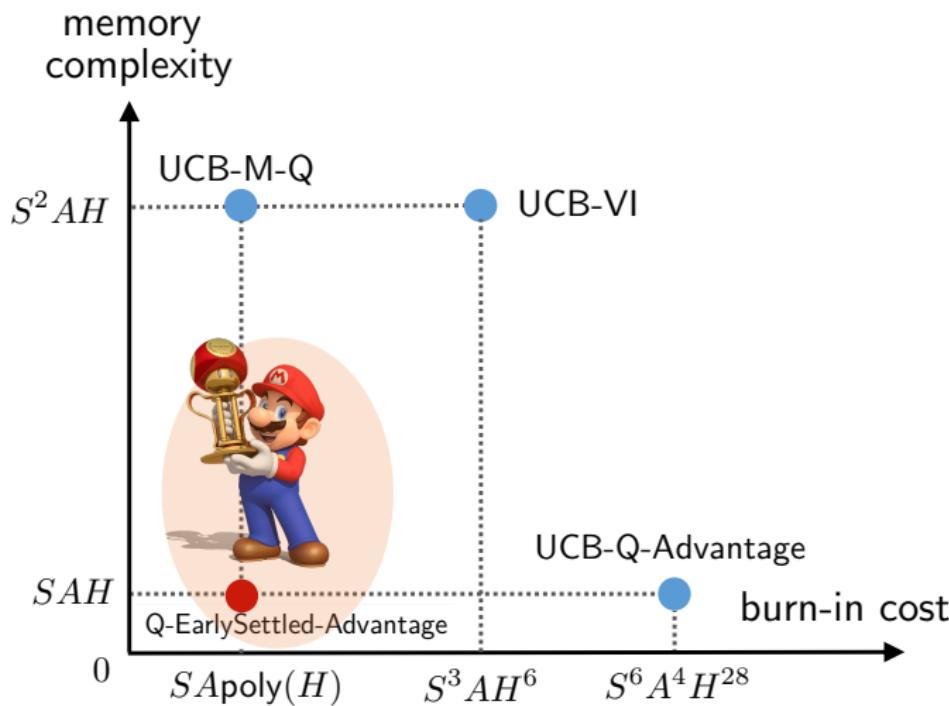
Theorem 6 (Li, Shi, Chen, Gu, Chi '21)

With high prob., Q-EarlySettled-Advantage achieves

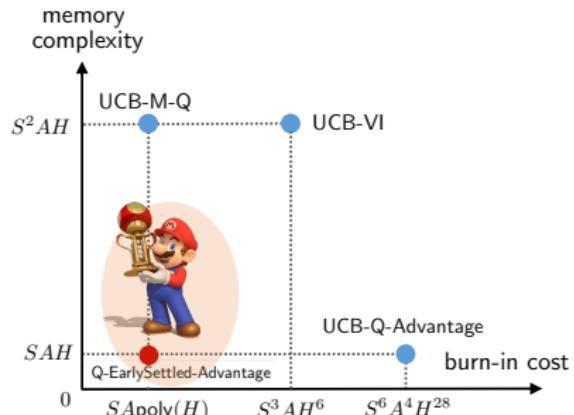
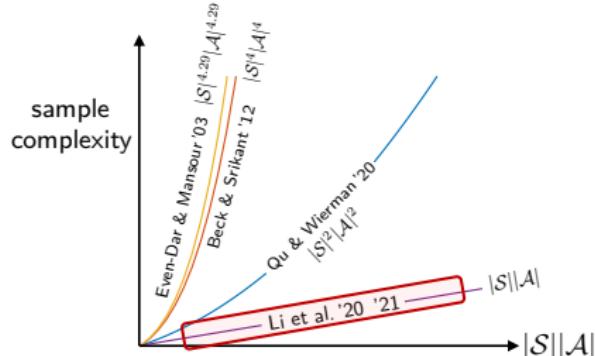
$$\text{Regret}(T) \leq \tilde{O}(\sqrt{H^2SAT} + H^6SA)$$

- regret-optimal with $\underbrace{\text{near-minimal burn-in cost}}_{S\text{Poly}(H)}$ in S and A
- memory-efficient $O(SAH)$
- computationally efficient: runtime $O(T)$

Comparisons of regret-optimal algorithms



Summary of this part



Model-free RL can achieve memory efficiency, computational efficiency, and sample efficiency at once!

— with some burn-in cost though

Reference 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
- "*Learning from delayed rewards,*" C. Watkins, 1989
- "*Q-learning,*" C. Watkins, P. Dayan, *Machine learning*, 1992
- "*Learning to predict by the methods of temporal differences,*" R. Sutton, *Machine learning*, 1988
- "*Analysis of temporal-difference learning with function approximation,*" B. van Roy, J. Tsitsiklis, *IEEE transactions on automatic control*, 1997
- "*Learning Rates for Q-learning,*" E. Even-Dar, Y. Mansour, *Journal of machine learning Research*, 2003

Reference II

- "*The asymptotic convergence-rate of Q-learning,*" C. Szepesvari, *NeurIPS*, 1998
- "*Stochastic approximation with cone-contractive operators: Sharp ℓ_∞ bounds for Q-learning,*" M. Wainwright, arXiv:1905.06265, 2019
- "*Is Q-Learning minimax optimal? A tight sample complexity analysis,*" G. Li, Y. Wei, Y. Chi, Y. Gu, Y. Chen, arXiv:2102.06548, 2021
- "*Accelerating stochastic gradient descent using predictive variance reduction,*" R. Johnson, T. Zhang, *NeurIPS*, 2013.
- "*Variance-reduced Q-learning is minimax optimal,*" M. Wainwright, arXiv:1906.04697, 2019
- "*Asynchronous stochastic approximation and Q-learning,*" J. Tsitsiklis, *Machine learning*, 1994

Reference III

- "*On the convergence of stochastic iterative dynamic programming algorithms,*" T. Jaakkola, M. Jordan, S. Singh, *Neural computation*, 1994
- "*Error bounds for constant step-size Q-learning,*" C. Beck, R. Srikant, *Systems and control letters*, 2012
- "*Sample complexity of asynchronous Q-learning: sharper analysis and variance reduction,*" G. Li, Y. Wei, Y. Chi, Y. Gu, Y. Chen, *NeurIPS* 2020
- "*Finite-time analysis of asynchronous stochastic approximation and Q-learning,*" G. Qu, A. Wierman, *COLT* 2020.
- "*Pessimistic Q-learning for offline reinforcement learning: Towards optimal sample complexity,*" L. Shi, G. Li, Y. Wei, Y. Chen, Y. Chi, arXiv:2202.13890, 2022.

Reference IV

- "*The efficacy of pessimism in asynchronous Q-learning,*" Y. Yan, G. Li, Y. Chen, J. Fan, arXiv:2203.07368, 2022.
- "*Asymptotically efficient adaptive allocation rules,*" T. L. Lai, H. Robbins, *Advances in applied mathematics*, vol. 6, no. 1, 1985.
- "*Is Q-learning provably efficient?*" C. Jin, Z. Allen-Zhu, S. Bubeck, and M. Jordan, *NeurIPS* 2018.
- "*Almost optimal model-free reinforcement learning via reference-advantage decomposition,*" Z. Zhang, Y. Zhou, X. Ji, *NeurIPS* 2020.
- "*Breaking the sample complexity barrier to regret-optimal model-free reinforcement learning,*" G. Li, L. Shi, Y. Chen, Y. Gu, Y. Chi, *NeurIPS* 2021.