Federated Reinforcement Learning with Environment Heterogeneity

Hao Jin
jin.hao@pku.edu.cn
Peking University

Yang Peng
pengyang@pku.edu.cn
Peking University

Wenhao Yang
yangwenhaosms@pku.edu.cn
Peking University

Shusen Wang
shusenwang@xiaohongshu.com
Xiaohongshu Inc.

Zhihua Zhang
zhzhang@math.pku.edu.cn
Peking University

Abstract

We study a Federated Reinforcement Learning (FedRL) problem in which n agents collaboratively learn a single policy without sharing the trajectories they collected during agent-environment interaction. We stress the constraint of environment heterogeneity, which means n environments corresponding to these n agents have different state transitions. To obtain a value function or a policy function which optimizes the overall performance in all environments, we propose two federated RL algorithms, `QAvg` and `PAvg`. We theoretically prove that these algorithms converge to suboptimal solutions, while such suboptimality depends on how heterogeneous these n environments are. Moreover, we propose a heuristic that achieves personalization by embedding the n environments into n vectors. The personalization heuristic not only improves the training but also allows for better generalization to new environments.

1 Introduction

In recent years, reinforcement learning (RL) Sutton et al. (1998) has made unprecedented progresses in solving challenging problems such as playing Go game Hessel et al. (2018); Silver et al. (2016, 2017) and controlling robots Fan et al. (2018); Levine et al. (2016). Traditionally, when handling such problems, one typically assumes that the environment has a fixed state transition. However, in some real-life applications, an agent is expected to simultaneously deal with different state transitions in multiple environments. For example, a drone is expected to perform well under different weather conditions of the physical environment (*e.g.*, wind speed and wind direction), which may affect the state transition. In this way, the learning of the drone policy falls beyond the traditional assumption mentioned earlier.

In this paper, n agents are assumed to be located in n environments which have the same state space $\mathcal{S}$, action space $\mathcal{A}$, reward function R, but different state transitions $\{\mathcal{P}_i\}_{i=1}^n$. After incorporating environment heterogeneity into FedRL, we are mainly concerned with the following two problems. First, it is natural to ask how to learn a single policy performing uniformly well in these n environments Doshi-Velez and Konidaris (2016); Killian et al. (2017). However, any single policy is inevitably suboptimal compared with the optimal policy in each environment because of the environment heterogeneity. Second, we wonder how to additionally develop a *personalized* policy in each environment, which is better than the globally learned policy. To address these issues, collaboration among these n agents is necessary: interaction with any single environment is limited in diversity to learn for all n environments; samples from each individual environment are also limited in quantity to learn a locally optimal policy. Therefore, it is important to figure out how to achieve collaboration among n agents in the setting of FedRL when deriving efficient solutions to these two issues. It is worth noting that we additionally do not allow agents to communicate their interactions with individual environments in order to protect privacy embedded in their local experiences.

The setting of FedRL with environment heterogeneity is common in real-life applications. Smart home devices are deployed in families with different using preference and habits, while service providers are interested in how to provide better experience via improving the policy loaded in these devices. Viewing the policy as a RL agent, users with different using habits can be

Proceedings of the 25th International Conference on Artificial Intelligence and Statistics (AISTATS) 2022, Valencia, Spain. PMLR: Volume 151.

regarded as environments with different state transitions, which means they may response differently even to the same action. Moreover, data collected in any certain device is usually not enough in the application to independently learn a reliable policy, while images and audios collected by each device are sometimes inaccessible for the service providers out of privacy issues. In this way, the policy training for these smart devices fits into the framework of FedRL with environment heterogeneity, and the second problem of personalization in our setting perfectly describes the dilemma of service providers in improving performance of different users without accessing their data.

To learn a uniformly good policy, we follow the approach of letting the agents share their models and propose two model-free algorithms, `QAvg` and `PAvg`. These algorithms iteratively perform local updates on the agent side and global aggregation on the server side. Different from the extant work in FedRL, we emphasize the role of environment heterogeneity and theoretically analyze effectiveness of these algorithms. Our theories show that both `QAvg` and `PAvg` converge to a suboptimal solution and the suboptimality is affected by the degree of environment heterogeneity in FedRL. Based on theoretical effectiveness of `QAvg` and `PAvg`, we also derive `DQNAvg` and `DDPGAvg` as extensions of methods with Q networks and policy networks, *i.e.*, `DQN` and `DDPG`. Moreover, we carry out numerical experiments on several tabular environments to verify theoretical results of `QAvg` and `PAvg`, and compare `DQNAvg` (`DDPGAvg`) with `DQN` (`DDPG`) in harder tasks of control.

To achieve personalization in different environments, we propose a heuristic with slight modification to structures of `DQNAvg` and `DDPGAvg`. Specifically, we embed each environment into a low-dimension vector to capture its specific state transition. During the training of FedRL, these n agents periodically aggregate their parameters except their embedding layers. Along with the learned aggregated network, the private embedding layer enables each agent to achieve better performance in its individual environment. Such personalization heuristic also enables us to generalize the learned model in FedRL to any novel environment. Instead of updating all parameters of the model, we only need to adjust the low-dimension embedding layer for the novel environment. Empirical experiments have shown that our proposed heuristic not only improves training performance of the learned models in `DQNAvg` and `DDPGAvg`, but also helps to achieve stable generalization within few updates in the novel environment.

In summary, this paper offers the following main contributions:

- We propose `QAvg` and `PAvg` to solve the task of federated reinforcement learning (FedRL) with environment heterogeneity, where environments have different state transitions.
- We theoretically analyze the convergence of `QAvg` and `PAvg`, discuss relations between their convergent performance and environment heterogeneity in FedRL, and extend the averaging strategy to derive `DQNAvg` and `DDPGAvg` for more complicated environments.
- We propose a heuristic idea to achieve personalization in FedRL, which utilizes embedding layers to capture the specific state transition in individual environment. We have also empirically shown that such heuristic helps to generalize the learned model in FedRL to new environments in a stable and easy way.

2 Related Work

Classical RL methods. Traditionally, reinforcement learning (RL) assumes the environment has a fixed state transition and seeks to maximize the cumulative rewards in the environment Sutton et al. (1998); Watkins and Dayan (1992). The environment is usually modelled as a standard MDP, $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, R, \mathcal{P}, \gamma \rangle$ Bellman (1957); Bertsekas et al. (1995). The objective function is formulated as

$$g_{d_0}(\pi) = \mathbb{E}\Bigg[\sum_{t=1}^{\infty} \gamma^t R(s_t, a_t) \;\Bigg|\; s_0 \sim d_0, a_t \sim \pi(\cdot|s_t), \\ s_{t+1} \sim \mathcal{P}_i(\cdot|s_t, a_t) \Bigg],$$

where d_0 represents the initial state distribution. To solve the problem, there are many model-free methods such as Q-learning Watkins and Dayan (1992) and policy gradient (PG) Sutton et al. (1999). Under the setting of a standard MDP, prior works Agarwal et al. (2019); Sutton et al. (1998) have proved their convergence to the optimal policy.

HiP-MDP and MTRL. FedRL is closely related to Hidden Parameter Markov Decision Processes (HiP-MDP) Doshi-Velez and Konidaris (2016); Killian et al. (2017) and Multi-Task Reinforcement Learning (MTRL) Espeholt et al. (2018); Mnih et al. (2016); Teh et al. (2017). HiP-MDP assumes the existence of latent variables which decide the state transition of an environment. In Doshi-Velez and Konidaris (2016); Killian et al. (2017) it explicitly learns the natural distribution of latent variables with a generative network and considers Bayesian reinforcement learning. FedRL is similar to HiP-MDP when talking about the source of

environment heterogeneity, but it additionally has constraints on privacy issues, which does not allow agents to share their collected experiences. MTRL assumes that the n agents located in different environments are different and they perform different tasks. FedRL can be viewed as a special case of MTRL where the agents perform the same task. While in Zeng et al. (2020) it also concentrates on methods of policy averaging, our work additionally focuses on the specific personalization problem in the federated setting.

Federated Learning. FL, also known as federated optimization, allows local devices to collaboratively train a model without data sharing Mahajan et al. (2018). To reduce the communication cost in FL, many communication-efficient algorithms have been proposed, *e.g.*, `FedAvg` Mahajan et al. (2018) and `FedProx` Sahu et al. (2018). The communication-efficient FL algorithms let each client locally update the model using its local data and periodically aggregate the local models. Our proposed algorithms bear a resemblance with the FL algorithms: an agent performs multiple local updates between two communications. Similar methods have also been previously studied in contexts of FedRL Liu et al. (2019); Nadiger et al. (2019); Wang et al. (2020); Zhuo et al. (2019): Wang et al. (2020) simply analyzes the convergence speed of policy gradient in FedRL tasks without considering environment heterogeneity while Liu et al. (2019); Nadiger et al. (2019); Zhuo et al. (2019) mainly concentrates on applications in specific scenarios. Moreover, Nadiger et al. (2019) considers personalization of FedRL in a specific application.

Personalized Federated Learning. FL seeks to learn a single model that performs uniformly well on all n local datasets, while personalized FL aims to learn n models specialized for the n local datasets. Many personalized FL methods have been developed: Arivazhagan et al. (2019) designed a neural network architecture with personalization layers which are not shared; Deng et al. (2020); Mansour et al. (2020) viewed the global model as the interpolation of local models; Bui et al. (2019) introduced a technical called private embedding. In this paper, we extend personalization in federated learning to the context of FedRL, which means we aim to additionally learn n different policies for each environment with the collaboration among agents.

3 Federated Reinforcement Learning

Suppose n agents respectively interact with n independent environments. The environments have different state transitions $\{\mathcal{P}_i\}_{i=1}^n$ but the same state space $\mathcal{S}$, action space $\mathcal{A}$, and reward function R. These environments are modelled as Markov Decision Processes (MDPs), $\mathcal{M}_i = \langle \mathcal{S}, \mathcal{A}, R, \mathcal{P}_i, \gamma \rangle$, for $i = 1, \cdots, n$.

The goal of Federated Reinforcement Learning (FedRL) is letting the n agents jointly learn a policy function or a value function that performs uniformly well across the n environments. Due to privacy constraints, the n agents cannot share their collected experience. Policy-based FedRL can be formulated as the following optimization problem:

$$\max_{\pi} \left\{ g_{d_0}(\pi) \triangleq \frac{1}{n} \sum_{i=1}^{n} \mathbb{E}\left[\sum_{t=1}^{\infty} \gamma^t R(s_t, a_t) | s_0 \sim d_0, a_t \sim \pi(\cdot|s_t), s_{t+1} \sim \mathcal{P}_i(\cdot|s_t, a_t) \right] \right\}, \tag{1}$$

where d_0 represents the common initial state distribution in these n environments. If state transitions $\{\mathcal{P}_i\}_{i=1}^n$ are the same, the optimal policy π^* is independent of d_0 Bellman and Dreyfus (1959). However, if these state transitions are different, then the solution to Eq. (1) actually depends on d_0.

Theorem 1. *There exists a task of FedRL with the following properties. Assume that $\pi^\star \in \operatorname{argmax}_\pi g_{d_0}(\pi)$. There exist another initial state distribution d_0' and another policy $\tilde{\pi}$ such that $g_{d_0}(\tilde{\pi}) < g_{d_0}(\pi^\star)$, but $g_{d_0'}(\tilde{\pi}) > g_{d_0'}(\pi^\star)$.*

Theorem 1 shows that there does not exist an optimal policy $\pi^\star$ that dominates all policies for all d_0. We denote the solution to (1) by $\pi^\star_{d_0}$ which means the initial state distribution affects the optimal policy.

4 Algorithms: `QAvg` and `PAvg`

We propose two novel FedRL algorithms, `QAvg` and `PAvg`, for learning a value function and a policy function, respectively. We discuss tabular versions of `QAvg` and `PAvg`; versions of neural networks, such as `DQNAvg` and `DDPGAvg`, can be similarly implemented. These algorithms alternate between local computation and global aggregation. Specifically, each agent locally updates its value function or policy function for multiple times, and then the server averages these n functions of all agents. To improve the communication efficiency, the local updates are performed multiple times between two communications.

`QAvg` learns an $|\mathcal{S}| \times |\mathcal{A}|$ table by alternating between local updates and global aggregations. For $k = 1, \cdots, n$,

the k-th agent performs the following local update:

$$Q_{t+1}^k(s,a) \leftarrow (1-\eta_t)\cdot Q_t^k(s,a) + \eta_t \cdot \Big[R(s,a) + \gamma \sum_{s'} \mathcal{P}_k(s'|s,a) \max_{a' \in \mathcal{A}} Q_t^k(s',a')\Big].$$

In the equation, the superscript k indexes the environment $\mathcal{M}_k$, and the subscript t indexes the iteration. After several local updates, there is a global aggregation:

$$\bar{Q}_t(s,a) \leftarrow \frac{1}{n}\sum_{i=1}^n Q_t^i(s,a),\ \forall\, s,a;$$
$$Q_t^i(s,a) \leftarrow \bar{Q}_t(s,a),\ \forall\, s,a,k.$$

Throughout, only Q tables are communicated, while agents do not share their collected experience.

`PAvg` seeks to learn a $|\mathcal{S}|\times|\mathcal{A}|$ table, $\bar{\pi}(a|s)$. Each agent independently repeats the local update for multiple times:

$$\tilde{\pi}_{t+1}^k(a|s) \leftarrow \pi_t^k(a|s) + \frac{\partial g_{d_0,k}(\pi_t^k)}{\partial \pi(a|s)},\ \forall\, s,a,k;$$
$$\pi_{t+1}^k(\cdot|s) \leftarrow \underset{\Delta(\mathcal{A})}{\mathrm{Proj}}(\tilde{\pi}_{t+1}^k(\cdot|s)),\ \forall\, s,a,k.$$

Here, $g_{d_0,k}(\pi) = \mathbb{E}[\sum_{t=1}^\infty \gamma^t R(s_t,a_t)|s_0 \sim d_0, a_t \sim \pi(\cdot|s_t), s_{t+1} \sim \mathcal{P}_k(\cdot|s_t,a_t)]$ is the k-th agent's objective function, and $\mathrm{Proj}_{\Delta(\mathcal{A})}$ is the projector onto the simplex of action space $\Delta(\mathcal{A})$. Then, there is a global aggregation after several local updates:

$$\bar{\pi}_t(a|s) \leftarrow \frac{1}{n}\sum_{i=1}^n \pi_t^i(a|s),\ \forall\, s,a;$$
$$\pi_t^i(a|s) \leftarrow \bar{\pi}_t(a|s),\ \forall\, s,a,k.$$

Similar to `QAvg`, agents in `PAvg` only share their policy functions throughout the training process.

5 Theoretical Analyses

In this section we prove that both `QAvg` and `PAvg` converge to suboptima whose performance across the n environments are theoretically guaranteed. We also discuss how the suboptimality of convergent policies is affected by the environment heterogeneity in FedRL.

5.1 Notation

Imaginary environment $\mathcal{M}_I$. Let $\mathcal{P}_1, \cdots, \mathcal{P}_n$ be the state transition functions of the n environments. Define the average state transition:

$$\bar{\mathcal{P}}(s'|s,a) = \frac{1}{n}\sum_{k=1}^n \mathcal{P}_k(s'|s,a),\ \forall s,s' \in \mathcal{S},\ \forall a \in \mathcal{A}.$$

To analyze the convergence of proposed algorithms, we introduce the imaginary environment, $\mathcal{M}_I = \langle \mathcal{S}, \mathcal{A}, R, \bar{\mathcal{P}}, \gamma\rangle$. As its name suggests, the imaginary environment $\mathcal{M}_I$ does not have to be one of the n environments in FedRL, *i.e.*, $\mathcal{M}_I \notin \{\mathcal{M}_i\}_{i=1}^n$.

Environment heterogeneity. In FedRL, different environments $\{\mathcal{M}_i\}_{i=1}^n$ have different state transitions $\{\mathcal{P}_i\}_{i=1}^n$. Intuitively speaking, the closer these state-transitions are, the easier the problem. To quantify the environment heterogeneity, we define

$$\kappa_1 \triangleq \max_{s,\pi} \sum_{s'}\sum_{i=1}^n \left|\mathcal{P}_i^\pi(s'|s) - \frac{1}{n}\sum_{j=1}^n \mathcal{P}_j^\pi(s'|s)\right|,$$
$$\kappa_2 \triangleq \max_{\pi} \frac{1}{n}\sum_{i=1}^n \left\|\nabla_\pi g_{d_0,i}(\pi) - \frac{1}{n}\sum_{j=1}^n \nabla_\pi g_{d_0,j}(\pi)\right\|_2,$$

where $\mathcal{P}_k^\pi(s'|s) \triangleq \mathbb{E}_{A\sim\pi(\cdot|s)}[\mathcal{P}_k(s'|s,A)]$. If the state transitions in FedRL are close to each other, both κ_1 and κ_2 are small.

5.2 Theoretical Analysis of QAvg

`QAvg` is the federated version of Q-Learning. Traditional analysis of Q-learning claims that Q-learning converges to the Q function of optimal policy in that given environment. Similarly, theoretical analysis of `QAvg` mainly focuses on convergence performance of the averaged Q function, *i.e.*, $\bar{Q}_t$ shown in its aggregation at time t.

To better understand the convergence of `QAvg`, we return to the definition of $g_{d_0}(\pi)$. The objective of FedRL is decomposed as follows:

$$g_{d_0}(\pi) = \frac{1}{n}\mathbb{E}_{S_0\sim d_0}\left[\sum_{i=1}^n V_i^\pi(S_0)\right] = \mathbb{E}_{S_0\sim d_0}\left[\bar{V}^\pi(S_0)\right],$$

where $\{V_i^\pi\}_{i=1}^n$ are normally defined value functions of policy π in the n environments $\{\mathcal{M}_i\}_{i=1}^n$:

$$V_i^\pi(s) = \mathbb{E}_{\pi,\mathcal{P}_i}\left[\sum_{t=0}^\infty \gamma^t R(s_t,a_t)|s_0 = s, a_t \sim \pi(\cdot|s_t)\right],$$

and $\bar{V}^\pi$ is the averaged value function $\bar{V}^\pi = \frac{1}{n}\sum_{i=1}^n V_i^\pi$. The dependence of optimality in FedRL with the initial state distribution d_0 indicates that $\bar{V}^\pi$ is not the value function of any environment (otherwise, there exists optimal policy π^* independent with d_0).

Imaginary environment $\mathcal{M}_I$ is therefore introduced to element-wisely lower bound the values of $\bar{V}^\pi$. Specifically, the value function V_I^π of the policy π in the imaginary environment $\mathcal{M}_I$ is normally defined as follows:

$$V_I^\pi(s) = \mathbb{E}_{\pi,\bar{\mathcal{P}}}\left[\sum_{t=0}^\infty \gamma^t R(s_t,a_t)|s_0 = s, a_t \sim \pi(\cdot|s_t)\right],$$

and its relationship with the averaged value function $\bar{V}^\pi$ is mainly described in Lemmas 1 and 2. As long as the environments $\{\mathcal{M}_i\}_{i=1}^n$ are not too different from each other, the value function V_I^π manages to properly approximate $\bar{V}^\pi$.

Lemma 1. *For all state s and policy π, we have $\bar{V}^\pi(s) \geq V_I^\pi(s)$.*

Lemma 2. *Let κ_1 be the environment heterogeneity. For all s and π, we have*

$$\left|\bar{V}^\pi(s) - V_I^\pi(s)\right| \leq \frac{\gamma\kappa_1}{(1-\gamma)^2}.$$

After identifying V_I^π as a lower bound of $\bar{V}^\pi$, it is natural to consider the optimal policy π_I^* in the imaginary environment $\mathcal{M}_I$. Because of its optimality in $\mathcal{M}_I$, the value function $V_I^{\pi_I^*}$ dominates the value function of any other policy π, *i.e.*, $V_I^{\pi_I^*}(s) \geq V_I^\pi(s), \forall s$. In other words, π_I^* reaches the largest lower bound of the averaged value function $\bar{V}^\pi$.

Finally, we are ready to show the convergence results of `QAvg`. Taking the number of local updates as E, the algorithm with $E \geq 1$ not only converges, but also reaches the Q function of π_I^* in $\mathcal{M}_I$:

Theorem 2 (Convergence results of `QAvg`)**.** *Take $\bar{Q}_t$ as the average of distributed Q functions Q_t^k in the n environments at iteration t, i.e., $\bar{Q}_t = \frac{1}{n}\sum_{k=1}^n Q_t^k$. Let the number of local updates be E. Assume $Q_I^{\pi_I^*}$ is the Q function of optimal policy π_I^* in $\mathcal{M}_I$. Letting $\eta_t = \frac{2}{(1-\gamma)(t+E)}$, we have*

$$\left\|\bar{Q}_t - Q_I^{\pi_I^*}\right\|_\infty \leq \frac{16\gamma E}{(1-\gamma)^3(t+E)}.$$

Remark 1. *$E = 1$ makes a special variant of `QAvg`. This means the agents communicate after every local update of their Q functions. Although the heavy communication load makes `QAvg` with $E = 1$ quite impractical, it provides intuitions on how `QAvg` achieves the optimal Q function of π_I^*. The update of every local Q function in `QAvg` with $E = 1$ is formulated as follows:*

$$\begin{aligned} Q_{t+1}^j(s,a) &\leftarrow \frac{1}{n}\sum_{k=1}^n \Big[R(s,a) + \gamma\sum_{s'}\mathcal{P}_k(s'|s,a)\max_{a'} Q_t^k(s',a')\Big] \\ &= R(s,a) + \gamma\sum_{s'}\bar{\mathcal{P}}(s'|s,a)\max_{a'} Q_t^j(s',a'), \end{aligned}$$

where the last equality is because the local Q functions keep the same in `QAvg` with $E = 1$. In this way, every local Q function is updated as if the agent were trained in the imaginary environment $\mathcal{M}_I$.

Remark 2. *`QAvg` with $E = \infty$ corresponds to the algorithm which never communicates and simply averages those independently trained Q functions as the aggregated Q function. Neither its theoretical convergence nor its empirical performance is similar to that of `QAvg` with $E < \infty$.*

5.3 Theoretical Analysis of PAvg

`PAvg` directly views the policy π as optimization parameters in maximizing the objective function. In this way, the corresponding theoretical analysis focuses on the convergence of objective values $g_{d_0}(\bar{\pi}_t)$, where $\bar{\pi}_t$ represents the averaged policy shown in aggregation of `PAvg` at time t.

Theorem 3 (Convergence performance of `PAvg`)**.** *Denote L as the L-smoothness parameter of $g_{d_0}(\pi)$ w.r.t. π, E as the number of local updates, and κ_2 as the environment heterogeneity. Letting $\eta_t = \sqrt{\frac{E}{12L^2(t+E/3)}}$, we have that*

$$\max_{t=0,..,T-1} g_{d_0}(\bar{\pi}_t) \geq g_{d_0}(\pi_{d_0}^\star) - c\cdot\Big(\kappa_2 + \frac{1}{\sqrt{T}}\Big),$$

where c is constants and logarithmic factors of T.

Remark 3. *Here we discuss the effect of local iterations E on the convergence. The term $c\cdot(\kappa_2 + \frac{1}{\sqrt{T}})$ in the theorem is equal to*

$$C_1 + T^{-0.5}\cdot\big(C_2E^{-0.5} + C_3E^{0.5} + C_4E^{2.5}\big).$$

Here, C_1, C_2, C_3, C_4 are either independent of T and E or contain logarithmic factors of T and E, implying there exists an E that is the best for the convergence.

6 Personalized FedRL

We propose a heuristic method that allows for better training in each local environment and better generalization to novel environments. The idea is personalized FedRL, that is, instead of learning one policy for all the n agents, we learn n policies for the n agents, respectively. In this section, we consider deep FedRL; see Figure 1. We treat each environment as an ID and embed it into a low-dimension vector which is regarded as part of the state. The n agents share all the layers except the embedding layer.

After the training, the learned policy network may be applied to a never-seen-before environment. In the new environment, the embedding layer cannot be reused. We need to let the agent interact with the new environment in order to learn the low-dimensional vector. If the output of embedding is d-dimensional, we need to learn only d parameters. Therefore, to generalize the trained policy network to a new environment, we need to perform few-shot learning in the new environment to learn the low-dimension embedding.

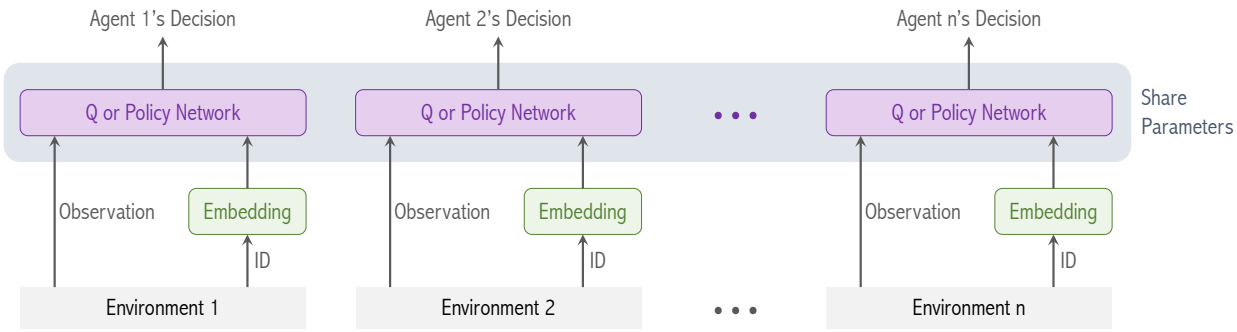

Figure 1: The figure shows the Q or policy networks of personalized FedRL. The networks, except the embedding layers, share parameters. To generalize the model to new environments, we keep the trained Q or policy networks but train the embedding layer from random initialization.

	RandomMDPs			WindyCliffs		
	QAvg	SoftPAvg	ProjPAvg	QAvg	SoftPAvg	ProjPAvg
$\kappa = 0$	$35.42_{\pm 0.05}$	$35.15_{\pm 0.05}$	$34.97_{\pm 0.05}$	$133.97_{\pm 0.00}$	$133.97_{\pm 0.00}$	$119.97_{\pm 0.34}$
$\kappa = 0.2$	$35.23_{\pm 0.05}$	$34.97_{\pm 0.05}$	$34.92_{\pm 0.05}$	$133.97_{\pm 0.00}$	$133.97_{\pm 0.00}$	$118.47_{\pm 0.34}$
$\kappa = 0.4$	$34.80_{\pm 0.05}$	$34.58_{\pm 0.05}$	$34.54_{\pm 0.05}$	$133.97_{\pm 0.00}$	$133.96_{\pm 0.00}$	$115.82_{\pm 0.34}$
$\kappa = 0.6$	$34.14_{\pm 0.06}$	$34.02_{\pm 0.06}$	$34.02_{\pm 0.06}$	$133.96_{\pm 0.00}$	$133.95_{\pm 0.00}$	$111.46_{\pm 0.35}$
$\kappa = 0.8$	$33.29_{\pm 0.06}$	$33.25_{\pm 0.06}$	$33.38_{\pm 0.06}$	$133.65_{\pm 0.03}$	$133.59_{\pm 0.03}$	$103.53_{\pm 0.36}$

Table 1: Impact of environment heterogeneity on convergent performance: larger κ indicates environments with larger environment heterogeneity, *i.e.*, $\{\mathcal{P}_k^\kappa\}_{k=1}^N$ with larger noise from $\mathcal{P}_0$; `QAvg` ($E = 4$), `SoftPAvg` ($E = 4$) and `ProjPAvg` ($E = 32$) are evaluated on the noiseless environment $\mathcal{P}_0$; each setting is repeated with $16,000$ random seeds, and we display the mean with standard error.

The benefit of the personalization heuristic is two-fold—better training and better generalization. Without personalization, we seek to learn one policy that performs uniformly well in all the n environments. Since one policy cannot achieve the optimal performance in every environment, the learned policy is suboptimal in every environment. With the n private embedding layers, the convergent model serves as n different policies for n policies; each policy best fits one environment. When the learned model is deployed to a never-seen-before environment, the few-shot learning of the embedding layer makes the policy quickly adapted to the new environment. The small number of parameters to be tuned also adds robustness to the generalization process.

7 Empirical Study

In this section, we firstly use tabular environments to verify our theories on `QAvg` and `PAvg`. Then, we evaluate the extensions to deep reinforcement learning, `DQNAvg` and `DDPGAvg`, which are more practical in real-world applications. Finally, we demonstrate that the personalization heuristic improves both training and generalization performance. [1]

[1] Our code of both tabular cases and deep cases have been released on https://github.com/pengyang7881187/FedRL

7.1 Settings

Environments. We construct a collection of heterogeneous environments by varying the state-transition parameters. For example, given the `CartPole` environment, we vary the length of pole. We use two types of tabular environments: first, random MDP with randomly generated state transition and reward function, and second, WindyCliff Paul et al. (2019) whose wind speed is uniformly sampled from $[S_{min}, S_{max}]$. We also use non-tabular environments in `Gym` Brockman et al. (2016): first, `CartPole` and `Acrobat` with varying length of pole, and second, `Hopper` and `Half-cheetah` with adjustable length of leg.

Control. For `QAvg`, after learning the averaged Q function $\bar{Q}(s, a)$, we use the deterministic policy, $\pi(s) = \text{argmax}_{a \in \mathcal{A}} \bar{Q}(s, a)$, for controlling the agent. For `PAvg`, we directly learn a stochastic policy, $\pi(a|s)$, that outputs the probability of taking action a. We use two types of `PAvg`: first, `ProjPAvg` denotes `PAvg` with projection operator, and second, `SoftPAvg` denotes `PAvg` with softmax activation function.

Deep FedRL. Deep Q Network (DQN) Mnih et al. (2015) and Deep Deterministic Policy Gradient (DDPG) Lillicrap et al. (2015) are two practical deep RL meth-

	WindyCliffs		
	QAvg	SoftPAvg	ProjPAvg
E=1	129.55±0.17	126.92±0.19	122.08±0.30
E=2	129.55±0.17	129.56±0.17	123.28±0.29
E=4	129.55±0.17	**129.65**±0.17	124.94±0.27
E=8	129.55±0.17	129.62±0.17	**126.03**±0.25
E=16	129.55±0.17	129.54±0.17	125.64±0.24
E=∞	129.12±0.17	127.01±0.18	90.92±0.41

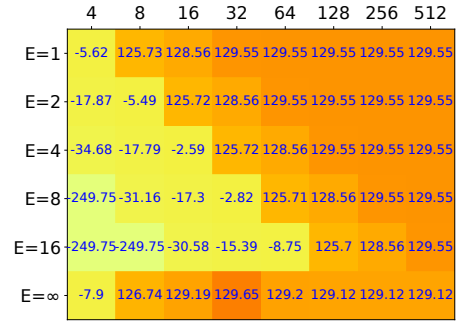

Figure 2: Impact of local update time E on convergent performance: larger E indicates less frequent communication while $E = \infty$ means agents do not communicate; **Left** shows the objective values of FedRL at convergence. **Right** shows the objective values of FedRL at different iterations during the training of `QAvg`s with different E.

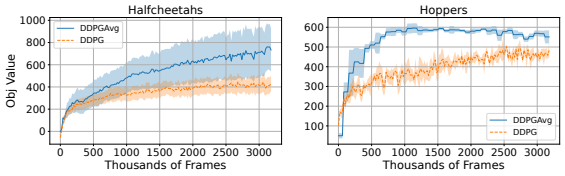

Figure 3: Acceleration of local training in federated setting: averaged local performance of locally trained policies is compared with averaged local performance of the policy trained in federated setting; we depict the mean as line and 1.65 times of standard error as shadow.

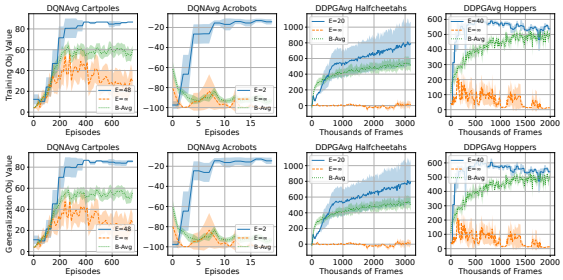

Figure 4: Training and generalization performance of DQNAvg and DDPGAvg in different tasks of FedRL: training performance refers to the objective value of FedRL, *i.e.*, averaged performance in N environments; generalization performance refers to the averaged performance in M environments with newly generated state-transitions; we depict the mean as line and 1.65 times of standard error as shadow.

ods. We extend our proposed `QAvg` and `PAvg` to DQN and DDPG; we call the extension `DQNAvg` and `DDPGAvg`. Specifically, `DQNAvg` periodically approximately aggregates Q functions via averaging parameters of local Q networks, while `DDPGAvg` periodically aggregates both critic networks and policy networks stored in local devices.

Baseline. The point of FedRL is to use all the agents' experience without directly sharing their experience. As opposed to FedRL, independent RL lets each agent perform RL without exchanging information with other agents. We use independent RL as the baseline for showing the usefulness of collaboration. Let `Baseline` be the step-wise averaged objective values of the n local models. In other words, `Baseline` represents the performance of a randomly selected local model in n involved environments.

7.2 Effect of Environment Heterogeneity

To check the impact of environment heterogeneity on convergent performance of our methods, we construct tasks of FedRL with various κ, which controls how different the state transitions are. Theorems 2 and 3 claim that larger environment heterogeneity, *i.e.*, κ with larger values, leads to larger performance gap with the optimal policy. Empirical results shown in Table 1 match such theoretical observations, and we discuss the experimental settings as below:

To get the control of environment heterogeneity with a scalar κ, we sample $N+1$ different state transitions $\{\mathcal{P}_k\}_{k=0}^N$ and then construct the environments with $\{\mathcal{P}_k^\kappa = \kappa\mathcal{P}_k + (1-\kappa)\mathcal{P}_0\}_{k=1}^N$. $\{\mathcal{P}_k^\kappa\}_{k=1}^N$ are N copies of $\mathcal{P}_0$ with noise, whose direction and intensity are respectively controlled with $\{\mathcal{P}_k\}_{k=1}^N$ and κ. With fixed $\{\mathcal{P}_k\}_{k=0}^\kappa$, we manage to construct environments with environmental heterogeneity controlled by κ. However, since the optimal policy for FedRL with $\{\mathcal{P}_k^\kappa\}_{k=1}^N$ is computationally intractable, we make the following approximations: the convergent performance is approximated as the performance in noiseless central environment with $\mathcal{P}_0$; the optimal policy is approximated as the optimal policy in the noiseless central environment $\mathcal{P}_0$, *i.e.*, the convergent policy with $\kappa = 0$. Each setting is repeated with $16,000$ random seeds.

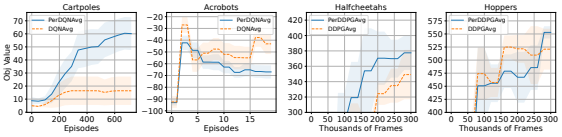

Figure 5: Improvement in local training with personalization heuristic: averaged local performance of `DQNAvg` and `DDPGAvg` is compared with the averaged local performance of `PerDQNAvg` and `PerDDPGAvg` with environment embeddings; we depict the mean as line and 1.65 times of standard error as shadow.

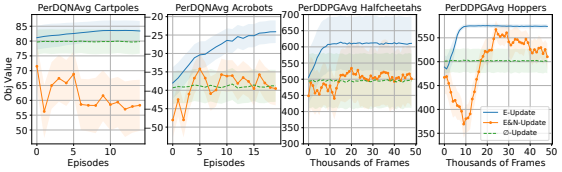

Figure 6: Impact of personalization heuristic on generalization performance: we compare different adjustment methods when fitting a novel environment given the learned convergent model; `E-Update` indicates only environment embeddings are adjusted, `E&N-Update` indicates both embeddings and policy network are adjusted, and `Φ-Update` keeps the learned model unchanged; we depict the mean as line and 1.65 times of standard error as shadow.

7.3 Effect of Communication Frequency

We are next to show the impact of communication frequency on the convergence of our methods. Specifically, the communication frequency is quantified by the number E of local updates between two consecutive communications. Empirical results in Figure 2 reveal that communication frequency indeed influences `QAvg` and `PAvg`, yet quite in different ways.

For `QAvg`, Theorem 2 reveals that the convergent Q table is free of E while communication frequency affects the convergence speed. Figure 2 **(Left)** confirms such a theoretical result with identical convergent values of `QAvg` with $E < \infty$, while Figure 2 **(Right)** shows that `QAvg` with larger E suffers from a lower convergence speed. For `PAvg`, Theorem 3 reveals that the performance gap to the optimal policy is affected by E and Remark 3 indicates the existence of optimal E. Figure 2 **(Left)** confirms that `PAvg`s, `ProjPAvg` and `SoftPAvg` have different convergent performance with different selection of E. Moreover, performance peaks at $E = 4, 8$ respectively for `SoftPAvg` and `ProjPAvg` indicate that E is a critical hyper-parameter in achieving the best convergent performance for `PAvg`s.

7.4 Experiments on Deep RL

Here we consider deep FedRL algorithms, `DQNAvg` and `DDPGAvg`, in more complicated FedRL tasks: `CartPoles` and `Acrobats` with discrete actions, `Halfcheetahs` and `Hoppers` with continuous actions. In these scenarios, it is impossible to directly quantify environment heterogeneity κ and we implicitly model it through sampling certain deciding parameters of state transitions from certain distribution. We first justify that the federated setting helps accelerate training in any individual environment, and then compare our methods with `Baseline` in terms of both training and generalization performance.

To figure out the impact of the federated setting on the training of any individual environment, we compare the averaged performance of local models in corresponding environments with and without communication with others. Although collected experience is not allowed to share, communication of policy is believed to transfer certain knowledge from others. As shown in Figure 3, policy communication indeed accelerates the local training and therefore alleviates the trouble of obtaining an efficient policy when data stored locally is limited.

Then we compare `DQNAvg` and `DDPGAvg` with their variants which do no communicate ($E = \infty$), and corresponding `Baselines`, *i.e.*, averaged performance of independently trained policies. In terms of training performance, `DQNAvg` and `DDPGAvg` manage to obtain higher objective values of FedRL, which indicates the convergent policy uniformly performs well on all involved environments in FedRL. Moreover, when faced with M environments with newly generated state transitions, the learned policies of `DQNAvg` and `DDPGAvg` also outperform their variants with $E = \infty$ and `Baselines`. Therefore, the convergent policies of our methods not only efficiently solve the task of FedRL, but also generalize well to similar but unseen environments.

7.5 Personalized FedRL

We are now to demonstrate how the heuristic mentioned in Section 6 helps the personalization in the training of FedRL and how the learned personalized model enables us to quickly fit to any unseen environment. `DQNAvg` and `DDPGAvg` with the heuristic are denoted as `PerDQNAvg` and `PerDDPGAvg`.

Figure 5 depicts the averaged performance of N local policies in their corresponding environments with and without personalization heuristic. Environment embeddings enable local policies to be personalized in the training process of `PerDQNAvg` and `PerDDPGAvg`, which helps to achieve better averaged local performance than the single aggregated policy of `DQNAvg` and `DDPGAvg`.

Moreover, when fitting the learned policy to any unseen environment, we merely adjust the environment embeddings from an averaged initialization. Figure 6 reveals that such adjustment is enough for a quick fit to the novel environment and outperforms adjustment of both embeddings and policy network.

8 Conclusion

We have studied Federated Reinforcement Learning (FedRL) and addressed two issues: how to learn a single policy with uniformly good performance in all n environments, and how to achieve personalization. The main difference from the existing FedRL work is that we assume that the n environments have different state-transition functions. We have proposed two algorithms, `QAvg` and `PAvg`, which are federated extensions of Q-Learning and policy gradient. Regarding their theoretical efficiency, we have analyzed their convergence and showed how environment heterogeneity affects the convergence. We have also proposed a heuristic approach for personalization in FedRL, where environment embeddings are used to capture any specific environment. Furthermore, such heuristic enables us to achieve generalization of convergent policies to fit any unseen environment via adjusting the embeddings.

Acknowledgments

Jin and Zhang have been supported by the National Key Research and Development Project of China (No. 2018AAA0101004) and Beijing Natural Science Foundation (Z190001).

References

Alekh Agarwal, Sham M Kakade, Jason D Lee, and Gaurav Mahajan. On the theory of policy gradient methods: Optimality, approximation, and distribution shift. *arXiv preprint arXiv:1908.00261*, 2019.

Manoj Ghuhan Arivazhagan, Vinay Aggarwal, Aaditya Kumar Singh, and Sunav Choudhary. Federated learning with personalization layers. *arXiv preprint arXiv:1912.00818*, 2019.

R Bellman. Dynamic programming princeton university press princeton. *New Jersey Google Scholar*, 1957.

Richard Bellman and Stuart Dreyfus. Functional approximations and dynamic programming. *Mathematical Tables and Other Aids to Computation*, pages 247–251, 1959.

Dimitri P Bertsekas, Dimitri P Bertsekas, Dimitri P Bertsekas, and Dimitri P Bertsekas. *Dynamic programming and optimal control*, volume 1. Athena scientific Belmont, MA, 1995.

Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. Openai gym, 2016.

Duc Bui, Kshitiz Malik, Jack Goetz, Honglei Liu, Seungwhan Moon, Anuj Kumar, and Kang G Shin. Federated user representation learning. *arXiv preprint arXiv:1909.12535*, 2019.

Yuyang Deng, Mohammad Mahdi Kamani, and Mehrdad Mahdavi. Adaptive personalized federated learning. *arXiv preprint arXiv:2003.13461*, 2020.

Finale Doshi-Velez and George Konidaris. Hidden parameter markov decision processes: A semiparametric regression approach for discovering latent task parametrizations. In *IJCAI: proceedings of the conference*, volume 2016, page 1432. NIH Public Access, 2016.

Lasse Espeholt, Hubert Soyer, Remi Munos, Karen Simonyan, Vlad Mnih, Tom Ward, Yotam Doron, Vlad Firoiu, Tim Harley, Iain Dunning, et al. Impala: Scalable distributed deep-rl with importance weighted actor-learner architectures. In *International Conference on Machine Learning*, pages 1407–1416. PMLR, 2018.

Linxi Fan, Yuke Zhu, Jiren Zhu, Zihua Liu, Orien Zeng, Anchit Gupta, Joan Creus-Costa, Silvio Savarese, and Li Fei-Fei. Surreal: Open-source reinforcement learning framework and robot manipulation benchmark. In *Conference on Robot Learning*, pages 767–782. PMLR, 2018.

Matteo Hessel, Joseph Modayil, Hado Van Hasselt, Tom Schaul, Georg Ostrovski, Will Dabney, Dan Horgan, Bilal Piot, Mohammad Azar, and David Silver. Rainbow: Combining improvements in deep reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

Taylor Killian, George Konidaris, and Finale Doshi-Velez. Robust and efficient transfer learning with hidden parameter markov decision processes. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017.

Sergey Levine, Chelsea Finn, Trevor Darrell, and Pieter Abbeel. End-to-end training of deep visuomotor policies. *The Journal of Machine Learning Research*, 17(1):1334–1373, 2016.

Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*, 2015.

Boyi Liu, Lujia Wang, Ming Liu, and Chengzhong Xu. Lifelong federated reinforcement learning: A learning architecture for navigation in cloud robotic systems. *arXiv preprint arXiv:1901.06455*, 2019.

Dhruv Mahajan, Nikunj Agrawal, S Sathiya Keerthi, Sundararajan Sellamanickam, and Léon Bottou. An efficient distributed learning algorithm based on effective local functional approximations. *Journal of Machine Learning Research*, 19(1):2942–2978, 2018.

Yishay Mansour, Mehryar Mohri, Jae Ro, and Ananda Theertha Suresh. Three approaches for personalization with applications to federated learning. *arXiv preprint arXiv:2002.10619*, 2020.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015.

Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In *International conference on machine learning*, pages 1928–1937. PMLR, 2016.

Chetan Nadiger, Anil Kumar, and Sherine Abdelhak. Federated reinforcement learning for fast personalization. In *2019 IEEE Second International Conference on Artificial Intelligence and Knowledge Engineering (AIKE)*, pages 123–127. IEEE, 2019.

Supratik Paul, Michael A Osborne, and Shimon Whiteson. Fingerprint policy optimisation for robust reinforcement learning. In *International Conference on Machine Learning*, pages 5082–5091. PMLR, 2019.

Anit Kumar Sahu, Tian Li, Maziar Sanjabi, Manzil Zaheer, Ameet Talwalkar, and Virginia Smith. Federated optimization for heterogeneous networks. *arXiv preprint arXiv:1812.06127*, 1(2):3, 2018.

David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of Go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016.

David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. Mastering the game of Go without human knowledge. *Nature*, 550(7676):354–359, 2017.

Richard S Sutton, Andrew G Barto, et al. *Introduction to reinforcement learning*, volume 135. MIT press Cambridge, 1998.

Richard S Sutton, David A McAllester, Satinder P Singh, Yishay Mansour, et al. Policy gradient methods for reinforcement learning with function approximation. In *NIPs*, volume 99, pages 1057–1063. Citeseer, 1999.

Yee Whye Teh, Victor Bapst, Wojciech Marian Czarnecki, John Quan, James Kirkpatrick, Raia Hadsell, Nicolas Heess, and Razvan Pascanu. Distral: Robust multitask reinforcement learning. *arXiv preprint arXiv:1707.04175*, 2017.

Xiaofei Wang, Chenyang Wang, Xiuhua Li, Victor CM Leung, and Tarik Taleb. Federated deep reinforcement learning for internet of things with decentralized cooperative edge caching. *IEEE Internet of Things Journal*, 7(10):9441–9455, 2020.

Christopher JCH Watkins and Peter Dayan. Q-learning. *Machine learning*, 8(3-4):279–292, 1992.

Sihan Zeng, Aqeel Anwar, Thinh Doan, Justin Romberg, and Arijit Raychowdhury. A decentralized policy gradient approach to multi-task reinforcement learning. *arXiv preprint arXiv:2006.04338*, 2020.

Hankz Hankui Zhuo, Wenfeng Feng, Qian Xu, Qiang Yang, and Yufeng Lin. Federated reinforcement learning. *arXiv preprint arXiv:1901.08277*, 2019.

Supplementary Materials

9 Proof of Theorem 1

We are next to find a task of FedRL having the following property: For some initial distribution d_0, $\forall \pi_1^\star \in \operatorname{argmax}_\pi g_{d_0}(\pi)$, there exist d_1 and $\tilde{\pi}$ such that $g_{d_0}(\tilde{\pi}) < g_{d_0}(\pi_1^\star)$, but $g_{d_1}(\tilde{\pi}) > g_{d_1}(\pi_1^\star)$.

Consider the task of FedRL composed of the following two environments:

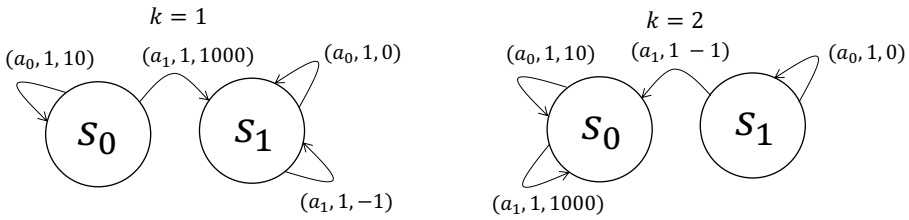

Figure 7: Counterexample in FedRL: The triple means (action, probability, reward) and $\gamma = 0.9$. Note that these two environments share the same action space $\{s_0, s_1\}$, same state space $\{a_0, a_1\}$, and same reward function.

Proof. In the task of FedRL mentioned in Figure 7, we use two real numbers $(p, q) \in [0, 1]^2$ to represent any policy π, where $p = \pi(a_0|s^0)$ and $q = \pi(a_0|s^1)$. Let the initial state distribution be $d_0 = (1, 0)$, which means s_0 is initial state. Therefore, the objective of FedRL is formulated as follows:

$$\max_\pi g_{d_0}(\pi) = \frac{1}{2}\{V_\pi^1(s_0) + V_\pi^2(s_0)\}.$$

It is a continuous function of policy $\pi = (p, q)$ whose support is compact. Therefore, the optimal policy exists. Such optimal policy $\pi^* = (p^*, q^*)$ is not unique, but we assert that $p^* < 1, q^* = 1$.

If $p^* = 1$, then the cumulative rewards in FedRL is $\sum_{t=0}^{\infty} \gamma^t 10 \approx 100$. $p = 0$ beats $p^* = 1$ since $p = 0$ earns 1000 at the very first step in both environments. In this way, we prove that $p^* < 1$. For the choices of q^*, if $q^* < 1$, there is positive probability to take a_1 at s_1. Since there is no probability to reach s_1 in the second environment when the initial state is s_0, we merely consider the first environment for the selection of q^*. $q^* < 1$ means there is positive probability to select a_1 at s_1. Yet selection of a_1 at s_1 leads to negative reward -1, which is obviously inferior to the selection of a_0 whose reward is 0 at s_1. Therefore, we prove that $q^* = 1$.

However, $\pi^* = (p^*, q^*)$ determined above is no longer the optimal solution when the initial state distribution changes to $d_1 = (0, 1)$. When starting from s_1, $q^* = 1$ means the agent never select a_1 at s_1 and the agent never reaches s_0 in the second environment. Although selection of a_1 leads to a negative reward of -2 in both environments, yet the positive reward of actions at s_0 obviously compensates for the loss of choosing a_1 at s_1. Therefore, $\pi^* = (p^*, q^*)$ above is no longer the optimal policy when the initial state distribution is formulated as $d_1 = (0, 1)$. □

However, the above example is, to some extent, tricky, since both of the involved environments in FedRL are not irreducible. Therefore, we propose the following example of FedRL with a positive lead probability $\tau > 0$ in Figure 8. We claim that if leak probability τ is small enough, the previous argument still holds.

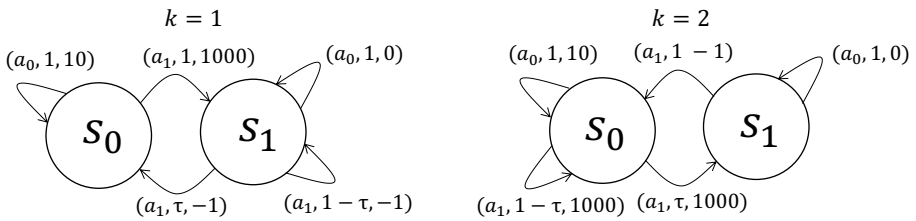

Figure 8: The modified tasks of FedRL with two connected and irreducible environments.

Proof. In the task of FedRL mentioned in Figure 8, we consider the following two initial state distributions $\{d_0 = (1,0), d_1 = (0,1)\}$. In this way, the objective functions in these two environments are denoted as $g_{d_0}(\pi) = \bar{V}_\pi^\tau(s_0)$ and $g_{d_1}(\pi) = \bar{V}_\pi^\tau(s_1)$, where τ is the leak probability and $\pi = (p, q)$. It is clear that both $g_{d_0}(\pi)$ and $g_{d_1}(\pi)$ are uniform continuous with respect to $(p, q, \tau) \in (0,1)^3$. We denote the set of optimal solutions w.r.t. these two initial state distributions as $\Gamma_0^\tau = \{(p_0^\tau, p_0^\tau)\}$ and $\Gamma_1^\tau = \{(p_1^\tau, p_1^\tau)\}$, and their objective values as $\tilde{M}_0^\tau$ and $\tilde{M}_0^\tau$. It is easy to see that both Γ_0^τ and Γ_1^τ are compact sets.

Taking FedRL described in Figure 7 as a special case with $\tau = 0$, we have already proved that $p_0^0 < 1, q_0^0 = 1, \forall (p_0^0, q_0^0) \in \Gamma_0^0$, and $q_1^0 < 1, \forall (p_1^0, q_1^0) \in \Gamma_1^0$. We claim that when τ is sufficiently small, there exists $\delta > 0$, `s.t.` $q_0^\tau > 1 - \delta > a_1^\tau$.

Firstly, we define $\alpha = \sum_{\Gamma_1^0} q_1^0$ with $\alpha < 1$ by the compactness of Γ_1^0. Then we derive $M_1 = \max_{q \geq \frac{1+\alpha}{2}} g_{d_1}^0(\pi)$ with $M_1 \leq \tilde{M}_1^0$. The uniform continuity of objective values w.r.t. τ tells us: $\exists \bar{\tau}_1 > 0$, $\forall \pi = (p, q)$, $\forall \tau < \bar{\tau}_1$, `s.t.` $|g_{d_1}^\tau(\pi) - g_{d_1}^0(\pi)| < \frac{\tilde{M}_1^0 - M_1}{4}$. Therefore, it is easy to tell $\forall \tau < \bar{\tau}_1, q_1^\tau < 1 - \delta$, where $\delta = \frac{1-\alpha}{2}$.

Following the same strategy, we are able to derive that $\exists \bar{\tau}_2 > 0, \forall \tau < \bar{\tau}_2, q_0^\tau > 1 - \delta$, which along with $\forall \tau < \bar{\tau}_1, q_1^\tau < 1 - \delta$ leads to a contradiction. □

10 Proof of Lemma1, 2

Proof of Lemma 1. The lower bound of weighted value function $\bar{V}_\pi$ is derived as:

$$\begin{aligned}
\bar{V}^\pi =& \frac{1}{n}\sum_{i=1}^n V_i^\pi = \frac{1}{n}\sum_{i=1}^n (I_{|\mathcal{S}|} - \gamma \mathcal{P}_i^\pi)^{-1} R^\pi \\
=& \frac{1}{n}\sum_{i=1}^n \sum_{k=0}^\infty (\gamma \mathcal{P}_i^\pi)^k R^\pi \\
\succcurlyeq & \sum_{k=0}^\infty (\gamma \frac{1}{n}\sum_{i=1}^n \mathcal{P}_i^\pi)^k R^\pi = (I_{|\mathcal{S}|} - \gamma \bar{\mathcal{P}}^\pi)^{-1} R^\pi = V_I^\pi,
\end{aligned}$$

where the second and fourth equalities come from $(I - A)^{-1} = \sum_{k=0}^\infty A^k$ with setting A^0 as I, and $\bar{V}^\pi \succcurlyeq V_I^\pi$ indicating $\bar{V}^\pi(s) \geq V_I^\pi(s)$. □

Proof of Lemma 2. In fact, by definition of $\bar{V}^\pi$, for any $s \in \mathcal{S}$, we have:

$$\begin{aligned}
\left|V_I^\pi(s) - \bar{V}^\pi(s)\right| &\leq \frac{1}{n}\sum_{k=1}^n \left|V_I^\pi(s) - \bar{V}_k^\pi(s)\right| \\
&\leq \frac{1}{n}\sum_{k=1}^n \left\|V_I^\pi - \bar{V}_k^\pi\right\|_\infty
\end{aligned}$$

By Bellman equation, we have:

$$\begin{aligned}
\left|V_I^\pi(s) - \bar{V}_k^\pi(s)\right| &= \gamma \left| \sum_{s'} \left(\frac{1}{n}\sum_{k=1}^n \mathcal{P}_k^\pi(s'|s) V_I^\pi(s') - \mathcal{P}_k^\pi(s'|s) V_k^\pi(s') \right) \right| \\
&\leq \frac{\kappa_1 \gamma}{1-\gamma} + \gamma \|V_I^\pi - \bar{V}_k^\pi\|_\infty
\end{aligned}$$

Thus, we have the final conclusion:

$$\|V_I^\pi - \bar{V}^\pi\|_\infty \leq \frac{\gamma\kappa_1}{(1-\gamma)^2}$$

□

11 Proof of Theorem 2

Denote the Bellman Operator in the k-th environment as:

$$\mathcal{T}_k Q(s,a) = R(s,a) + \gamma \sum_{s'} P_k(s'|s,a) \max_{a'} Q(s',a')$$

The average Bellman Operator as:

$$\mathcal{T} Q = \frac{1}{n}\sum_{k=1}^n \mathcal{T}_k Q$$

Theorem 4. *$\mathcal{T}$ is a γ-contractor. For any Q_1 and Q_2, it satisfies:*

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

Proof. By definition of $\mathcal{T}$, we have:

$$\begin{aligned}\|\mathcal{T}Q_1 - \mathcal{T}Q_2\|_\infty &\leq \frac{1}{n}\sum_{k=1}^n \|\mathcal{T}_k Q_1 - \mathcal{T}_k Q_2\|_\infty \\ &\leq \gamma \|\mathcal{T}_k Q_1 - \mathcal{T}_k Q_2\|_\infty\end{aligned}$$

□

By Theorem 4, there exists a fixed point Q^* satisfies $\mathcal{T}Q^* = Q^*$, which is also the optimal Q value function w.r.t. standard MDP with transition dynamics $\bar{\mathcal{P}}$.

Besides, we also define a general version of average Bellman Operator:

$$\mathcal{T}_E = \frac{1}{n}\sum_{k=1}^n \mathcal{T}_k^E$$

and a smooth version:

$$\widetilde{\mathcal{T}}_E = \frac{1}{n}\sum_{k=1}^n \prod_{t=1}^E (\lambda_t \mathcal{T}_k + (1-\lambda_t) Id)$$

In other words, the update rule is:

$$\begin{aligned}Q_{t+1}^k &= (1-\lambda_t) Q_t^k + \lambda_t \mathcal{T}_k Q_t^k \\ Q_{t+1}^k &= \begin{cases} \frac{1}{n}\sum_{k=1}^n Q_{t+1}^k, & \text{if } t+1 \in \mathcal{I}_E \\ Q_{t+1}^k, & \text{if } t+1 \notin \mathcal{I}_E \end{cases}\end{aligned}$$

We also denote $\bar{Q}_t = \frac{1}{n}\sum_{k=1}^n Q_t^k$.

Lemma 3 (One step recursion)**.** *We have:*

$$\|\bar{Q}_{t+1} - Q^*\|_\infty \leq (1-(1-\gamma)\lambda_t)\|\bar{Q}_t - Q^*\|_\infty + \frac{\lambda_t\gamma}{n}\sum_{k=1}^n \|Q_t^k - \bar{Q}_t\|_\infty$$

Proof. As $Q^* = \frac{1}{n}\sum_{k=1}^n \mathcal{T}_k Q^*$, we have:

$$\begin{aligned}
\|\bar{Q}_{t+1} - Q^*\|_\infty =& \|(1-\lambda_t)\bar{Q}_t + \frac{\lambda_t}{n}\sum_{k=1}^n \mathcal{T}_k Q_t^k - Q^*\|_\infty \\
=& \|(1-\lambda_t)(\bar{Q}_t - Q^*) + \frac{\lambda_t}{n}\sum_{k=1}^n (\mathcal{T}_k Q_t^k - \mathcal{T}_k Q^*)\|_\infty \\
\leq& (1-\lambda_t)\|\bar{Q}_t - Q^*\|_\infty + \frac{\lambda_t}{n}\sum_{k=1}^n \|\mathcal{T}_k Q_t^k - \mathcal{T}_k Q^*\|_\infty \\
\leq& (1-\lambda_t)\|\bar{Q}_t - Q^*\|_\infty + \frac{\gamma\lambda_t}{n}\sum_{k=1}^n \|Q_t^k - Q^*\|_\infty \\
\leq& (1-(1-\gamma)\lambda_t)\|\bar{Q}_t - Q^*\|_\infty + \frac{\gamma\lambda_t}{n}\sum_{k=1}^n \|Q_t^k - \bar{Q}_t\|_\infty
\end{aligned}$$

□

Lemma 4 (Value variance)**.** *Suppose $\lambda_t \leq 2\lambda_{t+E}$ and $Q_0^k \in [0, \frac{1}{1-\gamma}]$, we have:*

$$\frac{1}{n}\sum_{k=1}^n \|Q_t^k - \bar{Q}_t\|_\infty \leq \frac{4\lambda_t(E-1)}{(1-\gamma)}$$

Proof. Noting that $\mathbb{E}\|X - EX\|_\infty \leq 2\mathbb{E}\|X\|_\infty$, and for $\forall t$, there exists $t_0 \leq t$ and $t - t_0 \leq E-1$, such that $Q_{t_0}^k = \bar{Q}_{t_0}$. Thus we have:

$$\begin{aligned}
\frac{1}{n}\sum_{k=1}^n \|Q_t^k - \bar{Q}_t\|_\infty &= \frac{1}{n}\sum_{k=1}^n \|Q_t^k - \bar{Q}_{t_0} + \bar{Q}_{t_0} - \bar{Q}_t\|_\infty \\
&\leq \frac{2}{n}\sum_{k=1}^n \|Q_t^k - \bar{Q}_{t_0}\|_\infty \\
&= \frac{2}{n}\sum_{k=1}^n \|\sum_{t'=t_0}^{t-1} \lambda_{t'}\left(\mathcal{T}_k Q_{t'}^k - Q_{t'}^k\right)\|_\infty \\
&\leq \frac{2}{n}\sum_{k=1}^n \sum_{t'=t_0}^{t-1} \lambda_{t'}\|\mathcal{T}_k Q_{t'}^k - Q_{t'}^k\|_\infty \\
&\leq \frac{4\lambda_t(E-1)}{(1-\gamma)}
\end{aligned}$$

where the last inequality holds by $Q_t^k \in [0, \frac{1}{1-\gamma}]$. □

Proof of Theorem 2. By Lemma 3 and Lemma 4, we have:

$$\|\bar{Q}_{t+1} - Q^*\|_\infty \leq (1-(1-\gamma)\lambda_t)\|\bar{Q}_t - Q^*\|_\infty + \frac{4\lambda_t^2\gamma(E-1)}{(1-\gamma)}$$

To simplify, we denote $\Delta_{t+1} = \|\bar{Q}_{t+1} - Q^*\|_\infty$ and $C = \frac{4\gamma(E-1)}{(1-\gamma)}$, which leads to:

$$\Delta_{t+1} \leq (1-(1-\gamma)\lambda_t)\Delta_t + \lambda_t^2 \cdot C$$

By setting $\lambda_t = \frac{\alpha}{t+\beta}$, we will prove $\Delta_t \leq \frac{\zeta}{t+\beta}$ recursively:

$$\begin{aligned}
\Delta_{t+1} &\leq (1-(1-\gamma)\lambda_t)\frac{\zeta}{t+\beta} + \lambda_t^2 \cdot C \\
&= \frac{(t+\beta-1)\zeta}{(t+\beta)^2} + \frac{(1-(1-\gamma)\alpha)\zeta + \alpha^2 \cdot C}{(t+\beta)^2} \\
&\leq \frac{\zeta}{t+\beta+1}
\end{aligned}$$

Trivially, we can set $\alpha = \frac{2}{1-\gamma}$ and $\zeta = \frac{4C}{(1-\gamma)^2} = \frac{16\gamma(E-1)}{(1-\gamma)^3}$. Besides, to satisfy $\lambda_t \leq 2\lambda_{t+E}$, we can set $\beta = E$. Thus, we have:

$$\|\bar{Q}_t - Q^*\|_\infty \leq \frac{16\gamma(E-1)}{(1-\gamma)^3(t+E)}$$

□

12 Proof of Theorem 3

Theorem 5 (Global Optimality)**.** *Denote the optimal policy as π^*, for any given policy $\pi \in \Delta(\mathcal{A})^{\mathcal{S}}$, we have following inequality holds:*

$$\frac{1}{n}\sum_{k=1}^{n} V_k^{\pi^*}(\mu) - \frac{1}{n}\sum_{k=1}^{n} V_k^{\pi}(\mu) \leq 2(2L\eta+1)\rho\sqrt{|\mathcal{S}|}\left(\kappa + \|G^\eta(\pi)\|_2\right)$$

where

$$G^\eta(\pi) = \frac{\text{Proj}\left(\pi + \eta\frac{1}{n}\sum_{k=1}^{n}\nabla_\pi V_k^\pi(\mu)\right) - \pi}{\eta}$$
$$G_k^\eta(\pi) = \frac{\text{Proj}\left(\pi + \eta\nabla_\pi V_k^\pi(\mu)\right) - \pi}{\eta}$$

Proof. By definition, we have:

$$\begin{aligned}
\Delta(\pi) &= \frac{1}{n}\sum_{k=1}^{n} V_k^{\pi^*}(\mu) - \frac{1}{n}\sum_{k=1}^{n} V_k^{\pi}(\mu) \\
&= \frac{1}{n}\sum_{k=1}^{n}\left(V_k^{\pi^*}(\mu) - V_k^{\pi}(\mu)\right) \\
&= \frac{1}{n}\sum_{k=1}^{n}\frac{1}{1-\gamma}\mathbb{E}_{d_{\pi^*,\mu,k}}\langle\pi^*(\cdot|s), A_k^\pi(s,\cdot)\rangle \\
&= \frac{1}{n}\sum_{k=1}^{n}\frac{1}{1-\gamma}\mathbb{E}_{d_{\pi^*,\mu,k}}\langle\pi^*(\cdot|s) - \pi(\cdot|s), A_k^\pi(s,\cdot)\rangle \\
&= \frac{1}{n}\sum_{k=1}^{n}\frac{1}{1-\gamma}\mathbb{E}_{d_{\pi^*,\mu,k}}\langle\pi^*(\cdot|s) - \pi(\cdot|s), Q_k^\pi(s,\cdot)\rangle \\
&\leq \frac{1}{n}\sum_{k=1}^{n}\frac{1}{1-\gamma}\mathbb{E}_{d_{\pi^*,\mu,k}}\max_{\widetilde{\pi}}\langle\widetilde{\pi}(\cdot|s) - \pi(\cdot|s), Q_k^\pi(s,\cdot)\rangle \\
&\leq \frac{\rho}{n}\sum_{k=1}^{n}\max_{\widetilde{\pi}}\langle\widetilde{\pi} - \pi, \nabla_\pi V_k^\pi(\mu)\rangle \\
&\leq \frac{2\rho\sqrt{|\mathcal{S}|}}{n}\sum_{k=1}^{n}\max_{\pi+\delta\in\Delta(\mathcal{A})^{\mathcal{S}},\|\delta\|_2\leq 1}\delta^T\nabla_\pi V_k^\pi(\mu)
\end{aligned}$$

Denote $\pi_k^+ = \pi + \eta G_k^\eta(\pi)$, we have:

$$\begin{aligned}
&\max_{\pi+\delta\in\Delta(\mathcal{A})^{\mathcal{S}},\|\delta\|_2\leq 1}\delta^T\nabla_\pi V_k^\pi(\mu) \\
&\leq \left\|\nabla_\pi V_k^\pi(\mu) - \nabla_\pi V^{\pi_k^+}(\mu)\right\|_2 + \max_{\pi+\delta\in\Delta(\mathcal{A})^{\mathcal{S}},\|\delta\|_2\leq 1}\delta^T\nabla_\pi V^{\pi_k^+}(\mu) \\
&\leq (2L\eta+1)\|G_k^\eta(\pi)\|_2
\end{aligned}$$

Thus, we have:

$$\begin{aligned}\Delta(\pi) \leq & \frac{2(2L\eta+1)\rho\sqrt{|\mathcal{S}|}}{n}\sum_{k=1}^{n}\|G_k^\eta(\pi)\|_2 \\ \leq & \frac{2(2L\eta+1)\rho\sqrt{|\mathcal{S}|}}{n}\sum_{k=1}^{n}\|G_k^\eta(\pi)-G^\eta(\pi)\|_2 + 2(2L\eta+1)\rho\sqrt{|\mathcal{S}|}\|G^\eta(\pi)\|_2 \\ \leq & 2(2L\eta+1)\rho\sqrt{|\mathcal{S}|}\left(\kappa+\|G^\eta(\pi)\|_2\right)\end{aligned}$$

□

Denote the update rule as:

$$\begin{aligned}\pi_{t+1}^k &= \pi_t^k + \eta_t G_k^{\eta_t}(\pi_t^k) \\ \pi_{t+1}^k &= \begin{cases}\frac{1}{n}\sum_{k=1}^{n}\pi_{t+1}^k, & \text{if } t+1\in\mathcal{I}_E \\ \pi_{t+1}^k, & \text{if } t+1\notin\mathcal{I}_E\end{cases}\end{aligned}$$

And we also denote $\bar{\pi}_t = \frac{1}{n}\sum_{k=1}^{n}\pi_t^k$ and $\bar{\pi}_{t+1}^+ = \bar{\pi} + \eta_t G^{\eta_t}(\bar{\pi})$.

Lemma 5 (One step recursion)**.** *Measure the environment heterogeneity with κ_2, we have:*

$$\begin{aligned}F(\bar{\pi}_{t+1}) - F(\bar{\pi}_t) \geq & -\frac{\eta_t\kappa\sqrt{|\mathcal{A}|}}{(1-\gamma)^2} - \frac{\eta_t L\sqrt{|\mathcal{A}|}}{(1-\gamma)^2}\cdot\frac{1}{n}\sum_{k=1}^{n}\|\pi_t^k-\bar{\pi}_t\|_2 + (\eta_t-\eta_t^2 L)\|G^{\eta_t}(\bar{\pi}_t)\|_2^2 \\ & - 2\kappa^2\eta_t^2 L - \frac{2\eta_t^2 L^3}{n^2}\left(\sum_{k=1}^{n}\|\pi_t^k-\bar{\pi}_t\|_2\right)^2\end{aligned}$$

Proof. WLOG, we denote $F_k(\pi) = V_k^\pi(\mu)$ and $F(\pi) = \frac{1}{n}\sum_{k=1}^{n}F_k(\pi)$. By L-smoothness, we have:

$$F(\bar{\pi}_{t+1}) - F(\bar{\pi}_t) \geq \langle\nabla F(\bar{\pi}_t), \bar{\pi}_{t+1}-\bar{\pi}_t\rangle - \frac{L}{2}\|\bar{\pi}_{t+1}-\bar{\pi}_t\|_2^2$$

Noting that $\bar{\pi}_{t+1}-\bar{\pi}_t = \eta_t\frac{1}{n}\sum_{k=1}^{n}G_k^{\eta_t}(\pi_t^k)$, we have:

$$F(\bar{\pi}_{t+1}) - F(\bar{\pi}_t) \geq \eta_t\langle\nabla F(\bar{\pi}_t), \frac{1}{n}\sum_{k=1}^{n}G_k^{\eta_t}(\pi_t^k)\rangle - \frac{\eta_t^2 L}{2}\|\frac{1}{n}\sum_{k=1}^{n}G_k^{\eta_t}(\pi_t^k)\|_2^2$$

As $\bar{\pi}_{t+1}^+ = \bar{\pi}_t + \eta_t G^{\eta_t}(\bar{\pi}_t)$ and the first order stationary condition, we have:

$$\langle\bar{\pi}_{t+1}^+ - \bar{\pi}_t - \eta_t\nabla F(\bar{\pi}_t), \bar{\pi}_{t+1}^+ - \bar{\pi}_t\rangle \leq 0$$

which is equivalent with:

$$\langle G^{\eta_t}(\bar{\pi}_t) - \nabla F(\bar{\pi}_t), G^{\eta_t}(\bar{\pi}_t)\rangle \leq 0$$

Therefore, we have:

$$\begin{aligned}F(\bar{\pi}_{t+1}) - F(\bar{\pi}_t) \geq & \eta_t\langle\nabla F(\bar{\pi}_t), \frac{1}{n}\sum_{k=1}^{n}G_k^{\eta_t}(\pi_t^k) - G^{\eta_t}(\bar{\pi}_t)\rangle \\ & + \eta_t\|G^{\eta_t}(\bar{\pi}_t)\|_2^2 - \frac{\eta_t^2 L}{2}\|\frac{1}{n}\sum_{k=1}^{n}G_k^{\eta_t}(\pi_t^k)\|_2^2 \\ \geq & -\eta_t\|\nabla F(\bar{\pi}_t)\|_2\cdot\|\frac{1}{n}\sum_{k=1}^{n}G_k^{\eta_t}(\pi_t^k) - G^{\eta_t}(\bar{\pi}_t)\|_2 \\ & + \eta_t\|G^{\eta_t}(\bar{\pi}_t)\|_2^2 - \frac{\eta_t^2 L}{2}\|\frac{1}{n}\sum_{k=1}^{n}G_k^{\eta_t}(\pi_t^k)\|_2^2\end{aligned}$$

Noting that:

$$\begin{aligned}\|\frac{1}{n}\sum_{k=1}^{n}G_k^{\eta_t}(\pi_t^k)-G^{\eta_t}(\bar{\pi}_t)\|_2 \leq & \|\frac{1}{n}\sum_{k=1}^{n}G_k^{\eta_t}(\pi_t^k)-G^{\eta_t}(\pi_t^k)\|_2 \\ & +\|\frac{1}{n}\sum_{k=1}^{n}G^{\eta_t}(\pi_t^k)-G^{\eta_t}(\bar{\pi}_t)\|_2 \\ \leq & \kappa+\frac{L}{n}\sum_{k=1}^{n}\|\pi_t^k-\bar{\pi}_t\|_2\end{aligned}$$

Noting that $\|\nabla F\|_2 \leq \frac{\sqrt{|\mathcal{A}|}}{(1-\gamma)^2}$, we have:

$$\begin{aligned}F(\bar{\pi}_{t+1})-F(\bar{\pi}_t) \geq & -\frac{\eta_t\kappa\sqrt{|\mathcal{A}|}}{(1-\gamma)^2}-\frac{\eta_t L\sqrt{|\mathcal{A}|}}{(1-\gamma)^2}\cdot\frac{1}{n}\sum_{k=1}^{n}\|\pi_t^k-\bar{\pi}_t\|_2 \\ & +\eta_t\|G^{\eta_t}(\bar{\pi}_t)\|_2^2-\frac{\eta_t^2 L}{2}\|\frac{1}{n}\sum_{k=1}^{n}G_k^{\eta_t}(\pi_t^k)\|_2^2\end{aligned}$$

Besides, by $\|a+b\|_2^2 \leq 2\|a\|_2^2+2\|b\|_2^2$, we have:

$$\begin{aligned}\frac{1}{2}\|\frac{1}{n}\sum_{k=1}^{n}G_k^{\eta_t}(\pi_t^k)\|_2^2 \leq & \|\frac{1}{n}\sum_{k=1}^{n}G_k^{\eta_t}(\pi_t^k)-G^{\eta_t}(\bar{\pi}_t)\|_2^2+\|G^{\eta_t}(\bar{\pi}_t)\|_2^2 \\ \leq & (\kappa+\frac{L}{n}\sum_{k=1}^{n}\|\pi_t^k-\bar{\pi}_t\|_2)^2+\|G^{\eta_t}(\bar{\pi}_t)\|_2^2 \\ \leq & 2\kappa^2+\frac{2L^2}{n^2}\left(\sum_{k=1}^{n}\|\pi_t^k-\bar{\pi}_t\|_2\right)^2+\|G^{\eta_t}(\bar{\pi}_t)\|_2^2\end{aligned}$$

Gathering all these together, we have:

$$\begin{aligned}F(\bar{\pi}_{t+1})-F(\bar{\pi}_t) \geq & -\frac{\eta_t\kappa\sqrt{|\mathcal{A}|}}{(1-\gamma)^2}-\frac{\eta_t L\sqrt{|\mathcal{A}|}}{(1-\gamma)^2}\cdot\frac{1}{n}\sum_{k=1}^{n}\|\pi_t^k-\bar{\pi}_t\|_2 \\ & +(\eta_t-\eta_t^2 L)\|G^{\eta_t}(\bar{\pi}_t)\|_2^2 \\ & -2\kappa^2\eta_t^2 L-\frac{2\eta_t^2 L^3}{n^2}\left(\sum_{k=1}^{n}\|\pi_t^k-\bar{\pi}_t\|_2\right)^2\end{aligned}$$

□

Lemma 6 (Policy Variance)**.** *By $\|\nabla F_k\|_2 \leq \frac{\sqrt{\mathcal{A}}}{(1-\gamma)^2}$ and assuming $\eta_t \leq 2\eta_{t+E}$, we have:*

$$\frac{1}{n}\sum_{k=1}^{n}\|\pi_t^k-\bar{\pi}_t\|_2^2 \leq \frac{4\eta_t^2(E-1)^2|\mathcal{A}|}{(1-\gamma)^4}$$

Proof. For $\forall t$, there exists $t_0 \leq t$ and $t-t_0 \leq E-1$, such that $\pi_{t_0}^k=\bar{\pi}_{t_0}$. Thus we have:

$$\begin{aligned}\frac{1}{n}\sum_{k=1}^{n}\|\pi_t^k-\bar{\pi}_t\|_2^2 &= \frac{1}{n}\sum_{k=1}^{n}\|\pi_t^k-\bar{\pi}_{t_0}+\bar{\pi}_{t_0}-\bar{\pi}_t\|_2^2 \\ &\leq \frac{1}{n}\sum_{k=1}^{n}\|\pi_t^k-\bar{\pi}_{t_0}\|_2^2\end{aligned}$$

where the last inequality holds by $\mathbb{E}\|X-\mathbb{E}X\|_2^2 \le \mathbb{E}\|X\|_2^2$. Noting that:

$$\pi_t^k = \pi_{t_0}^k + \sum_{t'=t_0}^{t-1} \eta_{t'} G_k^{\eta_{t'}}(\pi_{t'-1}^k)$$

Thus, we have:

$$\begin{aligned}
\frac{1}{n}\sum_{k=1}^n \|\pi_t^k - \bar{\pi}_t\|_2^2 &\le \frac{1}{n}\sum_{k=1}^n \| \sum_{t'=t_0}^{t-1} \eta_{t'} G_k^{\eta_{t'}}(\pi_{t'-1}^k)\|_2^2 \\
&\le \frac{1}{n}\sum_{k=1}^n (t-t_0) \sum_{t'=t_0}^{t-1} \eta_{t'}^2 \|G_k^{\eta_{t'}}(\pi_{t'-1}^k)\|_2^2 \\
&\le \frac{4\eta_t^2 (E-1)^2 |\mathcal{A}|}{(1-\gamma)^4}
\end{aligned}$$

where the last inequality holds by $\|G_k^\eta(\pi)\|_2 \le \|\nabla F_k(\pi)\|_2$ □

Theorem 6 (Full version of Theorem 3)**.** *By setting* $\eta_t = \sqrt{\frac{E}{12L^2(t+E/3)}}$*, we have:*

$$\begin{aligned}
\min_{t=0,\dots,T-1} \|G^{\eta_t}(\bar{\pi}_t)\|_2^2 \le & \frac{2\kappa\sqrt{|\mathcal{A}|}}{(1-\gamma)^2} + \sqrt{\frac{24L^2}{E}} \cdot \frac{F(\bar{\pi}_T) - F(\bar{\pi}_0)}{\sqrt{T}} \\
& + \sqrt{\frac{E}{6L^2}} \cdot \left(\frac{2E(E-1)|\mathcal{A}|L}{(1-\gamma)^4} + 2\kappa^2 L \right) \cdot \frac{\log(1+\frac{3T}{E})}{\sqrt{T}} \\
& = \widetilde{O}\left(\frac{\kappa\sqrt{|\mathcal{A}|}}{(1-\gamma)^2} + \frac{|\mathcal{A}|L}{\sqrt{T}} \right)
\end{aligned}$$

where $\widetilde{O}(\cdot)$ *omits logarithmic terms and some constants.*

Proof. By Lemma 5, Lemma 6 and $\eta_t \le \frac{1}{2L}$, we have:

$$\begin{aligned}
F(\bar{\pi}_{t+1}) - F(\bar{\pi}_t) \ge & -\frac{\eta_t \kappa \sqrt{|\mathcal{A}|}}{(1-\gamma)^2} - \frac{\eta_t L\sqrt{|\mathcal{A}|}}{(1-\gamma)^2} \cdot \frac{1}{n}\sum_{k=1}^n \|\pi_t^k - \bar{\pi}_t\|_2 \\
& + (\eta_t - \eta_t^2 L)\|G^{\eta_t}(\bar{\pi}_t)\|_2^2 \\
& - 2\kappa^2 \eta_t^2 L - \frac{2\eta_t^2 L^3}{n^2}\left(\sum_{k=1}^n \|\pi_t^k - \bar{\pi}_t\|_2 \right)^2 \\
\ge & -\frac{\eta_t \kappa \sqrt{|\mathcal{A}|}}{(1-\gamma)^2} - \frac{2\eta_t^2 (E-1) L |\mathcal{A}|}{(1-\gamma)^4} \\
& + \frac{\eta_t}{2}\|G^{\eta_t}(\bar{\pi}_t)\|_2^2 - 2\kappa^2\eta_t^2 L - \frac{8\eta_t^4 (E-1)^2 |\mathcal{A}| L^3}{(1-\gamma)^4} \\
\ge & -\frac{\eta_t \kappa \sqrt{|\mathcal{A}|}}{(1-\gamma)^2} - \frac{2\eta_t^2 E(E-1)|\mathcal{A}|L}{(1-\gamma)^4} \\
& - 2\kappa^2\eta_t^2 L + \frac{\eta_t}{2}\|G^{\eta_t}(\bar{\pi}_t)\|_2^2
\end{aligned}$$

Summing over $t = 0, 1, ..., T-1$, we have:

$$\begin{aligned}
F(\bar{\pi}_T) - F(\bar{\pi}_0) \ge & -\frac{\kappa\sqrt{|\mathcal{A}|}}{(1-\gamma)^2}\sum_{t=0}^{T-1}\eta_t - \frac{2E(E-1)|\mathcal{A}|L}{(1-\gamma)^4}\sum_{t=0}^{T-1}\eta_t^2 \\
& - 2\kappa^2 L \sum_{t=0}^{T-1}\eta_t^2 + \min_{t=0,\dots,T-1}\|G^{\eta_t}(\bar{\pi}_t)\|_2^2 \sum_{t=0}^{T-1}\frac{\eta_t}{2}
\end{aligned}$$

Re-aranging above inequality, we have:

$$\min_{t=0,\ldots,T-1} \|G^{\eta_t}(\bar{\pi}_t)\|_2^2 \leq \frac{2\kappa\sqrt{|\mathcal{A}|}}{(1-\gamma)^2} + \frac{F(\bar{\pi}_T) - F(\bar{\pi}_0)}{\sum_{t=0}^{T-1}\eta_t} + \left(\frac{2E(E-1)|\mathcal{A}|L}{(1-\gamma)^4} + 2\kappa^2 L\right) \cdot \frac{\sum_{t=0}^{T-1}\eta_t^2}{\sum_{t=0}^{T-1}\eta_t}$$

By setting $\eta_t = \sqrt{\frac{E}{12L^2(t+E/3)}}$, which satisfying $\eta_t \leq \frac{1}{2L}$ and $\eta_t \leq 2\eta_{t+E}$, we have

$$\min_{t=0,\ldots,T-1} \|G^{\eta_t}(\bar{\pi}_t)\|_2^2 \leq \frac{2\kappa\sqrt{|\mathcal{A}|}}{(1-\gamma)^2} + \sqrt{\frac{24L^2}{E}} \cdot \frac{F(\bar{\pi}_T) - F(\bar{\pi}_0)}{\sqrt{T}} + \sqrt{\frac{E}{6L^2}} \cdot \left(\frac{2E(E-1)|\mathcal{A}|L}{(1-\gamma)^4} + 2\kappa^2 L\right) \cdot \frac{\log(1+\frac{3T}{E})}{\sqrt{T}}$$

where we use the following inequalities:

$$\sum_{t=0}^{T-1} \frac{1}{\sqrt{t+a}} \geq 2(\sqrt{T+a} - \sqrt{a}) = \frac{2T}{\sqrt{T+a}+\sqrt{a}} \geq \sqrt{\frac{T}{2}},$$
$$\sum_{t=0}^{T-1} \frac{1}{t+a} \leq \sum_{t=0}^{T-1} \log\left(1 + \frac{1}{t+a}\right) = \log\frac{T+a}{a}.$$

□

13 Details of Empirical Results

13.1 Details of constructed environments.

We construct tabular environments for ablation study on choices of E, and additionally modify several classical control tasks to evaluate deep methods and personalization heuristics.

- `Random MDPs` is composed of n environment, `Random MDP`. `Random MDPs` fix a randomly chosen reward function R and generate a set of transition dynamics for each environment. Specifically, we set $N = 5$ in the task of FedRL, and additionally sample $M = 20$ transition dynamics (element-wisely Bernoulli distributed), `i.e.` 20 novel environments of `Random MDP` with same R, to test performance of generalization; we set $\gamma = 0.9$; when testing the impact of environment heterogeneity, we evaluate `QAvg` and `SoftPAvg` with $E = 4$, and `ProjPAvg` with $E = 32$.
- `Windy Cliffs` is composed of n environment, `Windy Cliff`. `Windy Cliff` is a modified version of a classic gridworld example from Sutton et al. (1998): `Cliff Walking` environment. The agent is expected to arrive the goal as fast as possible while avoiding falling off the cliff. Just like the modified version considered in Paul et al. (2019), we introduce a structured random noise in the environment, intensity θ of wind blowing from north. Specifically, θ is uniformly sampled from $U_{[0,1]}$, which means the agent could end up going down even if she does not intend to do that with a probability of $\frac{\theta}{3}$. In our setting, we experiment with the map of size 4×4, and set the reward as 100 and -100 for achieving the goal and falling off the cliff. Similarly, we set $n = 5$ in the task of FedRL, and sample 20 novel environments of `Windy Cliff` to test performance of generalization; we set $\gamma = 0.95$; when testing the impact of environment heterogeneity, we evaluate `QAvg` and `SoftPAvg` with $E = 4$, and `ProjPAvg` with $E = 32$.
- `CartPoles`: We construct `CartPoles` from `CartPole`. Different pole length indicates different pole mass, which leads to different state transition. Specifically, the pole length follows the uniform distribution $\mathcal{U}_{[0.2,1.8]}$. Additionally, we choose $N = 5$, $M = 20$ and $\gamma = 0.99$ in the construction of FedRL.
- `Acrobats`: We construct `Acrobats` from `Acrobat`. Specifically, the mass of pole 1 follows the uniform distribution $\mathcal{U}_{[0.5,1.5]}$ when its pole length is fixed. Additionally, we choose $N = 5$, $M = 20$ and $\gamma = 0.99$ in the construction of FedRL.

- `Halfcheetahs`: We construct `Halfcheetahs` from `Halfcheetah`. Specifically, the pole length of `bthigh` follows the uniform distribution $\mathcal{U}_{[0.1005,0.1855]}$, and the pole length of `fthigh` follows the uniform distribution $\mathcal{U}_{[0.1005,0.1655]}$. Additionally, we choose $N = 5$, $M = 20$ and $\gamma = 0.99$ in the construction of FedRL.
- `Hoppers`: We construct `Hoppers` from `Hopper`. Specifically, the leg size follows the uniform distribution $\mathcal{U}_{[0.03,0.05]}$. Additionally, we choose $N = 5$, $M = 20$ and $\gamma = 0.99$ in the construction of FedRL.