Journal of Artificial Intelligence Research 74 (2022) 1565-1597 Submitted 06/2022; published 08/2022

Joint Optimization of Concave Scalarized Multi-Objective Reinforcement Learning with Policy Gradient Based Algorithm

Qinbo Bai BAI113@PURDUE.EDU
Purdue University

Mridul Agarwal AGARW180@PURDUE.EDU
Purdue University,

Vaneet Aggarwal VANEET@PURDUE.EDU
Purdue University

Abstract

Many engineering problems have multiple objectives, and the overall aim is to optimize a non-linear function of these objectives. In this paper, we formulate the problem of maximizing a non-linear concave function of multiple long-term objectives. A policy-gradient based model-free algorithm is proposed for the problem. To compute an estimate of the gradient, an asymptotically biased estimator is proposed. The proposed algorithm is shown to achieve convergence to within an ϵ of the global optima after sampling $\mathcal{O}(\frac{M^4\sigma^2}{(1-\gamma)^8\epsilon^4})$ trajectories where γ is the discount factor and M is the number of the agents, thus achieving the same dependence on ϵ as the policy gradient algorithm for the standard reinforcement learning.

1. Introduction

The standard formulation of reinforcement learning (RL), which aims to find the optimal policy to optimize the cumulative reward, has been well studied in the recent years. Compared with the model-based algorithms, model-free algorithms do not require the estimation of the transition dynamics and can be extended to the continuous space. Value function based algorithms such as Q-learning (Watkins & Dayan, 1992; Jin, Allen-Zhu, Bubeck, & Jordan, 2018), SARSA (Rummery & Niranjan, 1994), Temporal Difference (TD) (Sutton, 1988) and policy based algorithms such as policy gradient (Sutton, McAllester, Singh, & Mansour, 2000) and natural policy gradient (Kakade, 2001) have been proposed based on the Bellman Equation, which is a result of the additive structure for the standard RL.

However, many applications require more general non-linear reward functions. As an example, risk-sensitive objectives have been considered in (Mihatsch & Neuneier, 2002). (Hazan, Kakade, Singh, & Van Soest, 2019) studies the problem of maximizing the entropy of state-action distribution. Further, many realistic applications have multiple objectives, e.g., capacity and power usage in the communication system (Aggarwal, Bell, Elgabli, Wang, & Zhong, 2017), latency and energy consumption in queueing systems (Badita, Parag, & Aggarwal, 2020), efficiency and safety in robotic systems (Nishimura & Yonetani, 2020).

In this paper, we consider a setting that jointly optimizes a general concave function of the cumulative reward from multiple objectives.

$$\max_{\pi} f(J_1^{\pi}, \cdots, J_M^{\pi}) \tag{1}$$

where J_m^{π} is the value function following policy π for m^{th} objective and f is the general concave function. The detailed formulation can be found in Section 3. With this definition, a non-linear concave function of single objective becomes a special case. Further, fair allocation of resources among multiple users require a non-linear function of the rewards to each user (which correspond to the multiple objectives) (Lan, Kao, Chiang, & Sabharwal, 2010), and is thus a special case of this formulation. In the following, we provide two examples to better motivate our formulation.

Example 1. *(Communication System) In the communication system, there is a wireless scheduler to which M users are connected. Each user can exist in two states, good or bad. The action is the user to which the scheduler allocates the resource. This system has 2^M states with M actions. At time t, each user m achieves different rates $r_{m,t}$ based on their states and resource allocation. The joint objective function is proportional fairness or sum-logarithmic utility defined as:*

$$f(J_1^{\pi}, J_2^{\pi}, \cdots, J_M^{\pi}) = \sum\nolimits_{m=1}^{M} \log\left(J_m^{\pi}\right) \tag{2}$$

where J_m^{π} is the value function of user m using policy π.

Example 2. *(Queuing System) There is a server serving M queues with Poisson arrivals with different arrival rates. The system state is M dimensional vector of the length of the M queues. The action at each time is the queue which the server serves. At time t, each queue m achieves a reward of 1 unit if a customer from this queue is served. The joint objective function is α fairness utility (with $\alpha = 2$) defined as:*

$$f(J_1^{\pi}, J_2^{\pi}, \cdots, J_M^{\pi}) = -\sum\nolimits_{m=1}^{M} \frac{1}{J_m^{\pi}} \tag{3}$$

where J_m^{π} is the value function of queue m using policy π.

Note that in both the examples, the value of the function cannot be calculated using reward at time t (or $f(r_{1,t}, \cdots, r_{M,t})$ cannot be used for these problems), as the users which are not allocated wireless resource or the queues which are not served receive 0 reward and the function value is $-\infty$.

Such a setup was first considered in (Agarwal, Aggarwal, & Lan, 2022), where a model-based algorithm was proposed for the problem with provable regret guarantees. However, guarantees for model-free algorithm have not been studied to the best of our knowledge, which we focus on.

We note that the non-linear objective function looses the additive structure, and thus the Bellman's Equation does not work anymore in this setting (Agarwal et al., 2022; Zhang, Koppel, Bedi, Szepesvari, & Wang, 2020). Thus, the value function based algorithm do not directly work in this setup. This paper considers a policy-gradient approach and aim to show the global convergence of such policies. Recently, the authors of (Zhang et al., 2020; Zhang, Ni, Szepesvari, & Wang, 2021) considered the problem for a single-objective over finite state-action space. However, such a problem is open for continuous state action spaces, and for multiple objectives, which is the focus of this paper. In this paper, we consider a fundamental policy based algorithm, the vanilla policy gradient, and show the global convergence of this policy based on an efficient estimator of the gradient proposed in this paper.

We note that in standard reinforcement learning, Policy Gradient Theorem (Sutton et al., 2000) is used to propose an unbiased gradient estimator such as REINFORCE. However, such an approach can not directly give an unbiased estimator in our setting due to the presence of non-linear function (See Lemma 10). In this paper, we provide a biased estimator for the policy gradient. This biased estimator is then used to prove the global convergence of the policy gradient algorithm.

Our contribution can be summarized as follows.

- We consider a new problem statement in reinforcement learning, which aims to jointly optimize a multi-objective problem with concave utility. Such formulation has rarely been considered before.

- Due to the existence of concave utility, it is impossible to give an unbiased estimator. Thus, we propose a general biased gradient estimator, which can be applied to both tabular and continuous state-action spaces prove that the bias of the estimator decays at order $\mathcal{O}(1/\sqrt{n})$, where n is the number of trajectories sampled (See Remark 4).

- We prove the policy gradient algorithm with the proposed estimator converges to the global optimal with error ϵ using $\mathcal{O}(\frac{M^4\sigma^2}{(1-\gamma)^8\epsilon^4})$ samples, where M is the number of objectives, σ^2 is the variance defined in Assumption 5 and γ is the discount factor. As compared to the number of samples for standard RL with policy gradient algorithm (Liu, Zhang, Basar, & Yin, 2020), our result has the same dependence on ϵ.

- We also study our algorithm empirically. We observe that the proposed method performs better than a naive implementation of RL algorithms where reward at each time step is the value of the concave function of the individual rewards.

Further, even for the case when there is a non-linear function of a single objective, the approach and results are novel, and have not been considered in the prior works for continuous state-action spaces.

2. Related Work

Table 1 summarizes the key related works. The problem has been studied in the tabular model-based setup (Agarwal et al., 2022; Cheung, 2019). For the model-free approach, this is the first paper on guarantees on concave scalarized multi-objective infinite horizon reinforcement learning with large state-action space. As compared to the linear scalarization, biased estimator complicates the analysis, and the approach of finite state-action spaces do

	Works	Sample Complexity	Objective-Function	Multi-Objective	State Action Space
Model-Based	(Agarwal et al., 2022)	$\tilde{O}(M^2/\epsilon^2)$	Concave Scalarization	Yes	Finite
	(Cheung, 2019)	$\tilde{O}(1/\epsilon^2)$	Special Concave Scalarization [1]	Yes	Finite
Model-Free	(Zhang et al., 2020)	N/A [2]	Concave Utility[3]	No	Finite
	(Zhang et al., 2021)	$\tilde{O}(1/\epsilon^2)$	Concave Utility[3]	No	Finite
	This Work	$\tilde{O}(M^4/\epsilon^4)$	Concave Scalarization	Yes	Infinite
	(Liu et al., 2020)	$\tilde{O}(1/\epsilon^4)$	Reinforce	No	Infinite

Table 1: Overview of key related works for the problem in this paper. M is the number of objectives and ϵ is the gap between optimal objective and the objective function following the policy in the proposed algorithm.

not directly extend to our problem. Detailed comparison to the approaches is also provided in the following.

Policy Gradient with Cumulative Return: As the core result for policy based algorithms, Policy Gradient Theorem (Sutton et al., 2000) provides a method to obtain the gradient ascent direction for standard reinforcement learning with the policy parameterization. However, in general, the objective in the reinforcement learning is non-convex with respective to the parameters (Agarwal, Kakade, Lee, & Mahajan, 2020). Thus, the research on policy gradient algorithm focuses on the first order stationary point guarantees for a long time (Papini, Binaghi, Canonaco, Pirotta, & Restelli, 2018; Xu, Gao, & Gu, 2020a, 2020b). Recently, there is a line of interest on the global convergence result for reinforcement learning. (Zhang, Koppel, Zhu, & Basar, 2020) utilizes the idea of escaping saddle points in policy gradient and shows the convergence to the second order stationary points. (Agarwal et al., 2020) provides provable global convergence result for direct parameterization and softmax parameterization in the tabular case. For the restrictive parameterization, they propose a variant of NPG, Q-NPG and analyze the global convergence result with the function approximation error for both NPG and Q-NPG. (Mei, Xiao, Szepesvari, & Schuurmans, 2020) improves the convergence rate for policy gradient with softmax param-

[1] (Cheung, 2019) defines a specialized concave scalarization function, where $f(\boldsymbol{J}) = \frac{1}{M}\cdot\Big[\sum_{m=1}^{M} L_m J_m - \frac{L_0}{2}\min_{u\in U}\Big\{\sum_{m=1}^{M}(J_m - u_m)^2\Big\}\Big]$, where $L_0, \cdots, L_M$ are parameters and $U \in [0,1]^M$ is a convex compact set. The proposed algorithm and the achieved sample complexity is limited to above function and whether it can be extended to the general concave scalarization function is unknown.

[2] (Zhang et al., 2020) proposed the Varational Policy Gradient Algorithm to solve the problem. (Zhang et al., 2020)[Theorem 4.5] stated the algorithm requires $O(\epsilon^{-1})$ iterations to achieve ϵ-optimal policy. However, in each iteration, it needs to solve a min-max problem, which is costly even for estimating a single policy gradient.

[3] (Zhang et al., 2020, 2021) considered the concave utility function, where the objective is to maximize $g(\boldsymbol{\lambda})$, and $\boldsymbol{\lambda}$ is a cumulative discounted state-action occupancy measure. Setting $h_m(\boldsymbol{\lambda}) = \langle \boldsymbol{r}_m, \boldsymbol{\lambda}\rangle$ and defining $g(\boldsymbol{\lambda}) = f(h_1(\boldsymbol{\lambda}), \cdots, h_M(\boldsymbol{\lambda})) = f(\boldsymbol{J})$, their problem reduces to our formulation. Despite the formulation in (Zhang et al., 2020, 2021) is more general, the definition of the occupancy measure limits the state and action space to be finite.

eterization from $\mathcal{O}(1/\sqrt{t})$ to $\mathcal{O}(1/t)$ and shows a significantly faster linear convergence rate $\mathcal{O}(\exp(-t))$ for the entropy regularized policy gradient. With actor-critic method (Konda & Tsitsiklis, 2000), (Wang, Cai, Yang, & Wang, 2020) establishes the global optimal result for neural policy gradient method. (Bhandari & Russo, 2019) identifies the structure properties which shows that there are no sub-optimal stationary points for reinforcement learning. (Liu et al., 2020) proposes a general framework of the analysis for policy gradient type of algorithms and gives the sample complexity for PG, NPG and the variance reduced version of them. However, all of the above research have been done on the standard reinforcement learning, where the objective function is the direct summation of the reward. This paper focuses on a joint optimization of multi-objective problem, where multiple objectives are combined with a concave function.

Policy Gradient with General Objective Function: Even though standard reinforcement learning has been widely studied, there are few results on the policy gradient algorithm with a general objective function. Some special examples are variance-penalty (Huang & Kallenberg, 1994) and maximizing entropy (Hazan et al., 2019). Very recently, (Zhang et al., 2020, 2021) study the global convergence result of the policy gradient with general utilities. They consider the setting that the objective is a concave function of the state-action occupancy measure, which is similar to our setting. By the method of convex conjugate, (Zhang et al., 2020) proposed a variational policy gradient theorem to obtain the gradient ascent direction and gives the global convergences result of PG with general utilities. Despite enjoying a rate of $\mathcal{O}(1/t)$ in terms of iterations, their algorithm requires an additional saddle point problem to fulfill the gradient update and thus introduce extra computation complexity. (Zhang et al., 2021) further proposes the SIVR-PG algorithm and improves the convergence rate in the same setting. However, the SIVR-PG algorithm requires the estimation of state-action occupancy measure, which means that the algorithm can only be applied to the tabular setting. We note that our method does not have such limitation and thus can be applied even if the state and action space is large or continuous. Finally, note that (Zhang et al., 2020, 2021) improve the previous convergence rate for policy gradient by exploring the hidden convexity of the proposed problem. However, in order to utilize such convexity, they require the assumption that the inverse mapping of visitation measure $\lambda : \Theta \to \lambda(\Theta)$ exists and the Lipschitz property of such inverse mapping is assumed. It has been shown that such assumption holds for direct parameterization. However, such assumptions for continuous state-action space or other types of parameterization may not be valid.

3. Formulation

We consider an infinite horizon discounted Markov Decision Process (MDP) $\mathcal{M}$ defined by the tuple $(\mathcal{S}, \mathcal{A}, \mathbb{P}, r_1, r_2, \cdots, r_M, \gamma, \rho)$, where $\mathcal{S}$ and $\mathcal{A}$ denote the state and action space, respectively. $\mathbb{P} : \mathcal{S} \times \mathcal{A} \to \Delta^{\mathcal{S}}$ (where $\Delta^{\mathcal{S}}$ is a probability simplex over $\mathcal{S}$) denotes the transition probability distribution from a state-action pair to another state. M denotes the number of objectives and $r_m : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ denotes the reward for the m^{th} objective. $\gamma \in (0, 1)$ is the discounted factor and $\rho : \mathcal{S} \to \Delta^{\mathcal{S}}$ is the distribution for initial state. In this paper, we make following assumption.

Assumption 1. *The absolute value of the reward functions $r_m, m \in [M]$ is bounded by some constant. Without loss of generality, we assume $r_m \in [0,1], \forall m \in [M]$.*

Define a joint stationary policy $\pi : \mathcal{S} \to \Delta^{\mathcal{A}}$ that maps a state $s \in \mathcal{S}$ to a probability distribution of actions with a probability assigned to each action $a \in \mathcal{A}$. At the beginning of the MDP, an initial state $s_0 \sim \rho$ is given and the agent makes a decision $a_0 \sim \pi(\cdot|s_0)$. The agent receives M reward $r_m(s_0, a_0)$ and then transits to a new state $s_1 \sim \mathbb{P}(\cdot|s_0, a_0)$. We define the value function J_m^π for the m^{th} objective following policy π as a discounted sum of reward over infinite horizon.

$$J_m^\pi = \mathbf{E}_{\rho,\pi,\mathbb{P}}\Big[\sum_{t=0}^{\infty} \gamma^t r_m(s_t, a_t)\Big] \tag{4}$$

where $s_0 \sim \rho$, $a_t \sim \pi(\cdot|s_t)$ and $s_{t+1} \sim \mathbb{P}(\cdot|s_t, a_t)$. Similarly, we define the state value function $V_m^\pi(s)$ and state-action value function $Q_m^\pi(s,a)$

$$\begin{aligned} V_m^\pi(s) &= \mathbf{E}_{\pi,\mathbb{P}}\Big[\sum_{t=0}^{\infty} \gamma^t r_m(s_t, a_t)\Big| s_0 = s\Big] \\ Q_m^\pi(s,a) &= \mathbf{E}_{\pi,\mathbb{P}}\Big[\sum_{t=0}^{\infty} \gamma^t r_m(s_t, a_t)\Big| s_0 = s, a_0 = a\Big] \end{aligned} \tag{5}$$

The agent aims to maximize the joint objective function $f : \mathbb{R}^M \to \mathbf{R}$, which is a function of the long-term discounted reward of each objective. Formally, the problem is written as

$$\max_\pi f(J_1^\pi, J_2^\pi, \cdots, J_M^\pi) \tag{6}$$

We consider a policy-gradient based algorithm on this problem and parameterize the policy π as π_θ for some parameter $\theta \in \Theta$ such as softmax parameterization or a deep neural network. Commonly, the log-policy function $\log \pi_\theta(a|s)$ is called log-likelihood function and we make the following assumption.

Assumption 2. *The log-likelihood function is G-Lipschitz and B-smooth. Formally,*

$$\begin{aligned} &\|\nabla_\theta \log \pi_\theta(a|s)\| \le G \quad \forall \theta \in \Theta, \forall (s,a) \in \mathcal{S} \times \mathcal{A} \\ &\|\nabla_\theta \log \pi_{\theta_1}(a|s) - \nabla_\theta \log \pi_{\theta_2}(a|s)\| \le B\|\theta_1 - \theta_2\| \quad \forall \theta_1, \theta_2 \in \Theta, \forall (s,a) \in \mathcal{S} \times \mathcal{A} \end{aligned} \tag{7}$$

We consider all norms in this paper, unless explicitly mentioned, as L2-norm.

Remark 1. *The Lipschitz and smoothness properties for the log-likelihood are quite common in the field of policy gradient algorithm (Agarwal et al., 2020; Zhang et al., 2021; Liu et al., 2020). Such properties can also be verified for simple parameterization such as Gaussian policy.*

Define the value function vector $\boldsymbol{J}^{\pi_\theta} = (J_1^{\pi_\theta}, \cdots, J_M^{\pi_\theta})$. The original problem, Eq. (6), can be rewritten as

$$\max_{\theta \in \Theta} f(\boldsymbol{J}^{\pi_\theta}) \tag{8}$$

We make the following assumptions on the objective function f:

Assumption 3. *The objective function f is jointly concave. Hence for any arbitrary distribution $\mathcal{D}$, the following holds.*

$$f(\mathbf{E}_{\boldsymbol{x}\sim\mathcal{D}}[\boldsymbol{x}]) \geq \mathbf{E}_{\boldsymbol{x}\sim\mathcal{D}}[f(\boldsymbol{x})]) \quad \forall \boldsymbol{x} \in \mathbb{R}^M \tag{9}$$

Remark 2. *(Non-Concave Optimization) It is worth noticing that the above problem is a non-concave optimization problem despite the above joint-concave assumption on the objective function. This is because the parameterized value function $\boldsymbol{J}_m^{\pi_\theta}$ is non-concave with respect to θ (See Lemma 3.1 in (Agarwal et al., 2020)). Thus, the standard theory from convex optimization can't be directly applied to this problem.*

Assumption 4. *All partial derivatives of function f are assumed to be locally L_f-Lipschitz functions. Formally,*

$$\begin{gathered}|\frac{\partial f}{\partial x_i}(\boldsymbol{y}_1) - \frac{\partial f}{\partial x_i}(\boldsymbol{y}_2)| \leq L_f\|\boldsymbol{y}_1 - \boldsymbol{y}_2\| \\ \forall \boldsymbol{y}_1, \boldsymbol{y}_2 \in [0, \frac{1}{1-\gamma}]^M, \forall i \in [M]\end{gathered} \tag{10}$$

Remark 3. *By Assumption 1, $J_m^{\pi_\theta}$ is bounded in $[0, \frac{1}{1-\gamma}]$. Thus, it is enough to assume the locally Lipschitiz property for the partial derivatives of the objective. Such an assumption has also been adopted widely for the general objective function (Zhang et al., 2020, 2021).*

Finally, based on the Assumption. 4, we derive the following result for the objective function.

Lemma 1. *All partial derivative functions of f are locally bounded by a constant. Formally,*

$$\left|\frac{\partial f}{\partial x_i}(\boldsymbol{y})\right| \leq C \quad \forall \boldsymbol{y} \in [0, \frac{1}{1-\gamma}]^M, \forall i \in [M] \tag{11}$$

Proof. By Assumption 4, the partial derivative function is locally Lipschitz and thus is continuous on the set $[0, \frac{1}{1-\gamma}]^M$, which is compact. Since a continuous function with a compact set is bounded, the result follows. □

Further discussions on the assumptions are provided in Appendix J.

4. Policy Gradient Method for Joint Optimization

Policy gradient algorithm aims to update the parameter with the iteration

$$\theta^{k+1} = \theta^k + \eta\nabla_\theta f(\boldsymbol{J}^{\pi_{\theta^k}}) \tag{12}$$

where η is the step size. However, it is impossible to compute the true gradient because the transition dynamics is unknown in practice. Thus, an estimator for the true gradient is necessary. From the Chain Rule, the gradient for the objective function is (the detailed computation is in appendix B)

$$\nabla_\theta f(\boldsymbol{J}^{\pi_\theta}) = \mathbf{E}_{\tau\sim p(\tau|\theta)}\Big[\Big(\sum_{t=0}^{\infty}\nabla_\theta \log \pi_\theta(a_t|s_t)\Big)\Big(\sum_{m=1}^{M}\frac{\partial f}{\partial J_m^{\pi}}(\sum_{t=0}^{\infty}\gamma^t r_m(s_t, a_t))\Big)\Big] \tag{13}$$

In this section, we firstly propose a biased estimator and bound the bias. The policy-gradient algorithm is also formally described based on the estimator. Finally, we analyze some properties of the objective function, which will be used in the proof of the main result.

4.1 Proposed Estimator

The REINFORCE estimator of Eq. (13) for $\nabla_\theta f(\boldsymbol{J}^{\pi_\theta})$ can be considered as a sampled version of it, and it can be directly derived as

$$g(\tau_i, \tau_{j=1:N_2}|\theta) = \sum_{t=0}^{\infty} \nabla_\theta \log \pi_\theta(a_t^i|s_t^i)\Big(\sum_{m=1}^{M}(\frac{\partial f}{\partial J_m^\pi}\Big|_{J_m^\pi = \hat{J}_m^\pi}) \cdot (\sum_{h=0}^{\infty}\gamma^h r_m(s_h^i, a_h^i))\Big) \tag{14}$$

where

$$\hat{J}_m^\pi = \frac{1}{N_2}\sum_{j=1}^{N_2}\sum_{t=0}^{\infty}\gamma^t r_m(s_t^j, a_t^j) \tag{15}$$

and N_2 is the number of trajectories of τ_j that we need to sample to estimate $\frac{\partial f}{\partial J_m^\pi}$. Notice that the trajectories $\tau_i = (s_0^i, a_0^i, s_1^i, a_1^i, \cdots)$ and $\tau_j = (s_0^j, a_0^j, s_1^j, a_1^j, \cdots)$ are sampled independently from the distribution $p(\tau|\theta)$. However, notice that in general the proposed estimator is not unbiased due to the concavity of the function f (See Lemma 10 in Appendix D for detail). Moreover, the estimator in Eq. (14) is unachievable because it requires a sum over infinite range of t. Thus, we define a truncated version of Eq. (14) as

$$g(\tau_i^H, \tau_{j=1:N_2}^H|\theta) = \sum_{t=0}^{H-1} \nabla_\theta \log \pi_\theta(a_t^i|s_t^i)\Big(\sum_{m=1}^{M}(\frac{\partial f}{\partial J_m^\pi}\Big|_{J_m^\pi = \hat{J}_{m,H}^\pi}) \cdot (\sum_{h=0}^{H-1}\gamma^h r_m(s_h^i, a_h^i))\Big) \tag{16}$$

where

$$\hat{J}_{m,H}^\pi = \frac{1}{N_2}\sum_{j=1}^{N_2}\sum_{t=0}^{H-1}\gamma^t r_m(s_t^j, a_t^j) \tag{17}$$

Notice that removing the past reward from the return doesn't change the expectation value (Peters & Schaal, 2008). Thus, we can rewrite Eq. (16) as a PGT estimator.

$$g(\tau_i^H, \tau_{j=1:N_2}^H|\theta) = \sum_{t=0}^{H-1} \nabla_\theta \log \pi_\theta(a_t^i|s_t^i)\Big(\sum_{m=1}^{M}(\frac{\partial f}{\partial J_m^\pi}\Big|_{J_m^\pi = \hat{J}_{m,H}^\pi}) \cdot (\sum_{h=t}^{H-1}\gamma^h r_m(s_h^i, a_h^i))\Big) \tag{18}$$

We provide a lemma of equivalence for completeness and the proof is in Appendix C.

Lemma 2. *The expectation of PGT* (18) *and REINFORCE* (16) *are the same.*

In the remaining part of this paper, we denote $g(\tau_i^H, \tau_{j=1:N_2}^H|\theta)$ as $g(\tau_i^H, \tau_j^H|\theta)$ for simplicity. With this truncated estimator, the proposed algorithm is in Algorithm 1. In each iteration of policy gradient ascent, N_2 trajectories are sampled in line 3 and used to estimate the value function for each agent. Line 4 samples another N_1 trajectories independent of N_2 and uses Eq. (16) to calculate the gradient estimator. Line 5 and 6 perform one-step gradient descent using the gradient estimator.

Algorithm 1 Policy Gradient for Joint Optimization of Multi-Objective RL

```
Initialize $\theta^0$ and step size $\eta = \frac{1}{4L_J}$
for episode $k = 0, ..., K-1$ do
    Sample $N_2$ trajectories $\tau_j$ under policy $\theta^k$ of length $H$ and compute $\hat{J}^{\pi}_{m,H}$ by Eq. (17).
    Sample $N_1$ trajectories $\tau_i$ under policy $\theta^k$ of length $H$ and for each trajectory compute the gradient estimator $g(\tau_i^H, \tau_j^H|\theta^k)$ by Eq. (16)
    Compute the gradient update direction $\omega^k = \frac{1}{N_1}\sum_{i=1}^{N_1} g(\tau_i^H, \tau_j^H|\theta^k)$
    Update the parameter $\theta^{k+1} = \theta^k + \eta\omega^k$
end for
```

4.2 Bounding the Bias of the Truncated Estimator

To bound the bias of the proposed truncated estimator, we define three auxiliary functions.

$$\tilde{g}(\tau_i, \tau_j|\theta) = \sum_{t=0}^{\infty} \nabla_\theta \log \pi_\theta(a_t^i|s_t^i)\Bigg(\sum_{m=1}^{M}\big(\frac{\partial f}{\partial J_m^\pi}\big)\cdot\big(\sum_{h=t}^{\infty}\gamma^h r_m(s_h^i, a_h^i)\big)\Bigg) \tag{19}$$

$$\tilde{g}(\tau_i^H, \tau_j|\theta) = \sum_{t=0}^{H-1} \nabla_\theta \log \pi_\theta(a_t^i|s_t^i)\Bigg(\sum_{m=1}^{M}\big(\frac{\partial f}{\partial J_m^\pi}\big)\cdot\big(\sum_{h=t}^{H-1}\gamma^h r_m(s_h^i, a_h^i)\big)\Bigg) \tag{20}$$

$$\tilde{g}(\tau_i^H, \tau_j^H|\theta) = \sum_{t=0}^{H-1} \nabla_\theta \log \pi_\theta(a_t^i|s_t^i)\Bigg(\sum_{m=1}^{M}\big(\frac{\partial f}{\partial J_m^\pi}\bigg|_{J_m^\pi = J_{m,H}^\pi}\big)\cdot\big(\sum_{h=t}^{H-1}\gamma^h r_m(s_h^i, a_h^i)\big)\Bigg) \tag{21}$$

where $J^\pi_{m,H} = \mathbf{E}\Bigg[\sum_{t=0}^{H-1}\gamma^t r_m(s_t, a_t)\Bigg]$.

It should be noticed that Eq. (20) and (21) are different because the value function used in the partial derivatives are truncated in (21) but not in (20). Moreover, Eq. (21) and the proposed estimator in Eq. (16) are also different because (16) uses the empirical value for trajectories τ_j^H while Eq. (21) uses the expected value. We note that $\tilde{g}(\tau_i, \tau_j|\theta)$ is an unbiased estimator for $\nabla_\theta f(\boldsymbol{J}^{\pi_\theta})$. Thus, the bias of the truncated estimator Eq. (16) can be decomposed as

$$\begin{aligned}\mathbf{E}[g(\tau_i^H, \tau_j^H|\theta)] - \nabla_\theta f(\boldsymbol{J}^{\pi_\theta}) &= \mathbf{E}\underbrace{[g(\tau_i^H, \tau_j^H|\theta) - \tilde{g}(\tau_i^H, \tau_j^H|\theta)]}_{(I)} \\ + \mathbf{E}\underbrace{[\tilde{g}(\tau_i^H, \tau_j^H|\theta) - \tilde{g}(\tau_i^H, \tau_j|\theta)]}_{(II)} &+ \mathbf{E}\underbrace{[\tilde{g}(\tau_i^H, \tau_j|\theta) - \tilde{g}(\tau_i, \tau_j|\theta)]}_{(III)}\end{aligned} \tag{22}$$

which means the bias includes three parts: (I) denotes the bias coming from the finite samples of trajectories τ_j. (II) and (III) denote the bias due to the truncation of trajectories τ_j and τ_i, respectively. In the following, we give three lemmas to bound each of them. The detailed proofs are provided in Appendix E.

Lemma 3. *For any $\epsilon' > 0$ and $p \in (0,1)$, with probability at least $1-p$, if the number of samples for τ_j satisfies,*

$$N_2 \geq \frac{M(1-\gamma^H)^2}{2(1-\gamma)^2\epsilon'^2}\log(\frac{2MH}{p}) \tag{23}$$

then for each trajectory τ_i, the first part of bias for the proposed truncated estimator, Eq. (16)*, is bounded by*

$$\|g(\tau_i^H, \tau_j^H|\theta) - \tilde{g}(\tau_i^H, \tau_j^H|\theta)\| \leq MGL_f\frac{1-\gamma^H - H\gamma^H(1-\gamma)}{(1-\gamma)^2}\epsilon' \tag{24}$$

Lemma 4. *For each trajectory τ_i, the second part of bias for the proposed truncated estimator, Eq.* (16)*, is bounded by*

$$\|\tilde{g}(\tau_i^H, \tau_j^H|\theta) - \tilde{g}(\tau_i^H, \tau_j|\theta)\| \leq M^{3/2}GL_f\frac{1-\gamma^H - H\gamma^H(1-\gamma)}{(1-\gamma)^3}\gamma^H \tag{25}$$

Lemma 5. *For each trajectory τ_i, the third part of bias for the proposed truncated estimator, Eq.* (16)*, is bounded by*

$$\|\tilde{g}(\tau_i^H, \tau_j|\theta) - \tilde{g}(\tau_i, \tau_j|\theta)\| \leq MGC\frac{\gamma^H(1+H(1-\gamma))}{(1-\gamma)^2} \tag{26}$$

Remark 4. *Combining the Lemmas 3, 4, and 5, it is found that if the length of sampled trajectories is long enough, the bias of the proposed estimator decays as $\mathcal{O}(\frac{1}{\sqrt{N_2}})$.*

Further, note that the proposed estimator is asymptotically unbiased with respect to N_2 and H, as the bias reduces with increasing N_2 and H.

4.3 Properties of the Objective Function

Similar to the truncated estimator, we define a truncated version for the objective function as follows

$$f(\boldsymbol{J}_H^{\pi_\theta}) = f(\mathbf{E}[\sum_{t=0}^{H-1}\gamma^t r_1(s_t, a_t)], \cdots, \mathbf{E}[\sum_{t=0}^{H-1}\gamma^t r_M(s_t, a_t)])$$

In this subsection, we will give some properties of $f(\boldsymbol{J}^{\pi_\theta})$ and $f(\boldsymbol{J}_H^{\pi_\theta})$. The detailed proofs are provided in Appendix F. The following lemma shows the smoothness property for $f(\boldsymbol{J}^{\pi_\theta})$ and $f(\boldsymbol{J}_H^{\pi_\theta})$.

Lemma 6. *Both the objective function $f(\boldsymbol{J}^{\pi_\theta})$ and the truncated version $f(\boldsymbol{J}_H^{\pi_\theta})$ are L_J-smooth w.r.t. θ, where*

$$L_J = \frac{MCB}{(1-\gamma)^2}$$

It is reasonable to expect that the truncated objective function and the original one can be arbitrary close when the length of horizon is long enough, and the next lemma bounds the gap between original and truncated objective function.

Lemma 7. *The difference between the gradient of objective function and that of truncated version is bounded by*

$$\|\nabla_\theta f(\boldsymbol{J}^{\pi_\theta}) - \nabla_\theta f(\boldsymbol{J}_H^{\pi_\theta})\| \leq \frac{MG\gamma^H}{(1-\gamma)^2}\left[\sqrt{M}L_f\frac{1-\gamma^H - H\gamma^H(1-\gamma)}{1-\gamma} + C[1+H(1-\gamma)]\right] \tag{27}$$

In order to introduce the following result, it is helpful to define the state visitation measure

$$d_\rho^\pi := (1-\gamma)\mathbf{E}_{s_0\sim\rho}\left[\sum_{t=0}^{\infty}\gamma^t\mathbf{Pr}^\pi(s_t = s|s_0)\right] \tag{28}$$

where $\mathbf{Pr}^\pi(s_t = s|s_0)$ denotes the probability that $s_t = s$ with policy π starting from s_0. In the theoretical analysis of policy gradient for standard reinforcement learning, one key result is the performance difference lemma. In the multi-objective setting, a similar performance lemma is derived as follows.

Lemma 8. *The difference in the performance for any policies π_θ and $\pi_{\theta'}$is bounded as follows*

$$(1-\gamma)[f(\boldsymbol{J}^{\pi_\theta}) - f(\boldsymbol{J}^{\pi_{\theta'}})] \leq \sum_{m=1}^{M}\frac{\partial f(\boldsymbol{J}^{\pi_{\theta'}})}{\partial J_m^{\pi_{\theta'}}}\mathbf{E}_{s\sim d_\rho^{\pi_\theta}}\mathbf{E}_{a\sim\pi_\theta(\cdot|s)}\left[A_m^{\pi_{\theta'}}(s,a)\right] \tag{29}$$

where $A_m^\pi(s,a) = V_m^\pi(s) - Q_m^\pi(s,a)$ is the advantage function.

5. Main Result

Before stating the convergence result for the policy gradient algorithm, we describe the following assumptions which will be needed for the main result.

Assumption 5. *The auxiliary estimator $\tilde{g}(\tau_i^H, \tau_j^H|\theta)$ defined in Eq. (21) has bounded variance. Formally,*

$$Var(\tilde{g}(\tau_i^H, \tau_j^H|\theta)) := \mathbf{E}[\|\tilde{g}(\tau_i^H, \tau_j^H|\theta) - \mathbf{E}[\tilde{g}(\tau_i^H, \tau_j^H|\theta)]\|^2] \leq \sigma^2 \tag{30}$$

for any θ and $\tau_i^H, \tau_j^H \sim p^H(\cdot|\theta)$, where $p^H(\cdot|\theta)$ is a truncated version of $p(\cdot|\theta)$ defined in Eq. (40).

Remark 5. *In the standard reinforcement learning problem, it is common to assume that variance of the estimator is bounded (Liu et al., 2020), (Xu et al., 2020a) and (Xu et al., 2020b). Such assumption has been verified for Gaussian policy (Zhao, Hachiya, Niu, & Sugiyama, 2011) and (Pirotta, Restelli, & Bascetta, 2013). By Lemma 1, it can be verified similarly in the multi-objective setting.*

Assumption 6. *For all $\theta \in \mathbb{R}^d$, the Fisher information matrix induced by policy π_θ and initial state distribution ρ satisfies*

$$\begin{aligned} F_\rho(\theta) &= \mathbf{E}_{s\sim d_\rho^{\pi_\theta}}\mathbf{E}_{a\sim\pi_\theta}[\nabla_\theta\log\pi_\theta(a|s)\nabla_\theta\log\pi_\theta(a|s)^T] \\ &\succeq \mu_F\cdot\mathbf{I}_d \end{aligned} \tag{31}$$

for some constant $\mu_F > 0$

Remark 6. *The positive definiteness assumption is standard in the field of policy gradient based algorithms (Kakade, 2001; Peters & Schaal, 2008; Liu et al., 2020; Zhang et al., 2020). A common example which satisfies such assumption is Gaussian policy with mean parameterized linearly (See Appendix B.2 in (Liu et al., 2020)).*

Assumption 7. *Define the transferred function approximation error as below*

$$L_{d_\rho^{\pi^*},\pi^*}(\omega_*^\theta,\theta) = \mathbf{E}_{s\sim d_\rho^{\pi^*}}\mathbf{E}_{a\sim\pi^*(\cdot|s)}\left[\left(\nabla_\theta \log \pi_\theta(a|s)\cdot(1-\gamma)\omega_*^\theta - \sum_{m=1}^{M}\frac{\partial f(\boldsymbol{J}^{\pi_\theta})}{\partial J_m^{\pi_\theta}}A_m^{\pi_\theta}(s,a)\right)^2\right] \tag{32}$$

We assume that this error satisfies $L_{d_\rho^{\pi^*},\pi^*}(\omega_*^\theta,\theta) \le \epsilon_{bias}$ for any $\theta \in \Theta$, where π^* is the optimal policy and ω_*^θ is given as

$$\omega_*^\theta = \arg\min_\omega \mathbf{E}_{s\sim d_\rho^{\pi_\theta}}\mathbf{E}_{a\sim\pi_\theta(\cdot|s)}\left[[\nabla_\theta \log \pi_\theta(a|s)\cdot(1-\gamma)\omega - \sum_{m=1}^{M}\frac{\partial f(\boldsymbol{J}^{\pi_\theta})}{\partial J_m^{\pi_\theta}}A_m^{\pi_\theta}(s,a)]^2\right] \tag{33}$$

It can be shown that ω_*^θ is the exact Natural Policy Gradient (NPG) update direction.

Remark 7. *By Eq.* (32) *and* (33)*, the transferred function approximation error expresses an approximation error with distribution shifted to $(d_\rho^{\pi^*},\pi^*)$. With the softmax parameterization or linear MDP structure (Jin, Yang, Wang, & Jordan, 2020), it has been shown that $\epsilon_{bias} = 0$ (Agarwal et al., 2020). When parameterized by the restricted policy class, $\epsilon_{bias} > 0$ due to π_θ not containing all policies. However, for a rich neural network parameterization, the ϵ_{bias} is small (Wang et al., 2020). Similar assumption has been adopted in (Liu et al., 2020) and (Agarwal et al., 2020).*

Remark 8. *Due to there existing 7 assumption in the paper, we give a further discussion on all assumptions in Appendix J*

5.1 Global Convergence in Multi-Objective Setting

Inspired by the global convergence analysis framework for policy gradient in (Liu et al., 2020), we present a general framework for convergence analysis of non-linear multi-objective policy gradient in the following.

Lemma 9. *(Generalization of Proposition 4.5 in (Liu et al., 2020)) Suppose a general gradient ascent algorithm updates the parameter in the way*

$$\theta^{k+1} = \theta^k + \eta\omega^k \tag{34}$$

When Assumptions 2 and 7 hold, we have

$$\begin{aligned} f(\boldsymbol{J}^{\pi^*}) - \frac{1}{K}\sum_{k=0}^{K-1} f(\boldsymbol{J}^{\pi_{\theta^k}}) \le \frac{\sqrt{\epsilon_{bias}}}{1-\gamma} + \frac{G}{K}\sum_{k=0}^{K-1}\|(\omega^k-\omega_*^k)\|_2 \\ + \frac{B\eta}{2K}\sum_{k=0}^{K-1}\|\omega^k\|^2 + \frac{1}{\eta K}\mathbf{E}_{s\sim d_\rho^{\pi^*}}[KL(\pi^*(\cdot|s)\|\pi_{\theta^0}(\cdot|s))] \end{aligned} \tag{35}$$

where $\omega_*^k := \omega_*^{\theta^k}$ and is defined in Eq. (33)

Proof. We generalize the Proposition 4.5 in (Liu et al., 2020) by using the Lemma 8 and propose the framework of global convergence analysis in the joint optimization for multi-objective setting. Thus, the framework proposed in the Proposition 4.5 in (Liu et al., 2020) can be considered as a special case. The detailed proof is provided in Appendix G. □

Now, we provide the main result of global convergence for the policy gradient algorithm with multi-objective setting (with detailed proof in Appendix H).

Theorem 1. *For any $\epsilon > 0$, in the Policy Gradient Algorithm 1 with the proposed estimator in Eq.* (16)*, if step-size $\eta = \frac{1}{4L_J}$, the number of iteration $K = \mathcal{O}(\frac{M}{(1-\gamma)^2\epsilon})$, the length of each trajectory $H = \mathcal{O}\bigg(\log\frac{M}{(1-\gamma)\epsilon}\bigg)$, the number of samples $N_1 = \mathcal{O}(\frac{\sigma^2}{\epsilon})$ and $N_2 = \mathcal{O}(\frac{M^3}{(1-\gamma)^6\epsilon})$ achieves the following bound*

$$f(\boldsymbol{J}^{\pi^*}) - \frac{1}{K}\sum_{k=0}^{K-1} f(\boldsymbol{J}^{\pi_{\theta^k}}) \le \frac{\sqrt{\epsilon_{bias}}}{1-\gamma} + \epsilon \tag{36}$$

In other words, policy gradient algorithm needs $\mathcal{O}\big(\frac{M^4\sigma^2}{(1-\gamma)^8\epsilon^4}\big)$ trajectories.

6. Evaluations

6.1 Simulation Environment

To validate the understanding of our analysis, we perform evaluations using a queuing environment with multiple objectives and a concave utility combining the objectives. The environment is a server serving M queues with Poisson arrivals with different arrival rates. The system state is M dimensional vector of the length of the M queues. The action at each time is the queue which the server serves. At time t, each queue m achieves a reward of 1 unit if a customer from this queue is served. The joint objective function is α-fairness defined as:

$$f(\sum_t r_{1,t}, \cdots, \sum_t r_{K,t}) = -\sum\nolimits_{m=1}^{M} \frac{H}{\sum_{t=1}^{H}\gamma^{t-1} r_{m,t}}, \tag{37}$$

where H is the length of the episode set to 500 steps.

For our queuing environment, we consider a server serving customers coming from M queues. Each queue follows Poisson arrivals with different arrival rates given in Table 2. The server has access to the length of the queues. On observing the length of the queue, the server selects a queue to process. If the a customer from a queue is served, the queue gets a reward of 1 unit.

M	λ_1	λ_2	λ_3	λ_4	λ_5	λ_6	λ_7	λ_8
2	0.16	0.64	–	–	–	–	–	–
4	0.08	0.16	0.24	0.32	–	–	–	–
8	0.0125	0.0375	0.0625	0.0875	0.1125	0.1375	0.1625	0.1875

Table 2: Arrival rates of the multiple queues for Queuing system environment

6.2 Simulation Setup

We use softmax parameterization for implementing our policies. Further, we use PyTorch version 1.0.1 to implement the policies and perform gradient ascent. The experiments are run on a machine with Intel i9 processor with 36 logical cores running at 3.00 GHz each. The machines are equipped with Nvidia GeForce RTX 2080 GPU. Each of the 10 independent runs for both environment took about 500 seconds to finish. For the gradient ascent of objective, we used PyTorch's Adam (Kingma & Ba, 2015) optimization with learning rate of 0.005. Finally, we use the value of γ to be 0.9999.

6.3 Simulation Results

We study the impact of the number of trajectories used for gradient estimation. We keep the number of trajectories $N_1 = N_2 = N$ and vary N from $4, 16, 64$, and 256. We also vary the number of objectives M as $2, 4$, and 8. We observe the convergence rates for a softmax policy parameterization. We also compare our algorithm with a policy gradient algorithm which trains the actor using reward function $r_{train}(t)$ at each time t defined as,

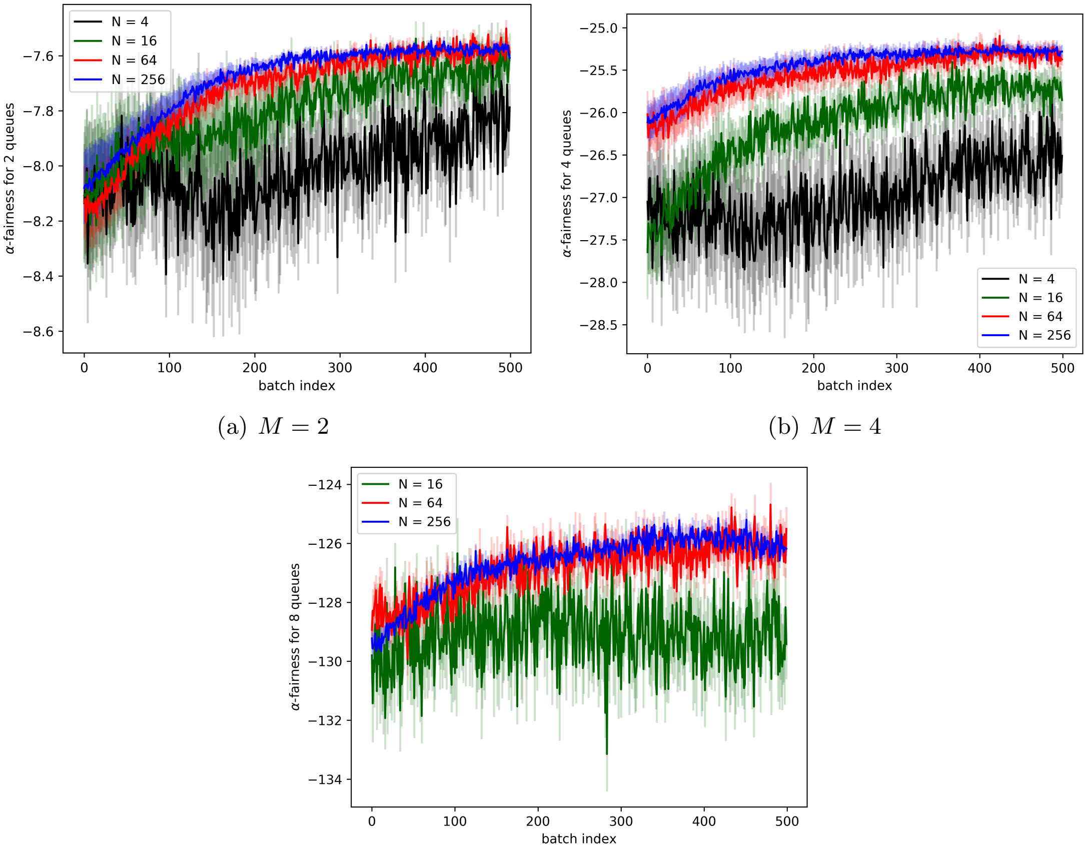
$$r_{train}(t) = -\sum_{m=1}^{M} \frac{t}{\sum_{\tau=1}^{t} \gamma^{\tau-1} r_{k,\tau}}. \tag{38}$$

To implement the policy gradient, we use the REINFORCE algorithm (Williams, 1992).

We plot the behavior of the policy gradient for joint optimization for different values of N in Figure 1. We run 10 independent iterations and plot the mean in solid lines and the shaded region is ± standard deviation. In Figure 1, for all values of M, we find that increasing N, the number of trajectories used for sampling gradient of the function, leads to faster convergence of the joint reward objective. We note that the objective value are in different scales, and hence we cannot directly compare the objective values for different M.

For $M = 2$ (Figure 1(a)), we note that the performance of $N = 256, 64$, and $N = 16$ are almost similar; but as compared to $N = 4$, the performance is significantly better. When M is increased to 4 (in Figure 1(b)), we observe that $N = 256$ and $N = 64$ are similar and $N = 256$ performs only marginally better as compared to $N = 64$. However, now $N = 16$ does not perform as well as $N = 256$ and $N = 64$ but the algorithm is still able to converge to the optimal policy with $N = 16$. Finally, for $M = 8$, we note that $N = 256$ again performs better than $N = 64$ and $N = 16$ with a lesser variance in the performance. However, for $M = 8$, the algorithm with $N = 16$ is not able to converge to the optimal policy. We infer that for joint optimization of multiple objectives, it is necessary to increase the number of trajectories as the number of objectives increase.

We now compare the performance of our proposed algorithm with the REINFORCE algorithm. We present the results in Figure 2, where we compare the REINFORCE algorithm with varying values for N to compute gradient estimate. We note that the REINFORCE algorithm does not learn a policy which maximizes the objective because the reward at each time step does not provide correct gradient estimate. The performance of the REINFORCE gradient estimate improves with increase in the number of trajectories N, but for same number of trajectories, the proposed Algorithm 1 performs significantly better. Based on the comparisons, we infer that using the proposed gradient estimator enables learning optimal policy which maximizes the function f.

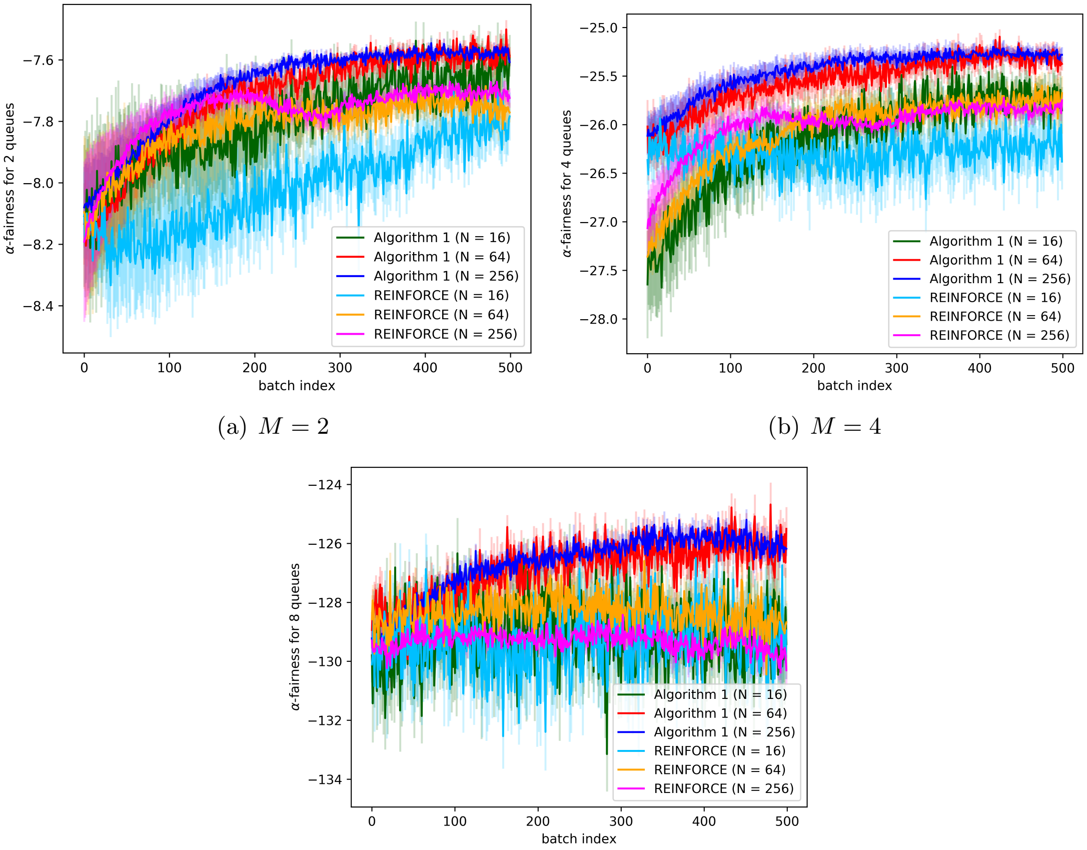


(a) $M = 2$

(b) $M = 4$

(c) $M = 8$

Figure 1: Convergence plots for the joint objective policy gradient algorithms for increasing number of queues M. As the number of trajectories N used for sampling gradient of the function increase, the convergence becomes steeper. Further, as the number of queues M increase, number of trajectories N is also required to increase to achieve similar performance of levels.



(a) $M = 2$

(b) $M = 4$

(c) $M = 8$

Figure 2: Comparison plots for the joint objective policy gradient algorithms and the REINFORCE algorithm for increasing number of queues M and varied number of samples N to compute gradient estimates. The REINFORCE algorithm is able to learn policies which improves the function value, but it does not achieves the policies as good as policies which our algorithm learns.

7. Conclusion

In this paper, we formulate a problem which optimizes a general concave function of multiple objectives. We propose a policy-gradient based approach for the problem, where an estimator for the gradient is used. We analyze the bias of the policy gradient estimator and show the global convergence result with a vanilla policy gradient algorithm. However, there are several limitations in the paper. Firstly, the local Lipschitz assumption for partial derivative of f is a bit strong. Secondly, the convergence rates w.r.t the number of objective is $\mathcal{O}(M^4)$, while a model-based algorithm (Agarwal et al., 2022) can achieve $\mathcal{O}(M^2)$. Further, extension of the proposed approach to evaluate the convergence rate guarantees of the Natural Policy Gradient and the variance reduced algorithms is an important future direction to reduce the sample complexity. Finally, the analysis of concave function of objectives with constraints is open for parametrized model-free setup, while has been studied for model-based setup (Agarwal, Bai, & Aggarwal, 2021), for model-free tabular setup (Bai, Bedi, Agarwal, Koppel, & Aggarwal, 2021), and for parametrized model-free setup with linear function of objectives (Bai, Bedi, Agarwal, Koppel, & Aggarwal, 2022).

Appendix A. Symbol Summary

Symbol	Definition	Reference
$\mathcal{S}, \mathcal{A}$	State and Action space	Section 3
$\mathbb{P}$	transition dynamics	Section 3
r_m	reward function for m^{th} objective	Section 3
M	Number of objectives	Section 3
γ	discounted factor	Section 3
ρ	distribution for initial state	Section 3
J_m^π	Expected value function for m^{th} objective	Eq. (4)
$V_m^\pi(s)$	State value function for m^{th} objective	Eq. (5)
$Q_m^\pi(s,a)$	State-action value function for m^{th} objective	Eq. (5)
$A_m^\pi(s,a)$	Advantage function for m^{th} objective	Lemma 8
G	Lipschitz constant for log-likelihood function	Assumption 2
B	smooth constant for log-likelihood function	Assumption 2
L_f	Lipschitz constant for partial derivatives of function f	Assumption 4
C	Bound on partial derivatives of function f	Lemma 1
H	truncation on proposed estimator	Section 4
L_J	smooth constant for objective function	Lemma 6
σ^2	bound on variance of auxiliary estimator	Assumption 5
μ_F	positive definitive constant for Fisher information matrix	Assumption 6
ϵ_{bias}	bias of transferred function approximation error	Assumption 7
η	learning rate of policy gradient	Algorithm 1
N_1, N_2	Number of samples for estimator	Algorithm 1
K	number of iterations of policy gradient	Algorithm 1

Table 3: Overview of symbols defined in the paper

Appendix B. Computation of the Gradient of Objective

$$\nabla_\theta f(\boldsymbol{J}^{\pi_\theta}) = \sum_{m=1}^{M} \frac{\partial f(\boldsymbol{J}^{\pi_\theta})}{\partial J_m^{\pi_\theta}} \nabla_\theta J_m^{\pi_\theta} \tag{39}$$

Define $\tau = (s_0, a_1, s_1, a_1, s_2, a_2 \cdots)$ as a trajectory, whose distribution induced by policy π_θ is $p(\tau|\theta)$ that can be expressed as

$$p(\tau|\theta) = \rho(s_0) \prod_{t=0}^{\infty} \pi_\theta(a_t|s_t) P(s_{t+1}|s_t, a_t) \tag{40}$$

Define $R_m(\tau) = \sum_{t=0}^{\infty} \gamma^t r_m(s_t, a_t)$ as the cumulative reward for m^{th} objective following the trajectory τ. Then, the expected return $J_m^\pi(\theta)$ can also be expressed as

$$J_m^{\pi_\theta} = \mathbf{E}_{\tau \sim p(\tau|\theta)}[R_m(\tau)]$$

and the gradient can be calculated as

$$\begin{aligned}\nabla_\theta J_m^\pi(\theta) &= \int_\tau R_m(\tau) p(\tau|\theta) d\tau = \int_\tau R_m(\tau) \frac{\nabla_\theta p(\tau|\theta)}{p(\tau|\theta)} p(\tau|\theta) d\tau \\ &= \mathbf{E}_{\tau\sim p(\tau|\theta)}\big[\nabla_\theta \log p(\tau|\theta) R_m(\tau)\big]\end{aligned}$$

Notice that $\nabla_\theta \log p(\tau|\theta)$ is independent of the transition dynamics and thus

$$\nabla_\theta f(\boldsymbol{J}^{\pi_\theta}) = \mathbf{E}_{\tau\sim p(\tau|\theta)}\Bigg[\Big(\sum_{t=0}^{\infty} \nabla_\theta \log \pi_\theta(a_t|s_t)\Big)\Big(\sum_{m=1}^{M} \frac{\partial f}{\partial J_m^\pi}(\sum_{t=0}^{\infty} \gamma^t r_m(s_t, a_t))\Big)\Bigg] \tag{41}$$

Appendix C. Proof of Equivalence between PGT and REINFORCE Estimator in Lemma 2

Proof. Notice that the difference between PGT and REINFORCE can be expressed as below

$$\sum_{t=0}^{H} \nabla_\theta \log(\pi_\theta(a_t^i|s_t^i))\Big(\sum_{m=1}^{M}(\frac{\partial f}{\partial J_m^\pi}\Big|_{J_m^\pi=\hat{J}_{m,H}^\pi})(\sum_{h=0}^{t-1}\gamma^h r_m(s_h^i, a_h^i))\Big) \tag{42}$$

Thus, it is sufficient to show the expectation of above equation is $\mathbf{0}$. Divide the trajectory τ_i into two parts $\tau_{i,1} = (s_0^i, a_0^i, \cdots, s_{t-1}^i, a_{t-1}^i)$ and $\tau_{i,2} = (s_t^i, a_t^i, \cdots)$. Then,

$$\begin{aligned}&\mathbf{E}_{\tau_i,\tau_j\sim p(\tau|\theta)}\Bigg[\sum_{t=0}^{H} \nabla_\theta \log(\pi_\theta(a_t^i|s_t^i))\Big(\sum_{m=1}^{M}(\frac{\partial f}{\partial J_m^\pi}\Big|_{J_m^\pi=\hat{J}_{m,H}^\pi})(\sum_{h=0}^{t-1}\gamma^h r_m(s_h^i, a_h^i))\Big)\Bigg] \\ &= \sum_{t=0}^{H}\mathbf{E}_{\tau_{i,1}}\Bigg\{\mathbf{E}_{\tau_{i,2}}\Bigg[\nabla_\theta \log(\pi_\theta(a_t^i|s_t^i))\Big(\sum_{m=1}^{M}\mathbf{E}_{\tau_j}(\frac{\partial f}{\partial J_m^\pi}\Big|_{J_m^\pi=\hat{J}_{m,H}^\pi})(\sum_{h=0}^{t-1}\gamma^h r_m(s_h^i, a_h^i))\Big)\Bigg]\Bigg|\tau_{i,1}\Bigg\} \\ &= \sum_{t=0}^{H}\mathbf{E}_{\tau_{i,1}}\Bigg\{\mathbf{E}_{\tau_{i,2}}\Bigg[\nabla_\theta \log(\pi_\theta(a_t^i|s_t^i))\Bigg]\Big(\sum_{m=1}^{M}\mathbf{E}_{\tau_j}(\frac{\partial f}{\partial J_m^\pi}\Big|_{J_m^\pi=\hat{J}_{m,H}^\pi})(\sum_{h=0}^{t-1}\gamma^h r_m(s_h^i, a_h^i))\Big)\Bigg|\tau_{i,1}\Bigg\} \\ &= \sum_{t=0}^{H}\mathbf{E}_{\tau_{i,1}}\Bigg\{\mathbf{0}\cdot\Big(\sum_{m=1}^{M}\mathbf{E}_{\tau_j}(\frac{\partial f}{\partial J_m^\pi}\Big|_{J_m^\pi=\hat{J}_{m,H}^\pi})(\sum_{h=0}^{t-1}\gamma^h r_m(s_h^i, a_h^i))\Big)\Bigg|\tau_{i,1}\Bigg\} = \mathbf{0}\end{aligned} \tag{43}$$

where the first step holds because τ_i, τ_j are dependent and the law of total expectation. The second equality holds because the summation of reward is a constant conditioned on $\tau_{i,1}$. The last step is true because

$$\mathbf{E}_{\tau_{i,2}}\Big[\nabla_\theta \log(\pi_\theta(a_t^i|s_t^i))\Big] = \mathbf{E}_{s_t^i}\Big[\int_{\mathcal{A}} \nabla_\theta \log(\pi_\theta(a_t^i|s_t^i))\pi_\theta(a_t^i|s_t^i)da\Big] \tag{44}$$

$$= \mathbf{E}_{s_t^i}\Big[\int_{\mathcal{A}} \nabla_\theta \pi_\theta(a_t^i|s_t^i)da\Big] = \mathbf{E}_{s_t^i}[\nabla_\theta \mathbf{1}] = \mathbf{0} \tag{45}$$

□

Appendix D. Proof for the Bias of Estimator in Eq. (14)

Lemma 10. *In general, the proposed estimator, Eq.* (14)*, is biased w.r.t.* $\nabla_\theta f(\boldsymbol{J}^{\pi_\theta})$*. The only exception is when the partial derivatives* $\frac{\partial f}{\partial J_m^\pi}$ *are linear w.r.t. each variable* J_n^π *for all* $m, n \in [M]$*.*

Proof. By the law of total expectation

$$\begin{aligned}
\mathbf{E}_{\tau_i,\tau_{j=1:N_2}}[g(\tau_i|\theta)] &= \mathbf{E}_{\tau_i,\tau_{j=1:N_2}}\Bigg[\sum_{t=0}^{\infty}\nabla_\theta \log \pi_\theta(a_t^i|s_t^i)\Bigg(\sum_{m=1}^{M}\Big(\frac{\partial f}{\partial J_m^\pi}\Big|_{J_m^\pi=\hat{J}_m^\pi}\Big)\Big(\sum_{h=t}^{\infty}\gamma^h r_m(s_h^i,a_h^i)\Big)\Bigg)\Bigg] \\
&= \mathbf{E}_{\tau_i}\Bigg\{\mathbf{E}_{\tau_{j=1:N_2}}\Bigg[\sum_{t=0}^{\infty}\nabla_\theta \log \pi_\theta(a_t^i|s_t^i)\Bigg(\sum_{m=1}^{M}\Big(\frac{\partial f}{\partial J_m^\pi}\Big|_{J_m^\pi=\hat{J}_m^\pi}\Big)\Big(\sum_{h=t}^{\infty}\gamma^h r_m(s_h^i,a_h^i)\Big)\Bigg)\Bigg]\Bigg|\tau_i\Bigg\} \\
&= \mathbf{E}_{\tau_i}\Bigg\{\sum_{t=0}^{\infty}\nabla_\theta \log \pi_\theta(a_t^i|s_t^i)\Bigg(\sum_{m=1}^{M}\mathbf{E}_{\tau_{j=1:N_2}}\Bigg[\frac{\partial f}{\partial J_m^\pi}\Big|_{J_m^\pi=\hat{J}_m^\pi}\Bigg]\Big(\sum_{h=t}^{\infty}\gamma^h r_m(s_h^i,a_h^i)\Big)\Bigg)\Bigg|\tau_i\Bigg\} \\
&\overset{(*)}{\neq} \mathbf{E}_{\tau_i}\Bigg\{\sum_{t=0}^{\infty}\nabla_\theta \log \pi_\theta(a_t^i|s_t^i)\Bigg(\sum_{m=1}^{M}\frac{\partial f}{\partial J_m^\pi}\Big(\sum_{h=t}^{\infty}\gamma^h r_m(s_h^i,a_h^i)\Big)\Bigg)\Bigg\} \\
&= \nabla_\theta f(J_1^\pi(s), J_2^\pi(s), \cdots, J_M^\pi(s))
\end{aligned} \tag{46}$$

Notice that the key step (*) holds because

$$\begin{aligned}
\mathbf{E}_{\tau_{j=1:N_2}}\Bigg[\frac{\partial f}{\partial J_m^\pi}\Big|_{J_m^\pi=\hat{J}_m^\pi}\Bigg] &= \mathbf{E}_{\tau_{j=1:N_2}}\Bigg[\frac{\partial f}{\partial J_m^\pi}\Big(\frac{1}{N_2}\sum_{j=1}^{N_2}\sum_{t=0}^{\infty}\gamma^t r_1(s_t^j,a_t^j), \cdots, \frac{1}{N_2}\sum_{j=1}^{N_2}\sum_{t=0}^{\infty}\gamma^t r_M(s_t^j,a_t^j)\Big)\Bigg] \\
&\neq \frac{\partial f}{\partial J_m^\pi}\Bigg(\mathbf{E}_{\tau_{j=1:N_2}}\Bigg[\frac{1}{N_2}\sum_{j=1}^{N_2}\sum_{t=0}^{\infty}\gamma^t r_1(s_t^j,a_t^j)\Bigg], \cdots, \mathbf{E}_{\tau_{j=1:N_2}}\Bigg[\frac{1}{N_2}\sum_{j=1}^{N_2}\sum_{t=0}^{\infty}\gamma^t r_M(s_t^j,a_t^j)\Bigg]\Bigg) \\
&= \frac{\partial f}{\partial J_m^\pi}(J_1^\pi, \cdots, J_M^\pi)
\end{aligned} \tag{47}$$

Eq. 39 cannot hold with an equality except when the partial derivatives are linear. However, this doesn't hold for any general concave function. □

Appendix E. Bounding the Bias for the Proposed Estimator

E.1 Proof for Lemma 3

Proof. By the triangle inequality, Assumptions 1 and 2, we have

$$\begin{aligned}\|g(\tau_i^H,\tau_j^H|\theta)-\tilde{g}(\tau_i^H,\tau_j^H|\theta)\| &= \Big\|\sum_{t=0}^{H-1}\nabla_\theta\log\pi_\theta(a_t^i|s_t^i)\Big(\sum_{m=1}^{M}\big(\frac{\partial f}{\partial J_m^\pi}\Big|_{J_m^\pi=\hat{J}_{m,H}^\pi}-\frac{\partial f}{\partial J_m^\pi}\Big|_{J_m^\pi=J_{m,H}^\pi}\big)\\ &\quad\big(\sum_{h=t}^{H-1}\gamma^h r_m(s_h^i,a_h^i)\big)\Big)\Big\|\\ &\le \frac{G}{1-\gamma}\Big|\sum_{t=0}^{H-1}(\gamma^t-\gamma^H)\Big(\sum_{m=1}^{M}\big(\frac{\partial f}{\partial J_m^\pi}\Big|_{J_m^\pi=\hat{J}_{m,H}^\pi}-\frac{\partial f}{\partial J_m^\pi}\Big|_{J_m^\pi=J_{m,H}^\pi}\big)\Big)\Big|\\ &\le G\frac{1-\gamma^H-H\gamma^H(1-\gamma)}{(1-\gamma)^2}\sum_{m=1}^{M}\Big|\frac{\partial f(\hat{J}_{m,H}^\pi)}{\partial J_m^\pi}-\frac{\partial f(J_{m,H}^\pi)}{\partial J_m^\pi}\Big|\\ &\le GML_f\frac{1-\gamma^H-H\gamma^H(1-\gamma)}{(1-\gamma)^2}\|\hat{\boldsymbol{J}}_H^\pi-\boldsymbol{J}_H^\pi\|\end{aligned} \tag{48}$$

where the last step follows from Assumption 4. Moreover, an entry in the difference $\hat{\boldsymbol{J}}_H^\pi-\boldsymbol{J}_H^\pi$ can be bounded as

$$\begin{aligned}|\hat{J}_{m,H}^\pi-J_{m,H}^\pi| &= \Big|\frac{1}{N_2}\sum_{j=1}^{N_2}\sum_{t=0}^{H-1}\gamma^t r_m(s_t,a_t)-\mathbf{E}\big[\sum_{t=0}^{H-1}\gamma^t r_m(s_t,a_t)\big]\Big|\\ &\le \sum_{t=0}^{H-1}\gamma^t\Big|\frac{1}{N_2}\sum_{j=1}^{N_2}r_m(s_t,a_t)-\mathbf{E}[r_m(s_t,a_t)]\Big|\end{aligned} \tag{49}$$

By Hoeffding Lemma, if we have $N_2\ge\frac{M(1-\gamma^H)^2}{2(1-\gamma)^2\epsilon'^2}\log(\frac{2MH}{p})$, then

$$P\Bigg(\Big|\frac{1}{N_2}\sum_{j=1}^{N_2}r_m(s_t,a_t)-\mathbf{E}[r_m(s_t,a_t)]\Big|\ge\frac{(1-\gamma)\epsilon'}{(1-\gamma^H)\sqrt{M}}\Bigg)\le 2\exp(-\frac{2N_2^2\frac{(1-\gamma)^2\epsilon'^2}{(1-\gamma^H)^2M}}{\sum_{j=1}^{N_2}(1-0)^2})\le\frac{p}{MH} \tag{50}$$

Finally, by using an union bound, with probability at least $1-p$, we have

$$\Big|\frac{1}{N_2}\sum_{j=1}^{N_2}r_m(s_t,a_t)-\mathbf{E}[r_m(s_t,a_t)]\Big|\le\frac{(1-\gamma)\epsilon'}{(1-\gamma^H)\sqrt{M}}\quad\forall m\in[M],\forall t\in[0,H-1] \tag{51}$$

Substituting Eq. (51) back into (49), we have $|\hat{J}_{m,H}^\pi-J_{m,H}^\pi|\le\frac{\epsilon'}{\sqrt{M}}$ and thus $\|\boldsymbol{J}_H^\pi-\hat{\boldsymbol{J}}_H^\pi\|_2\le\epsilon'$, which gives the result in the statement of the Lemma. □

E.2 Proof for Lemma 4

Proof. Similar to Eq. (48), we have

$$\|\tilde{g}(\tau_i^H, \tau_j^H|\theta) - \tilde{g}(\tau_i^H, \tau_j|\theta)\| \leq GML_f \frac{1-\gamma^H - H\gamma^H(1-\gamma)}{(1-\gamma)^2} \|\boldsymbol{J}_H^\pi - \boldsymbol{J}^\pi\| \tag{52}$$

By triangle inequality, the element of $\boldsymbol{J}_H^\pi - \boldsymbol{J}^\pi$ can be bounded by

$$\begin{aligned} |J_{m,H}^\pi - J_m^\pi| &\leq \left|\mathbf{E}\big[\sum_{t=0}^{\infty}\gamma^t r_m(s_t,a_t)\big] - \mathbf{E}\big[\sum_{t=0}^{H-1}\gamma^t r_m(s_t,a_t)\big]\right| \\ &\leq \sum_{t=H}^{\infty}\gamma^t \left|\mathbf{E}[r_m(s_t,a_t)]\right| \leq \frac{\gamma^H}{1-\gamma} \end{aligned} \tag{53}$$

where the last step holds by Assumption 1. Substituting Eq (53) back into (52) gives the result in the statement of the Lemma. □

E.3 Proof for Lemma 5

Proof. By the triangle inequality,

$$\begin{aligned} &\|\tilde{g}(\tau_i^H, \tau_j|\theta) - g(\tau_i, \tau_j|\theta)\| \\ &= \|\sum_{t=0}^{\infty}\nabla_\theta \log\pi_\theta(a_t^i|s_t^i)\Big(\sum_{m=1}^{M}\frac{\partial f}{\partial J_m^\pi}\big(\sum_{h=t}^{\infty}\gamma^h r_m(s_h^i,a_h^i)\big)\Big) \\ &- \sum_{t=0}^{H-1}\nabla_\theta \log\pi_\theta(a_t^i|s_t^i)\Big(\sum_{m=1}^{M}\frac{\partial f}{\partial J_m^\pi}\big(\sum_{h=t}^{\infty}\gamma^h r_m(s_h^i,a_h^i)\big)\Big) \\ &+ \sum_{t=0}^{H-1}\nabla_\theta \log\pi_\theta(a_t^i|s_t^i)\Big(\sum_{m=1}^{M}\frac{\partial f}{\partial J_m^\pi}\big(\sum_{h=t}^{\infty}\gamma^h r_m(s_h^i,a_h^i)\big)\Big) \\ &- \sum_{t=0}^{H-1}\nabla_\theta \log\pi_\theta(a_t^i|s_t^i)\Big(\sum_{m=1}^{M}\frac{\partial f}{\partial J_m^\pi}\big(\sum_{h=t}^{H-1}\gamma^h r_m(s_h^i,a_h^i)\big)\Big)\| \\ &\leq \|\sum_{t=H}^{\infty}\nabla_\theta \log\pi_\theta(a_t^i|s_t^i)\Big(\sum_{m=1}^{M}\frac{\partial f}{\partial J_m^\pi}\big(\sum_{h=t}^{\infty}\gamma^h r_m(s_h^i,a_h^i)\big)\Big)\| \\ &+ \|\sum_{t=0}^{H-1}\nabla_\theta \log\pi_\theta(a_t^i|s_t^i)\Big(\sum_{m=1}^{M}\frac{\partial f}{\partial J_m^\pi}\big(\sum_{h=H}^{\infty}\gamma^h r_m(s_h^i,a_h^i)\big)\Big)\| \\ &\leq \frac{MGC\gamma^H}{(1-\gamma)^2} + \frac{MGCH\gamma^H}{(1-\gamma)} = MGC\frac{\gamma^H(1+H(1-\gamma))}{(1-\gamma)^2} \end{aligned}$$

where the last inequality holds by Lemma 1 and Assumption 2. □

Appendix F. Proof for Properties of the Objective Function

F.1 Proof for Lemma 6

Proof. In order to show the smoothness, it is sufficient to bound $\|\nabla_\theta^2 f(\boldsymbol{J}^{\pi_\theta})\|$ and $\|\nabla_\theta^2 f(\boldsymbol{J}_H^{\pi_\theta})\|$. By Eq. (13), we have

$$\begin{aligned}\|\nabla_\theta^2 f(\boldsymbol{J}^{\pi_\theta})\| &= \|\mathbf{E}_{\tau\sim p(\tau|\theta)}\Big[\sum_{t=0}^{\infty}\nabla_\theta^2\log\pi_\theta(a_t|s_t)\Big(\sum_{m=1}^{M}\frac{\partial f}{\partial J_m^\pi}\big(\sum_{h=t}^{\infty}\gamma^h r_m(s_h,a_h)\big)\Big)\Big]\| \\ &\le \frac{MC}{(1-\gamma)}\sum_{t=0}^{\infty}\gamma^t\|\nabla_\theta^2\log\pi_\theta(a_t|s_t)\| \le \frac{MCB}{(1-\gamma)^2}\end{aligned} \tag{54}$$

where the last inequality holds by the Assumption 2. The smoothness property for the truncated version $f(\boldsymbol{J}_H^{\pi_\theta})$ can be proved similarly. □

F.2 Proof for Lemma 7

Proof. Notice that $\tilde{g}(\tau_i,\tau_j|\theta)$ is an unbiased estimator for $\nabla_\theta f(\boldsymbol{J}^{\pi_\theta})$. Moreover, $\tilde{g}(\tau_i^H,\tau_j^H)$ is an unbiased estimator for $\nabla_\theta f(\boldsymbol{J}_H^{\pi_\theta})$. Thus,

$$\begin{aligned}\|\nabla_\theta f(\boldsymbol{J}^{\pi_\theta})-\nabla_\theta f(\boldsymbol{J}_H^{\pi_\theta})\| &\overset{(a)}{=} \|\mathbf{E}[\tilde{g}(\tau_i,\tau_j|\theta)-\tilde{g}(\tau_i^H,\tau_j^H|\theta)]\| \le \mathbf{E}\|\tilde{g}(\tau_i,\tau_j|\theta)-\tilde{g}(\tau_i^H,\tau_j^H|\theta)\| \\ &\overset{(b)}{\le} \mathbf{E}\|\tilde{g}(\tau_i,\tau_j|\theta)-\tilde{g}(\tau_i^H,\tau_j|\theta)\|+\mathbf{E}\|\tilde{g}(\tau_i^H,\tau_j|\theta)-\tilde{g}(\tau_i^H,\tau_j^H|\theta)\| \\ &\overset{(c)}{\le} M^{3/2}GL_f\frac{1-\gamma^H-H\gamma^H(1-\gamma)}{(1-\gamma)^3}\gamma^H+MGC\frac{\gamma^H[1+H(1-\gamma)]}{(1-\gamma)^2}\end{aligned} \tag{55}$$

where the step (a) and (b) hold by the triangle inequality. Step (c) holds by the Lemma 4 and 5 □

F.3 Proof for Lemma 8

Proof. By the concavity of the function f, we have

$$\begin{aligned}f(\boldsymbol{J}^{\pi_\theta}) &\le f(\boldsymbol{J}^{\pi_{\theta'}})+\nabla_{\boldsymbol{J}^{\pi_{\theta'}}}f(\boldsymbol{J}^{\pi_{\theta'}})^T(\boldsymbol{J}^{\pi_\theta}-\boldsymbol{J}^{\pi_{\theta'}}) \\ &= f(\boldsymbol{J}^{\pi_{\theta'}})+\sum_{m=1}^{M}\frac{\partial f(\boldsymbol{J}^{\pi_{\theta'}})}{\partial J_m^{\pi_{\theta'}}}(J_m^{\pi_\theta}-J_m^{\pi_{\theta'}}) \\ &= f(\boldsymbol{J}^{\pi_{\theta'}})+\sum_{m=1}^{M}\frac{\partial f(\boldsymbol{J}^{\pi_{\theta'}})}{\partial J_m^{\pi_{\theta'}}}\frac{1}{1-\gamma}\mathbf{E}_{s\sim d_\rho^{\pi_\theta}}\mathbf{E}_{a\sim\pi_\theta(\cdot|s)}\big[A_m^{\pi_{\theta'}}(s,a)\big]\end{aligned} \tag{56}$$

where the last step comes from the policy gradient theorem (Sutton et al., 2000) for the standard reinforcement learning. Finally, we get the desired result by rearranging terms. □

Appendix G. Proof of Lemma 9

Proof. Starting with the definition of KL divergence,

$$\begin{aligned}
&\mathbf{E}_{s\sim d_\rho^{\pi^*}}[KL(\pi^*(\cdot|s)\|\pi_{\theta^k}(\cdot|s)) - KL(\pi^*(\cdot|s)\|\pi_{\theta^{k+1}}(\cdot|s))]\\
=&\mathbf{E}_{s\sim d_\rho^{\pi^*}}\mathbf{E}_{a\sim\pi^*(\cdot|s)}\left[\log\frac{\pi_{\theta^{k+1}(a|s)}}{\pi_{\theta^k}(a|s)}\right]\\
\overset{(a)}{\geq}&\mathbf{E}_{s\sim d_\rho^{\pi^*}}\mathbf{E}_{a\sim\pi^*(\cdot|s)}[\nabla_\theta\log\pi_{\theta^k}(a|s)\cdot(\theta^{k+1}-\theta^k)] - \frac{B}{2}\|\theta^{k+1}-\theta^k\|^2\\
=&\eta\mathbf{E}_{s\sim d_\rho^{\pi^*}}\mathbf{E}_{a\sim\pi^*(\cdot|s)}[\nabla_\theta\log\pi_{\theta^k}(a|s)\cdot\omega^k] - \frac{B\eta^2}{2}\|\omega^k\|^2\\
=&\eta\mathbf{E}_{s\sim d_\rho^{\pi^*}}\mathbf{E}_{a\sim\pi^*(\cdot|s)}[\nabla_\theta\log\pi_{\theta^k}(a|s)\cdot\omega^k_*] + \eta\mathbf{E}_{s\sim d_\rho^{\pi^*}}\mathbf{E}_{a\sim\pi^*(\cdot|s)}[\nabla_\theta\log\pi_{\theta^k}(a|s)\cdot(\omega^k-\omega^k_*)] - \frac{B\eta^2}{2}\|\omega^k\|^2\\
=&\eta[f(\boldsymbol{J}^{\pi^*}) - f(\boldsymbol{J}^{\pi_{\theta^k}})] + \eta\mathbf{E}_{s\sim d_\rho^{\pi^*}}\mathbf{E}_{a\sim\pi^*(\cdot|s)}[\nabla_\theta\log\pi_{\theta^k}(a|s)\cdot\omega^k_*] - \eta[f(\boldsymbol{J}^{\pi^*}) - f(\boldsymbol{J}^{\pi_{\theta^k}})]\\
&+\eta\mathbf{E}_{s\sim d_\rho^{\pi^*}}\mathbf{E}_{a\sim\pi^*(\cdot|s)}[\nabla_\theta\log\pi_{\theta^k}(a|s)\cdot(\omega^k-\omega^k_*)] - \frac{B\eta^2}{2}\|\omega^k\|^2\\
\overset{(b)}{=}&\eta[f(\boldsymbol{J}^{\pi^*}) - f(\boldsymbol{J}^{\pi_{\theta^k}})] + \frac{\eta}{1-\gamma}\mathbf{E}_{s\sim d_\rho^{\pi^*}}\mathbf{E}_{a\sim\pi^*(\cdot|s)}\left[\nabla_\theta\log\pi_{\theta^k}(a|s)\cdot(1-\gamma)\omega^k_* - \sum_{m=1}^{B}\frac{\partial f(\boldsymbol{J}^{\pi_{\theta^k}})}{\partial J_m^{\pi_{\theta^k}}}A_m^{\pi_{\theta^k}}(s,a)\right]\\
&+\eta\mathbf{E}_{s\sim d_\rho^{\pi^*}}\mathbf{E}_{a\sim\pi^*(\cdot|s)}[\nabla_\theta\log\pi_{\theta^k}(a|s)\cdot(\omega^k-\omega^k_*)] - \frac{B\eta^2}{2}\|\omega^k\|^2\\
\overset{(c)}{\geq}&\eta[f(\boldsymbol{J}^{\pi^*}) - f(\boldsymbol{J}^{\pi_{\theta^k}})]\\
&-\frac{\eta}{1-\gamma}\sqrt{\mathbf{E}_{s\sim d_\rho^{\pi^*}}\mathbf{E}_{a\sim\pi^*(\cdot|s)}\left[\left(\nabla_\theta\log\pi_{\theta^k}(a|s)\cdot(1-\gamma)\omega^k_* - \sum_{m=1}^{B}\frac{\partial f(\boldsymbol{J}^{\pi_{\theta^k}})}{\partial J_m^{\pi_{\theta^k}}}A_m^{\pi_{\theta^k}}(s,a)\right)^2\right]}\\
&-\eta\mathbf{E}_{s\sim d_\rho^{\pi^*}}\mathbf{E}_{a\sim\pi^*(\cdot|s)}\|\nabla_\theta\log\pi_{\theta^k}(a|s)\|_2\|(\omega^k-\omega^k_*)\| - \frac{B\eta^2}{2}\|\omega^k\|^2\\
\overset{(d)}{\geq}&\eta[f(\boldsymbol{J}^{\pi^*}) - f(\boldsymbol{J}^{\pi_{\theta^k}})] - \frac{\eta\sqrt{\epsilon_{bias}}}{1-\gamma} - \eta G\|(\omega^k-\omega^k_*)\| - \frac{B\eta^2}{2}\|\omega^k\|^2
\end{aligned}\tag{57}$$

where the step (a) holds by Assumption 2 and step (b) holds by Lemma 8. Step (c) uses the convexity of the function $f(x)=x^2$. Finally, step (d) comes from the Assumption 7. Rearranging items, we have

$$\begin{aligned}
f(\boldsymbol{J}^{\pi^*}) - f(\boldsymbol{J}^{\pi_{\theta^k}}) \leq& \frac{\sqrt{\epsilon_{bias}}}{1-\gamma} + G\|(\omega^k-\omega^k_*)\| + \frac{B\eta}{2}\|\omega^k\|^2\\
&+\frac{1}{\eta}\mathbf{E}_{s\sim d_\rho^{\pi^*}}[KL(\pi^*(\cdot|s)\|\pi_{\theta^k}(\cdot|s)) - KL(\pi^*(\cdot|s)\|\pi_{\theta^{k+1}}(\cdot|s))]
\end{aligned}\tag{58}$$

Summing from $k=0$ to $K-1$ and dividing by K, we get the desired result. □

Appendix H. Proof for Theorem 1

In this part, we prove the Theorem 1 by bounding the three terms on the right hand side of Eq. (35). These terms are: the difference between the update direction $\frac{G}{K}\sum_{k=0}^{K-1}\|(\omega^k -$

$\omega_*^k)\|$, norm of estimated gradient $\frac{M\eta}{2K}\sum_{k=0}^{K-1}\|\omega^k\|^2$, and the term about KL divergence $\frac{1}{\eta K}\mathbf{E}_{s\sim d_\rho^{\pi^*}}[KL(\pi^*(\cdot|s)\|\pi_{\theta^0}(\cdot|s))]$

H.1 Bounding the Difference Between the Update Directions

Recall the estimated policy gradient update direction is

$$\omega^k = \frac{1}{N_1}\sum_{i=1}^{N_1} g(\tau_i^H, \tau_j^H|\theta) \tag{59}$$

and the true natural policy gradient update direction is

$$\omega_*^k = F_\rho(\theta_k)^\dagger \nabla_\theta f(\boldsymbol{J}^{\pi_\theta}) \tag{60}$$

We define an auxiliary update direction as

$$\tilde{\omega}^k = \frac{1}{N_1}\sum_{i=1}^{N_1} \tilde{g}(\tau_i^H, \tau_j^H|\theta) \tag{61}$$

Thus, we can decompose the difference as

$$\begin{aligned}
&\left(\frac{1}{K}\sum_{k=0}^{K-1}\mathbf{E}\|\omega^k-\omega_*^k\|\right)^2 \le \frac{1}{K}\sum_{k=0}^{K-1}\left(\mathbf{E}\|\omega^k-\omega_*^k\|\right)^2 \le \frac{1}{K}\sum_{k=0}^{K-1}\mathbf{E}\Big[\|\omega_k-\omega_*^k\|^2\Big]\\
&= \frac{1}{K}\sum_{k=0}^{K-1}\mathbf{E}\Big[\|(\omega^k-\tilde{\omega}^k)+(\tilde{\omega}^k-\nabla_\theta f(\boldsymbol{J}_H^{\pi_\theta}))+(\nabla_\theta f(\boldsymbol{J}_H^{\pi_\theta})-\nabla_\theta f(\boldsymbol{J}^{\pi_\theta}))+(\nabla_\theta f(\boldsymbol{J}^{\pi_\theta})-F_\rho(\theta^k)^\dagger\nabla_\theta f(\boldsymbol{J}^{\pi_\theta}))\|^2\Big]\\
&\le \frac{4}{K}\sum_{k=0}^{K-1}\mathbf{E}\Big[\|\omega^k-\tilde{\omega}^k\|^2\Big]+\frac{4}{K}\sum_{k=0}^{K-1}\mathbf{E}\Big[\|\tilde{\omega}^k-\nabla_\theta f(\boldsymbol{J}_H^{\pi_\theta})\|^2\Big]+\frac{4}{K}\sum_{k=0}^{K-1}\mathbf{E}\Big[\|\nabla_\theta f(\boldsymbol{J}^{\pi_\theta})-\nabla_\theta f(\boldsymbol{J}_H^{\pi_\theta})\|^2\Big]\\
&+\frac{4}{K}\sum_{k=0}^{K-1}\mathbf{E}\Big[\|\nabla_\theta f(\boldsymbol{J}^{\pi_\theta})-F_\rho(\theta^k)^\dagger\nabla_\theta f(\boldsymbol{J}^{\pi_\theta})\|^2\Big]
\end{aligned} \tag{62}$$

The different terms in the above are bounded as follows:

- Bounding $\mathbf{E}\Big[\|\omega^k-\tilde{\omega}^k\|^2\Big]$: By Lemma 3, with N_2 large enough, for any τ_i and θ, we have

  $$\|g(\tau_i^H|\theta)-\tilde{g}(\tau_i^H|\theta)\| \le MGL_f\frac{1-\gamma^H-H\gamma^H(1-\gamma)}{(1-\gamma)^2}\epsilon' \tag{63}$$

  Thus,

  $$\begin{aligned}\|\omega^k-\tilde{\omega}^k\| &= \|\frac{1}{N_1}\sum_{i=1}^{N_1}(g(\tau_i^H|\theta^k)-\tilde{g}(\tau_i^H|\theta^k))\| \le \frac{1}{N_1}\sum_{i=1}^{N_1}\|(g(\tau_i^H|\theta^k)-\tilde{g}(\tau_i^H|\theta^k))\|\\
  &\le MGL_f\frac{1-\gamma^H-H\gamma^H(1-\gamma)}{(1-\gamma)^2}\epsilon' \le \frac{MGL_f}{(1-\gamma)^2}\epsilon'\end{aligned} \tag{64}$$

Thus,

$$\mathbf{E}\bigg[\|\omega^k - \tilde{\omega}^k\|^2\bigg] \leq \frac{M^2G^2L_f^2}{(1-\gamma)^4}\epsilon'^2 \tag{65}$$

- Bounding $\mathbf{E}\bigg[\|\tilde{\omega}^k - \nabla_\theta f(\boldsymbol{J}_H^{\pi_\theta})\|^2\bigg]$: Notice that $\tilde{g}(\tau^H|\theta)$ is an unbiased estimator for $\nabla_\theta f(\boldsymbol{J}_H^{\pi_\theta})$ and thus by Assumption 5, we have $\mathbf{E}\bigg[\|\omega^k - \tilde{\omega}^k\|^2\bigg] \leq \frac{\sigma^2}{N_1}$

- Bounding $\mathbf{E}\bigg[\|\nabla_\theta f(\boldsymbol{J}^{\pi_\theta}) - \nabla_\theta f(\boldsymbol{J}_H^{\pi_\theta})\|^2\bigg]$: By Lemma 7, we have

$$\mathbf{E}\bigg[\|\nabla_\theta f(\boldsymbol{J}^{\pi_\theta}) - \nabla_\theta f(\boldsymbol{J}_H^{\pi_\theta})\|^2\bigg] \leq \frac{M^2G^2\gamma^{2H}}{(1-\gamma)^4}\bigg[\sqrt{M}L_f + C[1+H(1-\gamma)]\bigg]^2 \tag{66}$$

- Bounding $\mathbf{E}\bigg[\|\nabla_\theta f(\boldsymbol{J}^{\pi_\theta}) - F_\rho(\theta^k)^\dagger \nabla_\theta f(\boldsymbol{J}^{\pi_\theta})\|^2\bigg]$: By Assumption 6, we have

$$\begin{aligned}
&\mathbf{E}\bigg[\|\nabla_\theta f(\boldsymbol{J}^{\pi_\theta}) - F_\rho(\theta^k)^\dagger \nabla_\theta f(\boldsymbol{J}^{\pi_\theta})\|^2\bigg] \leq (1+\frac{1}{\mu_F})^2\mathbf{E}[\|\nabla_\theta f(\boldsymbol{J}^{\pi_k})\|^2] \\
&\leq (1+\frac{1}{\mu_F})^2\bigg(2\mathbf{E}[\|\nabla_\theta f(\boldsymbol{J}_H^{\pi_k})\|^2] + 2\mathbf{E}[\|\nabla_\theta f(\boldsymbol{J}^{\pi_k}) - \nabla_\theta f(\boldsymbol{J}_H^{\pi_k})\|^2]\bigg) \\
&\leq (1+\frac{1}{\mu_F})^2\bigg(2\mathbf{E}[\|\nabla_\theta f(\boldsymbol{J}^{\pi_k})] + \frac{2M^2G^2\gamma^{2H}}{(1-\gamma)^4}\bigg[\sqrt{M}L_f + C[1+H(1-\gamma)]\bigg]^2\bigg)
\end{aligned} \tag{67}$$

Finally, we obtain the bound

$$\begin{aligned}
&\bigg(\frac{1}{K}\sum_{k=0}^{K-1}\mathbf{E}\|\omega^k - \omega_*^k\|\bigg)^2 \leq 4\frac{M^2G^2L_f^2}{(1-\gamma)^4}\epsilon'^2 + 4\frac{\sigma^2}{N_1} + 4\frac{M^2G^2\gamma^{2H}}{(1-\gamma)^4}\bigg[\sqrt{M}L_f + C[1+H(1-\gamma)]\bigg]^2 \\
&+ 4(1+\frac{1}{\mu_F})^2\bigg(\frac{2}{K}\sum_{k=0}^{K-1}\mathbf{E}[\|\nabla_\theta f(\boldsymbol{J}^{\pi_k})] + \frac{2M^2G^2\gamma^{2H}}{(1-\gamma)^4}\bigg[\sqrt{M}L_f + C[1+H(1-\gamma)]\bigg]^2\bigg) \\
&\overset{(a)}{=} (1+2(1+\frac{1}{\mu_F})^2)4\frac{M^2G^2\gamma^{2H}}{(1-\gamma)^4}\bigg[\sqrt{M}L_f + C[1+H(1-\gamma)]\bigg]^2 + 4\frac{M^2G^2L_f^2}{(1-\gamma)^4}\epsilon'^2 + 4\frac{\sigma^2}{N_1} \\
&+ 8(1+\frac{1}{\mu_F})^2\frac{\frac{\mathbf{E}[f(\boldsymbol{J}_H(\theta^K))-f(\boldsymbol{J}_H(\theta^0))]}{K} + (\eta+2L_J\eta^2)[\frac{M^2G^2L_f^2}{(1-\gamma)^4}\epsilon'^2 + \frac{\sigma^2}{N_1}]}{\frac{\eta}{2} - L_J\eta^2} \\
&= (1+2(1+\frac{1}{\mu_F})^2)4\frac{M^2G^2\gamma^{2H}}{(1-\gamma)^4}\bigg[\sqrt{M}L_f + C[1+H(1-\gamma)]\bigg]^2 + (1+6(1+\frac{1}{\mu_F})^2)4\frac{M^2G^2L_f^2}{(1-\gamma)^4}\epsilon'^2 \\
&+ (1+6(1+\frac{1}{\mu_F})^2)4\frac{\sigma^2}{N_1} + 128(1+\frac{1}{\mu_F})^2L_J\frac{\mathbf{E}[f(\boldsymbol{J}_H(\theta^K)) - f(\boldsymbol{J}_H(\theta^0))]}{K}
\end{aligned} \tag{68}$$

where the step (a) requires the first-order stationary property Eq. (88) and it is proved in the Lemma 11 in the Appendix I. Given the fixed ϵ, choose the value for H, ϵ', N_1, K as follows,

$$\frac{1}{4}\left(\frac{\epsilon^2}{3G^2}\right) \geq (1+2(1+\frac{1}{\mu_F})^2)\frac{4M^2G^2\gamma^{2H}}{(1-\gamma)^4}\left[\sqrt{M}L_f + C[1+H(1-\gamma)]\right]^2 \tag{69}$$

$$\epsilon'^2 \leq \frac{1}{4(1+6(1+\frac{1}{\mu_F})^2)}\frac{(1-\gamma)^4}{M^2G^2L_f^2}\cdot\frac{1}{4}\left(\frac{\epsilon^2}{3G^2}\right) \tag{70}$$

$$N_1 \geq \frac{(1+6(1+\frac{1}{\mu_F})^2)4\sigma^2}{\frac{1}{4}\left(\frac{\epsilon^2}{3G^2}\right)} \tag{71}$$

$$K \geq \frac{128(1+\frac{1}{\mu_F})^2 L_J \mathbf{E}[f(\boldsymbol{J}_H(\theta^K)) - f(\boldsymbol{J}_H(\theta^0))]}{\frac{1}{4}\left(\frac{\epsilon^2}{3G^2}\right)} \tag{72}$$

then we have

$$\frac{G}{K}\sum_{k=0}^{K-1}\mathbf{E}[\|\omega^k - \omega_*^k]\| \leq \frac{\epsilon}{3} \tag{73}$$

Given the choice of H, N_1, ϵ', K, the dependence of N_1, N_2, K and H on $\sigma, \epsilon, 1-\gamma$ are as follows.

$$N_1 = \mathcal{O}(\frac{\sigma^2}{\epsilon^2}) \quad N_2 = \mathcal{O}(\frac{M^3}{(1-\gamma)^6\epsilon^2}) \quad K = \mathcal{O}(\frac{M}{(1-\gamma)^2\epsilon^2}) \quad H = \mathcal{O}(\log\frac{M}{(1-\gamma)\epsilon}) \tag{74}$$

H.2 Bounding the Norm of Estimated Gradient

$$\begin{aligned}\frac{B\eta}{2K}\sum_{k=0}^{K-1}\|\omega^k\|^2 &\leq \frac{B\eta}{2}\Big[\frac{3}{K}\sum_{k=0}^{K-1}\|\omega^k - \tilde{\omega}^k\|^2 + \frac{3}{K}\sum_{k=0}^{K-1}\|\tilde{\omega}^k - \nabla_\theta f(\boldsymbol{J}_H^{\pi_\theta})\|^2 + \frac{3}{K}\sum_{k=0}^{K-1}\|\nabla_\theta f(\boldsymbol{J}_H^{\pi_\theta}))\|^2\Big] \\ &\leq \frac{B\eta}{2}\Bigg[3\frac{M^2G^2L_f^2}{(1-\gamma)^4}\epsilon'^2 + 3\frac{\sigma^2}{N_1} + 3\frac{\frac{\mathbf{E}[f(\boldsymbol{J}_H(\theta^K))-f(\boldsymbol{J}_H(\theta^0))]}{K} + (\eta+2L_J\eta^2)[\frac{M^2G^2L_f^2}{(1-\gamma)^4}\epsilon'^2 + \frac{\sigma^2}{N_1}]}{\frac{\eta}{2} - L_J\eta^2}\Bigg] \\ &= B\eta\Bigg[6\frac{M^2G^2L_f^2}{(1-\gamma)^4}\epsilon'^2 + 6\frac{\sigma^2}{N_1} + 24L_J\frac{\mathbf{E}[f(\boldsymbol{J}_H(\theta^K)) - f(\boldsymbol{J}_H(\theta^0))]}{K}\Bigg]\end{aligned} \tag{75}$$

Given the fixed ϵ, choose the value for ϵ', N_1, K as follows,

$$\epsilon'^2 \leq \frac{(1-\gamma)^4}{M^2G^2L_f^2}\cdot\frac{1}{6B\eta}\left(\frac{\epsilon}{9}\right) \tag{76}$$

$$N_1 \geq \frac{54\sigma^2}{\epsilon} \tag{77}$$

$$K \geq \frac{216L_J\mathbf{E}[f(\boldsymbol{J}_H(\theta^K)) - f(\boldsymbol{J}_H(\theta^0))]}{\epsilon} \tag{78}$$

then we have

$$\frac{1}{K}\sum_{k=0}^{K-1}\mathbf{E}[\|\omega^k]\|^2 \leq \frac{\epsilon}{3} \tag{79}$$

Given the choice of ϵ', N_1, K, the dependence of N_1, N_2, K and H on $\sigma, \epsilon, 1-\gamma$ are as follows.

$$N_1 = \mathcal{O}(\frac{\sigma^2}{\epsilon}) \quad N_2 = \mathcal{O}(\frac{M^3}{(1-\gamma)^6\epsilon}) \quad K = \mathcal{O}(\frac{M}{(1-\gamma)^2\epsilon}) \quad H = \mathcal{O}(\log\frac{M}{(1-\gamma)\epsilon}) \tag{80}$$

H.3 Bounding the KL Divergence

It is obvious if we choose

$$K \geq \frac{3\mathbf{E}_{s\sim d_\rho^{\pi^*}}[KL(\pi^*(\cdot|s)\|\pi_{\theta^0})]}{\eta\epsilon(\cdot|s)} \tag{81}$$

then

$$\frac{1}{\eta K}\mathbf{E}_{s\sim d_\rho^{\pi^*}}[KL(\pi^*(\cdot|s)\|\pi_{\theta^0})] \leq \frac{\epsilon}{3} \tag{82}$$

In other word, the dependence of K on ϵ is

$$K = \mathcal{O}(\frac{B}{\epsilon}) \tag{83}$$

Appendix I. First Order Stationary Result for Policy Gradient

Lemma 11. *The policy gradient algorithm can achieve first-order stationary. More formally, if we choose the step size $\eta = \frac{1}{4L_J}$ and*

$$N_1 = \mathcal{O}(\frac{\sigma^2}{\epsilon}) \quad N_2 = \mathcal{O}(\frac{M^3}{(1-\gamma)^6\epsilon}) \quad K = \mathcal{O}(\frac{M}{(1-\gamma)^2\epsilon}) \tag{84}$$

then,

$$\frac{1}{K}\sum_{k=0}^{K-1}\mathbf{E}[\|\nabla_\theta f(\boldsymbol{J}_H(\theta^k))\|^2] \leq \epsilon \tag{85}$$

Proof. Recall the definition of ω^k and $\tilde{\omega}^k$ in Eq. (59) and (61), respectively. By Lemma 6, we have

$$\begin{aligned}
f(\boldsymbol{J}_H(\theta^{k+1})) &\geq f(\boldsymbol{J}_H(\theta^k)) + \left\langle \nabla_\theta f(\boldsymbol{J}_H(\theta^k)), \theta^{k+1} - \theta^k \right\rangle - \frac{L_J}{2}\|\theta^{k+1} - \theta^k\|^2 \\
&= f(\boldsymbol{J}_H(\theta^k)) + \eta \left\langle \nabla_\theta f(\boldsymbol{J}_H(\theta^k)), \omega^k \right\rangle - \frac{L_J \eta^2}{2}\|\omega^k\|^2 \\
&\overset{(a)}{=} f(\boldsymbol{J}_H(\theta^k)) + \eta \left\langle \nabla_\theta f(\boldsymbol{J}_H(\theta^k)), \omega^k - \nabla_\theta f(\boldsymbol{J}_H(\theta^k)) + \nabla_\theta f(\boldsymbol{J}_H(\theta^k)) \right\rangle \\
&\quad - \frac{L_J \eta^2}{2}\|\omega^k - \nabla_\theta f(\boldsymbol{J}_H(\theta^k)) + \nabla_\theta f(\boldsymbol{J}_H(\theta^k))\|^2 \\
&\overset{(b)}{\geq} f(\boldsymbol{J}_H(\theta^k)) + \eta\|\nabla_\theta f(\boldsymbol{J}_H(\theta^k))\|^2 - \eta| \left\langle \nabla_\theta f(\boldsymbol{J}_H(\theta^k)), \omega^k - \nabla_\theta f(\boldsymbol{J}_H(\theta^k)) \right\rangle | \\
&\quad - L_J\eta^2\Big(\|\omega^k - \nabla_\theta f(\boldsymbol{J}_H(\theta^k))\|^2 + \|\nabla_\theta f(\boldsymbol{J}_H(\theta^k))\|^2 \Big) \\
&\geq f(\boldsymbol{J}_H(\theta^k)) + \eta\|\nabla_\theta f(\boldsymbol{J}_H(\theta^k))\|^2 - \frac{\eta}{2}\|\nabla_\theta f(\boldsymbol{J}_H(\theta^k))\|^2 - \frac{\eta}{2}\|\omega^k - \nabla_\theta f(\boldsymbol{J}_H(\theta^k))\|^2 \\
&\quad - L_J\eta^2\Big(\|\omega^k - \nabla_\theta f(\boldsymbol{J}_H(\theta^k))\|^2 + \|\nabla_\theta f(\boldsymbol{J}_H(\theta^k))\|^2 \Big) \\
&= f(\boldsymbol{J}_H(\theta^k)) + (\frac{\eta}{2} - L_J\eta^2)\|\nabla_\theta f(\boldsymbol{J}_H(\theta^k))\|^2 - (\frac{\eta}{2} + L_J\eta^2)\|\omega^k - \nabla_\theta f(\boldsymbol{J}_H(\theta^k))\|^2 \\
&\overset{(c)}{\geq} f(\boldsymbol{J}_H(\theta^k)) + (\frac{\eta}{2} - L_J\eta^2)\|\nabla_\theta f(\boldsymbol{J}_H(\theta^k))\|^2 - (\eta + 2L_J\eta^2)\|\omega^k - \tilde{\omega}^k\|^2 \\
&\quad - (\eta + 2L_J\eta^2)\|\tilde{\omega}^k - \nabla_\theta f(\boldsymbol{J}_H(\theta^k))\|^2 \\
&\overset{(d)}{\geq} f(\boldsymbol{J}_H(\theta^k)) + (\frac{\eta}{2} - L_J\eta^2)\|\nabla_\theta f(\boldsymbol{J}_H(\theta^k))\|^2 - (\eta + 2L_J\eta^2)\frac{M^2G^2L_f^2}{(1-\gamma)^4}\epsilon'^2 \\
&- (\eta + 2L_J\eta^2)\|\tilde{\omega}^k - \nabla_\theta f(\boldsymbol{J}_H(\theta^k))\|^2
\end{aligned} \tag{86}$$

where the step (a) holds by $\theta^{k+1} = \theta^k + \eta\omega^k$. Step (b) and (c) holds by Cauchy-Schwarz Inequality. Step (d) holds by Lemma 3. Then, take expectation with respect to the trajectories τ_i, τ_j (Recall that θ^k, θ^{k+1} is a function of τ_i, τ_j), we have

$$\begin{aligned}
\mathbf{E}[f(\boldsymbol{J}_H(\theta^{k+1}))] &\geq \mathbf{E}[f(\boldsymbol{J}_H(\theta^k))] + (\frac{\eta}{2} - L_J\eta^2)\mathbf{E}[\|\nabla_\theta f(\boldsymbol{J}_H(\theta^k))\|^2] - (\eta + 2L_J\eta^2)\frac{M^2G^2L_f^2}{(1-\gamma)^4}\epsilon'^2 \\
&\quad - (\eta + 2L_J\eta^2)\mathbf{E}[\|\tilde{g}^k - \nabla_\theta f(\boldsymbol{J}_H(\theta^k))\|^2] \\
&\geq \mathbf{E}[f(\boldsymbol{J}_H(\theta^k))] + (\frac{\eta}{2} - L_J\eta^2)\mathbf{E}[\|\nabla_\theta f(\boldsymbol{J}_H(\theta^k))\|^2] - (\eta + 2L_J\eta^2)\frac{M^2G^2L_f^2}{(1-\gamma)^4}\epsilon'^2 \\
&\quad - (\eta + 2L_J\eta^2)\frac{\sigma^2}{N_1}
\end{aligned} \tag{87}$$

where the last step holds by Assumption 5. Notice that in Eq. (87), $\mathbf{E}[f(\boldsymbol{J}_H(\theta^{k+1}))]$ and $\mathbf{E}[f(\boldsymbol{J}_H(\theta^k))]$ give a recursive form. Thus, telescoping from $k = 0$ to $k = K - 1$, we have

$$\frac{\mathbf{E}[f(\boldsymbol{J}_H(\theta^K)) - f(\boldsymbol{J}_H(\theta^0))]}{K} \geq (\frac{\eta}{2} - L_J\eta^2)\frac{1}{K}\sum_{k=0}^{K-1}\mathbf{E}[\|\nabla_\theta f(\boldsymbol{J}_H(\theta^k))\|^2] - (\eta + 2L_J\eta^2)[\frac{M^2G^2L_f^2}{(1-\gamma)^4}\epsilon'^2 + \frac{\sigma^2}{N_1}] \tag{88}$$

and thus

$$\frac{1}{K}\sum_{k=0}^{K-1}\mathbf{E}[\|\nabla_\theta f(\boldsymbol{J}_H(\theta^k))\|^2] \leq \frac{\frac{\mathbf{E}[f(\boldsymbol{J}_H(\theta^K)) - f(\boldsymbol{J}_H(\theta^0))]}{K} + (\eta + 2L_J\eta^2)[\frac{M^2G^2L_f^2}{(1-\gamma)^4}\epsilon'^2 + \frac{\sigma^2}{N_1}]}{\frac{\eta}{2} - L_J\eta^2} \tag{89}$$

Taking $\eta = \frac{1}{4L_J}$ and letting $N_1 = \frac{18\sigma^2}{\epsilon}$, $K = \frac{48L_J\mathbf{E}[\|\nabla_\theta f(\boldsymbol{J}_H(\theta^K)) - \nabla_\theta f(\boldsymbol{J}_H(\theta^0))\|^2]}{\epsilon}$ and $\epsilon' = \frac{(1-\gamma)^2}{MGL_f}\sqrt{\frac{\epsilon}{6}}$, we have

$$\frac{1}{K}\sum_{k=0}^{K-1}\mathbf{E}[\|\nabla_\theta f(\boldsymbol{J}_H(\theta^k))\|^2] \leq \epsilon \tag{90}$$

Recalling the definition of N_2 in the statement of Lemma 3, we have

$$N_2 = \frac{6M^3G^2L_f^2(1-\gamma^H)^2}{(1-\gamma)^6\epsilon}\log(\frac{2MH}{p}) \tag{91}$$

Also, by the definition of L_J in the lemma 6

$$K = \frac{48MCB}{(1-\gamma)^2\epsilon}\mathbf{E}[\|\nabla_\theta f(\boldsymbol{J}_H(\theta^k))\|^2] \tag{92}$$

□

Appendix J. Further Discussion on all Asumptions

- Assumption 1 is related to the bound for reward and it can always be satisfied by scalarization or shifting.
- Assumptions 3 and 4 are about the function class. They require a concave function with local-Lipschitz partial derivatives. As we discussed in the limitation, many function with regularization such as $\log(x), -x^2, \sqrt{x}, \sin(x)$ will satisfy these conditions.
- The remaining 4 assumptions limit the policy parameterization
  - We would like to say assumption 5 can be implied by assumption 2. This is because $\tilde{g}(\tau_i^H, \tau_j^H|\theta)$ is bounded under assumption 2 and thus the variance is also bounded.
  - The property that the likelihood is smooth and the gradient of it is bounded can be satisfied by Gaussian policy (Appendix C in (Papini et al., 2018)) and log-linear policy class (Remark 6.7 in (Agarwal et al., 2020)).

- The positive definite property of Fisher matrix can also be satisfied by Gaussian Policy (Appendix B.2 in (Liu et al., 2020)) and log-linear policy class (Assumption 6.5 part 3 in (Agarwal et al., 2020)).
- For the last assumption, the intuition for $\epsilon_{bias} = 0$ is that the difference between Eq. 31 and 32 is only the distribution of state and action. What we want here is that Eq. 31 is equal to 0 for any distribution. Using any policy parameterizations with $\theta \in \mathbb{R}^d$, we have $|S| \times |A|$ equations (one corresponding to each state-action pair) with d variables. If $d = |S| \times |A|$, we will have $\epsilon_{bias} = 0$. Thus, any complete parameterization for tabular case will have $\epsilon_{bias} = 0$. For the general case, a linear MDP (Jin et al., 2020) will also give $\epsilon_{bias} = 0$ as long as we use the features of the linear MDP (Remark 6.4 in (Agarwal et al., 2020)) and both Gaussian policy and log-linear policy can be used.
- Above all, Gaussian policy and log-linear policy satisfy the above 4 assumptions.

References

Agarwal, A., Kakade, S. M., Lee, J. D., & Mahajan, G. (2020). Optimality and approximation with policy gradient methods in markov decision processes. In Abernethy, J., & Agarwal, S. (Eds.), *Proceedings of Thirty Third Conference on Learning Theory*, Vol. 125 of *Proceedings of Machine Learning Research*, pp. 64–66. PMLR.

Agarwal, M., Aggarwal, V., & Lan, T. (2022). Multi-objective reinforcement learning with non-linear scalarization. In *Proceedings of the 21st International Conference on Autonomous Agents and Multiagent Systems*, pp. 9–17.

Agarwal, M., Bai, Q., & Aggarwal, V. (2021). Concave utility reinforcement learning with zero-constraint violations. *arXiv preprint arXiv:2109.05439*.

Aggarwal, V., Bell, M. R., Elgabli, A., Wang, X., & Zhong, S. (2017). Joint energy-bandwidth allocation for multiuser channels with cooperating hybrid energy nodes. *IEEE Transactions on Vehicular Technology*, *66*(11), 9880–9889.

Badita, A., Parag, P., & Aggarwal, V. (2020). Optimal server selection for straggler mitigation. *IEEE/ACM Transactions on Networking*, *28*(2), 709–721.

Bai, Q., Bedi, A. S., Agarwal, M., Koppel, A., & Aggarwal, V. (2021). Achieving zero constraint violation for constrained reinforcement learning via primal-dual approach. *CoRR*, *abs/2109.06332*.

Bai, Q., Bedi, A. S., Agarwal, M., Koppel, A., & Aggarwal, V. (2022). Achieving zero constraint violation for constrained reinforcement learning via primal-dual approach. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36, pp. 3682–3689.

Bhandari, J., & Russo, D. (2019). Global optimality guarantees for policy gradient methods. *arXiv preprint arXiv:1906.01786*.

Cheung, W. C. (2019). Regret minimization for reinforcement learning with vectorial feedback and complex objectives. *Advances in Neural Information Processing Systems*, *32*, 726–736.

Hazan, E., Kakade, S., Singh, K., & Van Soest, A. (2019). Provably efficient maximum entropy exploration. In *International Conference on Machine Learning*, pp. 2681–2691. PMLR.

Huang, Y., & Kallenberg, L. C. M. (1994). On finding optimal policies for markov decision chains: A unifying framework for mean-variance-tradeoffs. *Mathematics of Operations Research*, *19*(2), 434–448.

Jin, C., Allen-Zhu, Z., Bubeck, S., & Jordan, M. I. (2018). Is q-learning provably efficient?. *Advances in neural information processing systems*, *31*.

Jin, C., Yang, Z., Wang, Z., & Jordan, M. I. (2020). Provably efficient reinforcement learning with linear function approximation. In *Conference on Learning Theory*, pp. 2137–2143. PMLR.

Kakade, S. (2001). A natural policy gradient. In *Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic*, NIPS'01, p. 1531–1538, Cambridge, MA, USA. MIT Press.

Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. In *ICLR (Poster)*.

Konda, V. R., & Tsitsiklis, J. N. (2000). Actor-critic algorithms. In *Advances in neural information processing systems*, pp. 1008–1014. Citeseer.

Lan, T., Kao, D., Chiang, M., & Sabharwal, A. (2010). An axiomatic theory of fairness in network resource allocation. In *2010 Proceedings IEEE INFOCOM*, pp. 1–9.

Liu, Y., Zhang, K., Basar, T., & Yin, W. (2020). An improved analysis of (variance-reduced) policy gradient and natural policy gradient methods. *Advances in Neural Information Processing Systems*, *33*, 7624–7636.

Mei, J., Xiao, C., Szepesvari, C., & Schuurmans, D. (2020). On the global convergence rates of softmax policy gradient methods. In *International Conference on Machine Learning*, pp. 6820–6829. PMLR.

Mihatsch, O., & Neuneier, R. (2002). Risk-sensitive reinforcement learning. *Machine learning*, *49*(2), 267–290.

Nishimura, M., & Yonetani, R. (2020). L2b: Learning to balance the safety-efficiency tradeoff in interactive crowd-aware robot navigation. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, p. 11004–11010. IEEE Press.

Papini, M., Binaghi, D., Canonaco, G., Pirotta, M., & Restelli, M. (2018). Stochastic variance-reduced policy gradient. In *International conference on machine learning*, pp. 4026–4035. PMLR.

Peters, J., & Schaal, S. (2008). Reinforcement learning of motor skills with policy gradients. *Neural Networks*, *21*(4), 682–697. Robotics and Neuroscience.

Pirotta, M., Restelli, M., & Bascetta, L. (2013). Adaptive step-size for policy gradient methods. In Burges, C. J. C., Bottou, L., Welling, M., Ghahramani, Z., & Weinberger, K. Q. (Eds.), *Advances in Neural Information Processing Systems*, Vol. 26. Curran Associates, Inc.

Rummery, G. A., & Niranjan, M. (1994). *On-line Q-learning using connectionist systems*, Vol. 37. University of Cambridge, Department of Engineering Cambridge, UK.

Sutton, R. S. (1988). Learning to predict by the methods of temporal differences. *Machine learning*, *3*(1), 9–44.

Sutton, R. S., McAllester, D., Singh, S., & Mansour, Y. (2000). Policy gradient methods for reinforcement learning with function approximation. In Solla, S., Leen, T., & Müller, K. (Eds.), *Advances in Neural Information Processing Systems*, Vol. 12. MIT Press.

Wang, L., Cai, Q., Yang, Z., & Wang, Z. (2020). Neural policy gradient methods: Global optimality and rates of convergence. In *International Conference on Learning Representations*.

Watkins, C. J., & Dayan, P. (1992). Q-learning. *Machine learning*, *8*(3-4), 279–292.

Williams, R. J. (1992). Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, *8*(3), 229–256.

Xu, P., Gao, F., & Gu, Q. (2020a). An improved convergence analysis of stochastic variance-reduced policy gradient. In *Uncertainty in Artificial Intelligence*, pp. 541–551. PMLR.

Xu, P., Gao, F., & Gu, Q. (2020b). Sample efficient policy gradient methods with recursive variance reduction. In *International Conference on Learning Representations*.

Zhang, J., Koppel, A., Bedi, A. S., Szepesvari, C., & Wang, M. (2020). Variational policy gradient method for reinforcement learning with general utilities. In *Advances in Neural Information Processing Systems*, Vol. 33, pp. 4572–4583.

Zhang, J., Ni, C., Szepesvari, C., & Wang, M. (2021). On the convergence and sample efficiency of variance-reduced policy gradient method. *Advances in Neural Information Processing Systems*, *34*, 2228–2240.

Zhang, K., Koppel, A., Zhu, H., & Basar, T. (2020). Global convergence of policy gradient methods to (almost) locally optimal policies. *SIAM Journal on Control and Optimization*, *58*(6), 3586–3612.

Zhao, T., Hachiya, H., Niu, G., & Sugiyama, M. (2011). Analysis and improvement of policy gradient estimation. *Advances in Neural Information Processing Systems*, *24*.