

ZERO-TH-ORDER POLICY GRADIENT FOR REINFORCEMENT LEARNING FROM HUMAN FEEDBACK WITHOUT REWARD INFERENCE

Qining Zhang & Lei Ying

Department of Electrical Engineering and Computer Science
University of Michigan – Ann Arbor
Ann Arbor, MI 48105, USA
{qiningz, leiyiing}@umich.edu

ABSTRACT

Reward inference (learning a reward model from human preferences) is a critical intermediate step in the *Reinforcement Learning from Human Feedback* (RLHF) pipeline for fine-tuning *Large Language Models* (LLMs). In practice, RLHF faces fundamental challenges such as distribution shift, reward model overfitting, and problem misspecification. An alternative approach is direct policy optimization without reward inference, such as *Direct Preference Optimization* (DPO), which provides a much simpler pipeline and has shown empirical success in LLM applications. However, DPO utilizes the closed-form expression between the optimal policy and the reward function, which is only suitable under the bandit setting or deterministic MDPs. This paper develops two RLHF algorithms without reward inference for *general* RL problems beyond bandits and deterministic MDPs, and *general* preference models beyond the Bradley-Terry model. The key idea is to estimate the local value function difference from human preferences and then approximate the policy gradient with a zeroth-order gradient approximator. For both algorithms, we establish polynomial convergence rates in terms of the number of policy gradient iterations, the number of trajectory samples, and human preference queries per iteration. Numerical experiments in stochastic environments validate the performance of our proposed algorithms, outperforming popular RLHF baselines such as DPO and PPO. Our paper shows that there exist provably efficient methods to solve general RLHF problems without reward inference.

1 INTRODUCTION

In the past decade, we have witnessed unprecedented success in applying *Reinforcement Learning* (RL) to many applications, such as video games (Knox & Stone, 2008; Warnell et al., 2018), recommendation and search (Zeng et al., 2016; Kohli et al., 2013), and autonomous driving (Kiran et al., 2022). RL studies the interaction between decision-making agents and an evolving dynamic environment. At each time step, the agent takes a certain decision (action) given the current state, and a reward signal to measure the quality of that decision is provided by the environment. The agent’s goal is to learn a policy to maximize the cumulative reward, and the quality of the learned policy will depend on the per-step reward function. In classic RL, this reward function is usually handcrafted by domain experts to ensure it aligns with human interests. However, the problem of identifying a “good” reward function, also referred to as *Inverse Reinforcement Learning* (IRL), is non-trivial and one of the most fundamental problems in the history of RL (Ng & Russell, 2000). In recent years, *Reinforcement Learning from Human Feedback* (RLHF) that uses human preference feedback as a signal to recover a reward function has emerged to fine-tune *Large Language Models* (LLMs), which has delivered significant success (Christiano et al., 2017; Wu et al., 2021; Nakano et al., 2021; Ziegler et al., 2019; Stiennon et al., 2020; Ouyang et al., 2022). RLHF follows the diagram shown in Fig. 1, which includes three major steps (Ouyang et al., 2022): (i) pre-train a policy neural network, (ii) collect sets of trajectory pairs and query human evaluators for preferences over trajectory pairs to train a reward model using maximum likelihood to align with human feed-

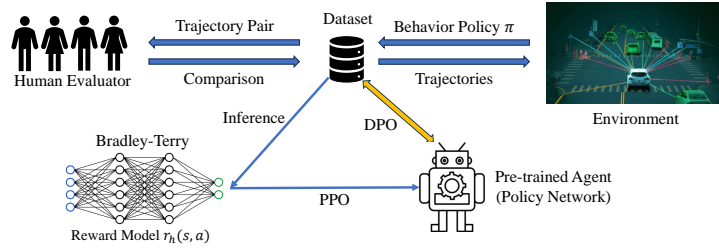


Figure 1: classic policy-based RLHF and DPO: classic RLHF, such as PPO, involves three steps: (i) policy pre-training, (ii) reward inference, and (iii) policy network training with reward model. DPO does not use a reward network but directly optimizes the policy network from human preferences.

back, and (iii) use policy-optimization-based RL algorithms such as PPO (Schulman et al., 2017) to fine-tune the policy network with the reward signals generated by the reward model. The reward inference intermediate step, which trains the reward network, is crucial to obtaining a high-quality policy through RL in the final step.

Drawbacks of Reward Inference. To train a good reward model, i.e., to infer the underlying per-step reward function from human feedback (Christiano et al., 2017; Wang et al., 2023), the most common approach is to assume the feedback is generated based on a preference model such as the *Bradley-Terry* model (Bradley & Terry, 1952), and then maximize the log-likelihood of the collected trajectory comparison dataset accordingly over all possible (parameterized) reward functions. This procedure is indeed analyzed in most theoretical RLHF papers for both offline (Zhu et al., 2023; Zhan et al., 2024a) and online settings (Saha et al., 2023; Zhan et al., 2024b; Wu & Sun, 2024; Wang et al., 2023; Du et al., 2024). However, several challenges occur in practice for reward model training, such as double problem misspecification, reward model evaluation without ground truth, distribution shift, and overfitting in joint reward model and policy training (Casper et al., 2023). These drawbacks are also reflected in the theoretical results, e.g., overfitting of the maximum likelihood estimator (MLE) in Zhu et al. (2024). Moreover, similar to the dilemma in IRL, the reward function that could explain human feedback is often not unique, especially when given limited training trajectories (Arora & Doshi, 2021; Ng & Russell, 2000). Some reward models may make it difficult for agents to learn a good RL policy.

DPO. To avoid the drawbacks of the reward inference in RLHF, Rafailov et al. (2023) proposed an algorithm called *Direct Preference Optimization* (DPO), which fine-tunes the LLM *directly* from human preferences. Based on the Bradley-Terry preference model and a closed-form expression of the optimal policy given a reference policy and the reward function, DPO constructs a loss function directly from human feedback for learning the optimal policy to avoid reward inference. This provides a much simpler pipeline and has great empirical performance (Rafailov et al., 2023; 2024a;b). However, the closed-form expression of the optimal policy that DPO builds on is only for non-parametric policies, and its theoretical justification only works for the bandit setting (Rafailov et al., 2023) or RL problems with deterministic transitions (Rafailov et al., 2024b). It remains an open question how to solve general RLHF problems without reward inference.

RLHF without Reward Inference. Recently, value-based RLHF algorithms without global reward inference have been theoretically developed and analyzed (Xu et al., 2020; Zhang et al., 2024a) based on a dueling bandit approach (Bengs et al., 2021). The results, however, only hold for MDPs in tabular settings with finite state and action spaces. Chen et al. (2022) studied the function approximation regime, but their algorithm requires both the true preference model and the transition kernel to belong to a known function class, which is also far from practice. The result also depends on the function class complexity, which is usually large for most function approximators in practice. So far, no provable policy-based algorithm in this category has been developed.

This paper addresses the following important question:

Does there exist a provably efficient RLHF approach that does not require a reward model and works for general RL problems such as stochastic MDPs or infinite state and action spaces?

1.1 MAIN CONTRIBUTIONS

DPO (Rafailov et al., 2023) establishes a direct connection between human preferences and RL based on the Bradley-Terry model and the optimal policy in closed form:

$$\pi^*(a|x) \propto \pi_{\text{ref}}(a|x) \exp\left(\frac{1}{\beta} r(x, a)\right), \quad (1)$$

where $r(x, a)$ is the reward in state x with action a , π_{ref} is a reference policy and π^* is the optimal policy. Based on the direction connection, the policy optimization can be formulated as a direct matching between human preference and the optimal policy with a log-likelihood loss function. In a recent paper (Rafailov et al., 2024a), it has been further shown that DPO solves a KL-divergence-constrained policy optimization problem for the *deterministic* token-level MDP for LLMs, where the next state is deterministic given the current state and action. For general RL problems with parameterized policies, equation 1 does not hold, and it is often hard, if not impossible, to obtain a “global” function like it that connects the optimal policy and the reward (hence human feedback).

This paper exploits the “local” relation between human feedback and policy optimization. In particular, given a policy π_θ and a perturbed version of the policy $\pi_{\theta+v}$ where v is a small perturbation vector, we use human feedback over the trajectories generated from both policies to inform the direction of a more preferred policy. Intuitively, if one trajectory is preferred over the other, the policy that generates this trajectory is likely to have a higher value. Then given a preference model such as the Bradley-Terry model, we can further estimate the value function differences of the two policies, $V(\pi_{\theta+v}) - V(\pi_\theta)$, where $V(\pi)$ is the value function associated with policy π . Finally, the value difference can be used as an estimator of policy gradient, $\nabla_\theta V(\pi_\theta)$, following the zeroth-order optimization approach (Nesterov & Spokoiny, 2017; Ghadimi & Lan, 2013) to improve the policy.

Based on this idea, this paper proposes two RLHF algorithms without reward inference: **Zeroth-Order Policy Gradient** (ZPG) and **Zeroth-Order Block-Coordinate Policy Gradient** (ZBCPG), both from Human Feedback. ZBCPG differs from ZPG in its policy perturbation rule, which has lower computational complexity and allows parallel optimization since one can sample multiple perturbed policies to perform policy gradient and aggregate the estimated gradient. Under mild assumptions, both algorithms have the following rate of convergence to a stationary policy:

$$\mathcal{O}\left(\frac{Hd}{T} + \frac{d^2\sqrt{\log M}}{\sqrt{M}} + \frac{Hd\sqrt{d}}{\sqrt{N}}\right),$$

where d is the dimension of policy network parameter θ , H is the planning horizon, T is the number of policy gradient steps, N is the number of policy perturbations each step, and M is the number of human queries for each pair of trajectories.

We remark that Tang et al. (2024b) proposes a similar approach towards utilizing human feedback and a zeroth-order gradient descent algorithm from ranking data. However, they assume an error-free ranking oracle over policies based on their value functions, which makes their problem a deterministic optimization problem and does not apply to trajectory preference data like in RLHF and DPO. This paper studies RLHF with trajectory preferences and quantifies the impacts of stochastic trajectories and human preferences on the rate of convergence of RLHF without reward inference.

2 PRELIMINARIES

Episodic RL. We consider an episodic RL problem $\mathcal{M} = (\mathbb{S}, \mathbb{A}, H, \mathbf{P}, \mu_0)$, where \mathbb{S} is the state space and \mathbb{A} is the action space (both can be continuous), H is the RL planning horizon, $\mathbf{P} = \{\mathbf{P}_h\}_{h=1}^H$ is the set of transition kernels, and μ_0 is the initial distribution. At the beginning of each episode, the agent will choose a policy π represented by H functions $\{\pi_h : \mathbb{S} \rightarrow \mathcal{P}(\mathbb{A})\}_{h=1}^H$, where $\mathcal{P}(\mathbb{A})$ denotes the set of all probability distributions over the action space. Then, an initial state s_1 is sampled from the initial distribution μ_0 . At step h , the agent takes an action $a_h = \pi_h(s_h)$ after observing state s_h . The environment then moves to the next state s_{h+1} sampled from the distribution $\mathbf{P}_h(\cdot | s_h, a_h)$ without revealing any reward feedback. We use τ to denote a trajectory with planning horizon H , i.e., $\tau = \{(s_h, a_h)\}_{h=1}^H$.

Trajectory Reward. we assume the expected reward of each trajectory τ is a general function $r(\tau)$ which maps any trajectory to a value in $[0, H]$ (Zhang et al., 2024a). Without loss of generality,

we scale the average per-step reward into $[0, 1]$, and the return of the trajectory does not necessarily need to be the sum of per-step rewards. For any given policy π , we can formulate the initial value function $V_1^\pi(s)$ as the expected reward of trajectories starting from s with policy π :

$$V_1^\pi(s) = \mathbb{E}_\pi[r(\tau) | s_1 = s] = \mathbb{E}[r(\tau) | s_1 = s, \{a_1, \dots, a_H\} \sim \pi].$$

The goal of the RL problem is to find a policy to maximize $V(\pi) = \mathbb{E}_{s \sim \mu_0}[V_1^\pi(s)]$.

Policy Parameterization. to address the large state space \mathbb{S} and action space \mathbb{A} in most RL problems, we assume access to a parameterized policy network $\mathbf{N}_\theta : \mathbb{S} \times [H] \rightarrow \mathcal{P}(\mathbb{A})$ which takes a state and a decision-making step as input, and then outputs the probability distribution of the next action. Here $\theta \in \mathbb{R}^d$ is the policy network parameter vector. Each parameter θ through the policy network will induce a policy, which we slightly abuse the notation and use π_θ to denote.

Human Feedback. The agent has access to human feedback that provides a preference based on the rewards of two trajectories. In each episode, the agent can choose two trajectories τ_0 and τ_1 to query human preference: one-bit feedback $o \in \{0, 1\}$. We assume the preference o is generated according to a known preference model where the probability of the outcome between two trajectories is determined by the difference in their rewards. Since the difference is not necessarily a value inside the unit interval, the preference model uses a *link* function $\sigma : \mathbb{R} \rightarrow [0, 1]$ to map these differences of rewards to actual probabilities, i.e.,

$$\mathbb{P}(\tau_1 \succ \tau_0) = \sigma(r(\tau_1) - r(\tau_0)), \quad (2)$$

where $\tau_1 \succ \tau_0$ is the event that the human feedback prefers τ_1 over τ_0 . The human feedback o , therefore, is a random sample from a Bernoulli distribution with $\mathbb{P}(o = 1) = \mathbb{P}(\tau_1 \succ \tau_0)$. The notion of link function comes from the dueling bandit literature to model preference with latent utility between arms, e.g., see Bengs et al. (2021, Section 3.2). This preference model has been used in dueling bandits (Bengs et al., 2021; Yue & Joachims, 2009; Kumagai, 2017; Ailon et al., 2014) as well as RLHF (Wang et al., 2023). One can see that one specific link function σ will define a specific preference model (Azari et al., 2012), i.e., replacing $\sigma(\cdot)$ with a logistic function, we recover the Bradley-Terry model (Bradley & Terry, 1952), which is commonly used in RLHF for both practical (Christiano et al., 2017; Ouyang et al., 2022; Rafailov et al., 2023) and theoretical works (Du et al., 2024; Zhan et al., 2024a;b). On the other hand, let $\sigma(\cdot)$ be the cumulative distribution function (CDF) of standard normal distribution, we obtain the *Probit* model (Thurstone, 1927) which has wide application in the study of social choice theory, economy, and psychology (Train, 2009; Greene, 2000). A detailed discussion on other preference models is provided in the appendix. The following assumption on the link function is standard in both dueling bandits (Bengs et al., 2021) and preference-based RL (Wang et al., 2023). One can easily verify that both the Bradley-Terry and Probit models satisfy this assumption.

Assumption 1 *The link function $\sigma(\cdot)$ in the preference model in equation 2 is bounded within $[0, 1]$ and strictly monotonically increasing on $[-H, H]$ with $\sigma(0) = 1/2$.*

Problem and Notations. Our goal is to find parameter θ that maximizes the value function, i.e., $\max_{\theta \in \mathbb{R}^d} V(\pi_\theta)$. For a scalar a , we use $\text{trim}[a|\Delta]$ to represent $\min\{\max\{a, \Delta\}, 1 - \Delta\}$. For a vector \mathbf{v} , $\text{trim}[\mathbf{v}|\Delta]$ represents the vector after applying the trim operator to each element respectively. Let $\mathbf{e}_i \in \mathbb{R}^d$ represent the unit vector with all zero elements but 1 on the i -th coordinate.

3 ZERO-ORDER POLICY GRADIENT ALGORITHMS FOR RLHF

In this section, we propose two RLHF algorithms without reward inference, motivated by the relation between preference and zeroth-order gradient. We first present ZPG, a stochastic gradient descent algorithm, and then ZBCPG, a stochastic block-coordinate descent algorithm.

3.1 ZPG: ZERO-ORDER POLICY GRADIENT FROM HUMAN FEEDBACK

Our first algorithm ZPG, consists of the following five steps in each policy gradient iteration:

- From the current policy π_{θ_t} , it first obtained a perturbed policy $\pi_{\theta_t + \mu \mathbf{v}_t}$ (line 2-3).
- Sample N pairs of trajectories under the two policies π_{θ_t} and $\pi_{\theta_t + \mu \mathbf{v}_t}$ (lines 5-6).

Algorithm 1: Zeroth-Order Policy Gradient from Human Feedback**Parameters:** initial parameter θ_0 , learning rate α , trim size Δ , perturbation distance μ .

```

1 for  $t = 1 : T$  do
2   sample  $v_t$  uniformly from a unit sphere  $\mathbb{S}^{d-1} = \{v \in \mathbb{R}^d \mid \|v\|_2 = 1\}$ ;
3   obtain a perturbed policy  $\pi_{\theta_t + \mu v_t}$ ;
4   for  $n = 1 : N$  do
5     sample trajectory  $\tau_{n,0} \sim \pi_{\theta_t}$ ;
6     sample trajectory  $\tau_{n,1} \sim \pi_{\theta_t + \mu v_t}$ ;
7     query  $M$  human evaluators with  $(\tau_{n,1}, \tau_{n,0})$  and obtain feedback  $[o_{n,1}, \dots, o_{n,M}]$ ;
8     estimate preference probability  $p_{t,n} = \text{trim} \left[ \sum_{m=1}^M \frac{o_{n,m}}{M} \mid \Delta \right]$ ;
9   estimate the policy gradient:  $\hat{g}_t = \frac{d}{\mu} \frac{\sum_{n=1}^N \sigma^{-1}(p_{t,n})}{N} v_t$ ;
10  update  $\theta_{t+1} = \theta_t + \alpha \hat{g}_t$ ;

```

- For each trajectory pair, say $(\tau_{n,1}, \tau_{n,0})$, obtain M independent human preferences (line 7) and estimate the probability that $\tau_{n,1}$ is preferred over $\tau_{n,0}$ (line 8), denoted by $p_{t,n}$.
- Use the N estimates $\{p_{t,n}\}_n$ and link function $\sigma(\cdot)$ to estimate the gradient \hat{g}_t (line 9).
- Update the current policy to a new policy θ_{t+1} using gradient ascent (line 10).

The pseudo-code is presented in Alg. 1. As we mentioned earlier, our approach uses human feedback in a way different from both the classic reward inference in RLHF and DPO. The reward inference uses human preferences to recover the *global* reward function, and DPO relates the human preference generation mechanism to the optimal policy. We view the human feedback as local information that points to the direction of a more preferred policy, i.e., the policy gradient direction. Some online RLHF algorithms, such as online DPO, also exploit similar local estimation viewpoints, i.e., using new trajectories of the current policy to locally improve the estimation of DPO loss and then proceed. However, we want to emphasize that the relation between the DPO loss and the optimal policy is still global, and it is limited to deterministic MDPs, which shows the novelty of our approach. The algorithm we propose has two key components: (i) a value function difference estimator from human preference, and (ii) a policy gradient estimator from value function difference.

Policy Gradient Approximation. At each iteration of the algorithm, it first samples a d -dimensional vector v_t from a unit sphere and then perturbs the policy network parameter θ_t along the direction of this sampled vector. Then, it uses the inner for loop to construct an estimation of the value function difference between the original policy and the perturbed policy, i.e., $V(\pi_{\theta_t + \mu v_t}) - V(\pi_{\theta_t})$. We then plug it into the zeroth-order stochastic gradient descent (SGD) algorithm proposed in (Ghadimi & Lan, 2013) to construct a zeroth-order approximation to the policy gradient, i.e.,

$$\nabla_{\theta} V(\pi_{\theta_t}) \approx \mathbb{E}_{v_t} \left[\frac{d}{\mu} (V(\pi_{\theta_t + \mu v_t}) - V(\pi_{\theta_t})) v_t \right].$$

We remark that the random vector v_t for each iteration can also be drawn from a normal distribution (Nesterov & Spokoiny, 2017), but the unit sphere is more numerically stable.

Value Function Inference. The inner for loop of Alg. 1 aims to estimate the value function difference between the perturbed policy $\pi_{\theta_t + \mu v_t}$ and current policy π_{θ_t} . The algorithm samples multiple trajectory pairs with both policies and for each pair, it queries humans multiple times to obtain pairwise preferences $[o_{n,1}, \dots, o_{n,M}]$. It then uses the preferences to construct a robust estimator $p_{t,n}$ to approximate the probability of comparison $\mathbb{P}(\tau_{n,1} \succ \tau_{n,0})$, which is further converted to an estimate of the value function difference based on the preference model in equation 2 as follows:

$$V(\pi_{\theta_t + \mu v_t}) - V(\pi_{\theta_t}) \approx \frac{1}{N} \sum_{n=1}^N \sigma^{-1}(p_{t,n}). \quad (3)$$

Querying humans multiple times ensures an accurate estimation of the reward gap between two trajectories. The reward gap of two trajectories is a random sample of the value function difference, so we sample multiple trajectories to ensure the average trajectory reward gap converges to the

Algorithm 2: Zeroth-Order Block-Coordinate Policy Gradient from Human Feedback

Parameters: initial parameter θ_0 , learning rate α , trim size Δ , perturbation distance μ , coordinate batch size K .

```

1 for  $t = 1 : T$  do
2   sample a set of  $K$  coordinates  $\mathbf{i}_t = [i_{t,1}, i_{t,2}, \dots, i_{t,K}]$  from  $\{1, 2, \dots, d\}$ ;
3   sample a set  $\boldsymbol{\lambda}_t = [\lambda_{t,1}, \lambda_{t,2}, \dots, \lambda_{t,K}]$  where each  $\lambda_{t,j}$  is uniformly sampled from  $\{-1, 1\}$ ;
4   construct the perturbation vector:  $\mathbf{v}_t = \frac{1}{\sqrt{K}} \sum_{j=1}^K \lambda_{t,j} \mathbf{e}_{i_{t,j}}$ ;
5   for  $n = 1 : N$  do
6     sample trajectory  $\tau_{n,0} \sim \pi_{\theta_t}$ ;
7     sample trajectory  $\tau_{n,1} \sim \pi_{\theta_t + \mu \mathbf{v}_t}$ ;
8     query  $M$  human evaluators with  $(\tau_{n,1}, \tau_{n,0})$  and obtain feedback  $[o_{n,1}, \dots, o_{n,M}]$ ;
9     estimate preference probability  $p_{t,n} = \text{trim} \left[ \sum_{m=1}^M \frac{o_{n,m}}{M} \mid \Delta \right]$ ;
10  estimate the policy gradient:  $\hat{\mathbf{g}}_t = \frac{d}{\mu} \frac{\sum_{n=1}^N \sigma^{-1}(p_{t,n})}{N} \mathbf{v}_t$ ;
11  update  $\theta_{t+1} = \theta_t + \alpha \hat{\mathbf{g}}_t$ ;

```

value function difference. To ensure finite variance after applying $\sigma^{-1}(\cdot)$ function, we trim $p_{t,n}$ with a small constant which can be set to $\min\{\sigma(-H), 1 - \sigma(H)\}$ in this case.

3.2 ZBCPG: ZEROth-ORDER BLOCK-COORDINATE POLICY GRADIENT FROM HUMAN FEEDBACK

In high-dimensional optimization problems, it is usually memory and operation-inefficient to approximate the full gradient and update all the parameters in the policy network at the same iteration step (Malladi et al., 2023; Zhang et al., 2024b), which motivates parameter-efficient fine-tuning (PEFT). The stochastic (block) coordinate descent approach naturally arises because of its ease of implementation, low memory requirements, and adaptability to distributed settings (Nesterov, 2012; Lu & Xiao, 2015). The same advantage also applies to RLHF when the number of parameters in the policy network is too large. Therefore, we propose ZBCPG, a block coordinate version of ZPG, which is summarized in Alg. 2. The key difference between ZBCPG and ZPG is the choice of the perturbation direction, where we use Rademacher noise instead of the normal perturbation in ZPG.

Zeroth-Order Block Coordinate Gradient Approximation. Instead of sampling from a sphere, which perturbs all parameters of the policy network, ZBCPG separates the sampling procedure into two simple parts: first, sample a minibatch of coordinates and then sample a zero-centered Bernoulli random variable for each coordinate, which still results in a valid gradient estimator.

$$\nabla_{\theta} V(\pi_{\theta_t}) \approx \mathbb{E}_{\mathbf{i}_t, \boldsymbol{\lambda}_t} \left[\frac{d}{K\mu} (V(\pi_{\theta_t + \mu \mathbf{v}_t}) - V(\pi_{\theta_t})) \mathbf{v}_t \right].$$

The block-coordinate approach allows us to (i) perturb a subset of parameters at each iteration, e.g., a specific layer of the policy network for fine-tuning, and (ii) have a parallel implementation where we have multiple gradient estimators $\hat{\mathbf{g}}_t$ when updating the policy. We will later show that both algorithms have similar provable convergence guarantees, but the analysis of ZBCPG is more challenging due to the perturbation mechanism.

4 THEORETICAL ANALYSIS: RATE OF CONVERGENCE

In this section, we provide theoretical performance guarantees for both ZPG and ZBCPG. We first provide technical assumptions on the preference generation model, the policy network, and the value function landscape, which are necessary for deriving theoretical insights.

4.1 ASSUMPTIONS

To infer the local reward difference from human preference probability through the link function $\sigma(\cdot)$, we impose the following assumption, which is satisfied by the Bradley-Terry model. A slightly weaker assumption is also adopted by Wang et al. (2023) and justified as a minimal requirement to learn the optimal policy. We use $\Delta = \min\{\sigma(-H), 1 - \sigma(H)\}$ as the trim constant.

Assumption 2 *The inverse link function $\sigma^{-1}(\cdot)$ is L -Lipchitz continuous on $[\Delta, 1 - \Delta]$.*

We further require the landscape of the value function and the policy network to be “regular”, and impose the following assumption, which is a standard assumption used in nonconvex optimization literature (Liu et al., 2019; Bernstein et al., 2018; Reddi et al., 2018).

Assumption 3 *The value function $V(\pi_\theta)$ for the policy network parameters θ is L -smooth on \mathbb{R}^d .*

Since a trajectory reward is bounded in $[0, H]$, $V(\pi_{\theta^*}) < \infty$, where θ^* is the global optimal solution. For simplicity, we assume L is the constant upper bound for both assumptions.

4.2 CONVERGENCE RATE AND SAMPLE COMPLEXITY

In this section, we present the theoretical guarantees for both ZPG and ZBCPG under all three assumptions mentioned in the previous sections. We aim to learn an ϵ -stationary policy π_θ with $\|\nabla_\theta V(\pi_\theta)\|_2^2 \leq \epsilon$, and study the convergence rate and sample complexity.

Theorem 1 *Choose the perturbation distance to be $\mu^2 = \Theta\left(\max\left\{\frac{1}{\sqrt{M}}, \frac{H}{\sqrt{dN}}\right\}\right)$ and learning rate to be $\alpha = \Theta(d^{-1})$. If $M = \Omega(H^2)$ and we randomly pick θ_R uniformly from the trajectory $\{\theta_0, \theta_1, \dots, \theta_{T-1}\}$, then the convergence rate of ZPG satisfies:*

$$\mathbb{E} [\|\nabla_\theta V(\pi_{\theta_R})\|_2^2] = \mathcal{O} \left(\frac{Hd}{T} + \frac{d^2 \sqrt{\log M}}{\sqrt{M}} + \frac{Hd\sqrt{d}}{\sqrt{N}} \right).$$

Theorem 2 *Choose the perturbation distance to be $\mu^2 = \Theta\left(\max\left\{\frac{1}{\sqrt{M}}, \frac{H}{\sqrt{dN}}\right\}\right)$ and learning rate to be $\alpha = \Theta(d^{-1})$. If $M = \Omega(H^2)$ and we randomly pick θ_R uniformly from the trajectory $\{\theta_0, \theta_1, \dots, \theta_{T-1}\}$, then the convergence rate of ZBCPG satisfies:*

$$\mathbb{E} [\|\nabla_\theta V(\pi_{\theta_R})\|_2^2] = \mathcal{O} \left(\frac{Hd}{T} + \frac{d^2 \sqrt{\log M}}{\sqrt{M}} + \frac{Hd\sqrt{d}}{\sqrt{N}} \right).$$

The complete proof of both theorems is presented in the appendix. Here we first provide insights into the choice of hyper-parameters and convergence rate results in both theorems, and then we discuss the challenges and technical novelties of our proof.

Insights behind the Convergence Rate. Both ZPG and ZBCPG have the same rate of convergence, which consists of three components: the zeroth-order gradient descent rate, the preference estimation error, and the value function approximation error

$$\underbrace{\frac{Hd}{T}}_{\text{Zeroth-Order Gradient Descent}} + \underbrace{\frac{d^2 \sqrt{\log M}}{\sqrt{M}}}_{\text{Preference Estimation}} + \underbrace{\frac{Hd\sqrt{d}}{\sqrt{N}}}_{\text{Value Function Approximation}}.$$

The second represents the error that occurs when using multiple human preferences $[o_{n,1}, \dots, o_{n,M}]$ to approximate the population-level human preference probability for given two trajectories, i.e., $\mathbb{P}(\tau_{n,1} \succ \tau_{n,0})$. This error will further result in a bias term after being plugged into the inverse link function $\sigma^{-1}(\cdot)$ to construct an estimation of the value function difference. The third term comes from the variance of using multiple trajectory rewards to approximate the value function of a policy. The first term represents the error resulting from zeroth-order stochastic gradient descent or blocked coordinate descent, which matches the state-of-the-art analysis result $\mathcal{O}(d/T)$ for

non-convex smooth function optimization (Nesterov & Spokoiny, 2017). Even though the final convergence rates are the same and we both use constant learning rates, how we choose the perturbation distance to obtain the rate differs from (Ghadimi & Lan, 2013). Specifically, they chose a small perturbation distance with $\mu^2 = \mathcal{O}(d/T)$ to make sure the zeroth-order approximation error is of lower order. However, this choice will not work for us, because our gradient estimate is biased due to the non-linear nature of the link function in preference estimation. If we choose the perturbation distance μ to be too small, the preference estimation error will be amplified by d/μ due to the formula of zeroth-order approximation \hat{g}_t . This phenomenon adds complication to our theoretical analysis. Our method is to use a moderate perturbation distance μ . Moreover, this moderate perturbation distance also balances the preference estimation and the value function approximation errors.

Based on the theorems, we have the following corollary that characterizes the sample complexity.

Corollary 1 *To learn an ϵ -stationary policy, the required number of human preference queries of ZPG and ZBCPG with proper hyperparameters satisfies*

$$TMN = \mathcal{O} \left(\frac{d^8 H^3}{\epsilon^5} \log \left(\frac{d}{\epsilon} \right) \right).$$

4.3 TECHNICAL CHALLENGES AND PROOF NOVELTIES

This section provides an overview of the proof of zeroth-order stochastic gradient descent used in (Ghadimi & Lan, 2013; Nesterov & Spokoiny, 2017; Gao et al., 2018; Liu et al., 2019) from a Lyapunov drift optimization perspective. We then show the major technical difficulties in applying such a framework to analyze both ZPG and ZBCPG, i.e., the gradient estimator is biased due to stochastic human preference. Then, we demonstrate our novel analysis techniques to resolve them.

Classic Proof of Zeroth-Order Optimization. To illustrate the procedure of the analysis of zeroth-order gradient estimate, we suppose we can query $V(\pi_\theta)$ for any θ . This procedure makes use of the randomized smoothing function $V_\mu(\theta)$ (Ghadimi & Lan, 2013; Gao et al., 2018) as $V_\mu(\pi_\theta) = \mathbb{E}_{v'} [V(\pi_{\theta+\mu v'})]$, where the random vector v' follows a uniform distribution over the unit Euclidean ball. It is shown in (Gao et al., 2018) that the zeroth-order gradient estimator used in ZPG, constructed from sampling v_t uniformly over a sphere, is an unbiased estimator of the smoothing function gradient, i.e.,

$$\nabla_\theta V_\mu(\pi_\theta) = \mathbb{E}_v \left[\frac{d}{\mu} (V(\pi_{\theta+\mu v}) - V(\pi_{\theta_t})) v \right],$$

where v is sampled from a unit sphere. The value and gradient of the smoothing function are different from the value function, but they are close as long as μ is small (Liu et al., 2018b):

$$|V_\mu(\pi_\theta) - V(\pi_\theta)| = \mathcal{O}(\mu^2); \quad \|\nabla_\theta V_\mu(\pi_\theta) - \nabla_\theta V(\pi_\theta)\|_2 = \mathcal{O}(\mu d). \quad (4)$$

The standard proof uses the randomized smoothing function $V_\mu(\pi_\theta)$ as the Lyapunov function and then bounds the drift given the stochastic gradient descent update rule when $\alpha = \Theta(1/d)$. Neglecting problem-independent constants, we have:

$$\begin{aligned} & V_\mu(\pi_{\theta_t}) - V_\mu(\pi_{\theta_{t+1}}) \\ & \leq -\underbrace{\alpha \|\nabla_\theta V_\mu(\pi_{\theta_t})\|_2^2}_{\text{Drift}} + \underbrace{\alpha \langle \nabla_\theta V_\mu(\pi_{\theta_t}), \nabla_\theta V_\mu(\pi_{\theta_t}) - \hat{g}_t \rangle}_{\text{1st Order: GradBias}} + \underbrace{\alpha^2 \|\hat{g}_t - \nabla_\theta V_\mu(\pi_{\theta_t})\|_2^2}_{\text{2nd Order: GradVar} \approx \mu^2 d^2}. \end{aligned}$$

Note the gradient estimator \hat{g}_t is unbiased and bounded, and the gradient of $V_\mu(\pi_\theta)$ is close to $\nabla V(\pi_\theta)$, taking a conditional expectation over the filtration before time t will result in:

$$\begin{aligned} & \mathbb{E}[V_\mu(\pi_{\theta_t}) | \mathcal{F}_t] - \mathbb{E}[V_\mu(\pi_{\theta_{t+1}}) | \mathcal{F}_t] \\ & \leq -\alpha \|\nabla_\theta V_\mu(\pi_{\theta_t})\|_2^2 + \alpha \langle \nabla_\theta V_\mu(\pi_{\theta_t}), \mathbb{E}[\nabla_\theta V_\mu(\pi_{\theta_t}) - \hat{g}_t] \rangle + \alpha^2 \mu^2 d^2 \\ & \leq -\alpha \|\nabla_\theta V(\pi_{\theta_t})\|_2^2 + \alpha \mu^2 d^2 + \alpha^2 \mu^2 d^2, \end{aligned}$$

where the last step uses equation 4 and the fact that the gradient is unbiased. Let us choose a small learning rate $\alpha = \Theta(1/d)$ and take an expectation with a telescoping sum to obtain:

$$\underbrace{\frac{\mathbb{E}[V_\mu(\pi_{\theta_0})] - \mathbb{E}[V_\mu(\pi_{\theta_T})]}{T}}_{\mathcal{O}(H/T)} \lesssim -\underbrace{\alpha \mathbb{E} \left[\frac{\sum_{t=1}^T \|\nabla_\theta V(\pi_{\theta_t})\|_2^2}{T} \right]}_{\text{Target}} + \alpha \mu^2 d^2.$$

A little manipulation will lead to the following bound, which can be made small when $\mu \approx \sqrt{1/dT}$.

$$\text{Target} = \mathcal{O}\left(\frac{H}{T\alpha} + \mu^2 d^2\right) = \mathcal{O}\left(\frac{Hd}{T} + \mu^2 d^2\right) = \mathcal{O}\left(\frac{Hd}{T}\right).$$

Amplified Gradient Biases for ZPG. If we directly apply the steps above to ZPG, we immediately run into the issue that our gradient estimator \hat{g}_t in expectation is biased even compared to the smoothing function gradient due to preference estimation. Moreover, the second-order gradient variance will be larger since we used the trajectory reward to estimate the value function. From concentration, we obtain an error bound of using preference to estimate the value function difference:

$$|\mathbb{E}[\sigma^{-1}(p_{t,n})] - (V(\pi_{\theta_t+\mu v_t}) - V(\pi_{\theta_t}))| \leq \tilde{\mathcal{O}}\left(\frac{1}{\sqrt{M}}\right),$$

where $\tilde{\mathcal{O}}$ hides logarithmic terms. This bias term will be amplified by d/μ and then added to the gradient estimation bias in the first-order drift term if plugged into the analysis:

$$\mathbb{E}[V_\mu(\pi_{\theta_t})|\mathcal{F}_t] - \mathbb{E}[V_\mu(\pi_{\theta_{t+1}})|\mathcal{F}_t] \leq \underbrace{-\alpha\|\nabla_{\theta}V(\pi_{\theta_t})\|_2^2 + \alpha\mu^2 d^2}_{\text{Same Drift as Before}} + \underbrace{\alpha \frac{d\|\nabla_{\theta}V_\mu(\pi_{\theta_t})\|_2}{\mu\sqrt{M}}}_{\text{Additional Bias}}.$$

Using the same perturbation distance as before, the additional bias will lead to an $\tilde{\mathcal{O}}(\sqrt{T/M})$ term in the final bound, which is small only when M is much larger than T and is much looser compared with ours. For example, letting $M = T^2$, the above bound is $\tilde{\mathcal{O}}(1/\sqrt{T})$ while ours is $\tilde{\mathcal{O}}(1/T)$.

Our approach to avoid this term in the final result is to make use of the gradient value $\nabla_{\theta}V(\pi_{\theta_t})$ in the first-order term to cancel out the additional bias on certain occasions. Specifically, we divide the trajectory of θ_t into two sets, one with a relatively large gradient and one with a relatively small gradient. For θ_t with a large gradient, we use a part of the negative drift to cancel out the additional bias, since the negative drift is the square of the gradient $\nabla_{\theta}V(\pi_{\theta_t})$, which is even larger. For θ_t with a small gradient, we know the bias term will be small and thus can provide a refined drift bound. Combining this analysis with a slightly larger perturbation distance μ , we will be able to balance the additional bias with gradient variance to cancel out the $\tilde{\mathcal{O}}(\sqrt{T/M})$ term and obtain the final result.

Implicit Smoothing Function for ZBCPG. Due to the choice of blocked perturbation vector sampling procedure, it is difficult to obtain the exact analytical expression of the smoothing function $V_\mu(\pi_{\theta})$ whose gradient is the expectation of gradient estimation \hat{g}_t for ZBCPG. This prohibits us from continuing to use $V_\mu(\pi_{\theta})$ as the Lyapunov function, as it is hard to analyze the gradient bias and the variance without an explicit target format. However, if we rethink the reason for introducing the smoothed function in zeroth-order optimization, we hope the gradient of the smoothed function will be unbiased to cancel out the first-order positive drift. However, this is already not true in the analysis of ZPG since we have gradient estimation bias from human feedback, but it is small enough on average to be controlled. If the gradient difference between the smoothed function $V_\mu(\pi_{\theta})$ and the vanilla value function $V(\pi_{\theta})$ is smaller than this additional bias, then we can use the original value function $V(\pi_{\theta})$ as the Lyapunov function at the cost of an additional bias besides preference estimation. Fortunately, this can be achieved through a carefully chosen perturbation distance μ to balance these two types of errors.

5 EXPERIMENTS

Algorithm	ZPG (Ours)	ZBCPG (Ours)	RM+PPO	DPO	Online DPO
Return	1.94 \pm 0.09	1.91 \pm 0.09	1.80 \pm 0.09	-4.13 \pm 0.09	1.71 \pm 0.09

Table 1: Last Iterate Policy Average Return with Bradley-Terry Feedback.

We study the empirical performance of ZPG and ZBCPG in a stochastic GridWorld environment, where details can be found in Appendix C. In our environment, the actions chosen by the agent may be reversed with certain probabilities due to imperfect control or environmental disturbances

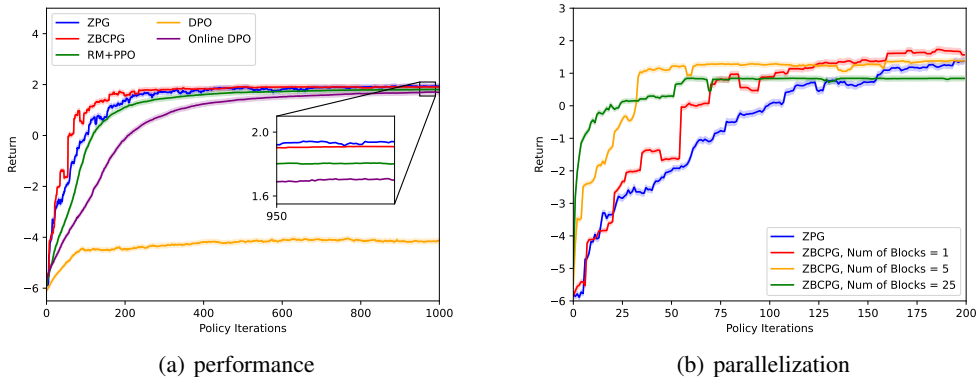


Figure 2: GridWorld with Bradley-Terry Feedback: (a) the trajectory return of ZPG, ZBCPG, and RLHF baselines, and (b) the return of ZBCPG with different parallelization levels. All results are averaged over 10^4 repetitions of policy evaluation, and shaded areas indicate confidence intervals.

such as wind or turbulence. The agent can query human evaluators for preference over two trajectories, and the human feedback is generated from the Bradley-Terry model with a logistic link function. We consider three baselines: (1) RM+PPO (Ouyang et al., 2022), (2) DPO for token-level MDP (Rafailov et al., 2024b), and (3) Online DPO for token-level MDP (Dong et al., 2024; Guo et al., 2024). All algorithms collect $N = 1000$ trajectory pairs between policy updates, and $M = 1000$ human experts evaluate each pair. The trajectory return for each iteration is compared in Fig. 2(a), and the return of the final policy is reported in Tab. 1. Both ZPG and ZBCPG perform better than the three baselines in both convergence rates and the quality of the last iterate policy. Compared to PPO, our algorithms converge to a better policy, partially because the reward model is inaccurate and the agent is not able to learn the optimal policy from it. It is also observed that vanilla DPO has a much worse performance than our proposed algorithms. This may result from two reasons: first, the DPO loss is valid only in a deterministic MDP, and second, DPO is constrained to the neighborhood of the sub-optimal reference policy. The online DPO algorithm improves over vanilla DPO but still has inferior performance due to the inherent model error of the DPO loss. This also shows the fundamental difference between stochastic and deterministic MDPs and the need to design RLHF algorithms for general RL problems. Moreover, online DPO converges much slower, partly because the DPO loss landscape becomes flat and hard to optimize when the weight of the KL constraint is small for better exploration. In Fig. 2(b), we also compare ZPG to distributed implementations of ZBCPG, where the panel of human evaluators is also separated into small groups for parallelization. It is shown that as the number of blocks increases, ZBCPG converges faster to a stationary policy. However, the number of human queries per pair of trajectories in each parallelization also decreases, which introduces a larger gradient bias and leads to a sub-optimal policy. Therefore, the trade-off between computation parallelization and accuracy should be taken carefully.

6 CONCLUSION

In this paper, we proposed two RLHF algorithms without reward inference based on a zeroth-order policy gradient called ZPG and ZBCPG, which train the policy network directly from human preferences without a global reward model. Both algorithms are shown to have a provable polynomial sample complexity to learn a stationary policy under mild conditions and exhibit nice empirical performances in environments with stochastic transitions, outperforming popular RLHF baselines.

ACKNOWLEDGMENTS

The work of Qining Zhang and Lei Ying is supported in part by NSF under grants 2112471, 2134081, 2207548, 2240981, and 2331780.

REFERENCES

- Nir Ailon, Zohar Karnin, and Thorsten Joachims. Reducing dueling bandits to cardinal bandits. In *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pp. 856–864, Beijing, China, 22–24 Jun 2014. PMLR.
- Riad Akrou, Marc Schoenauer, and Michele Sebag. Preference-based policy learning. In *Machine Learning and Knowledge Discovery in Databases*, pp. 12–27, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.
- Saurabh Arora and Prashant Doshi. A survey of inverse reinforcement learning: Challenges, methods and progress. *Artificial Intelligence*, 297:103500, 2021. ISSN 0004-3702. doi: <https://doi.org/10.1016/j.artint.2021.103500>.
- Mohammad Gheshlaghi Azar, Zhaohan Daniel Guo, Bilal Piot, Remi Munos, Mark Rowland, Michal Valko, and Daniele Calandriello. A general theoretical paradigm to understand learning from human preferences. In *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pp. 4447–4455. PMLR, 02–04 May 2024.
- Hossein Azari, David Parks, and Lirong Xia. Random utility theory for social choice. *Advances in Neural Information Processing Systems*, 25, 2012.
- Viktor Bengs, Robert Busa-Fekete, Adil El Mesaoudi-Paul, and Eyke Hullermeier. Preference-based online learning with dueling bandits: A survey. *Journal of Machine Learning Research*, 22(7): 1–108, 2021.
- Jeremy Bernstein, Yu-Xiang Wang, Kamyar Azizzadenesheli, and Animashree Anandkumar. signSGD: Compressed optimisation for non-convex problems. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 560–569. PMLR, 10–15 Jul 2018.
- Ralph Allan Bradley and Milton E. Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952. ISSN 00063444, 14643510.
- Róbert Busa-Fekete, Balázs Szörényi, Paul Weng, Weiwei Cheng, and Eyke Hüllermeier. Preference-based reinforcement learning: evolutionary direct policy search using a preference-based racing algorithm. *Machine learning*, 97:327–351, 2014.
- Hanqin Cai, Yuchen Lou, Daniel Mckenzie, and Wotao Yin. A zeroth-order block coordinate descent algorithm for huge-scale black-box optimization. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 1193–1203. PMLR, 18–24 Jul 2021.
- Stephen Casper, Xander Davies, Claudia Shi, Thomas Krendl Gilbert, Jérémy Scheurer, Javier Rando Ramirez, Rachel Freedman, Tomasz Korbak, David Lindner, Pedro Freire, et al. Open problems and fundamental limitations of reinforcement learning from human feedback. *Trans. Machine Learning Research (TMLR)*, 2023.
- Souradip Chakraborty, Amrit Bedi, Alec Koppel, Huazheng Wang, Dinesh Manocha, Mengdi Wang, and Furong Huang. PARL: A unified framework for policy alignment in reinforcement learning from human feedback. In *The Twelfth International Conference on Learning Representations*, 2024.
- Xiaoyu Chen, Han Zhong, Zhuoran Yang, Zhaoran Wang, and Liwei Wang. Human-in-the-loop: Provably efficient preference-based reinforcement learning with general function approximation. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 3773–3793. PMLR, 17–23 Jul 2022.
- Paul Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

- Edoardo Conti, Vashisht Madhavan, Felipe Petroski Such, Joel Lehman, Kenneth Stanley, and Jeff Clune. Improving exploration in evolution strategies for deep reinforcement learning via a population of novelty-seeking agents. *Advances in neural information processing systems*, 31, 2018.
- Hanze Dong, Wei Xiong, Bo Pang, Haoxiang Wang, Han Zhao, Yingbo Zhou, Nan Jiang, Doyen Sahoo, Caiming Xiong, and Tong Zhang. Rlhf workflow: From reward modeling to online rlhf. *arXiv preprint arXiv:2405.07863*, 2024.
- Yihan Du, Anna Winnicki, Gal Dalal, Shie Mannor, and R. Srikant. Exploration-driven policy optimization in RLHF: Theoretical insights on efficient data utilization. In *Forty-first International Conference on Machine Learning*, 2024.
- Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. Kto: Model alignment as prospect theoretic optimization. *arXiv preprint arXiv:2402.01306*, 2024.
- Xiang Gao, Bo Jiang, and Shuzhong Zhang. On the information-adaptive variants of the admm: an iteration complexity perspective. *Journal of Scientific Computing*, 76:327–363, 2018.
- Saeed Ghadimi and Guanghui Lan. Stochastic first-and zeroth-order methods for nonconvex stochastic programming. *SIAM journal on optimization*, 23(4):2341–2368, 2013.
- WH Greene. Modeling ordered choices: A primer, 2010.
- William H Greene. Econometric analysis 4th edition. *International edition, New Jersey: Prentice Hall*, pp. 201–215, 2000.
- Shangmin Guo, Biao Zhang, Tianlin Liu, Tianqi Liu, Misha Khalman, Felipe Llinares, Alexandre Rame, Thomas Mesnard, Yao Zhao, Bilal Piot, et al. Direct language model alignment from online ai feedback. *arXiv preprint arXiv:2402.04792*, 2024.
- Hamed Karimi, Julie Nutini, and Mark Schmidt. Linear convergence of gradient and proximal-gradient methods under the polyak-łojasiewicz condition. In Paolo Frasconi, Niels Landwehr, Giuseppe Manco, and Jilles Vreeken (eds.), *Machine Learning and Knowledge Discovery in Databases*, pp. 795–811, Cham, 2016. Springer International Publishing. ISBN 978-3-319-46128-1.
- Timo Kaufmann, Paul Weng, Viktor Bengs, and Eyke Hüllermeier. A survey of reinforcement learning from human feedback. *arXiv preprint arXiv:2312.14925*, 2023.
- Chinmaya Kausik, Mirco Mutti, Aldo Pacchiano, and Ambuj Tewari. A framework for partially observed reward-states in rlhf. *arXiv preprint arXiv:2402.03282*, 2024.
- Ravi Kiran, Ibrahim Sobh, Victor Talpaert, Patrick Mannion, Ahmad A. Al Sallab, Senthil Yogamani, and Patrick Pérez. Deep reinforcement learning for autonomous driving: A survey. *IEEE Transactions on Intelligent Transportation Systems*, 23(6):4909–4926, 2022. doi: 10.1109/TITS.2021.3054625.
- Bradley Knox and Peter Stone. Tamer: Training an agent manually via evaluative reinforcement. In *2008 7th IEEE International Conference on Development and Learning*, pp. 292–297, 2008. doi: 10.1109/DEVLRN.2008.4640845.
- Pushmeet Kohli, Mahyar Salek, and Greg Stoddard. A fast bandit algorithm for recommendation to users with heterogenous tastes. *Proceedings of the AAAI Conference on Artificial Intelligence*, 27(1):1135–1141, Jun. 2013. doi: 10.1609/aaai.v27i1.8463.
- Dingwen Kong and Lin Yang. Provably feedback-efficient reinforcement learning via active reward learning. In *Advances in Neural Information Processing Systems*, volume 35, pp. 11063–11078. Curran Associates, Inc., 2022.
- Wataru Kumagai. Regret analysis for continuous dueling bandit. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.
- Zihao Li, Zhuoran Yang, and Mengdi Wang. Reinforcement learning with human feedback: Learning dynamic choices via pessimism. *arXiv preprint arXiv:2305.18438*, 2023.

- Sijia Liu, Jie Chen, Pin-Yu Chen, and Alfred Hero. Zeroth-order online alternating direction method of multipliers: Convergence analysis and applications. In *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, volume 84 of *Proceedings of Machine Learning Research*, pp. 288–297. PMLR, 09–11 Apr 2018a.
- Sijia Liu, Bhavya Kailkhura, Pin-Yu Chen, Paishun Ting, Shiyu Chang, and Lisa Amini. Zeroth-order stochastic variance reduction for nonconvex optimization. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018b.
- Sijia Liu, Pin-Yu Chen, Xiangyi Chen, and Mingyi Hong. signSGD via zeroth-order oracle. In *International Conference on Learning Representations*, 2019.
- Zhaosong Lu and Lin Xiao. On the complexity analysis of randomized block-coordinate descent methods. *Mathematical Programming*, 152:615–642, 2015.
- Sadhika Malladi, Tianyu Gao, Eshaan Nichani, Alex Damian, Jason D Lee, Danqi Chen, and Sanjeev Arora. Fine-tuning language models with just forward passes. In *Advances in Neural Information Processing Systems*, volume 36, pp. 53038–53075. Curran Associates, Inc., 2023.
- Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, et al. Webgpt: Browser-assisted question-answering with human feedback. *arXiv preprint arXiv:2112.09332*, 2021.
- Yu. Nesterov. Efficiency of coordinate descent methods on huge-scale optimization problems. *SIAM Journal on Optimization*, 22(2):341–362, 2012. doi: 10.1137/100802001.
- Yurii Nesterov and Vladimir Spokoiny. Random gradient-free minimization of convex functions. *Found. Comput. Math.*, 17(2):527–566, apr 2017. ISSN 1615-3375. doi: 10.1007/s10208-015-9296-2.
- Andrew Y. Ng and Stuart J. Russell. Algorithms for inverse reinforcement learning. In *Proceedings of the Seventeenth International Conference on Machine Learning*, ICML ’00, pp. 663–670, San Francisco, CA, USA, 2000. Morgan Kaufmann Publishers Inc. ISBN 1558607072.
- Ellen Novoseller, Yibing Wei, Yanan Sui, Yisong Yue, and Joel Burdick. Dueling posterior sampling for preference-based reinforcement learning. In *Proceedings of the 36th Conference on Uncertainty in Artificial Intelligence (UAI)*, volume 124 of *Proceedings of Machine Learning Research*, pp. 1029–1038. PMLR, 03–06 Aug 2020.
- Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pp. 27730–27744. Curran Associates, Inc., 2022.
- Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems*, volume 36, pp. 53728–53741. Curran Associates, Inc., 2023.
- Rafael Rafailov, Yaswanth Chittipudi, Ryan Park, Harshit Sikchi, Joey Hejna, Bradley Knox, Chelsea Finn, and Scott Niekum. Scaling laws for reward model overoptimization in direct alignment algorithms. *arXiv preprint arXiv:2406.02900*, 2024a.
- Rafael Rafailov, Joey Hejna, Ryan Park, and Chelsea Finn. From r to q^* : Your language model is secretly a q -function. In *First Conference on Language Modeling*, 2024b.
- Ingo Rechenberg. Evolutionsstrategie. *Optimierung technischer Systeme nach Prinzipien der biologischen Evolution*, 1973.
- Sashank J. Reddi, Satyen Kale, and Sanjiv Kumar. On the convergence of adam and beyond. In *International Conference on Learning Representations*, 2018.

- Aadirupa Saha, Aldo Pacchiano, and Jonathan Lee. Dueling rl: Reinforcement learning with trajectory preferences. In *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics*, volume 206 of *Proceedings of Machine Learning Research*, pp. 6263–6289. PMLR, 25–27 Apr 2023.
- Tim Salimans, Jonathan Ho, Xi Chen, Szymon Sidor, and Ilya Sutskever. Evolution strategies as a scalable alternative to reinforcement learning. *arXiv preprint arXiv:1703.03864*, 2017.
- Marc Schoenauer, Riad Akrou, Michele Sebag, and Jean-Christophe Souplet. Programming by feedback. In *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pp. 1503–1511, Beijing, China, 22–24 Jun 2014. PMLR.
- John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.
- Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. Learning to summarize with human feedback. In *Advances in Neural Information Processing Systems*, volume 33, pp. 3008–3021. Curran Associates, Inc., 2020.
- Yunhao Tang, Zhaohan Daniel Guo, Zeyu Zheng, Daniele Calandriello, Remi Munos, Mark Rowland, Pierre Harvey Richemond, Michal Valko, Bernardo Avila Pires, and Bilal Piot. Generalized preference optimization: A unified approach to offline alignment. In *Forty-first International Conference on Machine Learning*, 2024a.
- Zhiwei Tang, Dmitry Rybin, and Tsung-Hui Chang. Zeroth-order optimization meets human feedback: Provable learning via ranking oracles. In *The Twelfth International Conference on Learning Representations*, 2024b.
- LL Thurstone. A law of comparative judgment. *Psychological Review*, 34(4):273, 1927.
- Kenneth E Train. *Discrete choice methods with simulation*. Cambridge university press, 2009.
- Yuanhao Wang, Qinghua Liu, and Chi Jin. Is rlhf more difficult than standard rl? *arXiv preprint arXiv:2306.14111*, 2023.
- Garrett Warnell, Nicholas Waytowich, Vernon Lawhern, and Peter Stone. Deep tamer: Interactive agent shaping in high-dimensional state spaces. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1), Apr. 2018. doi: 10.1609/aaai.v32i1.11485.
- Aaron Wilson, Alan Fern, and Prasad Tadepalli. A bayesian approach for policy learning from trajectory preference queries. In *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012.
- Christian Wirth, Johannes Fürnkranz, and Gerhard Neumann. Model-free preference-based reinforcement learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 30(1), Mar. 2016. doi: 10.1609/aaai.v30i1.10269.
- Christian Wirth, Riad Akrou, Gerhard Neumann, and Johannes Fürnkranz. A survey of preference-based reinforcement learning methods. *Journal of Machine Learning Research*, 18(136):1–46, 2017.
- Jeff Wu, Long Ouyang, Daniel Ziegler, Nisan Stiennon, Ryan Lowe, Jan Leike, and Paul Christiano. Recursively summarizing books with human feedback. *arXiv preprint arXiv:2109.10862*, 2021.
- Runzhe Wu and Wen Sun. Making RL with preference-based feedback efficient via randomization. In *The Twelfth International Conference on Learning Representations*, 2024.
- Tengyang Xie, Dylan J Foster, Akshay Krishnamurthy, Corby Rosset, Ahmed Awadallah, and Alexander Rakhlin. Exploratory preference optimization: Harnessing implicit q^* -approximation for sample-efficient rlhf. *arXiv preprint arXiv:2405.21046*, 2024.

- Wei Xiong, Hanze Dong, Chenlu Ye, Ziqi Wang, Han Zhong, Heng Ji, Nan Jiang, and Tong Zhang. Iterative preference learning from human feedback: Bridging theory and practice for RLHF under KL-constraint. In *Forty-first International Conference on Machine Learning*, 2024.
- Yichong Xu, Ruosong Wang, Lin Yang, Aarti Singh, and Artur Dubrawski. Preference-based reinforcement learning with finite-time guarantees. In *Advances in Neural Information Processing Systems*, volume 33, pp. 18784–18794. Curran Associates, Inc., 2020.
- Yisong Yue and Thorsten Joachims. Interactively optimizing information retrieval systems as a dueling bandits problem. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML ’09, pp. 1201–1208, New York, NY, USA, 2009. Association for Computing Machinery. ISBN 9781605585161. doi: 10.1145/1553374.1553527.
- Chunqiu Zeng, Qing Wang, Shekoofeh Mokhtari, and Tao Li. Online context-aware recommendation with time varying multi-armed bandit. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD ’16, pp. 2025–2034, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450342322. doi: 10.1145/2939672.2939878.
- Wenhao Zhan, Masatoshi Uehara, Nathan Kallus, Jason D. Lee, and Wen Sun. Provable offline preference-based reinforcement learning. In *The Twelfth International Conference on Learning Representations*, 2024a.
- Wenhao Zhan, Masatoshi Uehara, Wen Sun, and Jason D. Lee. Provable reward-agnostic preference-based reinforcement learning. In *The Twelfth International Conference on Learning Representations*, 2024b.
- Qining Zhang, Honghao Wei, and Lei Ying. Reinforcement learning from human feedback without reward inference: Model-free algorithm and instance-dependent analysis. *Reinforcement Learning Journal*, 3:1236–1251, 2024a.
- Yihua Zhang, Pingzhi Li, Junyuan Hong, Jiayang Li, Yimeng Zhang, Wenqing Zheng, Pin-Yu Chen, Jason D. Lee, Wotao Yin, Mingyi Hong, Zhangyang Wang, Sijia Liu, and Tianlong Chen. Revisiting zeroth-order optimization for memory-efficient LLM fine-tuning: A benchmark. In *Forty-first International Conference on Machine Learning*, 2024b.
- Yao Zhao, Rishabh Joshi, Tianqi Liu, Misha Khalman, Mohammad Saleh, and Peter J Liu. Slic-hf: Sequence likelihood calibration with human feedback. *arXiv preprint arXiv:2305.10425*, 2023.
- Banghua Zhu, Michael Jordan, and Jiantao Jiao. Principled reinforcement learning with human feedback from pairwise or k-wise comparisons. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 43037–43067. PMLR, 23–29 Jul 2023.
- Banghua Zhu, Michael Jordan, and Jiantao Jiao. Iterative data smoothing: Mitigating reward overfitting and overoptimization in RLHF. In *Forty-first International Conference on Machine Learning*, 2024.
- Daniel Ziegler, Nisan Stiennon, Jeffrey Wu, Tom Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*, 2019.

A RELATED WORK

We review recent developments in RLHF and zeroth-order optimization for both empirical and theoretical results. A thorough survey on RLHF can be found in Kaufmann et al. (2023) and Casper et al. (2023)

Empirical Studies of Direct RLHF in LLMs. DPO (Rafailov et al., 2024a) and SLiC-HF (Zhao et al., 2023) have empirically shown it is possible to directly learn an RL policy from human preference. In DPO, the authors solve a KL-divergence-constrained reward maximization problem for a prompt response generation problem similar to the contextual bandit. The optimal policy for this problem has a closed-form expression, and the reward of each prompt-response pair can be computed knowing the optimal policy and the reference policy. The authors then minimize a log-likelihood loss by plugging the reward expression into the Bradley-Terry model to measure the alignment between the policy and the human preference without training a reward network. Rafailov et al. (2024b) extends this loss to token-level MDP on the condition that the MDP transition is deterministic, i.e., the next state is a concatenation of the current state and action. This limits the DPO loss minimization approach to LLM problems. In general, MDPs with stochastic transitions, the DPO loss cannot be computed following the derivation in (Rafailov et al., 2024b). Azar et al. (2024) extends DPO to a wider class of RL problems, even avoiding the notion of an underlying reward function. Instead of maximizing the reward in a KL-constrained problem like DPO, the authors proposed to optimize a general non-decreasing function of the ground-truth population-level preference probability. Other variants are also studied. Ethayarajh et al. (2024) considers aligning policy with humans and designs loss from a prospect theory perspective, and Tang et al. (2024a) considers optimizing a general loss of the preference loss instead of the log-likelihood. Dong et al. (2024) and Xiong et al. (2024) proposed to obtain human feedback in an online fashion to mitigate the distribution-shift and over-parameterization phenomenon. Attempts to understand the theoretical performance of RLDHF algorithms such as DPO are made in (Azar et al., 2024), but the authors only showed the existence of optima of the loss function, without any policy optimality and sample complexity guarantees.

RLHF with Provable Sample Complexity. Two major approaches have been studied to learn the optimal policy from human preference data. The first is similar to the traditional RLHF paradigm, such as PPO used in empirical studies, which infers a reward function, or sometimes a utility function, and then trains an RL agent. This reward inference step is also called surrogate learning in the preference-based RL literature, i.e., see (Wirth et al., 2017, Sec. 2.4). The second approach directly optimizes the policy from human preferences. Empirical algorithms for both approaches have been developed without theoretical guarantees for a few years. For example, reward function or utility function estimation followed by an RL policy search algorithm has been proposed and studied in (Schoenauer et al., 2014; Wirth et al., 2016; Christiano et al., 2017), while direct policy learning from humans through trajectory sampling also has been proposed in using a heuristic evolutionary algorithm (Busa-Fekete et al., 2014; Akrouer et al., 2011), or from a Bayesian Markov chain Monte Carlo perspective (Wilson et al., 2012). However, it was not until recently that algorithms with provable theoretical guarantees were proposed. From the reward inference approach, Novoseller et al. (2020) took a Bayesian perspective and maintained a posterior distribution over the reward function and the transition kernel, which is computationally costly. Wang et al. (2023) assumes the reward function can be linearly parameterized with known feature embedding and proposes a preference-to-reward interface (a reward model) using an online policy update and a baseline trajectory for comparison. The authors theoretically showed that for a general known preference model setting, RLHF is no harder than RL. A similar analysis framework is adopted in contemporary theoretical works, e.g., (Saha et al., 2023; Zhan et al., 2024a;b; Zhu et al., 2023; Kong & Yang, 2022; Wu & Sun, 2024), for both online, offline, and hybrid RL problems, where the algorithms first learn the linear parameter of the reward function and perform value-based RL on the learned reward function. The analysis first characterizes the error of the reward parameter using the concentration of the MLE estimator and then propagates this error to the value-based RL algorithm. Specifically, Saha et al. (2023) considers the tabular RL setting and assumes a known feature embedding for each trajectory with a known transition kernel. Then, the agent is directed by RL to explore the feature direction where the uncertainty in the reward function is large. Wu & Sun (2024) extends the scenario to linear MDPs with unknown transition and uses least square value iteration to solve the RL problem. Zhu et al. (2023) extends the work (Saha et al., 2023) to the offline RL scenario where a pessimistic estimator of the reward parameter is used to combat the insufficient data coverage in offline settings.

Zhan et al. (2024b) replaces the linear reward parameterization with a general function class under the realizability condition. This also enables them to solve the unknown preference model as long as it is in the known function class. Zhan et al. (2024b) and Kong & Yang (2022) study the hybrid RL problem with human preference, where they first use an exploratory policy from an optimal design perspective to improve the coverage of the offline dataset and to extract more information useful for reward inference. Then, the problem will be solved under the general RL framework. All papers above took a value-based approach in RL and Du et al. (2024) analyzed natural policy gradient with reward inference. Direct policy learning from human preference has been less analyzed compared to reward inference approaches. In tabular MDPs, Xu et al. (2020) and Zhang et al. (2024a) reduce the RL problem to a sequence of dueling bandit problems. However, the approach is only suitable for MDPs with finite state and action spaces. For general MDPs, Chen et al. (2022) first uses function approximation for the mapping from trajectory pair to human preference, and learns the RL transition kernel from a least square estimator with this preference approximator. The optimal policy can then be learned using a dueling bandit approach. However, their results assume the true preference model and transition kernel are inside the known function class with small complexity, which is strong in real applications. RLHF has also been studied in other aspects. Li et al. (2023) studies RL from the human behavior dataset from a dynamic choice perspective. Chakraborty et al. (2024) formulated the reward inference and the policy optimization as a bilevel optimization problem, and Kausik et al. (2024) studies RLHF with partially observed rewards and states. Zhu et al. (2024) studies the overfitting issue in reward model training. Recently, Xie et al. (2024) combined DPO with optimistic exploration to design XPO in the function approximation regime with provable convergence. However, their algorithm and results are still dependent on the DPO loss, which only works for deterministic MDPs.

Zeroth-Order Optimization and Evolutionary Strategies. The zeroth-order optimization problem has been studied in the convex and non-convex optimization literature for more than a decade (Ghadimi & Lan, 2013; Nesterov & Spokoiny, 2017), where the stochastic gradient descent algorithm with two-point gradient estimator is most widely used. The convergence rate in smooth functions is first studied in (Ghadimi & Lan, 2013) in both convex and non-convex settings. In the convex setting, the algorithm finds the optimal point, while in the non-convex setting, the algorithm finds a stationary point. The rate of convergence for non-convex functions is improved in Nesterov & Spokoiny (2017). Variants of stochastic gradient descent, such as variance reduction techniques (Liu et al., 2018b) and ADMM (Liu et al., 2018a; Gao et al., 2018) have also been studied in the zeroth-order literature. (Cai et al., 2021) extends the zeroth-order method to blocked coordinate descent for computational efficiency and Liu et al. (2019) extends the method to analyze a signed version of SGD, which is more memory efficient in federated settings since each element of the gradient takes only one bit. In recent years, the study of evolutionary strategies (Rechenberg, 1973) brought zeroth-order optimization methods to optimize RL agents, which has achieved empirical success as a scalable alternative to classic RL algorithms such as Q-learning or policy gradient (Salimans et al., 2017; Conti et al., 2018). This strategy has also been applied in preference-based RL as well (Busa-Fekete et al., 2014; Akroun et al., 2011). However, the theoretical guarantees and provable algorithms remain under-explored. The zeroth-order optimization technique has been proposed in optimizing LLMs (Malladi et al., 2023; Zhang et al., 2024b), but they only implement it in the procedure of policy optimization from the reward network to avoid the heavy memory burden in back-propagation. In their studied problem, the loss function can be explicitly estimated or calculated, and thus can be queried to construct the zeroth-order gradient, which is more similar to classic evolutionary strategies. Our work is different in that the loss function, i.e., the value function, cannot be directly queried without reward feedback, so the zeroth-order algorithms cannot be directly used. However, we view human feedback as a natural source of zeroth-order information and apply the method directly from human preference.

Preference Models. The study of modeling the rationale of human decision-making has been pertinent for almost a century. The most popular and widely studied model in the literature is the random utility model (RUM) in social choice theory (Azari et al., 2012), which was first developed as early as 1920s (Thurstone, 1927). In RUM, for each decision-maker, each choice is associated with a utility that consists of two portions: a shared public utility, which is the same for all people, and a private utility (noise) unique to each person. People are modeled as utility maximizers and therefore will choose the choice that has the largest total utility. The private utility is assumed to follow a certain distribution among the population of decision-makers. It is not hard to see that the public utility will determine which choice is more preferred among the population, and the distribution of

the private utility will determine how much it is more preferred, i.e., the population preference probability. Different distribution assumptions give rise to different preference models with different link functions. If the private utility follows a normal distribution, the preference model is called the Probit model, where the link function is the cumulative distribution function of a normal distribution. The Probit model is the first proposed and studied preference model in the literature (Thurstone, 1927). Suppose the private utility follows the Gumbel distribution. We recover the Bradley-Terry model, also known as the logit model, with the logistic link function. Other preference models include the linear model with linear link function, the Weibull model, the Cauchy model, and the complementary log-log model, which have been studied in the literature for specific applications, e.g., see (Train, 2009; Greene, 2000; 2010) for a comprehensive review. In general, when the utility is close among choices, the Bradley-Terry model and the Probit model are often more accurate and easier to implement.

B DISCUSSION

Local and Global Convergence. Our results only establish the local convergence of both proposed algorithms. In general, considering the generality of stochastic MDPs and policy parameterization studied in our paper, it is extremely difficult to establish global convergence results without further assumptions beyond the ones that have already been made in the paper, such as smoothness. We conjecture that our proposed algorithms will have global convergence under additional assumptions on the value function, such as convexity or the Polyak-Lojasiewicz (PL) condition (Karimi et al., 2016). Again, these assumptions are often questionable in practice. Some contemporary works, such as Xie et al. (2024), established global convergence in the deterministic KL-constrained MDPs with additional coverability assumptions, a much easier and narrower model where the optimal policy can be explicitly expressed. Considering the difficulty and generality of our setting, we think local convergence is the best result that can be obtained, and a meaningful result showing the effectiveness of the proposed algorithm to a wider range of RL problems beyond LLMs. It remains an open question of how to design algorithms that achieve global optimality under minimal and practical assumptions.

Local and Global Reward Estimation. The main philosophical difference between our proposed algorithms and classic RLHF paradigms is the use of local reward information around the neighborhood of the current policy. This is completely different from RLHF with reward inference since reward models intend to approximate the global reward landscape of the MDP. Some recently developed online RLHF algorithms, such as online DPO (Guo et al., 2024) and XPO (Xie et al., 2024), also employ certain local reward information to estimate the DPO loss in each iteration. Specifically, the algorithms will explore the trajectories in the neighborhood of the current policy to construct a local estimation of the DPO loss. However, on the other hand, the relation between the DPO loss and the best policy is still global in deterministic MDPs studied in both papers. Our paper’s approach is purely local and does not rely on such global relations, which is incorrect in stochastic MDPs. It estimates the local policy gradient directly from human feedback and proceeds with gradient descent for policy updates. The use of local information in our approach is more general and may apply to a wider range of RL problems.

KL Regularization in RLHF. The KL-regularization is commonly used in RLHF for both empirical and theoretical studies, e.g., PPO (Ouyang et al., 2022), DPO (Rafailov et al., 2024a;b; Guo et al., 2024; Dong et al., 2024), and more recent algorithms (Xie et al., 2024; Xiong et al., 2024). On the positive side, the KL-regularization improves the training stability in RLHF tasks. For example, in LLM applications, the model pre-training procedure will provide powerful capabilities such as reasoning, and the regularization ensures that the updated policy is unlikely to deviate from the pre-training model and lose these capabilities. It also ensures the global relation between the optimal policy and the reward function in deterministic MDPs for LLM studies and motivates easier objectives, such as the DPO loss. On the other hand, in general RL problems, if the regularization is too large, it also limits the RL agent from the possibility of learning the best policy, which may be very different from the reference policy of the pre-trained model. It is also worth noting that KL-regularized MDPs may be easier to explore compared to their unregularized counterparts, especially for MDPs with sparse rewards. This is because the best policies for regularized MDPs are likely to be stochastic policies that exhibit certain exploration power due to stochasticity. In contrast, the optimal policy for unregularized counterparts is deterministic, which does not explore. It seems

there is no golden rule on whether KL-regularization is indeed needed in RLHF applications beyond LLMs. The answer is likely to depend on specific applications. Our algorithm can also be applied to KL-regularized MDPs after a slight modification in gradient estimation. In this case, the zeroth-order gradient of the policy will include the sum of two components, the value function difference estimation, and the KL regularization difference, i.e.,

$$(V(\pi_{\theta_t+\mu\mathbf{v}_t}) - \beta\text{KL}(\pi_{\theta_t+\mu\mathbf{v}_t}\|\pi_{\theta_0})) - (V(\pi_{\theta_t}) - \beta\text{KL}(\pi_{\theta_t}\|\pi_{\theta_0})),$$

since the objective is now the KL-regularized value function. Here, π_{θ_0} is the reference policy, and $\text{KL}(\pi_{\theta}\|\pi_{\theta_0})$ is the KL divergence between a policy parameterized by θ and the reference policy, i.e.,

$$\text{KL}(\pi_{\theta}\|\pi_{\theta_0}) = \mathbb{E}_{s_h, a_h \sim \pi_{\theta}} \left[\sum_{h=1}^H \log \frac{\pi_{\theta}(a_h|s_h)}{\pi_{\theta_0}(a_h|s_h)} \right].$$

Remark that human evaluators will only provide preference based on the trajectory returns, not the KL regularization. Therefore, the value function difference component can be estimated in the same way as the current algorithm, i.e., line 9 in Alg. 1. Then, the rest is to estimate the KL regularization difference between the perturbed policy $\pi_{\theta_t+\mu\mathbf{v}_t}$ and the current policy π_{θ_t} , concerning the reference policy π_{θ_0} , i.e.,

$$\begin{aligned} & \text{KL}(\pi_{\theta_t+\mu\mathbf{v}_t}\|\pi_{\theta_0}) - \text{KL}(\pi_{\theta_t}\|\pi_{\theta_0}) \\ &= \mathbb{E}_{s_h, a_h \sim \pi_{\theta_t+\mu\mathbf{v}_t}} \left[\sum_{h=1}^H \log \frac{\pi_{\theta_t+\mu\mathbf{v}_t}(a_h|s_h)}{\pi_{\theta_0}(a_h|s_h)} \right] - \mathbb{E}_{s_h, a_h \sim \pi_{\theta_t}} \left[\sum_{h=1}^H \log \frac{\pi_{\theta_t}(a_h|s_h)}{\pi_{\theta_0}(a_h|s_h)} \right]. \end{aligned}$$

This can be achieved by estimating the KL regularization term of both policies, which is conducted by evaluating the action probability of the reference policy π_{θ_0} on the trajectories generated from both policies. With both components, we can construct the zeroth-order gradient for the regularized MDP and conduct gradient descent similar to ZPG and ZBCPG.

Limitations. Both proposed algorithms require a large number of online human queries for accurate gradient estimation to ensure fast convergence to a good policy, both in theory and in experiments, which may limit their practicality. Such limitation is common in all online RLHF algorithms, including online DPO (Guo et al., 2024). On the other hand, online RLHF algorithms are shown to exhibit better performance, which is also observed in our experiments. It converges faster and requires fewer iterations. One possible approach to avoid this limitation is to replace human feedback with AI feedback similar to Guo et al. (2024). Another possible method is to replace the SGD-style update in the current algorithm with a momentum-based optimization scheme to reuse the human queries from previous policy iterations. The necessity of the current level of human queries will be further explored. Another limitation of our algorithms is the lack of strategic exploration, which may either fail to explore hard-to-reach trajectories or produce similar trajectories that are hard to provide preferences for. One possible way to incorporate better exploration practice into our proposed algorithms is to replace the SGD type of policy update with projected SGD, which forces the agent to explore before convergence. Solving both limitations will be our future direction.

C DETAILS OF THE NUMERICAL EXPERIMENT IN SEC. 5

In this section, we describe the GridWorld experiment environment used in our numerical evaluation and the details of the algorithm implementations.

Environment. Our GridWorld environment consists of 5×5 blocks, where the location of each block, i.e., the state space \mathbb{S} , is denoted as $(1, 1)$ to $(5, 5)$. For each block, we flip a fair coin. If the head shows, we will place a random reward on top, and if the tail shows, there will be no reward on this block. The reward, if placed in a block, is randomly sampled from a standard normal distribution, which means we allow a negative reward as punishment. Each episode consists of $H = 20$ steps, and at the start of each episode, an agent is positioned in block $(3, 3)$, i.e., the center of the GridWorld environment. At each step, the agent can choose from four actions that constitute the action space \mathbb{A} : going up, going down, going left, and going right. However, unlike classic GridWorld environments where the agent can perfectly control its movement, in our experiment, we assume the agent is subject to environment disturbances or imperfect control, to model the stochastic

transition and demonstrate the weaknesses of predominant RLHF frameworks such as DPO. We assume at each step, the action taken by the agent may be reversed with probability 0.4. For example, if the agent is in the center and chooses to go up, it will land on the block above the current block with a probability of 0.6 and on the block beneath it with a probability of 0.4. Similarly, if the agent chooses to go left, it will land on the block left next to it with a probability of 0.6 and on the block right next to it with a probability of 0.4. The motivation for imperfect control arises naturally from designing agents for turbulent environments, e.g., for robots operating on airplanes, cars, or ships, where continual wind or bumps may shift the agent’s action. The goal of the agent is to maximize the cumulative reward over the $H = 20$ time horizon, and the interaction is conducted episodically.

Policy Parameterization. For all algorithms implemented in our experiment, we used tabular policy softmax parameterization, i.e., each state-action pair (s, a) is equipped with a parameter $\xi_{s,a}$ and the policy $\pi(a|s)$ of taking action a at state s would follow:

$$\pi(a|s) = \frac{\exp(\xi_{s,a})}{\sum_a \exp(\xi_{s,a})}.$$

Human Feedback. In this experiment, we assume access to a human panel of size $M = 1000$, i.e., a group of human experts, can be recruited and queried at the same time for two trajectories, except for the distributed implementation of ZBCPG, which we will discuss in more detail in the next section. Each human expert, whom we refer to as a panelist, will generate a preference based on the standard Bradley-Terry model over the trajectory rewards. For example, for two trajectories τ_0 and τ_1 , each panelist would provide feedback following the distribution as follows:

$$\mathbb{P}(\tau_1 \succ \tau_0) = \frac{\exp(r(\tau_1))}{\exp(r(\tau_1)) + \exp(r(\tau_0))}, \quad \mathbb{P}(\tau_0 \succ \tau_1) = \frac{\exp(r(\tau_0))}{\exp(r(\tau_1)) + \exp(r(\tau_0))}.$$

C.1 EXPERIMENT COMPARING ZPG AND ZBCPG TO BASELINES

In this section, we describe the implementation of our proposed algorithms and the implementation of baseline algorithms such as DPO, online DPO, and PPO in Fig. 2(a). For all algorithms, we tune the learning rates and perturbation distances. We also ensure that the total number of trajectory samples and the total number of human queries are the same for fair comparison.

ZPG. We implemented ZPG according to Alg. 1 with trim size $\Delta = 0.001$ and batch size $N = 1000$ for $T = 1000$ iterations. After collecting N trajectory pairs for the perturbed policy and the current policy, all panelists were queried to obtain the preference of all trajectory pairs.

ZBCPG. We implemented ZBCPG according to Alg. 2 with the same setup as ZPG with trim size $\Delta = 0.001$ and batch size $N = 1000$ for $T = 1000$ iterations. The coordinate batch size K is chosen to be $5 \times 4 = 20$, i.e., at each time we perturb the parameter $\xi_{s,a}$ associated with all actions for 5 states. After collecting N trajectory pairs for the perturbed policy and the current policy, all panelists were queried to obtain the preference of all trajectory pairs.

RM+PPO. The PPO baseline contains two components: reward model training and policy training. The reward model we consider uses tabular parameterization. To train the reward model, we first used the randomly initialized policy to collect 5×10^5 trajectory pairs, which is half the number of trajectories used in ZPG and ZBCPG. Then, we queried all panelists for the preferences of all trajectory pairs and obtained the population preference probability for all pairs. After that, we used the SGD optimizer to train the reward model for 5 epochs under the loss in Christiano et al. (2017), which also used multiple human queries for each pair of trajectories. For the policy training, we used a standard online PPO with a KL regularization weight being 0.1. We also conducted $T = 1000$ policy iterations, and the agent collects $N = 1000$ trajectories between policy updates for improved training stability. The PPO loss in each iteration is optimized using SGD for 5 epochs.

DPO. To train the DPO agent, we also conducted $T = 1000$ policy iterations, and the agent collects $N = 1000$ trajectory pairs between policy updates. The KL regularization weight in the DPO loss is 0.1. For each pair of trajectories, the agent queried all panelists for preferences and then used the averaged preference probability to replace the single human preference in the vanilla DPO loss, similar to the loss in Christiano et al. (2017), to ensure fair comparison among algorithms. The DPO loss at each iteration is optimized using SGD for 5 epochs.

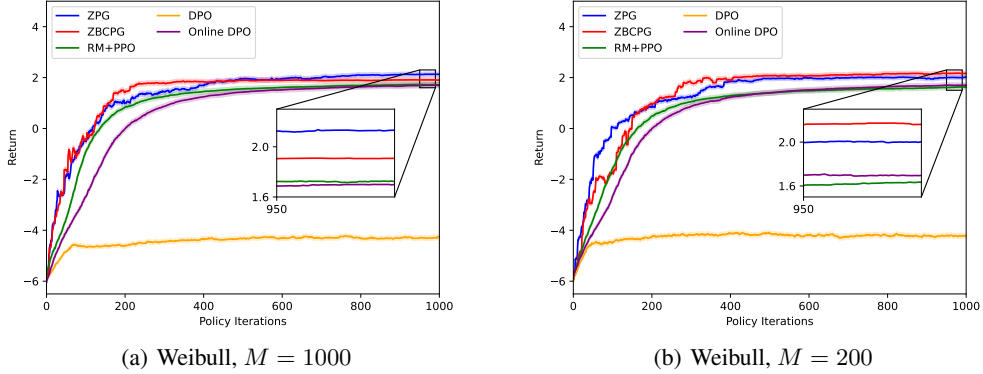


Figure 3: GridWorld with Weibull Feedback: (a) the return of ZPG, ZBCPG, and RLHF baselines under Weibull human feedback with panel size $M = 1000$, and (b) the trajectory return of ZPG, ZBCPG, and RLHF baselines under Weibull human feedback with panel size $M = 200$. All results are averaged over 10^4 repetitions of policy evaluation, and shaded areas indicate confidence intervals.

Online DPO. The implementation of online DPO is almost the same as offline DPO, except that we replaced the reference policy in the KL regularization with the current policy.

C.2 EXPERIMENT COMPARING ZPG TO DISTRIBUTED ZBCPG IMPLEMENTATION

In this section, we describe the distributed implementation of ZBCPG, which produced the results in Fig. 2(b). For distributed ZBCPG, we assume multiple parallel agents can collect trajectories and query humans at the same time, and the level of parallelization is controlled by the number of blocks in the legend of Fig. 2(b). For example, when the number of blocks is 5, we divide the parameters in the policy network into 5 blocks. Then, five perturbed policies are created by perturbing each block individually. These policies are distributed to 5 agents, and each agent will collect $N = 1000$ trajectories simultaneously. However, to ensure fairness in the human panel query, we also divide the complete 1000 people panel into 5 sub-panels that could work in parallel, where each sub-panel includes 200 panelists and will assist one agent with human preference queries. Therefore, the parallelized version of ZBCPG only has the advantage of sampling multiple trajectories at the same time. The human panel is also divided, so the overall number of human queries is the same. When the number of blocks is 5, we perturbed the parameters associated with all actions for 5 states at the same time, so each block contains $5 \times 4 = 20$ parameters. When the number of blocks is 25, we perturbed parameters associated with all actions for a single state at the same time, and each block contains 4 parameters. When the number of blocks is 1, we perturbed all parameters with Rademacher noise. The implementation of ZPG and other parameters is the same as in the previous section.

D ADDITIONAL NUMERICAL EXPERIMENT RESULTS

In this section, we conduct additional numerical experiments to evaluate the impact of the preference model link functions on our proposed algorithms and the baselines, as well as the influence of the size M of the human evaluator panel.

D.1 WEIBULL HUMAN FEEDBACK

For the first aspect, we consider a Weibull preference feedback mechanism (Train, 2009) which is anti-asymmetric between the two trajectories being compared, i.e., $\sigma(x) \neq 1 - \sigma(-x)$. For example, if two trajectories (τ_0, τ_1) are compared by a panelist with the Weibull preference model, it would

Algorithm	ZPG (Ours)	ZBCPG (Ours)	RM+PPO	DPO	Online DPO
Return	2.13 \pm 0.09	1.90 \pm 0.09	1.72 \pm 0.09	-4.25 \pm 0.09	1.69 \pm 0.09

Table 2: Last Iterate Policy Average Return under Weibull Feedback with $M = 1000$.

prefer the trajectory τ_1 with probability:

$$\mathbb{P}(\tau_1 \succ \tau_0) = \exp_2(-\exp_2(r(\tau_0) - r(\tau_1))).$$

This preference model would capture the bias effect presented in the revealing sequence to the human evaluators. Namely, human experts would unintentionally find it easier to distinguish if the better trajectory is presented first. The implementation of all algorithms is the same as the implementation in Sec. 5 except that we use the Weibull link function in both ZPG and ZBCPG. The results are shown in Fig. 3(a) and Tab. 2. It can be seen that both our algorithms perform much better than the baselines in terms of convergence rate and the quality of the last iterate policy. Even though the trend is similar to the Bradley-Terry feedback results shown in Sec. 5, the performance gap is slightly larger due to the additional model mismatch factor. Both DPO and Online DPO are designed explicitly under the Bradley-Terry model, which is inaccurate for the ground-truth Weibull feedback, and the same applies to the reward model training step in the PPO baseline. However, the performance is not significantly degrading, which implies the Weibull link function is not fundamentally different than the Bradley-Terry model and allows these algorithms to have a compelling performance still. It is also shown that our algorithms are more flexible with the different human preference models and deliver better quality.

D.2 INFLUENCE OF PANEL SIZE

Algorithm	ZPG (Ours)	ZBCPG (Ours)	RM+PPO	DPO	Online DPO
Return	2.02 \pm 0.08	2.16 \pm 0.09	1.63 \pm 0.09	-4.22 \pm 0.10	1.69 \pm 0.09

Table 3: Last Iterate Policy Average Return under Weibull Feedback with $M = 200$.

For the second aspect, i.e., to study the influence of panel size M on the quality of the learned policy, and to compare with our theoretical results presented in Theorem. 1 and Theorem 2, we conduct the same experiment with Weibull preference feedback with the same setup as the previous subsection where we only use $M = 200$ human evaluators for each trajectory pair, which is smaller than the $\mathcal{O}(H^2)$ guarantee required by theory. The results are shown in Fig. 3(b) and Tab. 3. It can be seen that even with a much smaller human panel, our proposed algorithms can still converge to a local optimal policy which has a better quality than all benchmarks. Comparing the quality of the last iterate policy in Tab. 3 to Tab. 2, it can also be deduced that the final policy found by our proposed algorithm is robust to the size of the human panel M , and the performance with less human evaluators does not significantly degenerate. However, it can also be seen from Fig. 3 that both ZPG and ZBCPG have relatively larger policy quality fluctuation over the training trajectory compared to the three baselines, and this fluctuation increases as the number of human experts M decreases, which poses stability concerns. This is partially because our proposed algorithms do not use KL regularization over the policies to constrain each gradient ascent step, which results in large policy alterations between iterations. This may be mitigated if we apply the KL regularization to our algorithms as discussed in Sec. B.

E PROOF BACKBONE FOR THEOREMS: PREFERENCE ESTIMATION ERROR

In the following sections, we provide proofs for both our main theorems, i.e., Theorem. 1 and Theorem. 2 for ZPG (Algorithm. 1) and ZBCPG (Algorithm. 2) respectively. Both algorithms follow the Lyapunov drift analysis framework, where the key to the final result relies on a tight bound on the drift. At each iteration t and for each trajectory pair, both algorithms estimate the population-level human preference probability $p_{t,n}$ and use it with the preference model in equation 2 to estimate the reward difference between trajectories, which is later averaged to estimate the value function

difference as in equation 3. Before presenting the proofs of the theorems, we first characterize the error incurred by preference estimation in the following Lemma:

Lemma 1 (Concentration of Reward Difference) *Suppose $\Delta = \min\{\sigma(-H), 1 - \sigma(H)\}$, for any trajectory pairs $(\tau_{n,0}, \tau_{n,1})$ that is queried from M human evaluators. We have:*

$$\mathbb{E} |\sigma^{-1}(p_{t,n}) - [r(\tau_{n,1}) - r(\tau_{n,0})]| \leq L \sqrt{\frac{2 \log M}{M}} + \frac{2H}{M^2}; \quad (5)$$

$$\mathbb{E} |\sigma^{-1}(p_{t,n}) - [r(\tau_{n,1}) - r(\tau_{n,0})]|^2 \leq \frac{2L^2 \log M}{M} + \frac{4H^2}{M^2}. \quad (6)$$

This lemma shows that our reward difference estimation from human preference is accurate as long as the number of human evaluators M is large. The proof of this lemma is provided in Sec. E.1

E.1 PROOF OF LEMMA. 1

Consider any iteration t and any trajectory pairs $(\tau_{n,0}, \tau_{n,1})$ generated by the original and perturbed policy in both ZPG and ZBCPG. We first bound the estimation error of population-level human preference $\mathbb{P}(\tau_{n,1} \succ \tau_{n,0})$ given any arbitrary trajectory pair in the following Lemma. 2. To formalize, let \bar{o}_n be the empirical average over the human feedback $o_{n,m}$, i.e.,

$$\bar{o}_n = \sum_{m=1}^M \frac{o_{n,m}}{M}.$$

Naturally, $\mathbb{E}[\bar{o}_n] = \mathbb{E}[o_{n,m}] = \mathbb{P}(\tau_{n,1} \succ \tau_{n,0})$. And we need to characterize this estimation error first before we prove Lemma. 1.

Lemma 2 (Concentration of Preference Probability) *For any $\delta < \frac{1}{4}$ and suppose $\Delta = \min\{\sigma(-H), 1 - \sigma(H)\}$, for any trajectory pairs $(\tau_{n,0}, \tau_{n,1})$ that is queried from M human evaluators, with probability at least $1 - \delta$, we have:*

$$|p_{t,n} - \mathbb{P}(\tau_{n,1} \succ \tau_{n,0})| = |\text{trim}[\bar{o}_n | \Delta] - \mathbb{P}(\tau_{n,1} \succ \tau_{n,0})| \leq \sqrt{\frac{\log \frac{1}{\delta}}{M}},$$

where $o_{n,m}$ is the feedback for the m -th human evaluator for this trajectory pair.

Proof. First, we notice that for two trajectories $\tau_{n,1}$ and $\tau_{n,0}$, the reward difference of the two trajectories is bounded, i.e., $|r(\tau_{n,1}) - r(\tau_{n,0})| \leq H$, which implies the population-level preference probability $\mathbb{P}(\tau_{n,1} \succ \tau_{n,0})$ is also bounded, i.e.,

$$\mathbb{P}(\tau_{n,1} \succ \tau_{n,0}) \in [\sigma(-H), \sigma(H)] \subset [\Delta, 1 - \Delta].$$

Note that \bar{o}_n may not be within $[\Delta, 1 - \Delta]$ due to the finite number of human evaluators. Overall, we may have the following three cases:

1. $\bar{o}_n \in [\Delta, 1 - \Delta]$: we have $|\text{trim}[\bar{o}_n | \Delta] - \mathbb{P}(\tau_{n,1} \succ \tau_{n,0})| = |\bar{o}_n - \mathbb{E}[\bar{o}_n]|$.
2. $\bar{o}_n > 1 - \Delta \geq \mathbb{P}(\tau_{n,1} \succ \tau_{n,0})$: we have
$$|\text{trim}[\bar{o}_n | \Delta] - \mathbb{P}(\tau_{n,1} \succ \tau_{n,0})| = 1 - \Delta - \mathbb{P}(\tau_{n,1} \succ \tau_{n,0}) \leq |\bar{o}_n - \mathbb{E}[\bar{o}_n]|.$$
3. $\bar{o}_n < \Delta \leq \mathbb{P}(\tau_{n,1} \succ \tau_{n,0})$: we have
$$|\text{trim}[\bar{o}_n | \Delta] - \mathbb{P}(\tau_{n,1} \succ \tau_{n,0})| = \mathbb{P}(\tau_{n,1} \succ \tau_{n,0}) - \Delta \leq |\bar{o}_n - \mathbb{E}[\bar{o}_n]|.$$

To summarize, we have $|\text{trim}[\bar{o}_n | \Delta] - \mathbb{P}(\tau_{n,1} \succ \tau_{n,0})| \leq |\bar{o}_n - \mathbb{E}[\bar{o}_n]|$. According to Hoeffding's inequality for a Bernoulli random variable, we have:

$$\begin{aligned} \mathbb{P} \left(|\text{trim}[\bar{o}_n | \Delta] - \mathbb{P}(\tau_{n,1} \succ \tau_{n,0})| > \sqrt{\frac{\log \frac{1}{\delta}}{M}} \right) &\leq \mathbb{P} \left(|\bar{o}_n - \mathbb{E}[\bar{o}_n]| > \sqrt{\frac{\log \frac{1}{\delta}}{M}} \right) \\ &\leq \exp \left(-2M \frac{\log \frac{1}{\delta}}{M} \right) \leq \delta, \end{aligned}$$

which concludes the proof. \blacksquare

Next, we are ready to bound the error between the reward difference estimator after plugging in the preference probability estimator $p_{t,n}$ to the inverse link function σ^{-1} in Lemma. 1. We first prove equation 5 and then prove equation 6. Let \mathcal{E}_n be the concentration event that is the event in Lemma. 2 holds. By Lemma. 2, when the concentration event does not hold, we can bound the term in the expectation of the left-hand side as:

$$|\sigma^{-1}(p_{t,n}) - [r(\tau_{n,1}) - r(\tau_{n,0})]| \leq 2H.$$

When the concentration event holds, we first notice that $r(\tau_1) - r(\tau_0) = \sigma^{-1}(\mathbb{P}(\tau_{n,1} \succ \tau_{n,0}))$. By definition, both $p_{t,n}$ and $\mathbb{P}(\tau_{n,1} \succ \tau_{n,0})$ belongs to the interval $[\Delta, 1 - \Delta]$. This allows us to use assumption 2 and we have:

$$\begin{aligned} |\sigma^{-1}(p_{t,n}) - [r(\tau_{n,1}) - r(\tau_{n,0})]| &= |\sigma^{-1}(p_{t,n}) - \sigma^{-1}(\mathbb{P}(\tau_{n,1} \succ \tau_{n,0}))| \\ &\leq L |p_{t,n} - \mathbb{P}(\tau_{n,1} \succ \tau_{n,0})| \\ &\leq L \sqrt{\frac{\log \frac{1}{\delta}}{M}}, \end{aligned}$$

where the last inequality uses the smoothness of the inverse link function, i.e., Lemma. 2. Combining both cases and let $\delta = 1/M^2$, we have:

$$\mathbb{E} |\sigma^{-1}(p_{t,n}) - [r(\tau_{n,1}) - r(\tau_{n,0})]| \leq \mathbb{P}(\mathcal{E}_n^c) L \sqrt{\frac{\log \frac{1}{\delta}}{M}} + \mathbb{P}(\mathcal{E}_n) 2H \leq L \sqrt{\frac{2 \log M}{M}} + \frac{2H}{M^2},$$

which concludes the proof of the first inequality. For the second inequality, we apply a similar procedure. Suppose the concentration event \mathcal{E}_n does not hold, then we can bound the term as:

$$|\sigma^{-1}(p_{t,n}) - [r(\tau_{n,1}) - r(\tau_{n,0})]|^2 \leq 4H^2.$$

This is because the inverse link function is bounded after applying the trim operator to the preference probability estimation. When the concentration holds, we also employ the continuity of the inverse link function as:

$$|\sigma^{-1}(p_{t,n}) - [r(\tau_{n,1}) - r(\tau_{n,0})]|^2 \leq \left(L \sqrt{\frac{\log \frac{1}{\delta}}{M}} \right)^2 = \frac{L^2 \log \frac{1}{\delta}}{M},$$

Combining both cases and let $\delta = 1/M^2$, we have:

$$\mathbb{E} |\sigma^{-1}(p_{t,n}) - [r(\tau_{n,1}) - r(\tau_{n,0})]|^2 \leq \mathbb{P}(\mathcal{E}_n^c) \frac{2L^2 \log M}{M} + \mathbb{P}(\mathcal{E}_n) 4H^2 \leq \frac{2L^2 \log M}{M} + \frac{4H^2}{M^2},$$

which concludes the proof of both inequalities.

F PROOF OF THEOREM. 1

Recall that we choose the smoothed value function $V_\mu(\pi_\theta)$ as the Lyapunov function, i.e.,

$$V_\mu(\pi_\theta) = \mathbb{E}_{\mathbf{v}'} [V(\pi_{\theta+\mu\mathbf{v}'})],$$

where the vector \mathbf{v}' is sampled from a d -dimensional uniform distribution over a unit ball. We first provide properties of the smoothed function, which is proved in Liu et al. (2018b, Lemma. 1).

Lemma 3 *Suppose \mathbf{v} is sampled from a uniform distribution over a unit sphere and \mathbf{v}' is sampled from a uniform distribution over a unit ball both in d -dimensional space, and Assumption. 3 holds. Then the smoothed value function $V_\mu(\pi_\theta)$ defined above satisfies:*

1. $V_\mu(\pi_\theta)$ is L -smooth and satisfies:

$$\nabla_\theta V_\mu(\pi_\theta) = \mathbb{E}_{\mathbf{v}} \left[\frac{d}{d\mu} (V(\pi_{\theta+\mu\mathbf{v}}) - V(\pi_\theta)) \mathbf{v} \right]. \quad (7)$$

2. For any $\theta \in \mathbb{R}^d$, the function value difference satisfies:

$$|V_\mu(\pi_\theta) - V(\pi_\theta)| \leq \frac{L\mu^2}{2}. \quad (8)$$

3. For any $\theta \in \mathbb{R}^d$, the gradient difference satisfies:

$$\|\nabla_\theta V_\mu(\pi_\theta) - \nabla_\theta V(\pi_\theta)\|_2 \leq \frac{\mu L d}{2}. \quad (9)$$

4. For any $\theta \in \mathbb{R}^d$, the gradient noise satisfies:

$$\mathbb{E}_v \left[\left\| \frac{d}{\mu} (V(\pi_{\theta+\mu v}) - V(\pi_\theta)) v \right\|_2^2 \right] \leq 2d \|\nabla_\theta V(\pi_\theta)\|_2^2 + \frac{\mu^2 L^2 d^2}{2}. \quad (10)$$

Now, we are ready to prove Theorem. 1. We choose $V_\mu(\pi_\theta)$ to be the Lyapunov function, where v is sampled from a unit ball. We first analyze its drift, from the smoothness of $V_\mu(\pi_\theta)$ by Lemma. 3, we have the following upper bound:

$$\begin{aligned} V_\mu(\pi_{\theta_t}) - V_\mu(\pi_{\theta_{t+1}}) &\leq \langle -\nabla_\theta V_\mu(\pi_{\theta_t}), \theta_{t+1} - \theta_t \rangle + \frac{L}{2} \|\theta_{t+1} - \theta_t\|_2^2 \\ &= -\alpha \langle \nabla_\theta V_\mu(\pi_{\theta_t}), \hat{g}_t \rangle + \frac{L\alpha^2}{2} \|\hat{g}_t\|_2^2 \\ &= -\alpha \|\nabla_\theta V_\mu(\pi_{\theta_t})\|_2^2 + \alpha \langle \nabla_\theta V_\mu(\pi_{\theta_t}), \nabla_\theta V_\mu(\pi_{\theta_t}) - \hat{g}_t \rangle + \frac{L\alpha^2}{2} \|\hat{g}_t\|_2^2. \end{aligned}$$

We bound the three terms above one by one, the first term can be bounded using equation 7 from Lemma. 3. We first have:

$$\begin{aligned} \|\nabla_\theta V(\pi_{\theta_t})\|_2^2 &= \|\nabla_\theta V(\pi_{\theta_t}) - \nabla_\theta V_\mu(\pi_{\theta_t}) + \nabla_\theta V_\mu(\pi_{\theta_t})\|_2^2 \\ &\leq 2\|\nabla_\theta V_\mu(\pi_{\theta_t}) - \nabla_\theta V(\pi_{\theta_t})\|_2^2 + 2\|\nabla_\theta V_\mu(\pi_{\theta_t})\|_2^2 \\ &\leq 2\|\nabla_\theta V_\mu(\pi_{\theta_t})\|_2^2 + \frac{\mu^2 L^2 d^2}{2}, \end{aligned}$$

where the first inequality uses $(a+b)^2 \leq 2a^2 + 2b^2$ and the second inequality uses equation 9. This implies:

$$\|\nabla_\theta V_\mu(\pi_{\theta_t})\|_2^2 \geq \frac{1}{2} \|\nabla_\theta V(\pi_{\theta_t})\|_2^2 - \frac{\mu^2 L^2 d^2}{4}.$$

And we have a bound on the negative drift term as:

$$-\alpha \|\nabla_\theta V_\mu(\pi_{\theta_t})\|_2^2 \leq -\frac{\alpha}{2} \|\nabla_\theta V(\pi_{\theta_t})\|_2^2 + \frac{\alpha \mu^2 L^2 d^2}{4}.$$

We take a conditional expectation of the drift over the natural filtration \mathcal{F}_t of time t and obtain:

$$\begin{aligned} \mathbb{E} [V_\mu(\pi_{\theta_t}) - V_\mu(\pi_{\theta_{t+1}}) | \mathcal{F}_t] &\leq -\frac{\alpha}{2} \|\nabla_\theta V(\pi_{\theta_t})\|_2^2 + \frac{\alpha \mu^2 L^2 d^2}{4} + \frac{L\alpha^2}{2} \underbrace{\mathbb{E} [\|\hat{g}_t\|_2^2 | \mathcal{F}_t]}_{\text{Var}_t} \\ &\quad + \alpha \underbrace{\langle \nabla_\theta V_\mu(\pi_{\theta_t}), \nabla_\theta V_\mu(\pi_{\theta_t}) - \mathbb{E} [\hat{g}_t | \mathcal{F}_t] \rangle}_{\text{Bias}_t}. \end{aligned} \quad (11)$$

Then, we analyze the two positive drift terms Bias_t for the first-order gradient bias and Var_t for the gradient noise separately and then construct a tight upper bound of the Lyapunov drift. The results are summarized in the following two lemmas, where the proofs are deferred to Sec. F.1 and Sec. F.2:

Lemma 4 For ZPG and let $M \geq 8(H/L)^{\frac{2}{3}}$, conditioned on the information filtration \mathcal{F}_t of any time t , the gradient bias can be upper bounded as follows:

$$\begin{aligned} \text{Bias}_t &= \langle \nabla_\theta V_\mu(\pi_{\theta_t}), \nabla_\theta V_\mu(\pi_{\theta_t}) - \mathbb{E} [\hat{g}_t | \mathcal{F}_t] \rangle \\ &\leq \frac{2dL}{\mu} \sqrt{\frac{\log M}{M}} \|\nabla_\theta V(\pi_{\theta_t})\|_2 + d^2 L^2 \sqrt{\frac{\log M}{M}}. \end{aligned}$$

Lemma 5 For ZPG and let $M \geq 4(H/L)^2$, conditioned on the information filtration \mathcal{F}_t of any time t , the gradient bias can be upper bounded as follows:

$$\text{Var}_t = \mathbb{E} [\|\hat{\mathbf{g}}_t\|_2^2 | \mathcal{F}_t] \leq 6d \|\nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t})\|_2^2 + \frac{3\mu^2 L^2 d^2}{2} + \frac{9d^2 L^2 \log M}{\mu^2 M} + \frac{12d^2 H^2}{\mu^2 N}.$$

Now we use Lemma. 4 and Lemma. 5 and combine the drift bound in equation 11 and obtain:

$$\begin{aligned} & \mathbb{E} [V_{\mu}(\pi_{\boldsymbol{\theta}_t}) - V_{\mu}(\pi_{\boldsymbol{\theta}_{t+1}}) | \mathcal{F}_t] \\ & \leq -\frac{\alpha}{2} \|\nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t})\|_2^2 + \frac{\alpha\mu^2 L^2 d^2}{4} + \alpha \text{Bias}_t + \frac{L\alpha^2}{2} \text{Var}_t \\ & \leq -\frac{\alpha}{2} \|\nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t})\|_2^2 + \frac{\alpha\mu^2 L^2 d^2}{4} + \frac{2\alpha dL}{\mu} \sqrt{\frac{\log M}{M}} \|\nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t})\|_2 + \alpha d^2 L^2 \sqrt{\frac{\log M}{M}} \\ & \quad + 3\alpha^2 dL \|\nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t})\|_2^2 + \alpha^2 \mu^2 L^3 d^2 + \frac{5\alpha^2 d^2 L^3 \log M}{\mu^2 M} + \frac{6\alpha^2 d^2 H^2 L}{\mu^2 N}. \end{aligned}$$

Let $\alpha \leq (12dL)^{-1}$, we can further simplify the drift bound as:

$$\begin{aligned} & \mathbb{E} [V_{\mu}(\pi_{\boldsymbol{\theta}_t}) - V_{\mu}(\pi_{\boldsymbol{\theta}_{t+1}}) | \mathcal{F}_t] \\ & \leq -\frac{\alpha}{2} \|\nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t})\|_2^2 + \frac{\alpha\mu^2 L^2 d^2}{4} + \frac{2\alpha dL}{\mu} \sqrt{\frac{\log M}{M}} \|\nabla_{\boldsymbol{\theta}} V_{\mu}(\pi_{\boldsymbol{\theta}_t})\|_2 \\ & \quad + \alpha d^2 L^2 \sqrt{\frac{\log M}{M}} + \frac{1}{4} \alpha \|\nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t})\|_2^2 + \alpha\mu^2 L^2 d + \frac{5\alpha^2 d^2 L^3 \log M}{\mu^2 M} + \frac{6\alpha^2 d^2 H^2 L}{\mu^2 N} \\ & \leq -\frac{\alpha}{4} \|\nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t})\|_2^2 + \frac{\alpha\mu^2 L^2 d^2}{2} + \frac{2\alpha dL}{\mu} \sqrt{\frac{\log M}{M}} \|\nabla_{\boldsymbol{\theta}} V_{\mu}(\pi_{\boldsymbol{\theta}_t})\|_2 + \alpha d^2 L^2 \sqrt{\frac{\log M}{M}} \\ & \quad + \frac{5\alpha^2 d^2 L^3 \log M}{\mu^2 M} + \frac{6\alpha^2 d^2 H^2 L}{\mu^2 N}, \end{aligned}$$

where the last inequality merges both positive and negative drifts on the square of the gradient together and assumes $d \geq 4$. When $\|\nabla_{\boldsymbol{\theta}} V_{\mu}(\pi_{\boldsymbol{\theta}_t})\|_2$ is large, typically, when we have:

$$\|\nabla_{\boldsymbol{\theta}} V_{\mu}(\pi_{\boldsymbol{\theta}_t})\|_2 \geq \frac{16dL}{\mu} \sqrt{\frac{\log M}{M}},$$

we have the positive first-order drift regarding the gradient norm is bounded by:

$$\frac{2\alpha dL}{\mu} \sqrt{\frac{\log M}{M}} \|\nabla_{\boldsymbol{\theta}} V_{\mu}(\pi_{\boldsymbol{\theta}_t})\|_2 \leq \frac{\alpha}{8} \|\nabla_{\boldsymbol{\theta}} V_{\mu}(\pi_{\boldsymbol{\theta}_t})\|_2^2,$$

and thus can be merged into the negative drift as follows:

$$\begin{aligned} & \mathbb{E} [V_{\mu}(\pi_{\boldsymbol{\theta}_t}) - V_{\mu}(\pi_{\boldsymbol{\theta}_{t+1}}) | \mathcal{F}_t] \\ & \leq -\frac{\alpha}{8} \|\nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t})\|_2^2 + \frac{\alpha\mu^2 L^2 d^2}{2} + \alpha d^2 L^2 \sqrt{\frac{\log M}{M}} + \frac{5\alpha^2 d^2 L^3 \log M}{\mu^2 M} + \frac{6\alpha^2 d^2 H^2 L}{\mu^2 N}. \end{aligned}$$

If on the other hand, the gradient is small, i.e., we have:

$$\|\nabla_{\boldsymbol{\theta}} V_{\mu}(\pi_{\boldsymbol{\theta}_t})\|_2 \leq \frac{16dL}{\mu} \sqrt{\frac{\log M}{M}},$$

and then we can upper bound the first-order drift regarding gradient norm in another way:

$$\frac{2\alpha dL}{\mu} \sqrt{\frac{\log M}{M}} \|\nabla_{\boldsymbol{\theta}} V_{\mu}(\pi_{\boldsymbol{\theta}_t})\|_2 \leq \frac{32\alpha d^2 L^2 \log M}{\mu^2 M}.$$

So in this case, the total drift can be upper bounded as:

$$\begin{aligned}
& \mathbb{E} [V_\mu(\pi_{\theta_t}) - V_\mu(\pi_{\theta_{t+1}}) | \mathcal{F}_t] \\
& \leq -\frac{\alpha}{4} \|\nabla_{\theta} V(\pi_{\theta_t})\|_2^2 + \frac{\alpha\mu^2 L^2 d^2}{2} + \frac{32\alpha d^2 L^2 \log M}{\mu^2 M} + \alpha d^2 L^2 \sqrt{\frac{\log M}{M}} \\
& \quad + \frac{5\alpha^2 d^2 L^3 \log M}{\mu^2 M} + \frac{6\alpha^2 d^2 H^2 L}{\mu^2 N} \\
& \leq -\frac{\alpha}{4} \|\nabla_{\theta} V(\pi_{\theta_t})\|_2^2 + \frac{\alpha\mu^2 L^2 d^2}{2} + \frac{33\alpha d^2 L^2 \log M}{\mu^2 M} + \alpha d^2 L^2 \sqrt{\frac{\log M}{M}} + \frac{6\alpha^2 d^2 H^2 L}{\mu^2 N}.
\end{aligned}$$

The last inequality uses $\alpha \leq (12dL)^{-1}$ and $d \geq 4$. So by taking a maximum of both bounds and taking an expectation, we have:

$$\begin{aligned}
& \mathbb{E} [V_\mu(\pi_{\theta_t})] - \mathbb{E} [V_\mu(\pi_{\theta_{t+1}})] \\
& \leq -\frac{\alpha}{8} \mathbb{E} [\|\nabla_{\theta} V(\pi_{\theta_t})\|_2^2] + \frac{33\alpha d^2 L^2 \log M}{\mu^2 M} + \frac{\alpha\mu^2 L^2 d^2}{2} + \alpha d^2 L^2 \sqrt{\frac{\log M}{M}} + \frac{6\alpha^2 d^2 H^2 L}{\mu^2 N}.
\end{aligned}$$

We then take a telescoping sum, which results in:

$$\begin{aligned}
& \mathbb{E} [V_\mu(\pi_{\theta_0})] - \mathbb{E} [V_\mu(\pi_{\theta_T})] \leq -\frac{\alpha}{8} \sum_{t=0}^{T-1} \mathbb{E} [\|\nabla_{\theta} V(\pi_{\theta_t})\|_2^2] \\
& \quad + \left(\frac{33\alpha d^2 L^2 \log M}{\mu^2 M} + \frac{\alpha\mu^2 L^2 d^2}{2} + \alpha d^2 L^2 \sqrt{\frac{\log M}{M}} + \frac{6\alpha^2 d^2 H^2 L}{\mu^2 N} \right) T.
\end{aligned}$$

We choose μ to balance the terms inside the parentheses. Specifically, let

$$\mu^2 = \max \left\{ 9\sqrt{\frac{\log M}{M}}, \frac{4H}{L\sqrt{dN}} \right\}, \quad \alpha = \frac{1}{12dL}.$$

Then, we have the following inequality on the positive drift:

$$\frac{\alpha\mu^2 L^2 d^2}{2} \geq \frac{33\alpha d^2 L^2 \log M}{\mu^2 M}, \quad \frac{\alpha\mu^2 L^2 d^2}{2} \geq \frac{6\alpha^2 d^2 H^2 L}{\mu^2 N}.$$

So we have the final results in Theorem. 1 as:

$$\begin{aligned}
& \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E} [\|\nabla_{\theta} V(\pi_{\theta_t})\|_2^2] \\
& \leq \frac{8(\mathbb{E} [V_\mu(\pi_{\theta_0})] - \mathbb{E} [V_\mu(\pi_{\theta_T})])}{T\alpha} + 12\mu^2 L^2 d^2 + 8d^2 L^2 \sqrt{\frac{\log M}{M}} \\
& \leq \frac{8(\mathbb{E} [V(\pi_{\theta_0})] - \mathbb{E} [V(\pi_{\theta_T})])}{T\alpha} + \frac{8\mu^2 L}{T\alpha} + 12\mu^2 L^2 d^2 + 8d^2 L^2 \sqrt{\frac{\log M}{M}} \\
& = \mathcal{O} \left(\frac{H L d}{T} + \frac{d^2 L^2 \sqrt{\log M}}{\sqrt{M}} + \frac{H L d \sqrt{d}}{\sqrt{N}} \right).
\end{aligned}$$

F.1 PROOF OF LEMMA. 4

We analyze the first-order gradient bias term Bias_t in the drift of equation 11. Notice that conditioned over \mathcal{F}_t , the randomness only comes from sampling the perturbation direction, sampling trajectories, and obtaining human feedback. By equation 7 and the definition of $\hat{\mathbf{g}}_t$, we have:

$$\begin{aligned}
& \mathbb{E} [\nabla_{\theta} V_\mu(\pi_{\theta_t}) - \hat{\mathbf{g}}_t | \mathcal{F}_t] \\
& = \mathbb{E} \left[\frac{d(V(\pi_{\theta_t + \mu \mathbf{v}_t}) - V(\pi_{\theta_t}))}{\mu} \mathbf{v}_t \middle| \mathcal{F}_t \right] - \mathbb{E} \left[\frac{d \sum_{n=1}^N \sigma^{-1}(p_{t,n})}{\mu N} \mathbf{v}_t \middle| \mathcal{F}_t \right].
\end{aligned}$$

Since each trajectory pair $(\tau_{n,0}, \tau_{n,1})$ is generated independently from other trajectories, we have:

$$\begin{aligned}
\mathbb{E} \left[\frac{d}{\mu} \frac{\sum_{n=1}^N \sigma^{-1}(p_{t,n})}{N} \mathbf{v}_t \middle| \mathcal{F}_t \right] &= \frac{d}{\mu} \mathbb{E} \left[\mathbb{E} \left[\frac{\sum_{n=1}^N \sigma^{-1}(p_{t,n})}{N} \middle| \mathbf{v}_t \right] \mathbf{v}_t \middle| \mathcal{F}_t \right] \\
&= \frac{d}{\mu} \mathbb{E} \left[\mathbb{E} \left[\sigma^{-1}(p_{t,n}) \middle| \mathbf{v}_t \right] \mathbf{v}_t \middle| \mathcal{F}_t \right] \\
&= \frac{d}{\mu} \mathbb{E} \left[\mathbb{E} \left[(r(\tau_{n,1}) - r(\tau_{n,0})) \middle| \mathbf{v}_t \right] \mathbf{v}_t \middle| \mathcal{F}_t \right] \\
&\quad - \frac{d}{\mu} \mathbb{E} \left[\mathbb{E} \left[(\sigma^{-1}(p_{t,n}) - [r(\tau_{n,1}) - r(\tau_{n,0})]) \middle| \mathbf{v}_t \right] \mathbf{v}_t \middle| \mathcal{F}_t \right],
\end{aligned}$$

where the first equality uses the law of total expectation. The second equality uses the independent nature of the trajectories generated by the same policy pair. In the last inequality, we aim to bound the difference between the reward difference estimate from human preference and the ground-truth reward difference, which can be constructed through concentration inequalities. Substitute it back into the bias, we have:

$$\begin{aligned}
\mathbb{E} [\nabla_{\boldsymbol{\theta}} V_{\mu}(\pi_{\boldsymbol{\theta}_t}) - \hat{\mathbf{g}}_t \middle| \mathcal{F}_t] &= \frac{d}{\mu} \mathbb{E} [\mathbb{E} [(V(\pi_{\boldsymbol{\theta}_t + \mu \mathbf{v}_t}) - V(\pi_{\boldsymbol{\theta}_t}) - r(\tau_{n,1}) + r(\tau_{n,0})) \middle| \mathbf{v}_t] \mathbf{v}_t \middle| \mathcal{F}_t] \\
&\quad + \frac{d}{\mu} \mathbb{E} [\mathbb{E} [(\sigma^{-1}(p_{t,n}) - [r(\tau_{n,1}) - r(\tau_{n,0})]) \middle| \mathbf{v}_t] \mathbf{v}_t \middle| \mathcal{F}_t],
\end{aligned}$$

Notice that in the inner expectation of the first term, the randomness comes from sampling trajectories given both policies $\pi_{\boldsymbol{\theta}_t + \mu \mathbf{v}_t}$ and $\pi_{\boldsymbol{\theta}_t}$, we have:

$$\begin{aligned}
&\mathbb{E} [(V(\pi_{\boldsymbol{\theta}_t + \mu \mathbf{v}_t}) - V(\pi_{\boldsymbol{\theta}_t}) - r(\tau_{n,1}) + r(\tau_{n,0})) \middle| \mathbf{v}_t] \\
&= V(\pi_{\boldsymbol{\theta}_t + \mu \mathbf{v}_t}) - \mathbb{E} [r(\tau_{n,1}) \middle| \tau_{n,1} \sim \pi_{\boldsymbol{\theta}_t + \mu \mathbf{v}_t}] + \mathbb{E} [r(\tau_{n,0}) \middle| \tau_{n,0} \sim \pi_{\boldsymbol{\theta}_t}] - V(\pi_{\boldsymbol{\theta}_t}) \\
&= 0.
\end{aligned}$$

The last equality is due to the definition of the value function, where trajectory returns are unbiased estimates of the value functions and the fact that $\tau_{n,1}$ and $\tau_{n,0}$ are generated by the perturbed policy and the original policy independently. Then, the bias drift only comes from the preference estimation error, and we will be able to plug it back and have:

$$\begin{aligned}
\text{Bias}_t &= \langle \nabla_{\boldsymbol{\theta}} V_{\mu}(\pi_{\boldsymbol{\theta}_t}), \nabla_{\boldsymbol{\theta}} V_{\mu}(\pi_{\boldsymbol{\theta}_t}) - \mathbb{E} [\hat{\mathbf{g}}_t \middle| \mathcal{F}_t] \rangle \\
&= \frac{d}{\mu} \mathbb{E} [(\sigma^{-1}(p_{t,n}) - [r(\tau_{n,1}) - r(\tau_{n,0})]) \langle \nabla_{\boldsymbol{\theta}} V_{\mu}(\pi_{\boldsymbol{\theta}_t}), \mathbf{v}_t \rangle \middle| \mathcal{F}_t] \\
&\leq \frac{d}{\mu} \|\nabla_{\boldsymbol{\theta}} V_{\mu}(\pi_{\boldsymbol{\theta}_t})\|_2 \mathbb{E} [\mathbb{E} [|\sigma^{-1}(p_{t,n}) - [r(\tau_{n,1}) - r(\tau_{n,0})]| \middle| \tau_{n,1}, \tau_{n,0}] \middle| \mathcal{F}_t] \\
&\leq \frac{d}{\mu} \|\nabla_{\boldsymbol{\theta}} V_{\mu}(\pi_{\boldsymbol{\theta}_t})\|_2 \left(L \sqrt{\frac{2 \log M}{M}} + \frac{2H}{M^2} \right) \\
&\leq \frac{2dL}{\mu} \|\nabla_{\boldsymbol{\theta}} V_{\mu}(\pi_{\boldsymbol{\theta}_t})\|_2 \sqrt{\frac{\log M}{M}} \\
&\leq \frac{2dL}{\mu} \sqrt{\frac{\log M}{M}} \|\nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t})\|_2 + d^2 L^2 \sqrt{\frac{\log M}{M}},
\end{aligned}$$

where the first inequality uses Cauchy-Schwarz inequality and the second inequality uses Lemma. 1. The second last inequality is obtained by choosing $M \geq 8(H/L)^{\frac{2}{3}}$. The final inequality is obtained by applying the third property of Lemma. 3.

F.2 PROOF OF LEMMA. 5

We aim for the variance term Var_t in the Lyapunov drift in equation 11. We first have an upper bound variance as follows:

$$\begin{aligned}
\text{Var}_t &= \mathbb{E} \left[\left\| \frac{d}{\mu} \frac{\sum_{n=1}^N \sigma^{-1}(p_{t,n})}{N} \mathbf{v}_t \right\|_2^2 \middle| \mathcal{F}_t \right] \\
&\leq 3 \underbrace{\mathbb{E} \left[\left\| \frac{d \sum_{n=1}^N (\sigma^{-1}(p_{t,n}) - [r(\tau_{n,1}) - r(\tau_{n,0})])}{\mu N} \mathbf{v}_t \right\|_2^2 \middle| \mathcal{F}_t \right]}_{\text{Var}_{t,1}} \\
&\quad + 3 \underbrace{\mathbb{E} \left[\left\| \frac{d}{\mu} \left(\frac{\sum_{n=1}^N (r(\tau_{n,1}) - r(\tau_{n,0}))}{N} - (V(\pi_{\theta_t + \mu \mathbf{v}_t}) - V(\pi_{\theta_t})) \right) \mathbf{v}_t \right\|_2^2 \middle| \mathcal{F}_t \right]}_{\text{Var}_{t,2}} \\
&\quad + 3 \underbrace{\mathbb{E} \left[\left\| \frac{d}{\mu} (V(\pi_{\theta_t + \mu \mathbf{v}_t}) - V(\pi_{\theta_t})) \mathbf{v}_t \right\|_2^2 \middle| \mathcal{F}_t \right]}_{\text{Var}_{t,3}}.
\end{aligned}$$

The first term comes from the human preference estimation error, which can be bounded using equation 6 from Lemma. 1 as follows:

$$\begin{aligned}
\text{Var}_{t,1} &= \mathbb{E} \left[\left\| \frac{d \sum_{n=1}^N (\sigma^{-1}(p_{t,n}) - [r(\tau_{n,1}) - r(\tau_{n,0})])}{\mu N} \mathbf{v}_t \right\|_2^2 \middle| \mathcal{F}_t \right] \\
&= \frac{d^2}{\mu^2 N^2} \mathbb{E} \left[\left\| \sum_{n=1}^N (\sigma^{-1}(p_{t,n}) - [r(\tau_{n,1}) - r(\tau_{n,0})]) \right\|_2^2 \right] \\
&\leq \frac{d^2}{\mu^2 N} \sum_{n=1}^N \mathbb{E} \left[|\sigma^{-1}(p_{t,n}) - [r(\tau_{n,1}) - r(\tau_{n,0})]|^2 \right] \\
&\leq \frac{d^2}{\mu^2} \left(\frac{2L^2 \log M}{M} + \frac{4H^2}{M^2} \right) \\
&\leq \frac{3d^2 L^2 \log M}{\mu^2 M},
\end{aligned}$$

where the first inequality uses Cauchy-Schwarz inequality, and the second inequality uses Lemma. 1. The final inequality is true by choosing $M \geq 4(H/L)^2$. The second term comes from using the empirical average reward to estimate the value function, similar to REINFORCE. So we have:

$$\begin{aligned}
\text{Var}_{t,2} &= \mathbb{E} \left[\left\| \frac{d}{\mu} \left(\frac{\sum_{n=1}^N (r(\tau_{n,1}) - r(\tau_{n,0}))}{N} - (V(\pi_{\theta_t + \mu \mathbf{v}_t}) - V(\pi_{\theta_t})) \right) \mathbf{v}_t \right\|_2^2 \middle| \mathcal{F}_t \right] \\
&= \frac{d^2}{\mu^2 N^2} \mathbb{E} \left[\left\| \sum_{n=1}^N \underbrace{\left((r(\tau_{n,1}) - r(\tau_{n,0})) - (V(\pi_{\theta_t + \mu \mathbf{v}_t}) - V(\pi_{\theta_t})) \right)}_{\mathbf{E}_{t,n}} \right\|_2^2 \middle| \mathcal{F}_t \right] \\
&= \frac{d^2}{\mu^2 N^2} \mathbb{E} \left[\left\| \sum_{n=1}^N \mathbf{E}_{t,n} \right\|_2^2 \middle| \mathcal{F}_t \right],
\end{aligned}$$

where the last equality is because \mathbf{v}_t is sampled from a unit ball. We open up the square as:

$$\begin{aligned}\mathbb{E} \left[\left| \sum_{n=1}^N \mathbf{E}_{t,n} \right|_2^2 \middle| \mathcal{F}_t \right] &= \sum_{n=1}^N \mathbb{E} \left[|\mathbf{E}_{t,n}|_2^2 \middle| \mathcal{F}_t \right] + \sum_{i \neq j} \mathbb{E} [\mathbb{E}[\mathbf{E}_{t,i} \mathbf{E}_{t,j} | \mathbf{v}_t] | \mathcal{F}_t] \\ &= \sum_{n=1}^N \mathbb{E} \left[|\mathbf{E}_{t,n}|_2^2 \middle| \mathcal{F}_t \right] + \sum_{i \neq j} \mathbb{E} [\mathbb{E}[\mathbf{E}_{t,i} | \mathbf{v}_t] | \mathcal{F}_t] \mathbb{E} [\mathbb{E}[\mathbf{E}_{t,j} | \mathbf{v}_t] | \mathcal{F}_t] \\ &= \sum_{n=1}^N \mathbb{E} \left[|\mathbf{E}_{t,n}|_2^2 \middle| \mathcal{F}_t \right].\end{aligned}$$

where the second inequality uses the independence between trajectories generated at the same step, and the last equality is due to $\mathbb{E}[(r(\tau_{n,1}) - r(\tau_{n,0})) - (V(\pi_{\theta_t + \mu \mathbf{v}_t}) - V(\pi_{\theta_t}))] = 0$ for any fixed perturbation direction \mathbf{v}_t from the definition of value function. So we have:

$$\begin{aligned}\text{Var}_{t,2} &= \frac{d^2}{\mu^2 N^2} \sum_{n=1}^N \mathbb{E} \left[|\mathbf{E}_{t,n}|_2^2 \middle| \mathcal{F}_t \right] \\ &= \frac{d^2}{\mu^2 N^2} \sum_{n=1}^N \mathbb{E} \left[|(r(\tau_{n,1}) - r(\tau_{n,0})) - (V(\pi_{\theta_t + \mu \mathbf{v}_t}) - V(\pi_{\theta_t}))|^2 \middle| \mathcal{F}_t \right] \\ &\leq \frac{4d^2 H^2}{\mu^2 N},\end{aligned}$$

where the last inequality uses the fact that both the difference of reward and the difference of value function are within $[-H, H]$. The last term to bound Var_t can be obtained from the property of the smoothed function in Lemma. 3 as:

$$\text{Var}_{t,3} = \mathbb{E} \left[\left\| \frac{d}{\mu} (V(\pi_{\theta_t + \mu \mathbf{v}_t}) - V(\pi_{\theta_t})) \mathbf{v}_t \right\|_2^2 \middle| \mathcal{F}_t \right] \leq 2d \|\nabla_{\theta} V(\pi_{\theta_t})\|_2^2 + \frac{\mu^2 L^2 d^2}{2}.$$

So, combining three terms, the variance drift can be bounded as:

$$\text{Var}_t \leq 6d \|\nabla_{\theta} V(\pi_{\theta_t})\|_2^2 + \frac{3\mu^2 L^2 d^2}{2} + \frac{9d^2 L^2 \log M}{\mu^2 M} + \frac{12d^2 H^2}{\mu^2 N}.$$

G PROOF OF THEOREM. 2

Due to the choice of the generation mechanism of the perturbation vector \mathbf{v}_t , we cannot construct an explicit expression for the smoothed value function $V_{\mu}(\pi_{\theta})$ in this setting, whose gradient is somewhat "unbiased" if the correct reward function is known. In this case, we use the original value function as the Lyapunov function and perform drift analysis as follows:

$$\begin{aligned}V(\pi_{\theta_t}) - V(\pi_{\theta_{t+1}}) &\leq \langle -\nabla_{\theta} V(\pi_{\theta_t}), \theta_{t+1} - \theta_t \rangle + \frac{L}{2} \|\theta_{t+1} - \theta_t\|_2^2 \\ &= -\alpha \langle \nabla_{\theta} V(\pi_{\theta_t}), \hat{\mathbf{g}}_t \rangle + \frac{L\alpha^2}{2} \|\hat{\mathbf{g}}_t\|_2^2 \\ &= -\alpha \|\nabla_{\theta} V(\pi_{\theta_t})\|_2^2 + \alpha \langle \nabla_{\theta} V(\pi_{\theta_t}), \nabla_{\theta} V(\pi_{\theta_t}) - \hat{\mathbf{g}}_t \rangle + \frac{L\alpha^2}{2} \|\hat{\mathbf{g}}_t\|_2^2.\end{aligned}$$

We also take a conditional expectation of the drift over the natural filtration \mathcal{F}_t of time t and obtain:

$$\begin{aligned}&\mathbb{E} [V(\pi_{\theta_t}) - V(\pi_{\theta_{t+1}}) | \mathcal{F}_t] \\ &\leq -\frac{\alpha}{2} \|\nabla_{\theta} V(\pi_{\theta_t})\|_2^2 + \underbrace{\frac{L\alpha^2}{2} \mathbb{E} [\|\hat{\mathbf{g}}_t\|_2^2 | \mathcal{F}_t]}_{\text{Var}_t} + \underbrace{\alpha \langle \nabla_{\theta} V(\pi_{\theta_t}), \nabla_{\theta} V(\pi_{\theta_t}) - \mathbb{E} [\hat{\mathbf{g}}_t | \mathcal{F}_t] \rangle}_{\text{Bias}_t}.\end{aligned}\quad (12)$$

Then, we analyze the two positive drift terms Bias_t for the first-order gradient bias and Var_t for the gradient noise separately and then construct a tight upper bound of the Lyapunov drift. The results are summarized in the following two lemmas, where the proofs are deferred to Sec. G.1 and Sec. G.2:

Lemma 6 For ZBCPG and let $M \geq 8(H/L)^{\frac{2}{3}}$, conditioned on the information filtration \mathcal{F}_t of any time t , the gradient bias can be upper bounded as follows:

$$\text{Bias}_t = \langle \nabla_{\theta} V(\pi_{\theta_t}), \nabla_{\theta} V(\pi_{\theta_t}) - \mathbb{E}[\hat{g}_t | \mathcal{F}_t] \rangle \leq \left(\frac{\mu L d}{2} + \frac{2dL}{\mu} \sqrt{\frac{\log M}{M}} \right) \|\nabla_{\theta} V(\pi_{\theta_t})\|_2.$$

Lemma 7 For ZBCPG and let $M \geq 4(H/L)^2$, conditioned on the information filtration \mathcal{F}_t of any time t , the gradient bias can be upper bounded as follows:

$$\text{Var}_t = \mathbb{E}[\|\hat{g}_t\|_2^2 | \mathcal{F}_t] \leq 6d \|\nabla_{\theta} V(\pi_{\theta_t})\|_2^2 + \frac{3\mu^2 L^2 d^2}{2} + \frac{9d^2 L^2 \log M}{\mu^2 M} + \frac{12d^2 H^2}{\mu^2 N}.$$

Now we use Lemma. 6 and Lemma. 7 and combine the drift bound in equation 12 and obtain:

$$\begin{aligned} & \mathbb{E}[V(\pi_{\theta_t}) - V(\pi_{\theta_{t+1}}) | \mathcal{F}_t] \\ & \leq -\frac{\alpha}{2} \|\nabla_{\theta} V(\pi_{\theta_t})\|_2^2 + \alpha \text{Bias}_t + \frac{L\alpha^2}{2} \text{Var}_t \\ & \leq -\frac{\alpha}{2} \|\nabla_{\theta} V(\pi_{\theta_t})\|_2^2 + \alpha \left(\frac{\mu L d}{2} + \frac{2dL}{\mu} \sqrt{\frac{\log M}{M}} \right) \|\nabla_{\theta} V(\pi_{\theta_t})\|_2 \\ & \quad + 3\alpha^2 dL \|\nabla_{\theta} V(\pi_{\theta_t})\|_2^2 + \alpha^2 \mu^2 L^3 d^2 + \frac{5\alpha^2 d^2 L^3 \log M}{\mu^2 M} + \frac{6\alpha^2 d^2 H^2 L}{\mu^2 N}. \end{aligned}$$

Let $\alpha \leq (12dL)^{-1}$, we can further simplify the drift bound as:

$$\begin{aligned} & \mathbb{E}[V(\pi_{\theta_t}) - V(\pi_{\theta_{t+1}}) | \mathcal{F}_t] \\ & \leq -\frac{\alpha}{2} \|\nabla_{\theta} V(\pi_{\theta_t})\|_2^2 + \alpha \left(\frac{\mu L d}{2} + \frac{2dL}{\mu} \sqrt{\frac{\log M}{M}} \right) \|\nabla_{\theta} V(\pi_{\theta_t})\|_2 \\ & \quad + \frac{1}{4} \alpha \|\nabla_{\theta} V(\pi_{\theta_t})\|_2^2 + \alpha^2 \mu^2 L^3 d^2 + \frac{5\alpha^2 d^2 L^3 \log M}{\mu^2 M} + \frac{6\alpha^2 d^2 H^2 L}{\mu^2 N} \\ & \leq -\frac{\alpha}{4} \|\nabla_{\theta} V(\pi_{\theta_t})\|_2^2 + \alpha \left(\frac{\mu L d}{2} + \frac{2dL}{\mu} \sqrt{\frac{\log M}{M}} \right) \|\nabla_{\theta} V(\pi_{\theta_t})\|_2 \\ & \quad + \alpha^2 \mu^2 L^3 d^2 + \frac{5\alpha^2 d^2 L^3 \log M}{\mu^2 M} + \frac{6\alpha^2 d^2 H^2 L}{\mu^2 N}, \end{aligned}$$

where the last inequality merges both positive and negative drifts on the square of the gradient together and assumes $d \geq 4$. When $\|\nabla_{\theta} V(\pi_{\theta_t})\|_2$ is large, typically, when we have:

$$\|\nabla_{\theta} V(\pi_{\theta_t})\|_2 \geq 8 \left(\frac{\mu L d}{2} + \frac{2dL}{\mu} \sqrt{\frac{\log M}{M}} \right),$$

we have the positive first-order drift, regarding the gradient norm is bounded by:

$$\left(\frac{\mu L d}{2} + \frac{2dL}{\mu} \sqrt{\frac{\log M}{M}} \right) \|\nabla_{\theta} V(\pi_{\theta_t})\|_2 \leq \frac{1}{8} \|\nabla_{\theta} V(\pi_{\theta_t})\|_2^2,$$

and thus can be merged into the negative drift as follows:

$$\begin{aligned} & \mathbb{E}[V(\pi_{\theta_t}) - V(\pi_{\theta_{t+1}}) | \mathcal{F}_t] \\ & \leq -\frac{\alpha}{8} \|\nabla_{\theta} V(\pi_{\theta_t})\|_2^2 + \alpha^2 \mu^2 L^3 d^2 + \frac{5\alpha^2 d^2 L^3 \log M}{\mu^2 M} + \frac{6\alpha^2 d^2 H^2 L}{\mu^2 N}. \end{aligned}$$

If on the other hand, the gradient is small, i.e., when we have:

$$\|\nabla_{\theta} V(\pi_{\theta_t})\|_2 \leq 8 \left(\frac{\mu L d}{2} + \frac{2dL}{\mu} \sqrt{\frac{\log M}{M}} \right),$$

and then we can upper bound the first-order drift regarding the gradient norm in another way:

$$\begin{aligned} \left(\frac{\mu L d}{2} + \frac{2dL}{\mu} \sqrt{\frac{\log M}{M}} \right) \|\nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t})\|_2 &\leq 8 \left(\frac{\mu L d}{2} + \frac{2dL}{\mu} \sqrt{\frac{\log M}{M}} \right)^2 \\ &\leq 4\mu^2 L^2 d^2 + \frac{64d^2 L^2 \log M}{\mu^2 M}. \end{aligned}$$

So in this case, the total drift can be upper bounded as:

$$\begin{aligned} &\mathbb{E} [V(\pi_{\boldsymbol{\theta}_t}) - V(\pi_{\boldsymbol{\theta}_{t+1}}) | \mathcal{F}_t] \\ &\leq -\frac{\alpha}{4} \|\nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t})\|_2^2 + 4\alpha\mu^2 L^2 d^2 + \alpha^2 \mu^2 L^3 d^2 + \frac{64\alpha d^2 L^2 \log M}{\mu^2 M} \\ &\quad + \frac{5\alpha^2 d^2 L^3 \log M}{\mu^2 M} + \frac{6\alpha^2 d^2 H^2 L}{\mu^2 N} \\ &\leq -\frac{\alpha}{4} \|\nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t})\|_2^2 + 5\alpha\mu^2 L^2 d^2 + \frac{64\alpha d^2 L^2 \log M}{\mu^2 M} + \frac{5\alpha^2 d^2 L^3 \log M}{\mu^2 M} + \frac{6\alpha^2 d^2 H^2 L}{\mu^2 N}. \end{aligned}$$

The last inequality uses $\alpha \leq (12dL)^{-1}$ and $d \geq 4$. So, combining both bounds and taking an expectation, we have:

$$\begin{aligned} &\mathbb{E} [V(\pi_{\boldsymbol{\theta}_t}) - V(\pi_{\boldsymbol{\theta}_{t+1}}) | \mathcal{F}_t] \\ &\leq -\frac{\alpha}{8} \mathbb{E} [\|\nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t})\|_2^2] + 5\alpha\mu^2 L^2 d^2 + \frac{5\alpha^2 d^2 L^3 \log M}{\mu^2 M} + \frac{64\alpha d^2 L^2 \log M}{\mu^2 M} + \frac{6\alpha^2 d^2 H^2 L}{\mu^2 N} \\ &\leq -\frac{\alpha}{8} \mathbb{E} [\|\nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t})\|_2^2] + 5\alpha\mu^2 L^2 d^2 + \frac{65\alpha d^2 L^2 \log M}{\mu^2 M} + \frac{6\alpha^2 d^2 H^2 L}{\mu^2 N}, \end{aligned}$$

where we use $\alpha \leq (12dL)^{-1}$ again in the last inequality. We then take a telescoping sum, which results in the following:

$$\begin{aligned} &\mathbb{E} [V(\pi_{\boldsymbol{\theta}_0})] - \mathbb{E} [V(\pi_{\boldsymbol{\theta}_T})] \\ &\leq -\frac{\alpha}{8} \sum_{t=0}^{T-1} \mathbb{E} [\|\nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t})\|_2^2] + \left(5\alpha\mu^2 L^2 d^2 + \frac{65\alpha d^2 L^2 \log M}{\mu^2 M} + \frac{6\alpha^2 d^2 H^2 L}{\mu^2 N} \right) T. \end{aligned}$$

We choose μ to balance the terms inside the parentheses. Specifically, let

$$\mu^2 = \max \left\{ 4\sqrt{\frac{\log M}{M}}, \frac{H}{3L\sqrt{dN}} \right\}, \quad \alpha = \frac{1}{12dL}.$$

Then, we have the following inequality on the positive drift:

$$5\alpha\mu^2 L^2 d^2 \geq \frac{65\alpha d^2 L^2 \log M}{\mu^2 M}, \quad 5\alpha\mu^2 L^2 d^2 \geq \frac{6\alpha^2 d^2 H^2 L}{\mu^2 N}.$$

So we have the final results in Theorem. 2 as:

$$\begin{aligned} \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E} [\|\nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t})\|_2^2] &\leq \frac{8(\mathbb{E} [V(\pi_{\boldsymbol{\theta}_0})] - \mathbb{E} [V(\pi_{\boldsymbol{\theta}_T})])}{T\alpha} + 120\mu^2 L^2 d^2 \\ &= \mathcal{O} \left(\frac{H L d}{T} + \frac{d^2 L^2 \sqrt{\log M}}{\sqrt{M}} + \frac{H L d \sqrt{d}}{\sqrt{N}} \right). \end{aligned}$$

G.1 PROOF OF LEMMA. 6

We analyze the first-order gradient bias term Bias_t in the drift of equation 12. Notice that conditioned over \mathcal{F}_t , the randomness only comes from sampling the perturbation coordinate set i_t , the

perturbation direction λ_t , sampling trajectories, and obtaining human feedback. By equation 7 and the definition of \hat{g}_t , we have:

$$\begin{aligned}
& \nabla_{\theta} V(\pi_{\theta_t}) - \mathbb{E} [\hat{g}_t | \mathcal{F}_t] \\
&= \nabla_{\theta} V(\pi_{\theta_t}) - \mathbb{E} \left[\mathbb{E} \left[\frac{d \sum_{n=1}^N \sigma^{-1}(p_{t,n})}{\mu N} \middle| \mathbf{v}_t \right] \mathbf{v}_t \middle| \mathcal{F}_t \right] \\
&= \nabla_{\theta} V(\pi_{\theta_t}) - \frac{d}{\mu} \mathbb{E} [\mathbb{E} [\sigma^{-1}(p_{t,n}) | \mathbf{v}_t] \mathbf{v}_t | \mathcal{F}_t] \\
&= \underbrace{\nabla_{\theta} V(\pi_{\theta_t}) - \mathbb{E} \left[\frac{d}{\mu} (V(\pi_{\theta_t + \mu \mathbf{v}_t}) - V(\pi_{\theta_t})) \mathbf{v}_t \middle| \mathcal{F}_t \right]}_{\text{Bias}_{t,1}} \\
&\quad + \frac{d}{\mu} \mathbb{E} \left[\underbrace{[(V(\pi_{\theta_t + \mu \mathbf{v}_t}) - V(\pi_{\theta_t})) - \mathbb{E} [r(\tau_{n,1}) - r(\tau_{n,0}) | \mathbf{v}_t]]}_{=0} \mathbf{v}_t \middle| \mathcal{F}_t \right] \\
&\quad + \underbrace{\frac{d}{\mu} \mathbb{E} [\mathbb{E} [(\sigma^{-1}(p_{t,n}) - [r(\tau_{n,1}) - r(\tau_{n,0})]) | \tau_{n,1}, \tau_{n,0}] \mathbf{v}_t | \mathcal{F}_t]}_{\text{Bias}_{t,2}},
\end{aligned}$$

where the second inequality uses the fact that conditioned on \mathbf{v}_t , each trajectory pair $(\tau_{n,1}, \tau_{n,0})$ is independently generated from policies $(\pi_{\theta_t + \mu \mathbf{v}_t}, \pi_{\theta_t})$. The last inequality is since conditioned on the policy $(\pi_{\theta_t + \mu \mathbf{v}_t}, \pi_{\theta_t})$, the expectation of each reward trajectory is the value function of the policy since $(\tau_{n,1}, \tau_{n,0})$ is independently generated from policies $(\pi_{\theta_t + \mu \mathbf{v}_t}, \pi_{\theta_t})$. Therefore, the Bias_t term is written as follows:

$$\text{Bias}_t = \langle \nabla_{\theta} V(\pi_{\theta_t}), \text{Bias}_{t,1} \rangle + \langle \nabla_{\theta} V(\pi_{\theta_t}), \text{Bias}_{t,2} \rangle.$$

We first analyze the second term, which comes from preference estimation. Similar to the proof of ZPG, we have:

$$\begin{aligned}
& \langle \nabla_{\theta} V(\pi_{\theta_t}), \text{Bias}_{t,2} \rangle \\
&= \frac{d}{\mu} \mathbb{E} [\mathbb{E} [(\sigma^{-1}(p_{t,n}) - [r(\tau_{n,1}) - r(\tau_{n,0})]) | \tau_{n,1}, \tau_{n,0}] \langle \nabla_{\theta} V(\pi_{\theta_t}), \mathbf{v}_t \rangle | \mathcal{F}_t] \\
&\leq \frac{d}{\mu} \|\nabla_{\theta} V(\pi_{\theta_t})\|_2 \mathbb{E} [\mathbb{E} [|\sigma^{-1}(p_{t,n}) - [r(\tau_{n,1}) - r(\tau_{n,0})]| | \tau_{n,1}, \tau_{n,0}] | \mathcal{F}_t] \\
&\leq \frac{d}{\mu} \|\nabla_{\theta} V(\pi_{\theta_t})\|_2 \left(L \sqrt{\frac{2 \log M}{M}} + \frac{2H}{M^2} \right) \\
&\leq \frac{2dL}{\mu} \sqrt{\frac{\log M}{M}} \|\nabla_{\theta} V(\pi_{\theta_t})\|_2,
\end{aligned}$$

where the first inequality uses Cauchy-Schwarz inequality and $\|\mathbf{v}_t\|_2 = 1$ and the second inequality uses Lemma. 1. The last inequality is obtained by choosing $M \geq 8(H/L)^{\frac{2}{3}}$. Then, we need to deal with the first-term inner product. We first make the following claim:

Lemma 8 *For any iteration t , the following is true:*

$$\frac{1}{d} \nabla_{\theta} V(\pi_{\theta_t}) = \mathbb{E}_{\mathbf{i}_t, \lambda_t} [\langle \nabla_{\theta} V(\pi_{\theta_t}), \mathbf{v}_t \rangle \mathbf{v}_t].$$

where the subscript indicates the randomness only comes from \mathbf{i}_t and λ_t .

Proof. To prove this equality, we start from the right-hand side:

$$\begin{aligned}\mathbb{E}_{\mathbf{i}_t, \boldsymbol{\lambda}_t} [\langle \nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t}), \mathbf{v}_t \rangle \mathbf{v}_t] &= \frac{1}{K} \mathbb{E}_{\mathbf{i}_t, \boldsymbol{\lambda}_t} \left[\left\langle \nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t}), \sum_{j=1}^K \lambda_{i_t, j} \mathbf{e}_{i_t, j} \right\rangle \sum_{l=1}^K \lambda_{i_t, l} \mathbf{e}_{i_t, l} \right] \\ &= \frac{1}{K} \mathbb{E}_{\mathbf{i}_t, \boldsymbol{\lambda}_t} \left[\sum_{j=1}^K \sum_{l=1}^K \langle \nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t}), \lambda_{i_t, j} \mathbf{e}_{i_t, j} \rangle \lambda_{i_t, l} \mathbf{e}_{i_t, l} \right] \\ &= \frac{1}{K} \mathbb{E}_{\mathbf{i}_t} \left[\sum_{j=1}^K \sum_{l=1}^K \langle \nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t}), \mathbf{e}_{i_t, j} \rangle \mathbf{e}_{i_t, l} \mathbb{E}_{\boldsymbol{\lambda}_t} [\lambda_{i_t, j} \lambda_{i_t, l}] \right],\end{aligned}$$

where the first equality is by the definition of \mathbf{v}_t , and the third equality is by the law of total expectation. For $j \neq l$, we can use the independence between different coordinates of $\boldsymbol{\lambda}_t$. Since the expectation of $\boldsymbol{\lambda}_t$ is zero, we have:

$$\mathbb{E}_{\boldsymbol{\lambda}_t} [\lambda_{i_t, j} \lambda_{i_t, l}] = \mathbb{E}_{\boldsymbol{\lambda}_t} [\lambda_{i_t, j}] \mathbb{E}_{\boldsymbol{\lambda}_t} [\lambda_{i_t, l}] = 0.$$

On the other hand, if $j = l$, since $\lambda_{i_t, j}$ is sampled from $\{-1, 1\}$, we have $\lambda_{i_t, j}^2 = 1$. Therefore, the right-hand side can be expressed as:

$$\begin{aligned}\mathbb{E}_{\mathbf{i}_t, \boldsymbol{\lambda}_t} [\langle \nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t}), \mathbf{v}_t \rangle \mathbf{v}_t] &= \frac{1}{K} \mathbb{E}_{\mathbf{i}_t} \left[\sum_{j=1}^K \langle \nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t}), \mathbf{e}_{i_t, j} \rangle \mathbf{e}_{i_t, j} \right] \\ &= \frac{1}{K} \mathbb{E}_{\mathbf{i}_t} \left[\sum_{i=1}^d \mathbb{1}_{i \in \mathbf{i}_t} \langle \nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t}), \mathbf{e}_i \rangle \mathbf{e}_i \right] \\ &= \frac{1}{K} \sum_{i=1}^d \mathbb{P}(i \in \mathbf{i}_t) \langle \nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t}), \mathbf{e}_i \rangle \mathbf{e}_i \\ &= \frac{1}{d} \sum_{i=1}^d \langle \nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t}), \mathbf{e}_i \rangle \mathbf{e}_i \\ &= \frac{1}{d} \nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t}),\end{aligned}$$

where the second last equality is because the probability of each coordinate being chosen by \mathbf{i}_t is exactly K/d . The proof is hence concluded. \blacksquare

With the help of the Lemma above, we can express the $\text{Bias}_{t,1}$ term as follows:

$$\begin{aligned}&\langle \nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t}), \text{Bias}_{t,1} \rangle \\ &= \left\langle \nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t}), \nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t}) - \frac{d}{\mu} \mathbb{E} [(V(\pi_{\boldsymbol{\theta}_t + \mu \mathbf{v}_t}) - V(\pi_{\boldsymbol{\theta}_t})) \mathbf{v}_t | \mathcal{F}_t] \right\rangle \\ &= \frac{d}{\mu} \left\langle \nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t}), \mathbb{E} \left[\frac{\mu}{d} \nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t}) - (V(\pi_{\boldsymbol{\theta}_t + \mu \mathbf{v}_t}) - V(\pi_{\boldsymbol{\theta}_t})) \mathbf{v}_t | \mathcal{F}_t \right] \right\rangle \\ &= -\frac{d}{\mu} \mathbb{E} [(V(\pi_{\boldsymbol{\theta}_t + \mu \mathbf{v}_t}) - V(\pi_{\boldsymbol{\theta}_t}) - \langle \nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t}), \mu \mathbf{v}_t \rangle) \langle \nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t}), \mathbf{v}_t \rangle | \mathcal{F}_t] \\ &\leq \frac{d}{\mu} \mathbb{E} [|V(\pi_{\boldsymbol{\theta}_t + \mu \mathbf{v}_t}) - V(\pi_{\boldsymbol{\theta}_t}) - \langle \nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t}), \mu \mathbf{v}_t \rangle| \|\nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t})\|_2 \|\mathbf{v}_t\|_2 | \mathcal{F}_t].\end{aligned}$$

By the smoothness of the value function V , we have:

$$|V(\pi_{\boldsymbol{\theta}_t + \mu \mathbf{v}_t}) - V(\pi_{\boldsymbol{\theta}_t}) - \langle \nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t}), \mu \mathbf{v}_t \rangle| \leq \frac{\mu^2 L}{2} \|\mathbf{v}_t\|_2 = \frac{\mu^2 L}{2}.$$

So we can substitute into the bias term and obtain:

$$\begin{aligned}\langle \nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t}), \text{Bias}_{t,1} \rangle &\leq \frac{d}{\mu} \mathbb{E} \left[\frac{\mu^2 L}{2} \|\nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t})\|_2 \middle| \mathcal{F}_t \right] \\ &= \frac{\mu L d}{2} \|\nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t})\|_2.\end{aligned}$$

Then, we obtain the upper bound of the gradient bias as follows:

$$\text{Bias}_t = \langle \nabla_{\theta} V(\pi_{\theta_t}), \text{Bias}_{t,1} + \text{Bias}_{t,2} \rangle \leq \left(\frac{\mu L d}{2} + \frac{2dL}{\mu} \sqrt{\frac{\log M}{M}} \right) \|\nabla_{\theta} V(\pi_{\theta_t})\|_2.$$

G.2 PROOF OF LEMMA. 7

We aim for the variance term Var_t in the Lyapunov drift in equation 12. We first have an upper bound variance as follows:

$$\begin{aligned} \text{Var}_t &= \mathbb{E} \left[\left\| \frac{d}{\mu} \frac{\sum_{n=1}^N \sigma^{-1}(p_{t,n})}{N} \mathbf{v}_t \right\|_2^2 \middle| \mathcal{F}_t \right] \\ &\leq 3 \underbrace{\mathbb{E} \left[\left\| \frac{d \sum_{n=1}^N (\sigma^{-1}(p_{t,n}) - [r(\tau_{n,1}) - r(\tau_{n,0})])}{\mu N} \mathbf{v}_t \right\|_2^2 \middle| \mathcal{F}_t \right]}_{\text{Var}_{t,1}} \\ &\quad + 3 \underbrace{\mathbb{E} \left[\left\| \frac{d}{\mu} \left(\frac{\sum_{n=1}^N (r(\tau_{n,1}) - r(\tau_{n,0}))}{N} - (V(\pi_{\theta_t + \mu \mathbf{v}_t}) - V(\pi_{\theta_t})) \right) \mathbf{v}_t \right\|_2^2 \middle| \mathcal{F}_t \right]}_{\text{Var}_{t,2}} \\ &\quad + 3 \underbrace{\mathbb{E} \left[\left\| \frac{d}{\mu} (V(\pi_{\theta_t + \mu \mathbf{v}_t}) - V(\pi_{\theta_t})) \mathbf{v}_t \right\|_2^2 \middle| \mathcal{F}_t \right]}_{\text{Var}_{t,3}}. \end{aligned}$$

The first term comes from the human preference estimation error, which can be bounded using equation 6 from Lemma. 1 as follows:

$$\begin{aligned} \text{Var}_{t,1} &= \mathbb{E} \left[\left\| \frac{d \sum_{n=1}^N (\sigma^{-1}(p_{t,n}) - [r(\tau_{n,1}) - r(\tau_{n,0})])}{\mu N} \mathbf{v}_t \right\|_2^2 \middle| \mathcal{F}_t \right] \\ &= \frac{d^2}{\mu^2 N^2} \mathbb{E} \left[\left| \sum_{n=1}^N (\sigma^{-1}(p_{t,n}) - [r(\tau_{n,1}) - r(\tau_{n,0})]) \right|^2 \|\mathbf{v}_t\|_2^2 \middle| \mathcal{F}_t \right] \\ &\leq \frac{d^2}{\mu^2 N} \sum_{n=1}^N \mathbb{E} \left[|\sigma^{-1}(p_{t,n}) - [r(\tau_{n,1}) - r(\tau_{n,0})]|^2 \right] \\ &\leq \frac{d^2}{\mu^2} \left(\frac{2L^2 \log M}{M} + \frac{4H^2}{M^2} \right) \\ &\leq \frac{3d^2 L^2 \log M}{\mu^2 M}, \end{aligned}$$

where the first inequality uses Cauchy-Schwarz inequality and the fact that $\|\mathbf{v}_t\|_2^2 = 1$, and the second inequality uses Lemma. 1. The final inequality is true by choosing $M \geq 4(H/L)^2$. The second term comes from using the empirical average reward to estimate the value function, similar

to REINFORCE. So we have:

$$\begin{aligned}
\text{Var}_{t,2} &= \mathbb{E} \left[\left\| \frac{d}{\mu} \left(\frac{\sum_{n=1}^N (r(\tau_{n,1}) - r(\tau_{n,0}))}{N} - (V(\pi_{\theta_t + \mu \mathbf{v}_t}) - V(\pi_{\theta_t})) \right) \mathbf{v}_t \right\|_2^2 \middle| \mathcal{F}_t \right] \\
&= \frac{d^2}{\mu^2 N^2} \mathbb{E} \left[\left\| \sum_{n=1}^N \underbrace{\left((r(\tau_{n,1}) - r(\tau_{n,0})) - (V(\pi_{\theta_t + \mu \mathbf{v}_t}) - V(\pi_{\theta_t})) \right)}_{\mathbf{E}_{t,n}} \right\|_2^2 \middle| \mathcal{F}_t \right] \\
&= \frac{d^2}{\mu^2 N^2} \mathbb{E} \left[\left\| \sum_{n=1}^N \mathbf{E}_{t,n} \right\|_2^2 \middle| \mathcal{F}_t \right],
\end{aligned}$$

where the last equality is because $\|\mathbf{v}_t\|_2^2 = 1$. We open up the square as:

$$\begin{aligned}
\mathbb{E} \left[\left\| \sum_{n=1}^N \mathbf{E}_{t,n} \right\|_2^2 \middle| \mathcal{F}_t \right] &= \sum_{n=1}^N \mathbb{E} \left[\|\mathbf{E}_{t,n}\|_2^2 \middle| \mathcal{F}_t \right] + \sum_{i \neq j} \mathbb{E} [\mathbb{E}[\mathbf{E}_{t,i} \mathbf{E}_{t,j} | \mathbf{v}_t] | \mathcal{F}_t] \\
&= \sum_{n=1}^N \mathbb{E} \left[\|\mathbf{E}_{t,n}\|_2^2 \middle| \mathcal{F}_t \right] + \sum_{i \neq j} \mathbb{E} [\mathbb{E}[\mathbf{E}_{t,i} | \mathbf{v}_t] | \mathcal{F}_t] \mathbb{E} [\mathbb{E}[\mathbf{E}_{t,j} | \mathbf{v}_t] | \mathcal{F}_t] \\
&= \sum_{n=1}^N \mathbb{E} \left[\|\mathbf{E}_{t,n}\|_2^2 \middle| \mathcal{F}_t \right].
\end{aligned}$$

where the second inequality uses the independence between trajectories generated at the same step, and the last equality is due to $\mathbb{E}[(r(\tau_{n,1}) - r(\tau_{n,0})) - (V(\pi_{\theta_t + \mu \mathbf{v}_t}) - V(\pi_{\theta_t}))] = 0$ for any fixed perturbation direction \mathbf{v}_t from the definition of value function. So we have:

$$\begin{aligned}
\text{Var}_{t,2} &= \frac{d^2}{\mu^2 N^2} \sum_{n=1}^N \mathbb{E} \left[\|\mathbf{E}_{t,n}\|_2^2 \middle| \mathcal{F}_t \right] \\
&= \frac{d^2}{\mu^2 N^2} \sum_{n=1}^N \mathbb{E} \left[|(r(\tau_{n,1}) - r(\tau_{n,0})) - (V(\pi_{\theta_t + \mu \mathbf{v}_t}) - V(\pi_{\theta_t}))|^2 \middle| \mathcal{F}_t \right] \\
&\leq \frac{4d^2 H^2}{\mu^2 N},
\end{aligned}$$

where the last inequality uses the fact that both the difference of reward and the difference of value function are within $[-H, H]$. Now we analyze the last term:

$$\begin{aligned}
\text{Var}_{t,3} &= \mathbb{E} \left[\left\| \frac{d}{\mu} (V(\pi_{\theta_t + \mu \mathbf{v}_t}) - V(\pi_{\theta_t})) \mathbf{v}_t \right\|_2^2 \middle| \mathcal{F}_t \right] \\
&= \frac{d^2}{\mu^2} \mathbb{E} \left[\|(V(\pi_{\theta_t + \mu \mathbf{v}_t}) - V(\pi_{\theta_t}) - \langle \nabla_{\theta} V(\pi_{\theta_t}), \mu \mathbf{v}_t \rangle) \mathbf{v}_t + \langle \nabla_{\theta} V(\pi_{\theta_t}), \mu \mathbf{v}_t \rangle \mathbf{v}_t\|_2^2 \middle| \mathcal{F}_t \right] \\
&\leq \frac{2d^2}{\mu^2} \mathbb{E} \left[\|(V(\pi_{\theta_t + \mu \mathbf{v}_t}) - V(\pi_{\theta_t}) - \langle \nabla_{\theta} V(\pi_{\theta_t}), \mu \mathbf{v}_t \rangle) \mathbf{v}_t\|_2^2 \middle| \mathcal{F}_t \right] \\
&\quad + \frac{2d^2}{\mu^2} \mathbb{E} \left[\|\langle \nabla_{\theta} V(\pi_{\theta_t}), \mu \mathbf{v}_t \rangle \mathbf{v}_t\|_2^2 \middle| \mathcal{F}_t \right].
\end{aligned}$$

The first term can be bounded using the smoothness of the value function as follows:

$$\begin{aligned}
&\mathbb{E} \left[\|(V(\pi_{\theta_t + \mu \mathbf{v}_t}) - V(\pi_{\theta_t}) - \langle \nabla_{\theta} V(\pi_{\theta_t}), \mu \mathbf{v}_t \rangle) \mathbf{v}_t\|_2^2 \middle| \mathcal{F}_t \right] \\
&= \mathbb{E} \left[|(V(\pi_{\theta_t + \mu \mathbf{v}_t}) - V(\pi_{\theta_t}) - \langle \nabla_{\theta} V(\pi_{\theta_t}), \mu \mathbf{v}_t \rangle)|_2^2 \|\mathbf{v}_t\|_2^2 \middle| \mathcal{F}_t \right] \\
&\leq \mathbb{E} \left[\frac{L^2 \mu^4}{4} \|\mathbf{v}_t\|_2^4 \middle| \mathcal{F}_t \right] \\
&= \frac{L^2 \mu^4}{4},
\end{aligned}$$

where the last equality uses $\|\mathbf{v}_t\|_2^2 = K$. The second term can be simplified as follows:

$$\begin{aligned}\mathbb{E} \left[\|\langle \nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t}), \mu \mathbf{v}_t \rangle \mathbf{v}_t \|_2^2 \middle| \mathcal{F}_t \right] &= \mu^2 \mathbb{E} \left[\nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t})^\top \mathbf{v}_t \mathbf{v}_t^\top \nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t}) \|\mathbf{v}_t\|_2^2 \middle| \mathcal{F}_t \right] \\ &= \mu^2 \nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t})^\top \mathbb{E} [\mathbf{v}_t \mathbf{v}_t^\top | \mathcal{F}_t] \nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t}).\end{aligned}$$

Notice that \mathbf{v}_t is a vector such that on the coordinates chosen by \mathbf{i}_t , the element is either $-1/\sqrt{K}$ or $1/\sqrt{K}$, and on other coordinates, it is zero. So $\mathbf{v}_t \mathbf{v}_t^\top$ will result in a diagonal matrix where on position $(i_{t,1}, i_{t,1}), \dots, (i_{t,K}, i_{t,K})$ the element is $1/K$ and 0 otherwise. Since each coordinate i will be chosen in each iteration in \mathbf{i}_t with probability K/d , taking an expectation over $\mathbf{v}_t \mathbf{v}_t^\top$ will result in a diagonal matrix where the diagonal elements are $1/d$. Therefore, we have:

$$\mathbb{E} \left[\|\langle \nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t}), \mu \mathbf{v}_t \rangle \mathbf{v}_t \|_2^2 \middle| \mathcal{F}_t \right] = \frac{\mu^2}{d} \nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t})^\top \nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t}) = \frac{\mu^2}{d} \|\nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t})\|_2^2.$$

So Let us combine both terms and we can bound $\text{Var}_{t,3}$ as follows:

$$\text{Var}_{t,3} \leq \frac{\mu^2 L^2 d^2}{2} + 2d \|\nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t})\|_2^2.$$

So, combining three terms, the variance drift can be bounded as:

$$\text{Var}_t \leq 6d \|\nabla_{\boldsymbol{\theta}} V(\pi_{\boldsymbol{\theta}_t})\|_2^2 + \frac{3\mu^2 L^2 d^2}{2} + \frac{9d^2 L^2 \log M}{\mu^2 M} + \frac{12d^2 H^2}{\mu^2 N}.$$