The Blessing of Heterogeneity in Federated Q-Learning: Linear Speedup and Beyond

Jiin Woo

Carnegie Mellon University

August 2023

Gauri Joshi
CMU

Yuejie Chi
CMU

Reinforcement learning (RL)

In RL, an agent learns optimal decisions by interacting with an environment.

Real-world applications: autonomous driving, game, clinical trials, ...

Challenges: Data and computation

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

clinical trials

autonomous driving

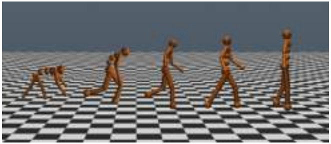
Challenges: Data and computation

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

clinical trials

autonomous driving

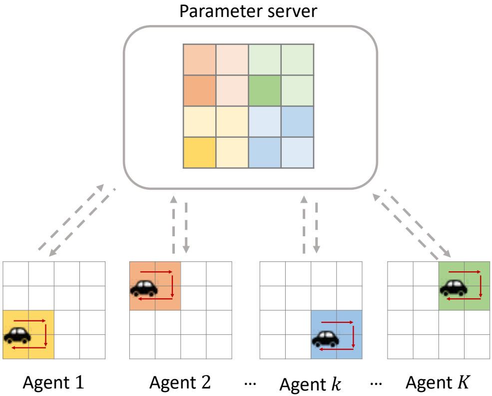
- Computational efficiency: Training RL algorithms might take a long time



many CPUs / GPUs / TPUs + computing hours

RL meets federated learning

Can we harness the power of federated learning?



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

This paper

Understand the sample efficiency of Q-learning in federated settings.

Linear speedup:

Can we achieve linear speedup when learning with multiple agents?

Communication efficiency:

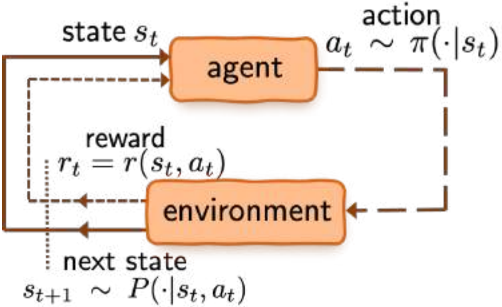
Can we perform multiple local updates to save communication?

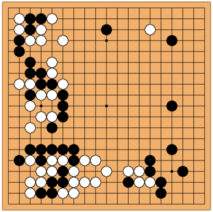
Taming heterogeneity:

How to combine heterogeneous local updates to accelerate learning?

Backgrounds:
Markov decision processes and Q-learning

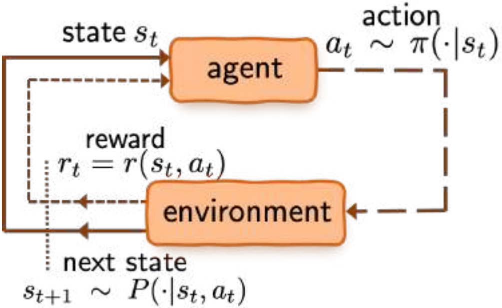
Markov decision process (MDP)

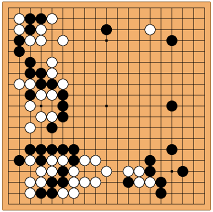




- $\mathcal{S}$: state space
- $\mathcal{A}$: action space

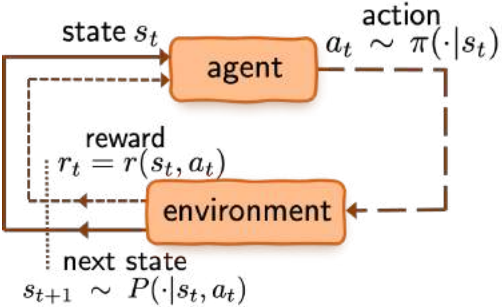
Markov decision process (MDP)

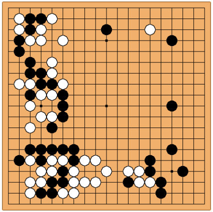




- $\mathcal{S}$: state space
- $\mathcal{A}$: action space
- $r(s, a) \in [0, 1]$: immediate reward

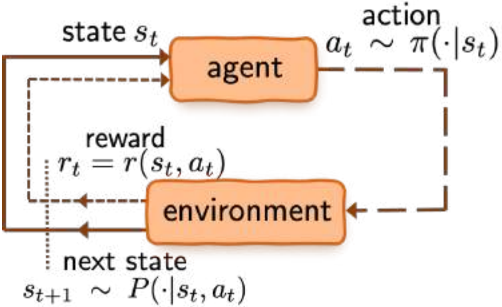
Markov decision process (MDP)

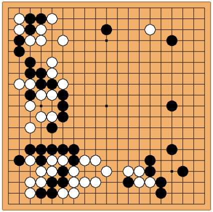




- $\mathcal{S}$: state space
- $\mathcal{A}$: action space
- $r(s, a) \in [0, 1]$: immediate reward
- $\pi(\cdot|s)$: policy (or action selection rule)

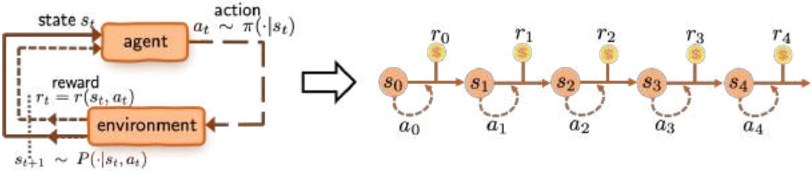
Markov decision process (MDP)





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

Value function



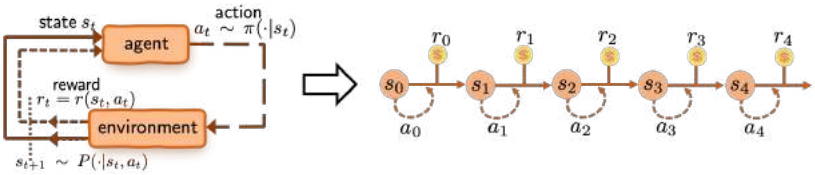
Value function of policy π:

$$\forall s \in \mathcal{S} : \qquad V^\pi(s) := \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r_t \,\middle|\, s_0 = s\right]$$

Q-function of policy π:

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

Value function



Value function of policy π:

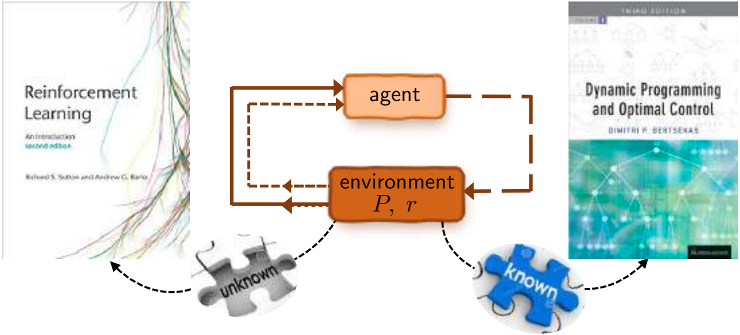
$$\forall s \in \mathcal{S}: \qquad V^\pi(s) := \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r_t \,\middle|\, s_0 = s\right]$$

Q-function of policy π:

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

- $\gamma \in [0,1)$ is the discount factor; $\frac{1}{1-\gamma}$ is effective horizon
- Expectation is w.r.t. the sampled trajectory under π

Searching for the optimal policy



Goal: find the optimal policy $\pi^\star$ that maximize $V^\pi(s)$

- optimal value / Q function: $V^\star := V^{\pi^\star}$, $Q^\star := Q^{\pi^\star}$
- optimal policy $\pi^\star(s) = \mathrm{argmax}_{a \in \mathcal{A}}\, Q^\star(s, a)$

Bellman's optimality principle

Bellman operator

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

- one-step look-ahead

Bellman's optimality principle

Bellman operator

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

- one-step look-ahead

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

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

γ-contraction of Bellman operator:

$$\|\mathcal{T}(Q_1) - \mathcal{T}(Q_2)\|_\infty \leq \gamma \|Q_1 - Q_2\|_\infty$$

Richard Bellman

Asynchronous Q-learning

Q-learning: Stochastic approximation for solving Bellman equation. With a transition sample (s_t, a_t, r_t, s_{t+1}), update Q_t as

$$Q_{t+1}(s_t, a_t) = (1-\eta)Q_t(s_t, a_t) + \eta \underbrace{(r_t + \gamma \max_{a' \in \mathcal{A}} Q_t(s_{t+1}, a'))}_{\mathcal{T}_t(Q_t)}, \quad t \geq 0$$

η: step size

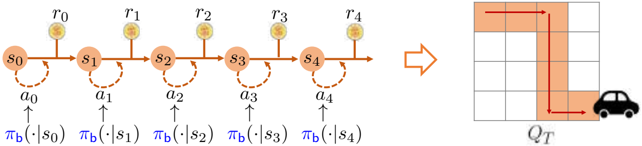
Asynchronous Q-learning

Q-learning: Stochastic approximation for solving Bellman equation. With a transition sample (s_t, a_t, r_t, s_{t+1}), update Q_t as

$$Q_{t+1}(s_t, a_t) = (1-\eta)Q_t(s_t, a_t) + \eta \underbrace{\left(r_t + \gamma \max_{a' \in \mathcal{A}} Q_t(s_{t+1}, a')\right)}_{\mathcal{T}_t(Q_t)}, \quad t \geq 0$$

η: step size

Asynchronous setting: Update single entry (s_t, a_t) along a *Markovian trajectory* generated by *behavior policy* π_{b}

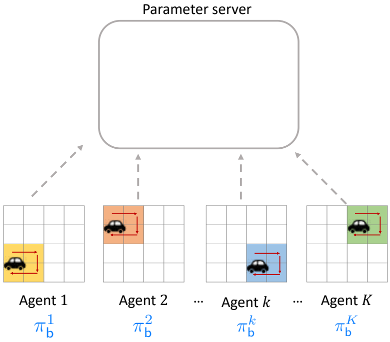


How to federate Q-learning?

Federated asynchronous Q-learning with local updates

- **Local update (agent):**
  Performs τ rounds of local Q-learning updates.

$$Q_{t+1}^{k}(s_t, a_t) \leftarrow (1-\eta)Q_t^{k}(s_t, a_t)+\eta\mathcal{T}_t(Q_t^{k})(s_t, a_t)$$



Local trajectories might be heterogeneous!

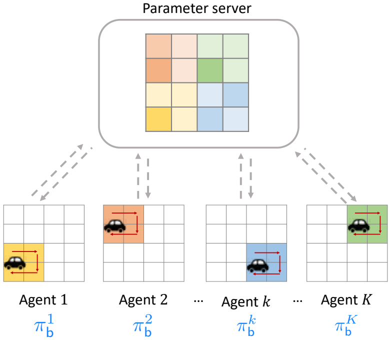
Federated asynchronous Q-learning with local updates

- **Local update (agent):**
  Performs τ rounds of local Q-learning updates.

$$Q^k_{t+1}(s_t, a_t) \leftarrow (1-\eta)Q^k_t(s_t, a_t) + \eta \mathcal{T}_t(Q^k_t)(s_t, a_t)$$

- **Periodic averaging (server):**
  Averages the local Q-tables.

$$Q_t = \frac{1}{K}\sum_{k=1}^{K} Q^k_t.$$



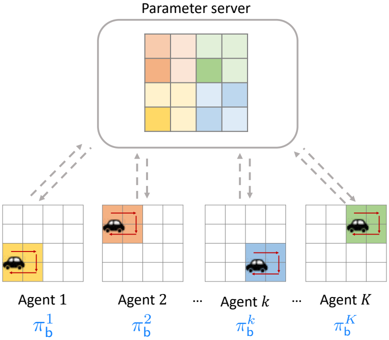
Federated asynchronous Q-learning with local updates

- **Local update (agent):**
  Performs τ rounds of local Q-learning updates.

$$Q_{t+1}^k(s_t, a_t) \leftarrow (1-\eta) Q_t^k(s_t, a_t) + \eta \mathcal{T}_t(Q_t^k)(s_t, a_t)$$

- **Periodic averaging (server):**
  Averages the local Q-tables.

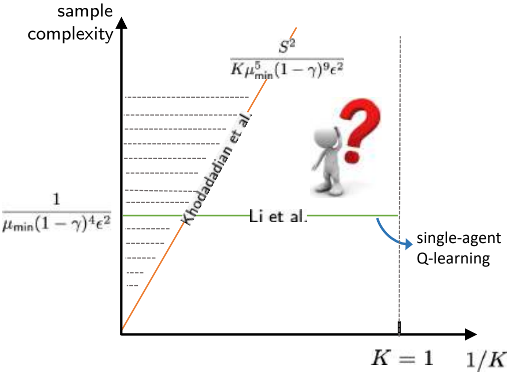
$$Q_t = \frac{1}{K} \sum_{k=1}^{K} Q_t^k.$$



Can we achieve faster convergence with heterogeneous local updates?

Sample complexity of federated Q-learning

Prior art



Unfavorable dependencies on salient problem parameters (γ, $\mu_{\min}$, $|\mathcal{S}|$)

Our theorem

Theorem (this work)

For sufficiently small $\epsilon > 0$, if τ is not too large, federated asynchronous Q-learning yields $\|\widehat{Q} - Q^\star\|_\infty \leq \epsilon$ with sample complexity at most

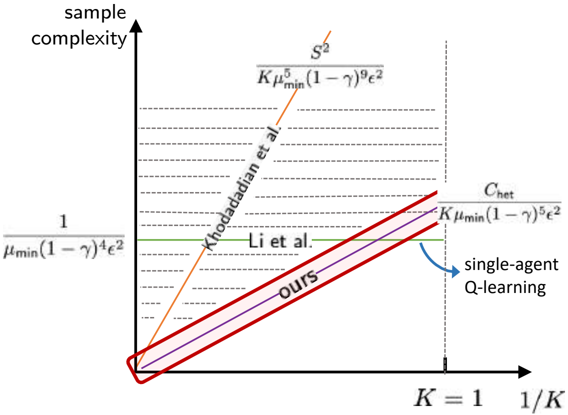
$$\widetilde{O}\left(\frac{C_{\mathsf{het}}}{K\mu_{\mathsf{min}}(1-\gamma)^5\epsilon^2}\right)$$

ignoring the burn-in cost that depends on the mixing times, where

$$\mu_{\mathsf{min}} := \min_{k,s,a} \underbrace{\mu_{\mathsf{b}}^k(s,a)}_{\text{stationary distribution}} \quad \text{and } C_{\mathsf{het}} := K \max_{k,s,a} \frac{\mu_{\mathsf{b}}^k(s,a)}{\sum_{k=1}^K \mu_{\mathsf{b}}^k(s,a)}.$$

- $1 \leq C_{\mathsf{het}} \leq \frac{1}{\mu_{\mathsf{min}}}$ measures the heterogeneity of local behavior policies.
- $C_{\mathsf{het}} \approx 1$ when the local behavior policies are similar.

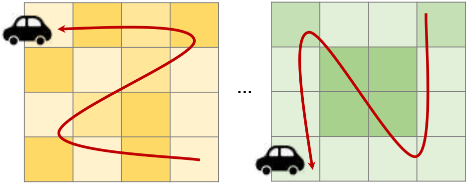
Near-optimal linear speedup



Linear speedup with near-optimal parameter dependencies!

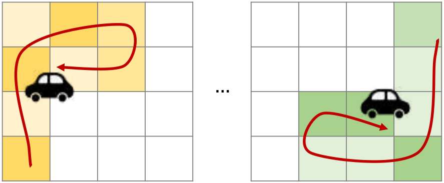
Curse of heterogeneity?

- **Full coverage:** The insufficient coverage of *just one* agent can significantly slow down the convergence (i.e. $\mu_{\text{min}} \approx 0$)



Curse of heterogeneity?

- **Full coverage:** The insufficient coverage of *just one* agent can significantly slow down the convergence (i.e. $\mu_{\text{min}} \approx 0$)
- **Curse of heterogeneity:** Performance degenerates when local behavior policies are heterogeneous (i.e. $C_{\text{het}} \gg 1$).

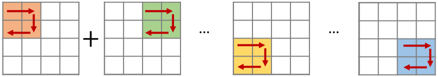


Is it possible to alleviate these limitations?

How to federate Q-learning
without the curse of heterogeneity?

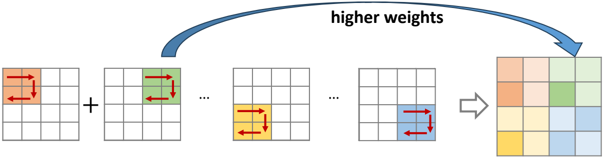
Importance averaging

Key observation: Not all updates are of same quality due to limited visits induced by the behavior policy.



Importance averaging

Key observation: Not all updates are of same quality due to limited visits induced by the behavior policy.



Importance averaging: Averages the local Q-values assigning higher weights on more frequently updated local values via

$$Q_t(s,a) = \sum_{k=1}^{K} \alpha_t^k(s,a) Q_t^k(s,a),$$

where

$$\alpha_t^k = \frac{(1-\eta)^{-N_{t-\tau,t}^k(s,a)}}{\sum_{k=1}^{K}(1-\eta)^{-N_{t-\tau,t}^k(s,a)}}, \quad N_{t-\tau,t}^k(s,a) = \text{number of visits in the sync period}.$$

Sample complexity of federated Q-learning with importance averaging

Our theorem

Theorem (this work)

For sufficiently small $\epsilon > 0$, if τ is not too large, federated asynchronous Q-learning with importance averaging yields $\|\widehat{Q} - Q^\star\|_\infty \leq \epsilon$ with sample complexity at most

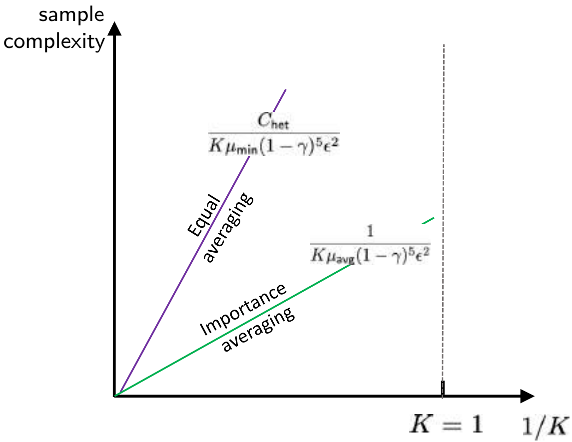
$$\widetilde{O}\left(\frac{1}{K\mu_{\mathsf{avg}}(1-\gamma)^5\epsilon^2}\right)$$

ignoring the burn-in cost that depends on the mixing times, where

$$\mu_{\mathsf{avg}} = \min_{s,a} \frac{1}{K}\sum_{k=1}^{K} \mu_{\mathsf{b}}^k(s,a).$$

- No performance degeneration due to heterogeneity (C_{het}).
- Near-optimal linear speedup.

Equal averaging versus importance averaging

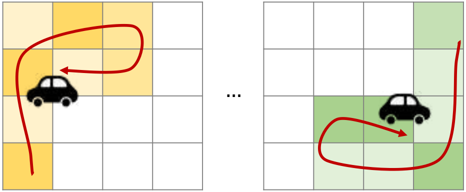


Faster convergence: $\mu_{\mathsf{avg}} \geq \mu_{\mathsf{min}}$

Partial-coverage

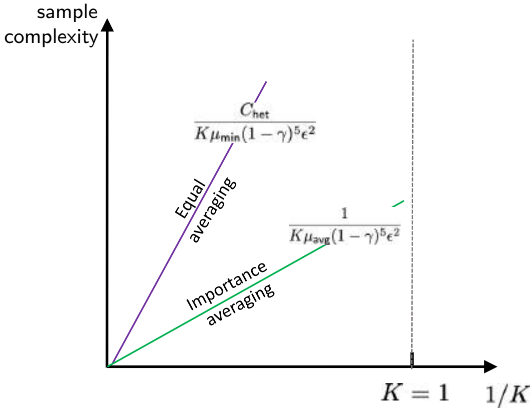
Partial coverage is enough as long as agents collectively cover the entire state-action space, i.e.,

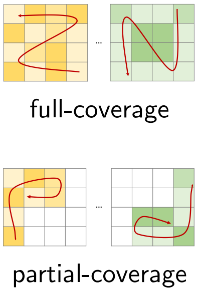
$$\mu_{\mathsf{avg}} = \min_{s,a} \frac{1}{K} \sum_{k=1}^{K} \mu_{\mathsf{b}}^{k}(s,a) > 0$$



No longer require full coverage of every individual agent!

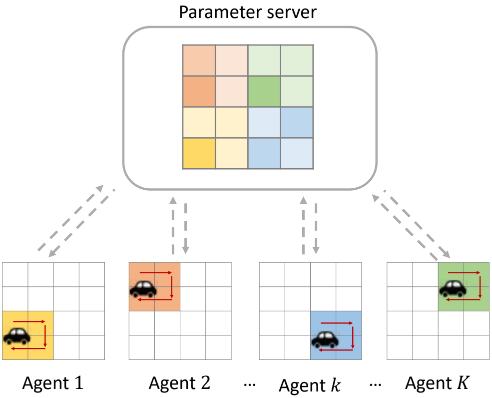
Blessing of heterogeneity





Overcome the insufficient coverage of individual agents
by exploiting heterogeneity!

Final remarks



Near-optimal linear speedup of federated Q-learning without full coverage of individual agents!

Thanks!

- The Blessing of Heterogeneity in Federated Q-Learning: Linear Speedup and Beyond, ICML 2023. (arXiv: 2305.10697)