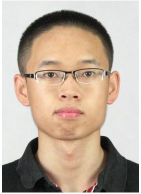
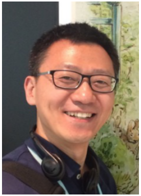
Breaking the Sample Complexity Barrier to Regret-Optimal Model-Free Reinforcement Learning



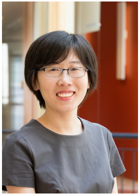
Gen Li
Princeton ECE

Laixi Shi
CMU ECE

Yuxin Chen
Princeton ECE



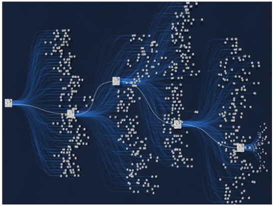
Yuantao Gu
Tsinghua EE



Yuejie Chi
CMU ECE

Reinforcement learning (RL): challenges

In RL, an agent learns by interacting with an environment



Challenges:

- explore or exploit in unknown environments
- credit assignment problem: delayed rewards or feedback
- enormous state and action space

Sample efficiency

Collecting data samples might be expensive or time-consuming in the face of enormous state/action space

clinical trials

autonomous driving

online ads

Sample efficiency

Collecting data samples might be expensive or time-consuming in the face of enormous state/action space

clinical trials

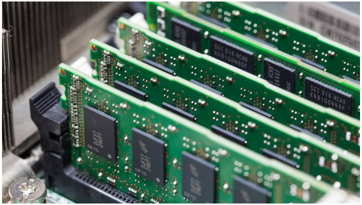
autonomous driving

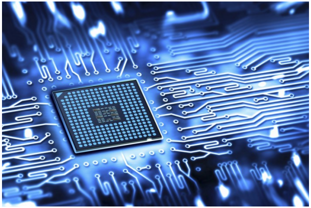
online ads

Calls for design of sample-efficient RL algorithms!

Memory efficiency

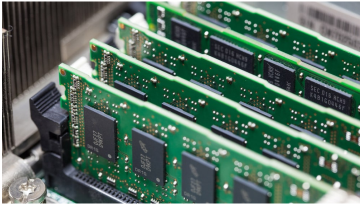
Running RL algorithms might impose huge memory requirement in the face of enormous state/action space

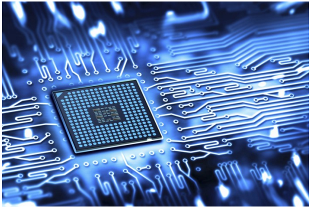




Memory efficiency

Running RL algorithms might impose huge memory requirement in the face of enormous state/action space

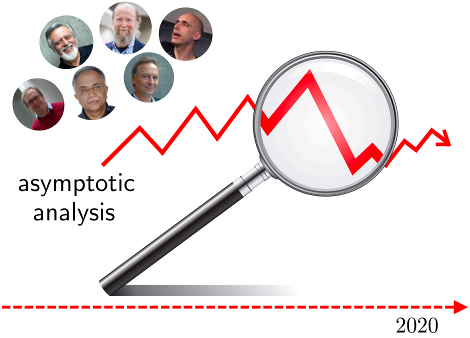




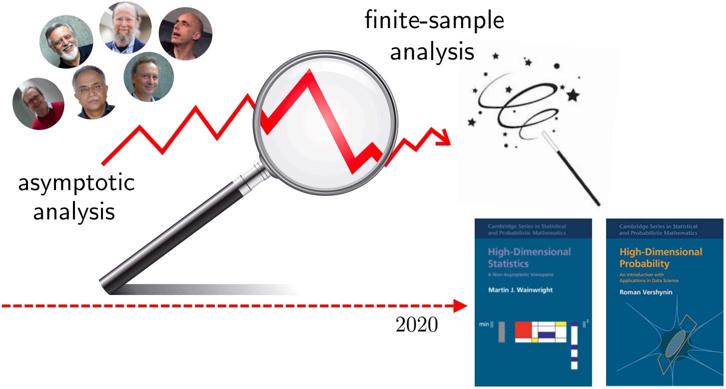
Calls for design of memory-efficient RL algorithms!

How to design sample- & memory-efficient algorithms?

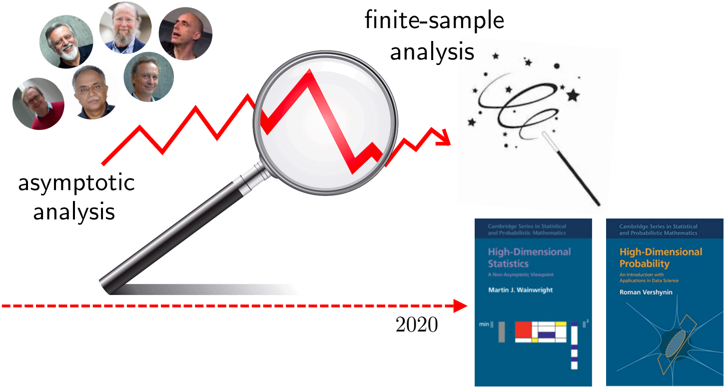
From asymptotic to non-asymptotic analyses



From asymptotic to non-asymptotic analyses



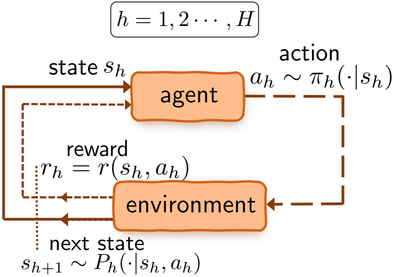
From asymptotic to non-asymptotic analyses



Non-asymptotic analyses play a key role in understanding sample & memory efficiency of modern RL

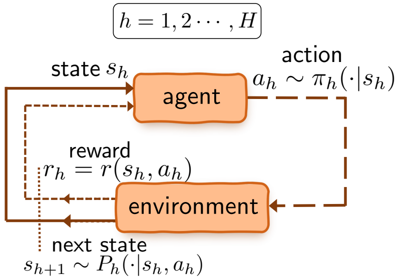
Background

Episodic Markov decision process (MDP)



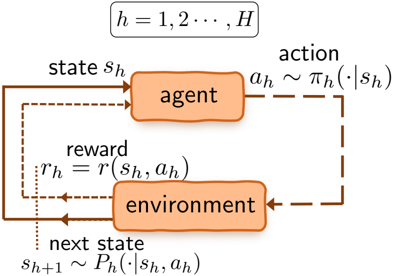
- H: horizon length

Episodic Markov decision process (MDP)



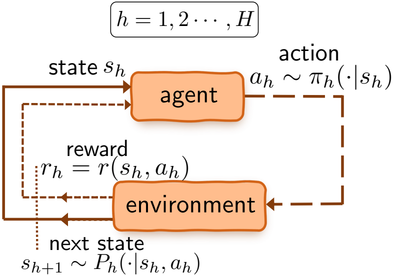
- H: horizon length
- $\mathcal{S}$: state space with size S
- $\mathcal{A}$: action space with size A

Episodic Markov decision process (MDP)



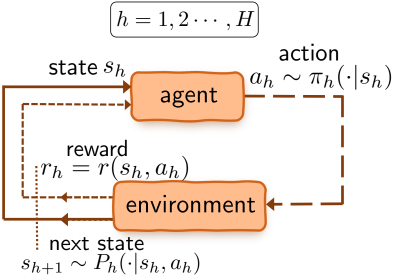
- H: horizon length
- $\mathcal{S}$: state space with size S
- $\mathcal{A}$: action space with size A
- $r_h(s_h, a_h) \in [0, 1]$: immediate reward in step h

Episodic Markov decision process (MDP)



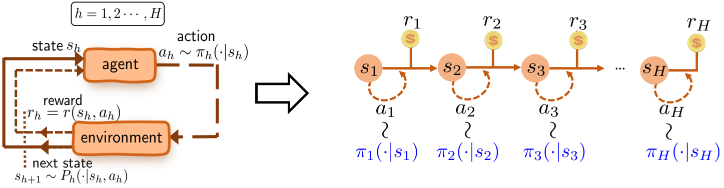
- H: horizon length
- $\mathcal{S}$: state space with size S
- $\mathcal{A}$: action space with size A
- $r_h(s_h, a_h) \in [0, 1]$: immediate reward in step h
- $\pi = \{\pi_h\}_{h=1}^{H}$: policy (or action selection rule)

Episodic Markov decision process (MDP)



- 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)
- $P_h(\cdot \mid s, a)$: transition probabilities in step h

Value function and Q-function of policy π

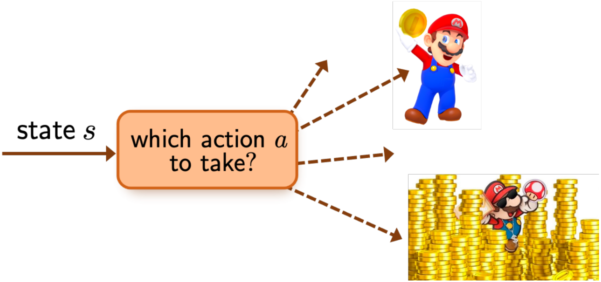


$$V_h^\pi(s) := \mathbb{E}\left[\sum_{t=h}^{H} r_h(s_h, a_h) \,\Big|\, s_h = s\right]$$

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

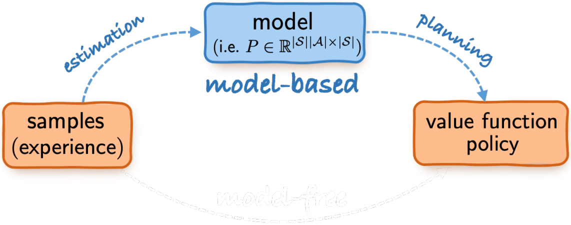
- execute policy π to generate sample trajectory

Optimal policy and optimal values



- Optimal policy $\pi^\star$: maximizing the value function
- Optimal value / Q function: $V_h^\star := V_h^{\pi^\star}$, $Q_h^\star := Q_h^{\pi^\star}$

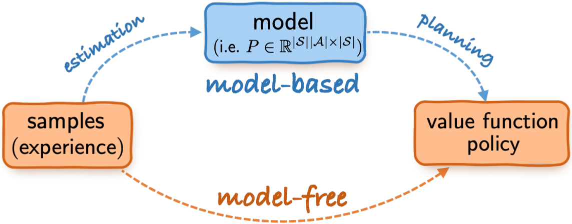
Model-based vs. model-free RL



Model-based approach ("plug-in")

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

Model-based vs. model-free RL



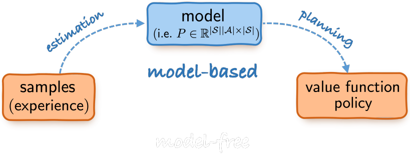
Model-based approach ("plug-in")

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

Model-free approach

— learning w/o modeling & estimating environment explicitly

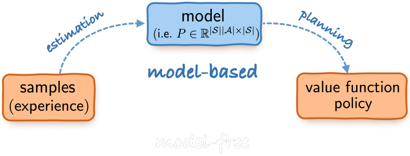
Model-free RL is often more memory-efficient



store transition kernel estimates

$\rightarrow$ $O(S^2AH)$ *memory*

Model-free RL is often more memory-efficient



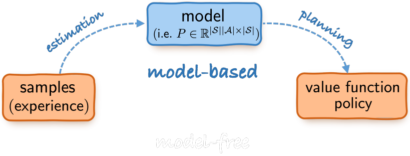
store transition kernel estimates

$\rightarrow$ $O(S^2AH)$ *memory*

maintain Q-estimates

$\rightarrow$ $O(SAH)$ *memory*

Model-free RL is often more memory-efficient



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

maintain Q-estimates
$\rightarrow$ $O(SAH)$ *memory*

Definition 1 (Jin et al. '18)

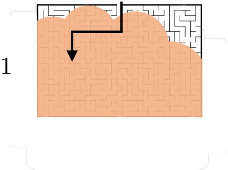
An RL algorithm is **model-free** if its space complexity is $o(S^2AH)$

Online RL and regret minimization

Online RL: interacting with real environments

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

episode 1



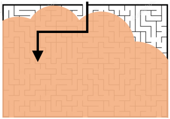
execute π^1

$\{s_h^1, a_h^1, r_h^1\}_{h=1}^H$

Online RL: interacting with real environments

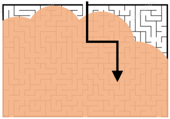
Sequentially execute MDP for K episodes, each consisting of H steps

episode 1



execute π^1 $\Longrightarrow$ $\{s_h^1, a_h^1, r_h^1\}_{h=1}^H$

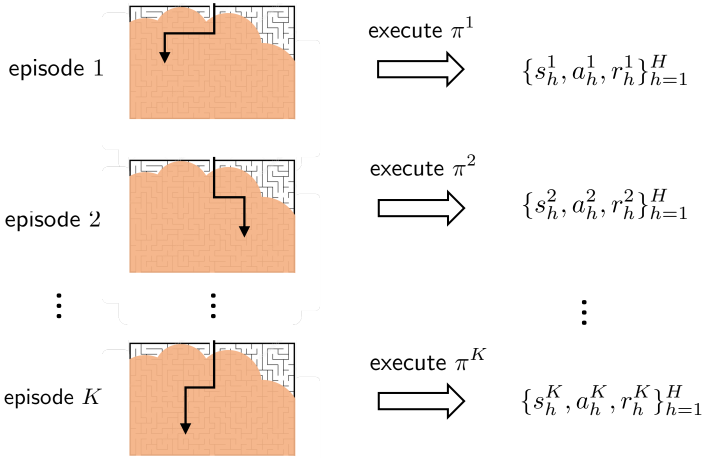
episode 2



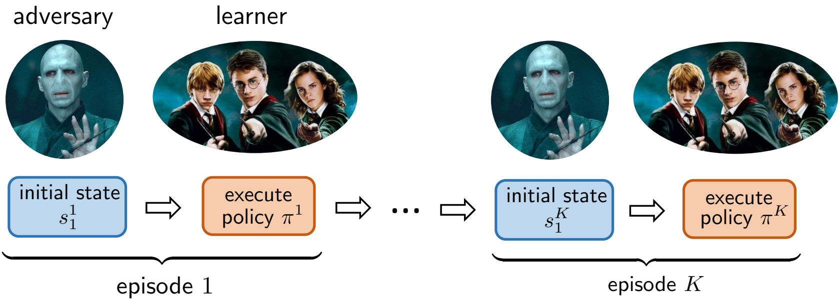
execute π^2 $\Longrightarrow$ $\{s_h^2, a_h^2, r_h^2\}_{h=1}^H$

Online RL: interacting with real environments

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



Regret: gap between learned policy & optimal policy



Regret: gap between learned policy & optimal policy

adversary

learner

initial state s_1^1 ⇨ execute policy π^1 ⇨ ⋯ ⇨ initial state s_1^K ⇨ execute policy π^K

episode 1 … episode K

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

$$\text{Regret}(\underbrace{T}_{\text{sample size: } KH}) := \sum_{k=1}^{K} \Big(V_1^{\star}(s_1^k) - V_1^{\pi^k}(s_1^k) \Big)$$

Lower bound

(Domingues et al. '21)

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

Existing algorithms

- UCB-VI: Azar et al. '17
- UBCV: 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

Lower bound

(Domingues et al. '21)

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

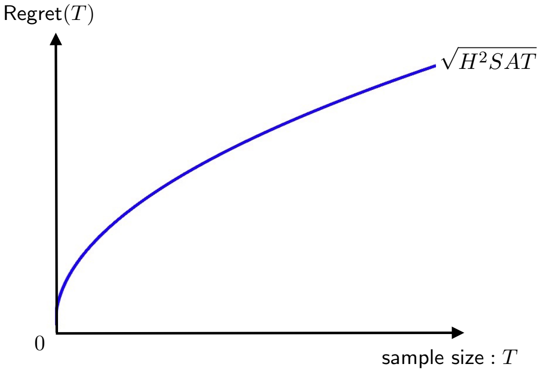
Existing algorithms

- UCB-VI: Azar et al. '17
- UBCV: 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

Which algorithms can achieve near-minimal regret?

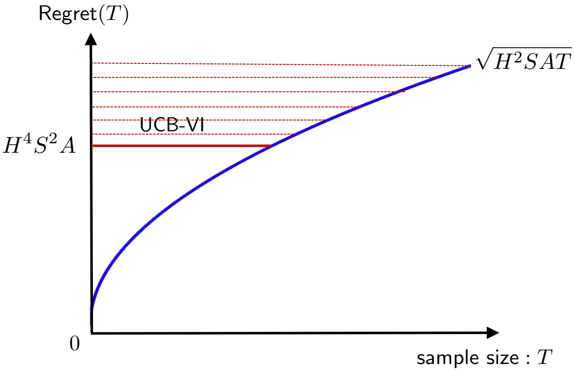
Prior art: Azar et al. '17

First method that is *asymptotically* regret-optimal: UCB-VI



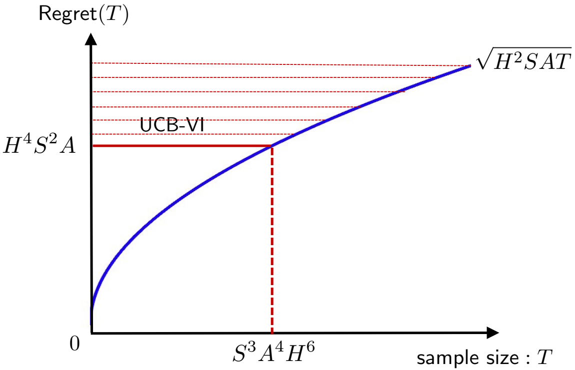
Prior art: Azar et al. '17

First method that is *asymptotically* regret-optimal: UCB-VI



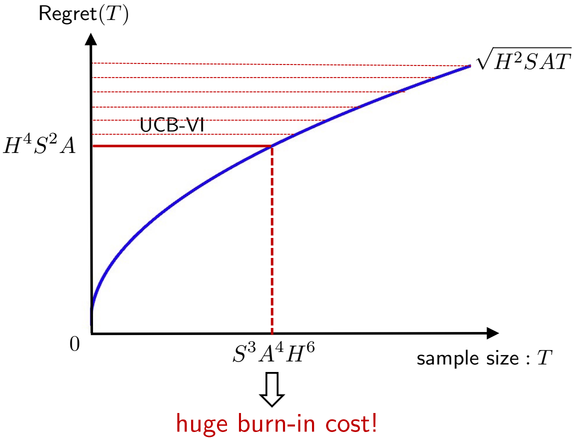
Prior art: Azar et al. '17

First method that is *asymptotically* regret-optimal: UCB-VI



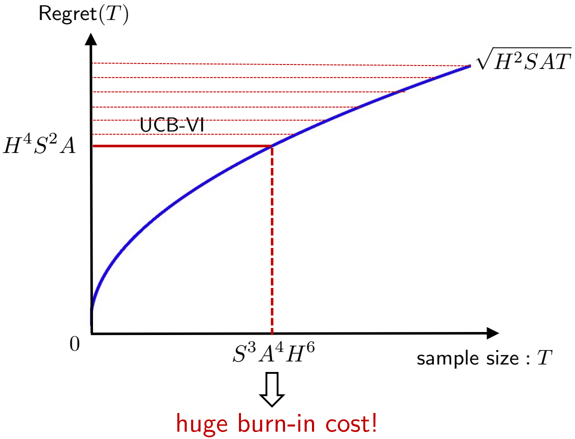
Prior art: Azar et al. '17

First method that is *asymptotically* regret-optimal: UCB-VI



Prior art: Azar et al. '17

First method that is *asymptotically* regret-optimal: UCB-VI



Issues: (1) large burn-in cost; (2) large memory complexity

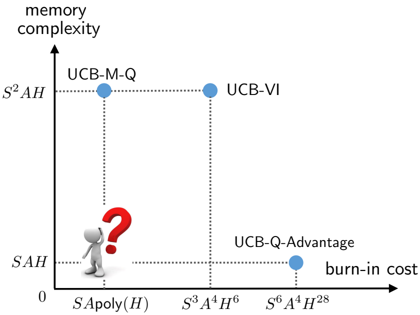
model-based: S^2AH

Prior art: other regret-optimal algorithms

Algorithm	Regret
UCB-VI (Azar et al., 2017)	$\sqrt{H^2SAT} + H^4S^2A$
UCB-M-Q (Menard et al., 2021)	$\sqrt{H^2SAT} + H^4SA$
UCB-Q-Advantage (Zhang et al., 2020)	$\sqrt{H^2SAT} + H^8S^2A^{\frac{3}{2}}T^{\frac{1}{4}}$

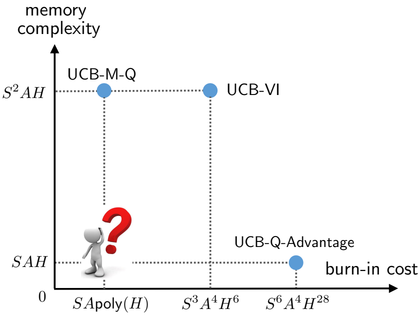
Prior art: other regret-optimal algorithms

Algorithm	Regret
UCB-VI (Azar et al., 2017)	$\sqrt{H^2SAT} + H^4S^2A$
UCB-M-Q (Menard et al., 2021)	$\sqrt{H^2SAT} + H^4SA$
UCB-Q-Advantage (Zhang et al., 2020)	$\sqrt{H^2SAT} + H^8S^2A^{\frac{3}{2}}T^{\frac{1}{4}}$



Prior art: other regret-optimal algorithms

Algorithm	Regret
UCB-VI (Azar et al., 2017)	$\sqrt{H^2SAT} + H^4S^2A$
UCB-M-Q (Menard et al., 2021)	$\sqrt{H^2SAT} + H^4SA$
UCB-Q-Advantage (Zhang et al., 2020)	$\sqrt{H^2SAT} + H^8S^2A^{\frac{3}{2}}T^{\frac{1}{4}}$



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

This work: an efficient model-free solution

Our algorithm: Q-EarlySettled-Advantage

Theorem 2 (Li, Shi, Chen, Gu, Chi, 2021)

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

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

with a memory complexity of $O(SAH)$

Our algorithm: Q-EarlySettled-Advantage

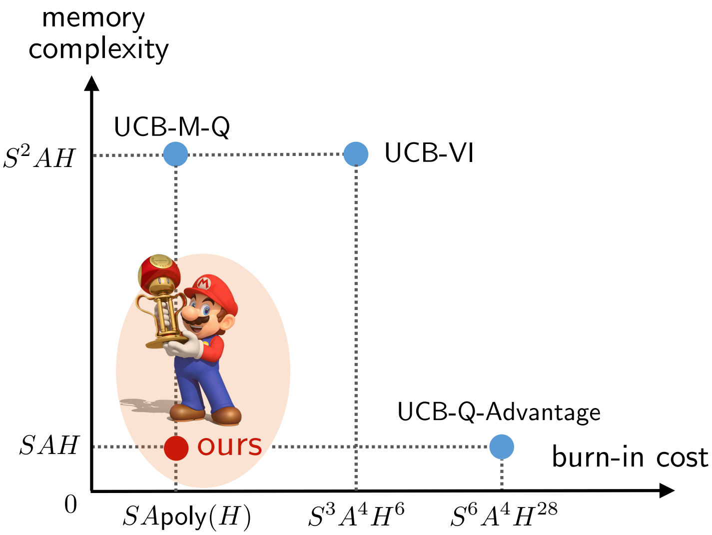
Theorem 2 (Li, Shi, Chen, Gu, Chi, 2021)

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

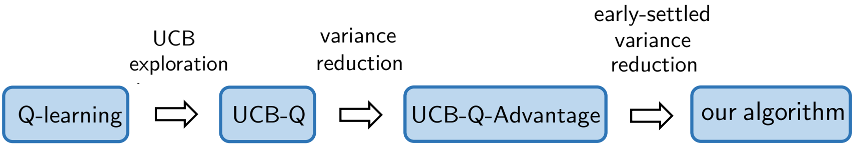
$$\text{Regret}(T) \lesssim \sqrt{H^2SAT} + H^6SA$$

with a memory complexity of $O(SAH)$

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


memory complexity
UCB-M-Q
UCB-VI
S^2AH
UCB-Q-Advantage
SAH
ours
burn-in cost
0
$SA\text{poly}(H)$
$S^3A^4H^6$
$S^6A^4H^{28}$

A glimpse of our algorithm design



A glimpse of our algorithm design

Q-learning: a classical model-free algorithm

Chris Watkins

Peter Dayan

Stochastic approximation for solving Bellman equation

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

Q-learning: a classical model-free algorithm

Chris Watkins

Peter Dayan

Stochastic approximation for solving Bellman equation

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

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

using sample in k-th episode

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

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

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

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

- $b_h(s,a)$: 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)

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

- $b_h(s,a)$: 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)

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

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

Regret$(T) \lesssim \sqrt{H^3 SAT}$ $\Longrightarrow$ 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 reference-advantage decomposition into UCB-Q:

$$Q_h(s_h, a_h) \leftarrow (1-\eta_k) Q_h(s_h, a_h) + \eta_k \underbrace{b_h(s_h, a_h)}_{\text{UCB bonus}}$$

$$+ \eta_k \Big(\underbrace{\mathcal{T}_k(Q_{h+1}) - \mathcal{T}_k(\overline{Q}_{h+1})}_{\text{advantage}} + \underbrace{\widehat{\mathcal{T}}(\overline{Q}_{h+1})}_{\text{reference}} \Big)(s_h, a_h)$$

- Reference $\overline{Q}_h$, batch estimate $\widehat{\mathcal{T}}(\overline{Q}_{h+1})$: help reduce variability

Q-learning with UCB and variance reduction

— *Zhang et al. '20*

Incorporates reference-advantage decomposition into UCB-Q:

$$Q_h(s_h, a_h) \leftarrow (1-\eta_k) Q_h(s_h, a_h) + \eta_k \underbrace{b_h(s_h, a_h)}_{\text{UCB bonus}}$$
$$+ \eta_k \Big(\underbrace{\mathcal{T}_k(Q_{h+1}) - \mathcal{T}_k(\overline{Q}_{h+1})}_{\text{advantage}} + \underbrace{\widehat{\mathcal{T}}(\overline{Q}_{h+1})}_{\text{reference}} \Big)(s_h, a_h)$$

- Reference $\overline{Q}_h$, batch estimate $\widehat{\mathcal{T}}(\overline{Q}_{h+1})$: 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:

$$Q_h(s_h,a_h) \leftarrow (1-\eta_k)Q_h(s_h,a_h) + \eta_k \underbrace{b_h(s_h,a_h)}_{\text{UCB bonus}} + \eta_k\Big(\underbrace{\mathcal{T}_k(Q_{h+1}) - \mathcal{T}_k(\overline{Q}_{h+1})}_{\text{advantage}} + \underbrace{\widehat{\mathcal{T}}(\overline{Q}_{h+1})}_{\text{reference}}\Big)(s_h,a_h)$$

- Reference $\overline{Q}_h$, batch estimate $\widehat{\mathcal{T}}(\overline{Q}_{h+1})$: help reduce variability

UCB-Q-Advantage is asymptotically regret-optimal

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

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 $\overline{Q}_h$

$\Downarrow$

Diagnosis of UCB-Q-Advantage

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

$\Downarrow$

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

$\Downarrow$

Diagnosis of UCB-Q-Advantage

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

$$\Downarrow$$

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

$$\Downarrow$$

Large burn-in cost

Diagnosis of UCB-Q-Advantage

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

$\Downarrow$

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

$\Downarrow$

Large burn-in cost

Diagnosis of UCB-Q-Advantage

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

⇓

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

⇓

Large burn-in cost

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

How to implement our early-settlement idea?

$$\overline{V}_h(s) - V_h^{\star}(s) \leq 1$$

How to implement our early-settlement idea?

$$\overline{V}_h(s) - V_h^{\star}(s) \leq 1$$

$$\Uparrow$$

$$\overline{V}_h(s) - V_h^{\mathsf{LCB}}(s) \leq 1 \quad \text{for some estimate } V_h^{\mathsf{LCB}} \leq V_h^{\star}$$

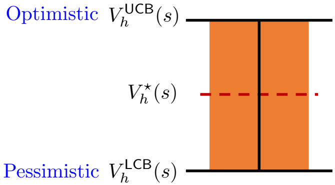
How to implement our early-settlement idea?

$$\overline{V}_h(s) - V_h^\star(s) \le 1$$

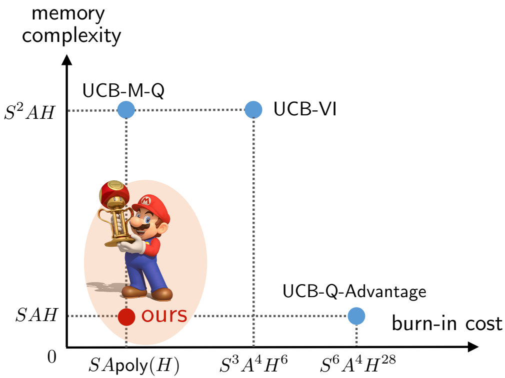
$$\Uparrow$$

$$\overline{V}_h(s) - V_h^{\mathsf{LCB}}(s) \le 1 \quad \text{for some estimate } V_h^{\mathsf{LCB}} \le V_h^\star$$

Q-EarlySettled-Advantage:
maintains auxiliary sequences V_h^{UCB} & V_h^{LCB} to help settle the reference early



Concluding remarks



Model-free algorithms can simultaneously achieve

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

Paper:

"Breaking the sample complexity barrier to regret-optimal model-free reinforcement learning," G. Li, L. Shi, Y. Chen, Y. Gu, Y. Chi, arXiv:2110.04645, NeurIPS 2021