

The Blessing of Heterogeneity in Federated Reinforcement Learning

Yuejie Chi

Yale Statistics & Data Science



Rutgers

October 2026



Jiin Woo
CMU→Amazon



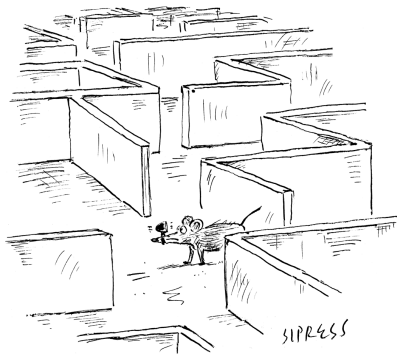
Sudeep Salgia
CMU→Meta



Gauri Joshi
CMU

Reinforcement learning (RL)

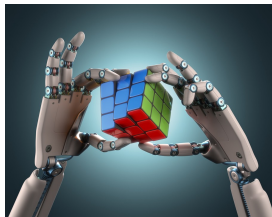
In RL, an agent learns by interacting with an *unknown* environment through trial-and-error to maximize long-term total reward.



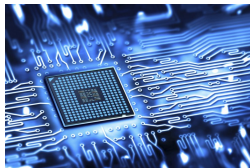
"Recalculating ... recalculating ..."



More successes of RL since AlphaGo



robotics



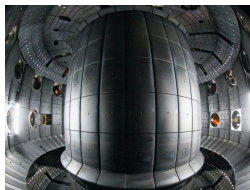
chip designs



resource management



strategic games

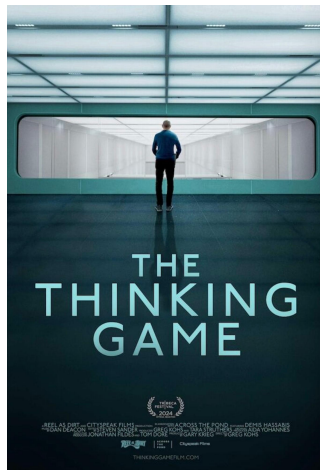


nuclear plant control



UAV and drones

One more: RL for thinking models



Turing Award Goes to 2 Pioneers of Artificial Intelligence

Andrew Barto and Richard Sutton developed reinforcement learning, a technique vital to chatbots like ChatGPT.



RL holds great promise in accelerating scientific, engineering and societal discoveries.

Sample efficiency

However, collecting data samples might be expensive or time-consuming.



clinical trials

Prompt:
Should I add chorizo
to my paella?

Response 1: Absolutely! ...
Response 2: In Valencian...

Feedback (ranking):
Response 1 is better than 2

LLM alignment



autonomous driving

Sample efficiency

However, collecting data samples might be expensive or time-consuming.



clinical trials

Prompt:
Should I add chorizo
to my paella?

Response 1: Absolutely! ...
Response 2: In Valencian...

Feedback (ranking):
Response 1 is better than 2

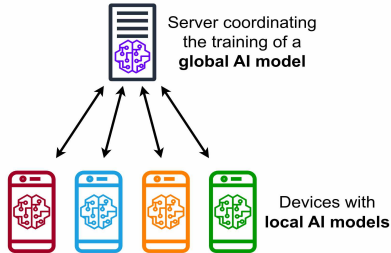
LLM alignment



autonomous driving

Calls for design of sample-efficient RL algorithms!

Can we harness the power of federated learning?



FORBES > INNOVATION > AI

IBM Federated Learning Research - Extracting Machine Learning Models From Multiple Data Pools

Kevin Krawiec Contributor
Tirias Research Contributor Group 0

Follow

Oct 18, 2020, 02:51pm EDT

How Apple personalizes Siri without hoovering up your data

The tech giant is using privacy-preserving machine learning to improve its voice assistant while keeping your data on your phone.

By Karen Hao

December 11, 2019

Five rival drugmakers trained one AI without sharing their data, and it beat the open model

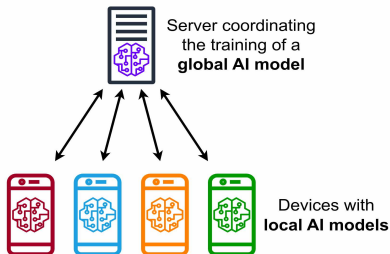


Edited By Tarun Mishra

Published: Oct 02, 2020, 04:30 IST | Updated: Oct 02, 2020, 04:30 IST



Can we harness the power of federated learning?



FORBES > INNOVATION > AI

IBM Federated Learning Research - Extracting Machine Learning Models From Multiple Data Pools

Kevin Krawell Contributor
Tirias Research Contributor Group 0

Follow

Oct 18, 2020, 02:51pm EDT

How Apple personalizes Siri without hoovering up your data

The tech giant is using privacy-preserving machine learning to improve its voice assistant while keeping your data on your phone.

By Karen Hao
December 11, 2019

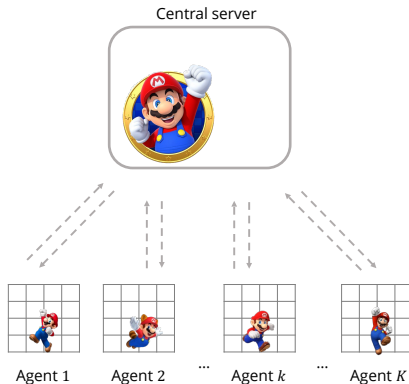
Five rival drugmakers trained one AI without sharing their data, and it beat the open model

Edited By Tarun Mishra
Published: Oct 02, 2020, 04:30 IST | Updated: Oct 02, 2020, 04:30 IST



Can we harness the power of federated learning for RL?

RL meets federated learning



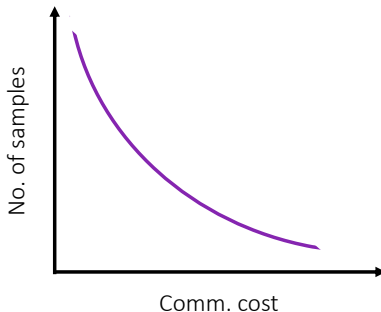
Federated reinforcement learning: enables multiple agents to collaboratively learn a global policy without sharing datasets.

Statistical-communication trade-offs

Statistical
benefits

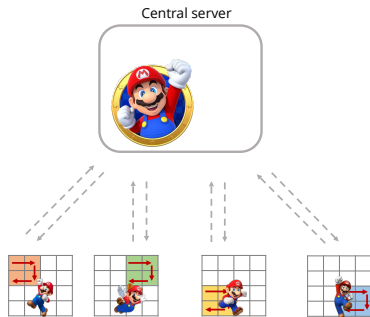


Communication
overhead

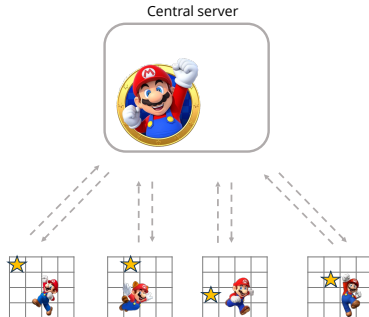


Is linear speedup possible? What is the price in communication?

Dealing with heterogeneity



Data heterogeneity



Task heterogeneity

This talk: federated RL

Statistical
benefits



Communication
overhead

Linear speedup:

Can we achieve linear speedup when learning with multiple agents?

Communication efficiency:

What is the minimum amount of communication to achieve speedup?

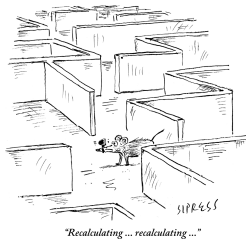
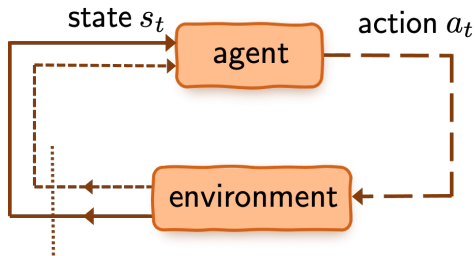
Taming heterogeneity:

What if the data and tasks across agents are heterogeneous?

*Backgrounds:
Markov decision processes*

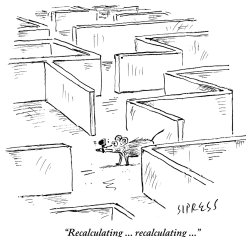
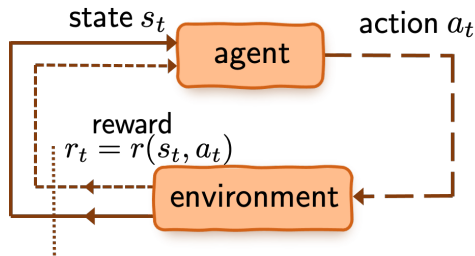


Markov decision process (MDP)



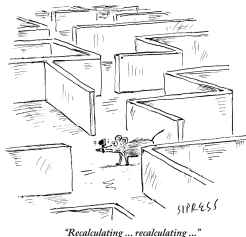
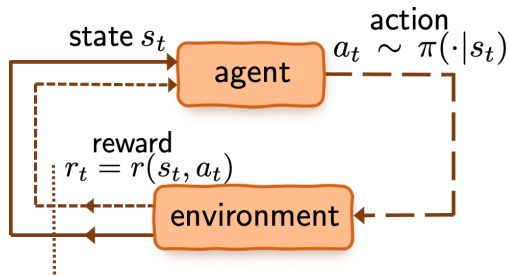
- $\mathcal{S}$: state space
- $\mathcal{A}$: action space

Markov decision process (MDP)



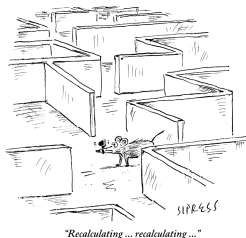
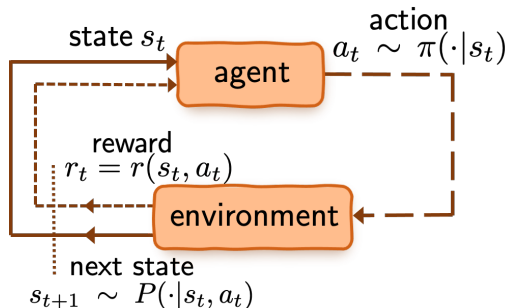
- $\mathcal{S}$: state space
- $\mathcal{A}$: action space
- $r(s, a) \in [0, 1]$: immediate reward

Markov decision process (MDP)



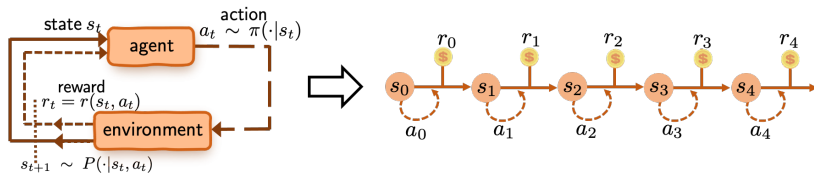
- $\mathcal{S}$: state space
- $\mathcal{A}$: action space
- $r(s, a) \in [0, 1]$: immediate reward
- $\pi(\cdot | s)$: policy (or action selection rule)

Markov decision process (MDP)

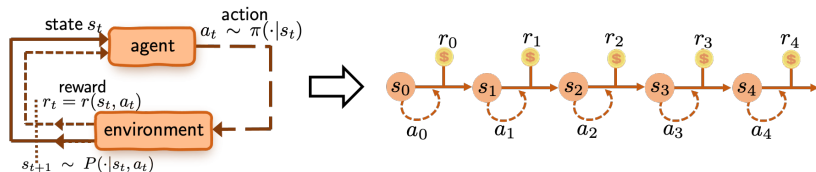


- $\mathcal{S}$: state space
- $\mathcal{A}$: action space
- $r(s, a) \in [0, 1]$: immediate reward
- $\pi(\cdot|s)$: policy (or action selection rule)
- $P(\cdot|s, a)$: transition probabilities

Goal: value maximization



Goal: value maximization



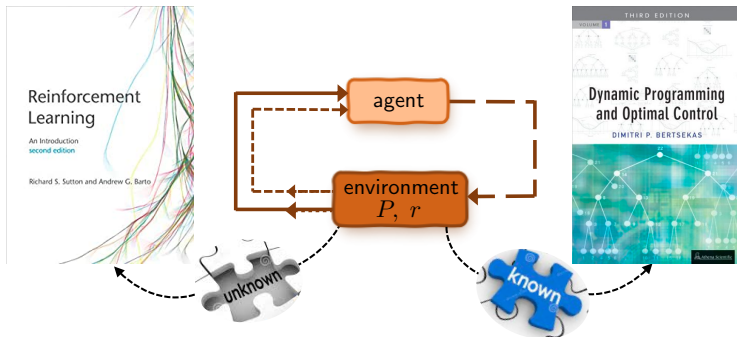
Value function and **Q-function** of policy π :

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

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

- $\gamma \in [0, 1)$ 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



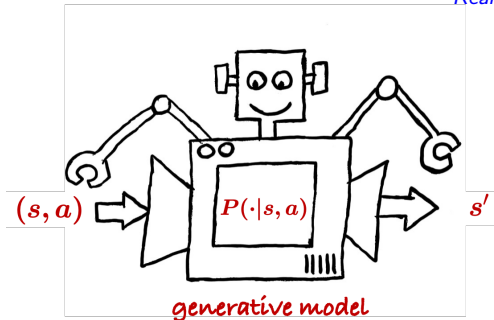
Goal: find the optimal policy π^* that maximize $V^{\pi}(s)$

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

*Federated Q-learning: statistical and
communication complexities*

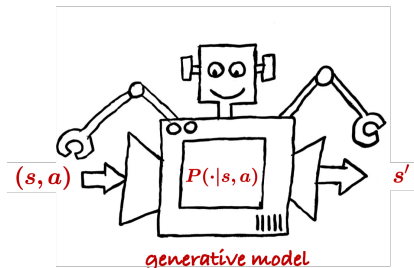
A generative model / simulator

— Kearns & Singh, 1999



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

Q-learning with a generative model

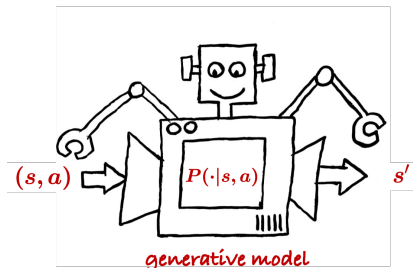


Stochastic approximation for solving Bellman equation $Q^* = \mathcal{T}(Q^*)$ using samples collected from the generative model:

$$Q_{t+1}(s, a) = (1 - \eta)Q_t(s, a) + \eta \mathcal{T}_t(Q_t)(s, a), \quad t \geq 0$$

draw the transition (s, a, s') for all (s, a)

Q-learning with a generative model



Stochastic approximation for solving Bellman equation $Q^* = \mathcal{T}(Q^*)$ using samples collected from the generative model:

$$\underbrace{Q_{t+1}(s, a) = (1 - \eta)Q_t(s, a) + \eta \mathcal{T}_t(Q_t)(s, a)}_{\text{draw the transition } (s, a, s') \text{ for all } (s, a)}, \quad t \geq 0$$

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

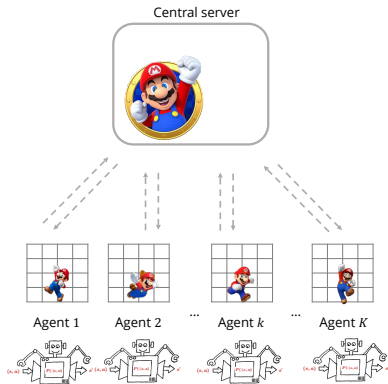
$$\mathcal{T}(Q)(s, a) = r(s, a) + \gamma \mathbb{E}_{s' \sim P(\cdot|s, a)} \left[\max_{a'} Q(s', a') \right]$$

Federated Q-learning with local updates

- **Local updates:** the agent k performs τ rounds of local Q-learning updates:

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

and sends it to the server.



Federated Q-learning with local updates

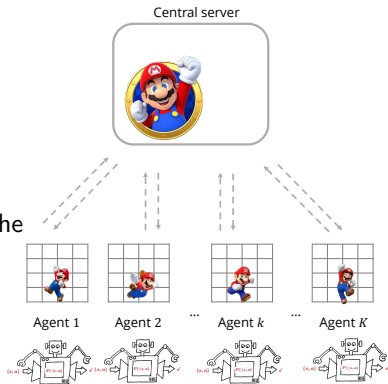
- **Local updates:** the agent k performs τ rounds of local Q-learning updates:

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

and sends it to the server.

- **Periodic averaging:** the server averages the local updates and sends it back to agents:

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



Federated Q-learning with local updates

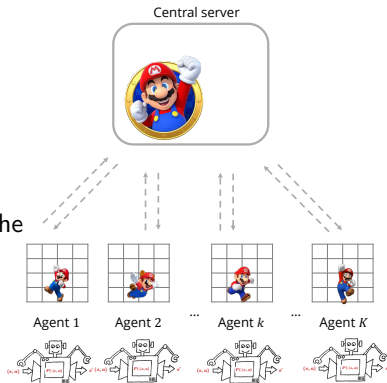
- **Local updates:** the agent k performs τ rounds of local Q-learning updates:

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

and sends it to the server.

- **Periodic averaging:** the server averages the local updates and sends it back to agents:

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



Can we achieve faster convergence, i.e. linear speedup, with low communication complexity?

Federated Q-learning with local updates

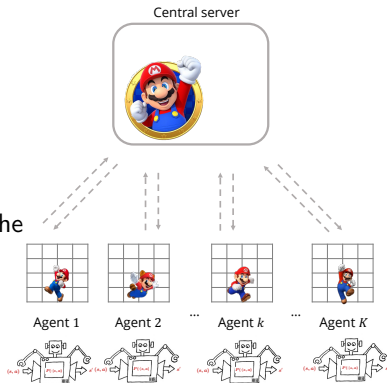
- **Local updates:** the agent k performs τ rounds of local Q-learning updates:

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

and sends it to the server.

- **Periodic averaging:** the server averages the local updates and sends it back to agents:

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



Can we achieve faster convergence, i.e. linear speedup, with low communication complexity?

Yes!!

Linear speedup of federated Q-learning

Theorem (Jiin, Joshi, Chi, ICML 2023)

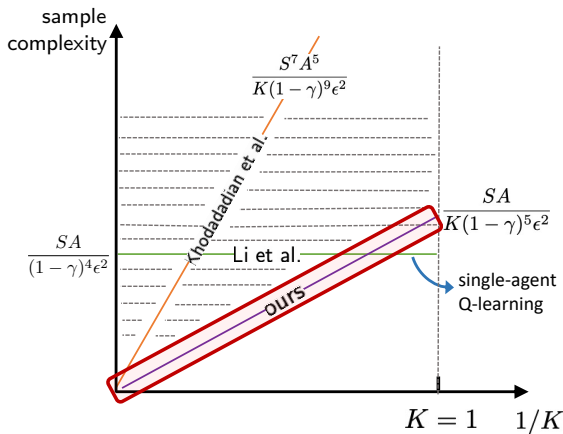
For any $0 < \varepsilon \leq \frac{1}{1-\gamma}$, federated synchronous Q-learning yields $\|\widehat{Q} - Q^*\|_\infty \leq \varepsilon$ with sample complexity *at most*

$$\tilde{O}\left(\frac{|\mathcal{S}||\mathcal{A}|}{K(1-\gamma)^5\varepsilon^2}\right)$$

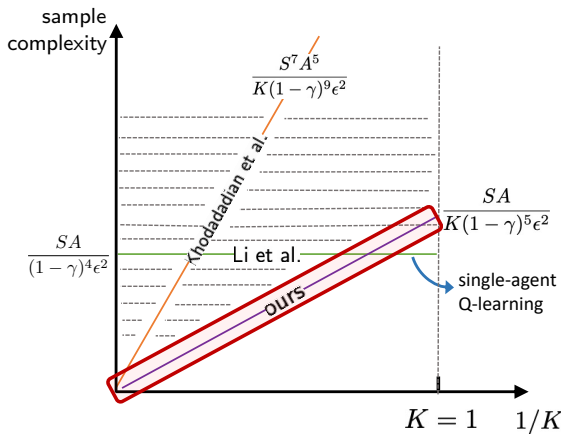
as long as $\tau - 1 \leq \frac{1}{\eta} \min\left\{\frac{1-\gamma}{8\gamma}, \frac{1}{K}\right\}$ and $\eta = \tilde{O}(K(1-\gamma)^4\varepsilon^2)$.

- **Linear speedup** compared with the single-agent sample complexity $\tilde{O}\left(\frac{|\mathcal{S}||\mathcal{A}|}{(1-\gamma)^4\varepsilon^2}\right)$.
- Similar speedup for *average-reward* MDPs, as well (Jiao et al., 2026).

Comparison with prior art

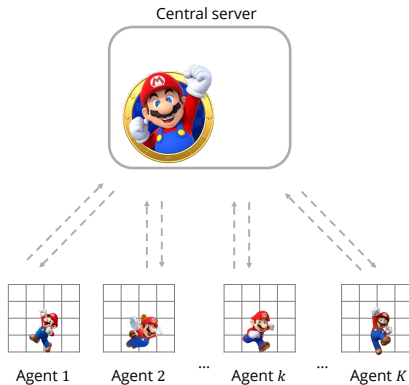


Comparison with prior art



Linear speedup with near-optimal parameter dependencies!

Communication bottleneck



The price of communication: how much communication do we need to pay to achieve the linear speedup?

A communication lower bound

Theorem (Salgia and Chi, NeurIPS 2024; informal)

For a wide family of federated Q-learning algorithm with intermittent communication, regardless of the choice of synchronization schedules, the number of communication rounds needs to be at least

$$\tilde{\Omega}\left(\frac{1}{1-\gamma}\right)$$

*in order to achieve **any speedup** with respect to the number of agents.*

- A similar lower bound holds for the number of communication *bits*.
- This is the first communication complexity barrier established for federated RL algorithms.

Key idea

$$\mathbb{E}[(\widehat{Q} - Q^*)^2] = \underbrace{\mathbb{E}[(\mathbb{E}[\widehat{Q}] - Q^*)^2]}_{\text{Bias}} + \underbrace{\mathbb{E}[(\widehat{Q} - \mathbb{E}[\widehat{Q}])^2]}_{\text{Variance}}$$

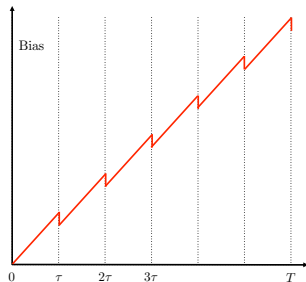
- **Variance** exhibits linear speedup on average.
- **Bias** increases between two communication rounds. Averaging has a small compensating effect, but the overall bias is independent of K .

Key idea

$$\mathbb{E}[(\widehat{Q} - Q^*)^2] = \underbrace{\mathbb{E}[(\mathbb{E}[\widehat{Q}] - Q^*)^2]}_{\text{Bias}} + \underbrace{\mathbb{E}[(\widehat{Q} - \mathbb{E}[\widehat{Q}])^2]}_{\text{Variance}}$$

- **Variance** exhibits linear speedup on average.
- **Bias** increases between two communication rounds. Averaging has a small compensating effect, but the overall bias is independent of K .

$\text{Bias} \propto \tau = \Omega(T(1 - \gamma))$
 $\Downarrow$
bias dominates the **variance**
 $\Downarrow$
no collaboration gain

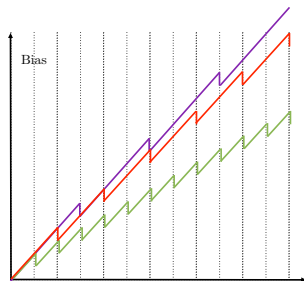


Key idea

$$\mathbb{E}[(\widehat{Q} - Q^*)^2] = \underbrace{\mathbb{E}[(\mathbb{E}[\widehat{Q}] - Q^*)^2]}_{\text{Bias}} + \underbrace{\mathbb{E}[(\widehat{Q} - \mathbb{E}[\widehat{Q}])^2]}_{\text{Variance}}$$

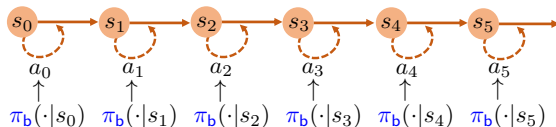
- **Variance** exhibits linear speedup on average.
- **Bias** increases between two communication rounds. Averaging has a small compensating effect, but the overall bias is independent of K .

$\text{Bias} \propto \tau = \Omega(T(1 - \gamma))$
 $\Downarrow$
bias dominates the **variance**
 $\Downarrow$
no collaboration gain



The blessing of data heterogeneity in federated RL

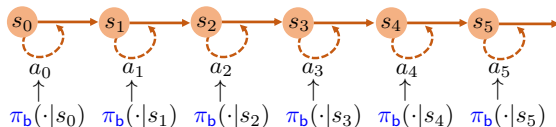
Q-learning following a behavior policy



Stochastic approximation for solving Bellman equation $Q^* = \mathcal{T}(Q^*)$ using samples collected from a behavior policy π_b :

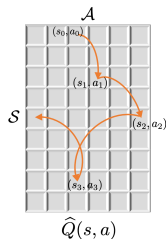
$$Q_{t+1}(s_t, a_t) = \underbrace{(1 - \eta)Q_t(s_t, a_t) + \eta \mathcal{T}_t(Q_t)(s_t, a_t)}_{\text{only update } (s_t, a_t)\text{-th entry}}, \quad t \geq 0$$

Q-learning following a behavior policy



Stochastic approximation for solving Bellman equation $Q^* = \mathcal{T}(Q^*)$ using samples collected from a behavior policy π_b :

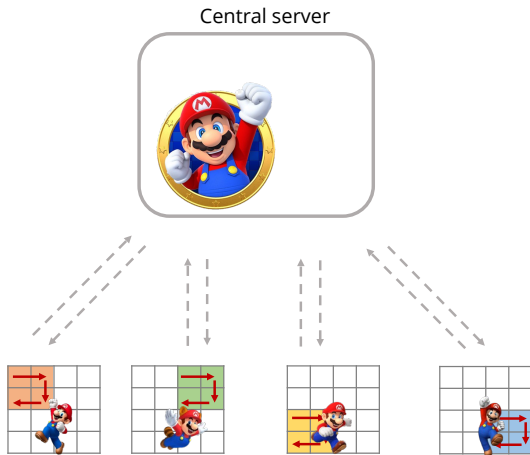
$$Q_{t+1}(s_t, a_t) = \underbrace{(1 - \eta)Q_t(s_t, a_t) + \eta \mathcal{T}_t(Q_t)(s_t, a_t)}_{\text{only update } (s_t, a_t)\text{-th entry}}, \quad t \geq 0$$



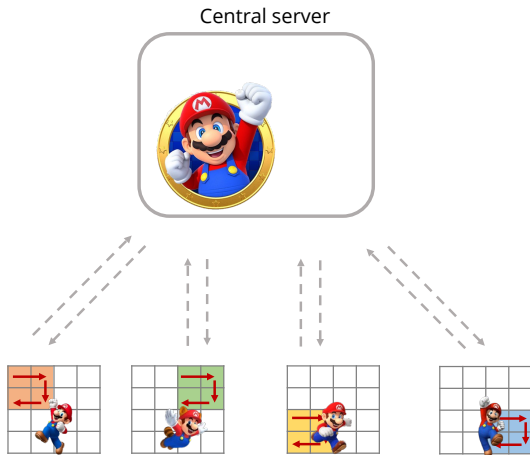
$$\mathcal{T}_t(Q)(s_t, a_t) = r(s_t, a_t) + \gamma \max_{a'} Q(s_{t+1}, a')$$

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

Tackling data heterogeneity



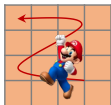
Tackling data heterogeneity



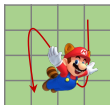
Can we achieve faster convergence with heterogeneous local behavior policies with low communication complexity?

The benefit of collaboration?

Single-agent RL requires **full coverage** of every agent over the entire state-action space...

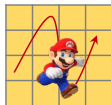


Agent 1



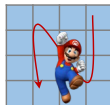
Agent 2

...



Agent k

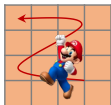
...



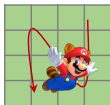
Agent K

The benefit of collaboration?

Single-agent RL requires **full coverage** of every agent over the entire state-action space...

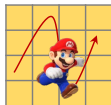


Agent 1



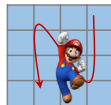
Agent 2

...



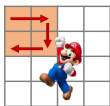
Agent k

...

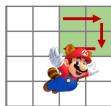


Agent K

However, the power of collaboration really shines if we only need...

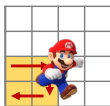


Agent 1



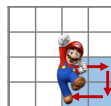
Agent 2

...



Agent k

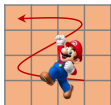
...



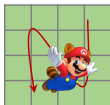
Agent K

The benefit of collaboration?

Single-agent RL requires **full coverage** of every agent over the entire state-action space...

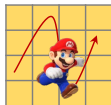


Agent 1



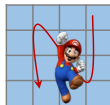
Agent 2

...



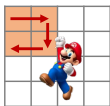
Agent k

...

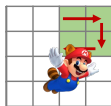


Agent K

However, the power of collaboration really shines if we only need...

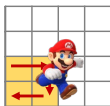


Agent 1



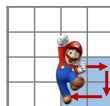
Agent 2

...



Agent k

...



Agent K

Is collaborative coverage enough for federated Q-learning?

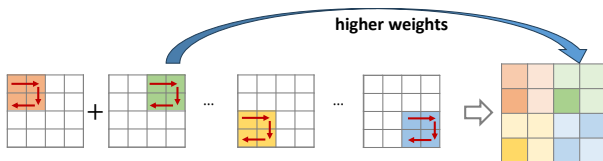
Importance averaging

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: the server averages the local updates based on importance via

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

where

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

Our theorem

Collaborative coverage: minimum entry of the average stationary distribution

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

Theorem (Woo, Joshi, Chi, JMLR 2025)

For sufficiently small $\varepsilon > 0$, federated asynchronous Q-learning *with importance averaging* yields $\|\widehat{Q} - Q^*\|_{\infty} \leq \varepsilon$ with at most

$$\tilde{O}\left(\frac{1}{K\mu_{\text{avg}}(1-\gamma)^5\varepsilon^2}\right)$$

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

Our theorem

Collaborative coverage: minimum entry of the average stationary distribution

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

Theorem (Woo, Joshi, Chi, JMLR 2025)

For sufficiently small $\varepsilon > 0$, federated asynchronous Q-learning *with importance averaging* yields $\|\widehat{Q} - Q^*\|_{\infty} \leq \varepsilon$ with at most

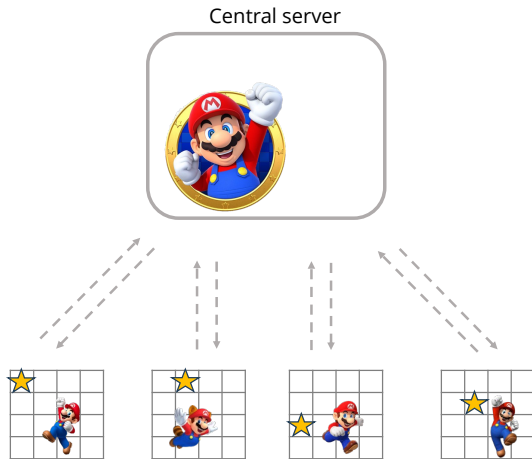
$$\tilde{O}\left(\frac{1}{K\mu_{\text{avg}}(1-\gamma)^5\varepsilon^2}\right)$$

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

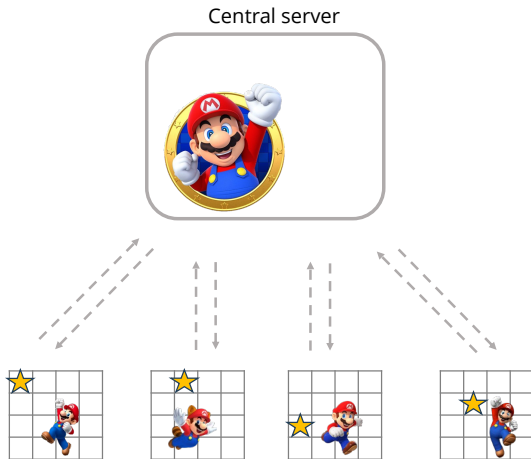
- Similar results can be developed for the offline setting, too.

*Task heterogeneity (aka Personalization) in
federated RL*

Tackling task heterogeneity

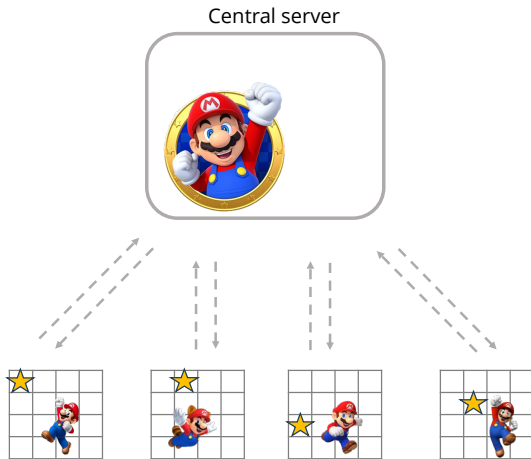


Tackling task heterogeneity



- All agents share the same state-action transition kernel P .
- Each agent collects its own *private* reward w.r.t. different local tasks.

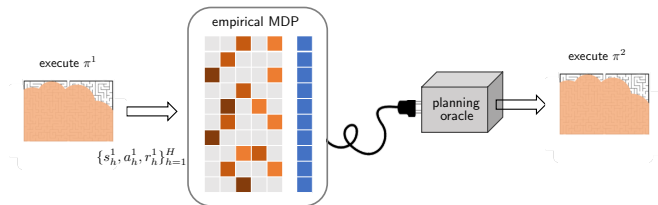
Tackling task heterogeneity



Can we enable collaborative learning of heterogeneous tasks while achieving linear speedup?

Personalized federated UCB-VI

Server updates: aggregate global visitation counts $N_h(s, a)$ and kernel estimate $\widehat{P}_{h,s,a}$.



Local exploration via personalized UCB: apply **optimistic** value iteration using the greedy policy $\pi_h^k = \text{greedy}(Q_h^k)$:

$$Q_h^k(s, a) \leftarrow \min \left\{ H - h + 1, \underbrace{r_h^k(s_h, a_h)}_{\text{immediate reward}} + \underbrace{\widehat{P}_{h,s,a} V_{h+1}^k}_{\text{next step's value}} + \underbrace{b_h^k(s_h, a_h)}_{\text{bonus}} \right\},$$

with $V_h(s) = \max_{a \in \mathcal{A}} Q_h(s, a)$ and $b_h^k(s_h, a_h) \asymp \sqrt{\frac{\text{Var}_{\widehat{P}_{h,s,a}}(V_{h+1}^k)}{N_h(s, a)}}$.

Our theorem

Regret: let π_t^k is the policy of agent k at episode t .

$$\text{Regret}(T) = \sum_{k=1}^K \sum_{t=1}^T \underbrace{\mathbb{E}_{s_1 \sim \rho} \left[V_1^{k,*}(s_1) - V_1^{k,\pi_t^k}(s_1) \right]}_{\text{personalized gap of agent } k}.$$

Theorem (Woo, Joshi, Chi, NeurIPS 2026)

Let the agents perform the personalized federated UCB-VI (PF-UCBVI) algorithm with a confidence parameter $\delta \in (0, 1)$. Then, with probability at least $1 - \delta$, the total regret satisfies:

$$\text{Regret}(T) \lesssim \sqrt{SAKTH^3},$$

up to logarithmic factors and burn-in cost.

Our theorem

Regret: let π_t^k is the policy of agent k at episode t .

$$\text{Regret}(T) = \sum_{k=1}^K \underbrace{\sum_{t=1}^T \mathbb{E}_{s_1 \sim \rho} \left[V_1^{k,*}(s_1) - V_1^{k,\pi_t^k}(s_1) \right]}_{\text{personalized gap of agent } k}.$$

Theorem (Woo, Joshi, Chi, NeurIPS 2026)

Let the agents perform the personalized federated UCB-VI (PF-UCBVI) algorithm with a confidence parameter $\delta \in (0, 1)$. Then, with probability at least $1 - \delta$, the total regret satisfies:

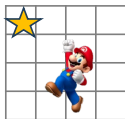
$$\text{Regret}(T) \lesssim \sqrt{SAKTH^3},$$

up to logarithmic factors and burn-in cost.

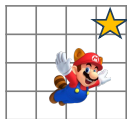
- Matches the UCB-VI performance but allows personalized regret.
- Linear speedup of average per-agent regret $\mathcal{O}(\sqrt{SAKTH^3/K})$.

The blessing of reward diversity

One agent's exploitation can become another agent's exploration!

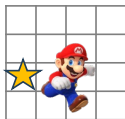


Agent 1



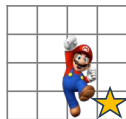
Agent 2

...



Agent k

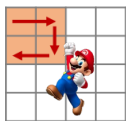
...



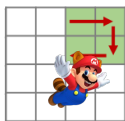
Agent K

The blessing of reward diversity

One agent's exploitation can become another agent's exploration!

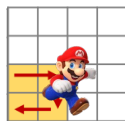


Agent 1



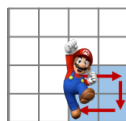
Agent 2

...



Agent k

...

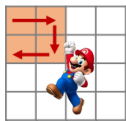


Agent K

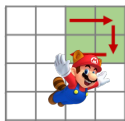
- The greedy action of all agents can collectively lead to sufficient coverage of the entire state-action space.

The blessing of reward diversity

One agent's exploitation can become another agent's exploration!

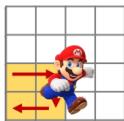


Agent 1



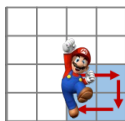
Agent 2

...



Agent k

...



Agent K

- The greedy action of all agents can collectively lead to sufficient coverage of the entire state-action space.

Can reward diversity replace the need of exploration?

Our theorem

Reward-induced coverage: Let $\tilde{\pi}^k$ be the greedy policy induced by any permissible state-action value function $\tilde{Q}_h^k$. There exists a strictly positive constant $\lambda_{\min} > 0$ such that

$$\min_{(s,a) \in \mathcal{S} \times \mathcal{A}} \frac{1}{K} \sum_{k=1}^K d_h^{\tilde{\pi}^k}(s, a) \geq \lambda_{\min},$$

where $d_h^{\tilde{\pi}^m}$ is the visitation probability of $\tilde{\pi}^m$ under *true* P .

Theorem (Woo, Joshi, Chi, NeurIPS 2026)

With probability at least $1 - \delta$, the total regret of PF-EFVI (removing the exploration bonus from PF-UCBVI) satisfies:

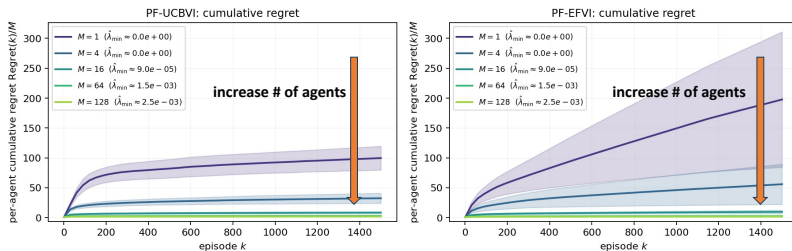
$$\text{Regret}(T) \lesssim \frac{H^5 \log(SAHKT/\delta)}{K \lambda_{\min}} \sum_{k=1}^K \frac{1}{\Delta_{\min}^k} + \frac{H \log(4SAKH/\delta)}{\lambda_{\min}^2},$$

where $\Delta_{\min}^k = \min_{(s,a,h): \Delta_h^k(s,a) > 0} (V_h^{k,*}(s) - Q_h^{k,*}(s,a))$ for agent k is the instance-dependent sub-optimality gap.

- Under sufficient reward diversity, the regret grows as $\tilde{O}(\log T)$.

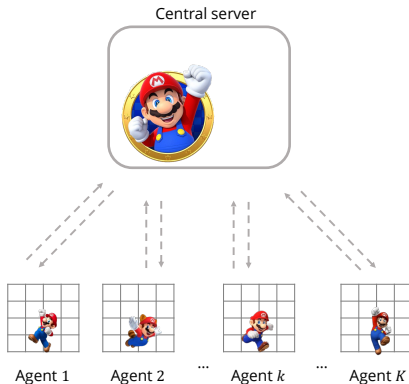
Numerical illustration

Random MDP $P_h(\cdot | s, a) \sim \text{Dirichlet}(\alpha \cdot \mathbf{1})$, and each agent assigns a random subset of (s, a, h) with reward 1 (and 0 elsewhere).



- While exploration is indispensable when the number of agents is small, enough reward diversity makes it possible to reduce active exploration – **research opportunity between the two regimes!**

Summary



Synergy of statistics and RL: federated RL unleashes the collaborative power of agents even under heterogeneity!

Thanks!

- Exploration via Exploitation: The Blessing of Reward Diversity in Personalized Federated RL, *NeurIPS 2026*.
- Sample Complexity of Average-Reward Q-Learning: From Single-agent to Federated Reinforcement Learning, *arXiv:2601.13642*.
- The Blessing of Heterogeneity in Federated Q-Learning: Linear Speedup and Beyond, *JMLR 2025*.
- The Sample-Communication Complexity Trade-off in Federated Q-Learning, *NeurIPS 2024*, **oral**.
- Federated Offline Reinforcement Learning: Collaborative Single-Policy Coverage Suffices, *ICML 2024*.



<https://yuejiechi.github.io/>