Multi-Step Preference Optimization via Two-Player Markov Games

Yongtao Wu[*] [†] **Luca Viano**[*] [†] **Yihang Chen**[‡] **Zhenyu Zhu**[†]

Quanquan Gu[‡] [§] **Volkan Cevher**[†] [§]

Abstract

Reinforcement Learning from Human Feedback (RLHF) has been highly successful in aligning large language models with human preferences. While prevalent methods like DPO have demonstrated strong performance, they frame interactions with the language model as a bandit problem, which limits their applicability in real-world scenarios where multi-turn conversations are common. Additionally, DPO relies on the Bradley-Terry model assumption, which does not adequately capture the non-transitive nature of human preferences. In this paper, we address these challenges by modeling the alignment problem as a two-player constant-sum Markov game, where each player seeks to maximize their winning rate against the other across all steps of the conversation. Our approach Multi-step Preference Optimization (`MPO`) is built upon the natural actor-critic framework [32]. We further develop `OMPO` based on the optimistic online gradient descent algorithm [36, 19]. Theoretically, we provide a rigorous analysis for both algorithms on convergence and show that `OMPO` requires $\mathcal{O}(\epsilon^{-1})$ policy updates to converge to an ϵ-approximate Nash equilibrium. We also validate the effectiveness of our method through experiments on the multi-turn conversations dataset in MT-bench-101.

1 Introduction

In recent years, the integration of large-language models (LLMs) [8, 1, 43] into various applications has highlighted the need for advanced preference alignment methods [52, 40, 5, 31, 35]. As models increasingly engage in complex decision making or reasoning scenarios, e.g., GPT-4o and o1[1], the ability to align their outputs with user preferences has received more attention. However, existing works on reinforcement learning from human feedback (RLHF) focus mostly on one-step preference [35, 26, 27, 3, 49, 50], which neglects indispensable intermediate preferences within the answer and limits the model's alignment ability. For example, in multi-round conversations, alignment must occur at each turn to meet user needs. Similarly, in mathematical reasoning with chain-of-thought prompting, step-by-step validation is essential to ensure accuracy in the final result. The reliance on final-output feedback in most existing RLHF methods [46, 38] neglects these intermediate steps, highlighting the need for multi-step preference optimization to enhance alignment capabilities.

Meanwhile, earlier alignment methods e.g., DPO and its variants step-DPO [21, 24], typically model the pairwise preference by the Bradley-Terry model [7], which assigns a score for each answer based on its preference. This assumption of the model cannot capture the non-transitive preference, which is often observed in the averaged human preferences from the population [44, 15]. While a recent line of work has modeled the alignment process under the framework of general preference [3, 27, 49, 37], and thus bypasses the BT model assumption, the challenge of multi-step preference optimization remains underexplored. In this paper, we address this gap by making the following contribution:

† EPFL ‡ UCLA * Equal contribution § Equal mentorship

[1] https://openai.com/o1

NeurIPS 2024 Workshop on Language Gamification.

- We formulate multi-step preference optimization as a two-player partially observable Markov game. Unlike [46, 41, 38] who focus on the preference feedback at the final state, we assume that the preference signal is received at each step. Such feedback allows the model to better identify which steps are correct or erroneous, potentially enhancing learning efficiency and accuracy.
- We propose Multi-step Preference Optimization (`MPO`) based on the natural actor-critic framework and Optimistic Multi-step Preference Optimization (`OMPO`), built upon the optimistic online gradient descent. Theoretically, we show that `OMPO` requires $\mathcal{O}(\epsilon^{-1})$ policy updates to converge to an ϵ-approximate Nash equilibrium, compared to $\mathcal{O}(\epsilon^{-2})$ by the algorithms provided in [46, 41, 38]. Our result cannot be trivially extended by [2] due to the partially observable nature of Markov game. We bypass this difficulty by parameterizing the game over occupancy measures.
- We provide practical implementations of both `MPO` and `OMPO` for LLM alignment. Numerical results show that the proposed methods achieve considerable improvement on multi-turn conversation datasets, such as MT-bench-101, compared to the multi-step variant of DPO.

2 Multi-step RLHF as two-player Markov game

We define the prompt to LLM as x and the answer from LLM as a. For a multi-turn conversation with turn H, the prompts are denoted by a sequence $(x_1, \ldots, x_H)$ and the answers are denoted by a sequence $(a_1, \ldots, a_H)$. The concatenation of a prompt x and an answer a is denoted by $[x, a]$ and can be generalized to the concatenation of multiple prompts and answers, e.g., $[x_1, a_1, \ldots, x_H, a_H]$. For any two sentences, e.g., $[x, a]$ and $[x', a']$, we define a preference oracle as $o([x, a], [x', a']) \in \{0, 1\}$, which can provide preference feedback with 0-1 scores, where 1 means the conversation $[x, a]$ is preferred and 0 otherwise. We denote $\mathbb{P}([x, a] \succ [x', a']) = \mathbb{E}[o([x, a] \succ [x', a'])]$ as the probability that the conversation $[x, a]$ is preferred over $[x', a']$. Moreover, we have $\mathbb{P}([x, a] \succ [x', a']) = 1 - \mathbb{P}([x', a'] \succ [x, a])$. An autoregressive LLM is denoted by $\pi(a|x)$.

We can cast the multi-step alignment process as a finite-horizon MDP. We define $s_h = [x_1, a_1, \ldots, x_{h-1}, a_{h-1}, x_h]$ as the state at $h > 1$. We define the action a_h as the answer given s_h. Particularly, we have $s_1 = x_1$. The prompt in the next state is sampled under the transition $x_{h+1} \sim f(\cdot|s_h, a_h)$, which is equivalent to $s_{h+1} \sim f(\cdot|s_h, a_h)$. The equivalence comes from the fact $s_{h+1} = [s_h, a_h, x_{h+1}]$ by using the concatenation operator between sentences. The terminal state is s_{H+1}. Next, we define the pair-wise reward function of two state-action pairs as the preference of two trajectories: $r(s_h, a_h, s'_h, a'_h) = \mathbb{P}([s_h, a_h] \succ [s'_h, a'_h])$. We define the initial state distribution ν_1 is a distribution over the initial prompt x_1. Note that each state in $\mathcal{S}$ is a pair of s_h and s'_h generated by two policies. Our goal is to identify the Nash equilibrium of the following two-player Markov game:

$$(\pi^*, \pi^*) = \arg\max_{\pi} \min_{\pi'} \mathbb{E}_{s_1 \sim \nu_1, s_h, a_h, s'_h, a'_h} \Big[\sum_{h=1}^{H} r(s_h, a_h, s'_h, a'_h) \Big], \tag{Game}$$

where $s_1 = s'_1 = x_1, a_h \sim \pi(\cdot|s_h)$, $a'_h \sim \pi'(\cdot|s'_h)$, $s_h \sim f(\cdot|s_{h-1}, a_{h-1})$, $s'_h \sim f(\cdot|s'_{h-1}, a'_{h-1})$. By Lemma 1, we can rewrite Eq. (Game) as follows:

$$(\pi^*, \pi^*) = \arg\max_{\pi} \min_{\pi'} \mathbb{E}_{s_1 \sim \nu_1} V^{\pi, \pi'}(s_1, s_1)\,, \tag{1}$$

with the value function is defined at Appx. A. Due to the constrained space, we defer the definition of value function, Q-function, and occupancy measures d_h to Appx. A.

3 `MPO` with natural actor-critic

This section presents our first method to find an approximate solution to equation Game. In order to find an ϵ-approximate Nash equilibrium, the MPO method builds upon Lemma 2 which decomposes the difference of two value functions to the Q function at each step. By setting $\pi' = \overline{\pi} = \pi^t$ in Lemma 2 and $\pi = \pi^\star$ and summing from $t = 1$ to T we obtain:

$$\sum_{t=1}^{T} \left\langle \nu_1, V^{\pi^\star, \pi^t} - V^{\pi^t, \pi^t} \right\rangle = \mathbb{E}_{s_1 \sim \nu_1} \sum_{h=1}^{H} \sum_{t=1}^{T} \mathbb{E}_{s \sim d_h^{\pi^\star}|s_1} \left[\left\langle \mathbb{E}_{s', a' \sim d_h^{\pi^t}|s_1} Q_h^{\pi^t, \pi^t}(s, \cdot, s', a'), \pi_h^\star(\cdot|s) - \pi_h^t(\cdot|s) \right\rangle \right].$$

Since the sum over t commutes with the expectation, we see that we can decompose the global regret $\sum_{t=1}^{T} \left\langle \nu_1, V^{\pi^\star, \pi^t} - V^{\pi^t, \pi^t} \right\rangle$ into a weighted sum of local regrets at each stage. Therefore, we can

Algorithm 1 `MPO` (Theory Version)

input: reference policy π^1, preference oracle $\mathbb{P}$, learning rate $\beta = \sqrt{\frac{\log \underline{\pi}^{-1}}{T H^2}}$, total iteration T
for $t = 1, 2, \ldots, T$ **do**

$$\pi_h^{t+1}(a|s) \propto \pi_h^t(a|s) \exp\left[\beta \mathbb{E}_{s',a' \sim d_h^{\pi^t}|s_1(s)} Q_h^{\pi^t,\pi^t}(s,a,s',a')\right] \quad \forall h \in [H], \quad \forall s, a.$$

end for
output: $\bar{\pi}^T$ (such that $d_h^{\bar{\pi}^T} = \frac{1}{T}\sum_{t=1}^T d_h^{\pi^t}, \ \forall h \in [H]$.).

Algorithm 2 `OMPO` (Theory Version)

input: occupancy measure of reference policy π^1 denoted as d^1, preference oracle $\mathbb{P}$ (i.e. reward function r), learning rate β, Bregman divergence $\mathbb{D}$, iteration T
for $t = 1, 2, \ldots, T$ **do**

$$d_h^{t+1} = \arg\max_{d \in \mathcal{F}_{s_1}} \beta \left\langle d, 2\mathbb{E}_{s',a' \sim d_h^t} r(\cdot,\cdot,s',a') - \mathbb{E}_{s',a' \sim d_h^{t-1}} r(\cdot,\cdot,s',a') \right\rangle - \mathbb{D}(d, d_h^t) \quad \forall h \in [H] \ \forall s_1.$$

end for
$\pi_h^{\text{out}}(a|s) = \frac{\bar{d}_h(s,a|s_1)}{\sum_a \bar{d}_h(s,a|s_1)}$ with $\bar{d}_h = T^{-1}\sum_{t=1}^T d_h^t$ for all $h \in [H]$ for the unique s_1 from which s is reachable.
Output : π^{out}

control the global regret implementing at each state online mirror descent updates ([47], [30, Chapter 6], [9]), i.e., implementing the following update[2]:

$$\pi_h^{t+1}(\cdot|s) = \arg\max_{\pi} \langle \pi(\cdot|s), \mathbb{E}_{s',a' \sim d_h^{\pi^t}|s_1(s)} Q_h^{\pi^t,\pi^t}(s,\cdot,s',a') \rangle - \beta D(\pi(\cdot|s) || \pi_h^t(\cdot|s)) \,,$$

with learning rate β. The solution is $\pi_h^{t+1}(a|s) \propto \pi_h^t(a|s) \exp\{\beta \mathbb{E}_{s',a' \sim d_h^{\pi^t}|s_1(s)} Q_h^{\pi^t,\pi^t}(s,a,s',a')\}$, which corresponds to natural actor-critic [32] that utilizes a softmax-based method for updating policies. The number of policy updates needed by `MPO` (see Alg. 1) can be bounded as follows.

Theorem 1. *Consider Alg. 1 and assume that the reference policy is uniformly lower bounded by $\underline{\pi}$, then there exists a policy $\bar{\pi}^T$ such that $d_h^{\bar{\pi}^T} = \frac{1}{T}\sum_{t=1}^T d_h^{\pi^t}, \forall h \in [H]$, and it holds that for $T = \frac{16 H^4 \log \underline{\pi}^{-1}}{\epsilon^2}$ the policy pair $(\bar{\pi}^T, \bar{\pi}^T)$ is an ϵ-approximate Nash equilibrium. Therefore, Alg. 1 outputs an ϵ-approximate Nash equilibrium after $\frac{16 H^4 \log \underline{\pi}^{-1}}{\epsilon^2}$ policy updates.*

4 Optimistic `MPO`: `OMPO`

In this section, we propose an alternative algorithm based on the optimistic gradient descent method and by reformulating the optimization problem over occupancy measures. Here, we show that optimistic online mirror descent with one projection [19] with an appropriately chosen regularizer can be used to solve approximately the following program which corresponds to Game lifted to the space of the occupancy measures.

$$(d^\star, d^\star) = \arg\max_{d \in \tilde{\mathcal{F}}} \min_{d' \in \tilde{\mathcal{F}}} \mathbb{E}_{s_1 \sim \nu_1} \sum_{h=1}^H \sum_{s,a,s',a'} d_h(s,a|s_1) r(s,a,s',a') d'_h(s',a'|s_1) \,,$$

where $\tilde{\mathcal{F}}$ is the product set of the Bellman flow constraints for a particular initial state, i.e. $\tilde{\mathcal{F}} = \times_{s_1 \in \text{supp}(\nu_1)} \mathcal{F}_{s_1}$. We also introduced the Bellman flow constraints for a specific initial state $\mathcal{F}_{s_1} = \left\{ d = (d_1, \ldots, d_H) : \sum_a d_{h+1}(s,a) = \sum_{s',a'} f(s|s',a') d_h(s',a'), d_1(s) = \mathbb{1}\{s = s_1\} \right\}$.

The policy pair $(\pi^\star, \pi^\star)$ solution of Game can be retrieved from the occupancy measure pair $(d^\star, d^\star)$ as $\pi^\star(a|s) = \frac{d^\star(s,a|s_1)}{\sum_a d^\star(s,a|s_1)}$. Our idea is to apply the optimistic algorithm from [19] to the reformu-

[2]We denote as $s_1(s)$ the only initial state that can lead to s. This is motivated by practical LLM training, where system prompts such as "user" and "assistant" are inserted before every x_h and a_h, respectively. As a result, one can infer a unique s_1 for every s.

Table 1: Evaluation results on MT-bench-101 dataset. We can observe that both of the proposed algorithms MPO and OMPO considerably outperform the baseline in terms of the score.

Model		Perceptivity					Adaptability						Interactivity	
		Memory	Understanding		Interference		Rephrasing		Reflection		Reasoning		Questioning	
	Avg.	CM	SI	AR	TS	CC	CR	FR	SC	SA	MR	GR	IC	PI
Base (Mistral-7B-Instruct)	6.223	7.202	7.141	7.477	7.839	8.294	6.526	6.480	4.123	4.836	4.455	5.061	5.818	5.641
DPO (iter=1)	6.361	7.889	6.483	7.699	8.149	8.973	7.098	7.423	3.448	**6.123**	3.421	4.492	5.639	5.858
DPO (iter=2)	6.327	7.611	6.206	8.106	8.052	9.111	6.670	7.153	3.494	5.884	3.360	4.691	5.837	6.078
DPO (iter=3)	5.391	6.019	4.521	6.890	6.631	8.177	5.437	5.723	3.448	5.295	3.142	4.015	5.256	5.529
SPPO (iter=1)	6.475	7.432	7.464	7.714	8.353	8.580	6.917	6.714	4.136	5.055	4.403	5.400	6.036	5.966
SPPO (iter=2)	6.541	7.516	7.496	7.808	8.313	8.731	7.077	6.867	4.136	5.281	4.488	5.477	6.098	5.751
SPPO (iter=3)	6.577	7.575	7.547	7.944	8.365	8.797	7.040	6.865	4.442	5.185	4.346	5.394	6.092	5.906
Step-DPO (iter=1)	6.433	7.463	7.054	7.790	8.157	8.593	6.827	6.748	4.234	4.849	4.236	5.519	5.982	6.171
Step-DPO (iter=2)	6.553	7.616	7.043	7.925	8.147	8.662	6.790	6.878	4.331	5.048	4.366	**5.734**	**6.391**	6.254
Step-DPO (iter=3)	6.442	7.665	7.023	7.767	8.016	8.589	6.723	6.581	4.305	5.014	4.153	5.453	6.202	**6.257**
MPO* (iter=1)	6.630	7.624	**7.846**	8.085	8.398	8.947	7.105	7.286	4.208	4.993	4.377	5.264	6.179	5.873
MPO* (iter=2)	6.735	7.838	7.723	8.196	**8.590**	9.027	7.347	7.209	4.240	5.137	4.469	5.531	6.181	6.061
MPO* (iter=3)	6.733	**7.868**	7.686	**8.289**	8.510	9.078	7.330	7.529	**4.461**	4.829	4.225	5.366	6.198	6.155
OMPO$\star$ (iter=2)	6.736	7.733	7.723	8.257	8.478	9.122	7.300	7.421	4.123	5.288	**4.506**	5.513	6.179	5.923
OMPO$\star$ (iter=3)	**6.776**	7.649	7.792	8.281	8.578	**9.136**	**7.424**	**7.635**	4.377	5.308	4.312	5.455	6.187	5.954

lation of Game over occupancy measures, we present the resulting algorithm, i.e., OMPO, in Alg. 2.

Remark 1. *Lifting the problem to the occupancy measures turns out to be of fundamental importance to have each agent learning a policy conditioned only on their own state. This is different from the standard literature on Markov Games [12, 48, 2] that assumes that both agents share a common state.*

As the next theorem shows, in the ideal case where the updates can be computed exactly, Alg. 2 finds an ϵ-approximate Nash equilibrium using fewer updates compared to Alg. 1 and to [41, Algorithm 1].

Theorem 2 (Convergence of OMPO)**.** *Consider Alg. 2 and assume the occupancy measure of the reference policy is uniformly lower bounded by $\underline{d}$. Moreover, let $\mathbb{D}$ be $1/\lambda$ strongly convex, i.e. $\mathbb{D}(p||q) \geq \frac{\|p-q\|_1^2}{2\lambda}$. Then, setting $T = \frac{10H \log \underline{d}^{-1}}{\beta\epsilon}$ and $\beta \leq \frac{1}{\sqrt{2\lambda}}$, we ensure that the output of Alg. 2 is an ϵ-approximate Nash equilibrium. Therefore, we need at most $\frac{10H \log \underline{d}^{-1}}{\beta\epsilon}$ policy updates.*

In addition, not only [41, Algorithm 1] but also OMPO can be implemented using only one player since in a constant sum game, the max and min player produce the same iterates. The result is formalized as follows and the proof is deferred to Appx. E.5.

Theorem 3. *Consider a constant sum two-player Markov games with reward such that $r(s, a, s', a') = 1 - r(s', a', s, a)$, then for each $s_1 \in \operatorname{supp}(\nu_1)$ the updates for d in Alg. 2 coincides with the updates for the min player that uses the updates*

$$d_h^{t+1}(a|s) = \arg\min_{d \in \mathcal{F}_{s_1}} \beta \left\langle d, 2\mathbb{E}_{s',a' \sim d_h^t} r(s', a', \cdot, \cdot) - \mathbb{E}_{s',a' \sim d_h^{t-1}} r(s', a', \cdot, \cdot) \right\rangle + \mathbb{D}(d, d_h^t)\,.$$

Furthermore, we can avoid the projection over the set $\mathcal{F}$ implementing this update on the policy space (see Appendix F). We achieve such result following the techniques developed in [6, 45].

5 Experiments

In this section, we validate the proposed algorithm with multi-turn conversations in MT-bench-101 [4]. We choose Mistral-7B-Instruct-v0.2 as the base model [18]. We select iterative DPO [13], iterative SPPO [49], and iterative Step-DPO as our baselines. We use a pre-trained model PairRM[3] as the preference oracle. Each round of dialogue is rated on a scale of 1 to 10 by GPT-4o mini, with the mean score reported for each dialogue. All methods are run for a total of 3 iterations. We use the practical algorithms of MPO and OMPO in Alg. 3 and 4. Detailed set-up and hyperparameters can be found at Appx. H. The results are summarized in Tab. 1, showing significant improvements over the baselines with the proposed MPO and OMPO approaches.

[3] https://huggingface.co/llm-blender/PairRM

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[2] Ahmet Alacaoglu, Luca Viano, Niao He, and Volkan Cevher. A natural actor-critic framework for zero-sum markov games. In *International Conference on Machine Learning*, pages 307–366. PMLR, 2022.

[3] 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 *International Conference on Artificial Intelligence and Statistics*, pages 4447–4455. PMLR, 2024.

[4] Ge Bai, Jie Liu, Xingyuan Bu, Yancheng He, Jiaheng Liu, Zhanhui Zhou, Zhuoran Lin, Wenbo Su, Tiezheng Ge, Bo Zheng, et al. Mt-bench-101: A fine-grained benchmark for evaluating large language models in multi-turn dialogues. *arXiv preprint arXiv:2402.14762*, 2024.

[5] Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.

[6] Joan Bas-Serrano, Sebastian Curi, Andreas Krause, and Gergely Neu. Logistic q-learning. In *International conference on artificial intelligence and statistics*, pages 3610–3618. PMLR, 2021.

[7] 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.

[8] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[9] Nicolo Cesa-Bianchi and Gábor Lugosi. *Prediction, learning, and games*. Cambridge university press, 2006.

[10] Chao-Kai Chiang, Tianbao Yang, Chia-Jung Lee, Mehrdad Mahdavi, Chi-Jen Lu, Rong Jin, and Shenghuo Zhu. Online optimization with gradual variations. In Shie Mannor, Nathan Srebro, and Robert C. Williamson, editors, *Proceedings of the 25th Annual Conference on Learning Theory*, volume 23 of *Proceedings of Machine Learning Research*, pages 6.1–6.20, Edinburgh, Scotland, 25–27 Jun 2012. PMLR.

[11] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

[12] Constantinos Daskalakis, Dylan J Foster, and Noah Golowich. Independent policy gradient methods for competitive reinforcement learning. *Advances in neural information processing systems*, 33:5527–5540, 2020.

[13] 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.

[14] Roy Fox, Ari Pakman, and Naftali Tishby. Taming the noise in reinforcement learning via soft updates. *arXiv preprint arXiv:1512.08562*, 2015.

[15] Martin Gardner. Mathematical games. *Scientific american*, 222(6):132–140, 1970.

[16] Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. *NeurIPS*, 2021.

[17] Jiwoo Hong, Noah Lee, and James Thorne. Orpo: Monolithic preference optimization without reference model. *arXiv preprint arXiv:2403.07691*, 2(4):5, 2024.

[18] Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023.

[19] Pooria Joulani, András György, and Csaba Szepesvári. A modular analysis of adaptive (non-)convex optimization: Optimism, composite objectives, and variational bounds. In Steve Hanneke and Lev Reyzin, editors, *Proceedings of the 28th International Conference on Algorithmic Learning Theory*, volume 76 of *Proceedings of Machine Learning Research*, pages 681–720. PMLR, 15–17 Oct 2017.

[20] Sham Kakade and John Langford. Approximately optimal approximate reinforcement learning. In *Proceedings of the Nineteenth International Conference on Machine Learning*, pages 267–274, 2002.

[21] Xin Lai, Zhuotao Tian, Yukang Chen, Senqiao Yang, Xiangru Peng, and Jiaya Jia. Step-dpo: Step-wise preference optimization for long-chain reasoning of llms. *arXiv preprint arXiv:2406.18629*, 2024.

[22] Ilya Loshchilov and Frank Hutter. SGDR: Stochastic gradient descent with warm restarts. In *International Conference on Learning Representations*, 2017.

[23] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

[24] Zimu Lu, Aojun Zhou, Ke Wang, Houxing Ren, Weikang Shi, Junting Pan, and Mingjie Zhan. Step-controlled dpo: Leveraging stepwise error for enhanced mathematical reasoning. *arXiv preprint arXiv:2407.00782*, 2024.

[25] Yura Malitsky and Matthew K Tam. A forward-backward splitting method for monotone inclusions without cocoercivity. *SIAM Journal on Optimization*, 30(2):1451–1472, 2020.

[26] Yu Meng, Mengzhou Xia, and Danqi Chen. Simpo: Simple preference optimization with a reference-free reward. *arXiv preprint arXiv:2405.14734*, 2024.

[27] Rémi Munos, Michal Valko, Daniele Calandriello, Mohammad Gheshlaghi Azar, Mark Rowland, Zhaohan Daniel Guo, Yunhao Tang, Matthieu Geist, Thomas Mesnard, Andrea Michi, et al. Nash learning from human feedback. In *Forty-first International Conference on Machine Learning*, 2024.

[28] Gergely Neu, Anders Jonsson, and Vicenç Gómez. A unified view of entropy-regularized markov decision processes, 2017.

[29] Gergely Neu and Julia Olkhovskaya. Online learning in mdps with linear function approximation and bandit feedback. *Advances in Neural Information Processing Systems*, 34:10407–10417, 2021.

[30] Francesco Orabona. A modern introduction to online learning, 2023.

[31] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

[32] Jan Peters and Stefan Schaal. Natural actor-critic. *Neurocomputing*, 71(7-9):1180–1190, 2008.

[33] Leonid Denisovich Popov. A modification of the arrow-hurwitz method of search for saddle points. *Mat. Zametki*, 28(5):777–784, 1980.

[34] M. L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, Inc., USA, 1st edition, 1994.

[35] 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. *Advances in Neural Information Processing Systems*, 36, 2023.

[36] Alexander Rakhlin and Karthik Sridharan. Online learning with predictable sequences. In *Conference on Learning Theory*, pages 993–1019. PMLR, 2013.

[37] Corby Rosset, Ching-An Cheng, Arindam Mitra, Michael Santacroce, Ahmed Awadallah, and Tengyang Xie. Direct nash optimization: Teaching language models to self-improve with general preferences. *arXiv preprint arXiv:2404.03715*, 2024.

[38] Lior Shani, Aviv Rosenberg, Asaf Cassel, Oran Lang, Daniele Calandriello, Avital Zipori, Hila Noga, Orgad Keller, Bilal Piot, Idan Szpektor, et al. Multi-turn reinforcement learning from preference human feedback. *arXiv preprint arXiv:2405.14655*, 2024.

[39] Lloyd S Shapley. Stochastic games. *Proceedings of the national academy of sciences*, 39(10):1095–1100, 1953.

[40] 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. *Advances in Neural Information Processing Systems*, 33:3008–3021, 2020.

[41] Gokul Swamy, Christoph Dann, Rahul Kidambi, Steven Wu, and Alekh Agarwal. A minimaximalist approach to reinforcement learning from human feedback. In *Forty-first International Conference on Machine Learning*, 2024.

[42] Yunhao Tang, Zhaohan Daniel Guo, Zeyu Zheng, Daniele Calandriello, Rémi Munos, Mark Rowland, Pierre Harvey Richemond, Michal Valko, Bernardo Ávila Pires, and Bilal Piot. Generalized preference optimization: A unified approach to offline alignment. *arXiv preprint arXiv:2402.05749*, 2024.

[43] Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

[44] Amos Tversky. Intransitivity of preferences. *Psychological review*, 76(1):31, 1969.

[45] Luca Viano, Angeliki Kamoutsi, Gergely Neu, Igor Krawczuk, and Volkan Cevher. Proximal point imitation learning. *Advances in Neural Information Processing Systems*, 35:24309–24326, 2022.

[46] Yuanhao Wang, Qinghua Liu, and Chi Jin. Is rlhf more difficult than standard rl? a theoretical perspective. *Advances in Neural Information Processing Systems*, 2023.

[47] Manfred K Warmuth, Arun K Jagota, et al. Continuous and discrete-time nonlinear gradient descent: Relative loss bounds and convergence. In *Electronic proceedings of the 5th International Symposium on Artificial Intelligence and Mathematics*, volume 326. Citeseer, 1997.

[48] Chen-Yu Wei, Chung-Wei Lee, Mengxiao Zhang, and Haipeng Luo. Last-iterate convergence of decentralized optimistic gradient descent/ascent in infinite-horizon competitive markov games. In *Conference on Learning Theory*, pages 4259–4299. PMLR, 2021.

[49] Yue Wu, Zhiqing Sun, Huizhuo Yuan, Kaixuan Ji, Yiming Yang, and Quanquan Gu. Self-play preference optimization for language model alignment. *arXiv preprint arXiv:2405.00675*, 2024.

[50] Yuheng Zhang, Dian Yu, Baolin Peng, Linfeng Song, Ye Tian, Mingyue Huo, Nan Jiang, Haitao Mi, and Dong Yu. Iterative nash policy optimization: Aligning llms with general preferences via no-regret learning. *arXiv preprint arXiv:2407.00617*, 2024.

[51] Brian D Ziebart. *Modeling purposeful adaptive behavior with the principle of maximum causal entropy*. Carnegie Mellon University, 2010.

[52] Daniel M Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*, 2019.

Acknowledgements

This work was supported by Hasler Foundation Program: Hasler Responsible AI (project number 21043). This work was supported by the Swiss National Science Foundation (SNSF) under grant number 200021_205011. This work is funded (in part) through a PhD fellowship of the Swiss Data Science Center, a joint venture between EPFL and ETH Zurich. This research was sponsored by the Army Research Office and was accomplished under Grant Number W911NF-24-1-0048.

Contents of the appendix

The Appendix is organized as follows:

- In Appx. A, we summarize the symbols and notation used in this paper.
- Preliminaries on single-step RLHF can be found in Appx. B.
- A detailed discussion of related work is present in Appx. C.
- In Appx. D, we give several auxiliary lemmas for our analysis.
- In Appx. E, we provide the proofs for the theoretical results.
- Appx. F shows the implementation of Alg. 2 with updates over policies.
- The practical version of `MPO` and `OMPO` are present in Appx. G.
- Additional experimental detail and result are given in Appx. H.
- Limitation and future work can be found at Appx. J.

A Symbols and Notation

We include the core symbols and notation in Tab. 2 to facilitate the understanding of our work.

Table 2: Core symbols and notations used in this paper.

Symbol	Dimension(s) & range	Definition
x_h	-	Prompt at step h
a_h	-	Answer (action) at step h
s_h	-	State at step h
$s_1(s_h)$	-	The only initial state that can lead to s_h
π		Language model (policy)
ν_1		Initial distribution of state s_1
$d_h^\pi(s,a)$	$[0,1]$	Occupancy measure of π at stage h
f		Transition function
$\Pr(s_h = s, a_h = a)$	$[0,1]$	Joint probability of $s_h = a$ and $a_h = a$
o	$\{0,1\}$	Preference oracle
$\mathbb{P}([s,a],[s',a'])$	$[0,1]$	Winning probability of $[s,a]$ against $[s',a']$
$D(p\|q)$		KL divergence of two probability distributions p and q
$\mathbb{D}(p\|q)$		Bregman Divergences between two points q and p.
$\mathcal{D}_t$		Dataset buffet at iteration t
$\Delta_\mathcal{X}$	$[0,1]^{\lvert\mathcal{X}\rvert}$	Set of probability distributions over the set $\mathcal{X}$
$\mathcal{O}$, o, Ω and Θ	-	Standard Bachmann–Landau order notation

Definition 1 (Value function)**.** *We define the pair-wise value function as follows:*

$$V_h^{\pi,\pi'}(s,s') = \mathbb{E}\Big[\sum_{\hat{h}=h}^{H} r(s_{\hat{h}}, a_{\hat{h}}, s'_{\hat{h}}, a'_{\hat{h}}) | s_h = s, s'_h = s'\Big],$$

where $a_{\hat{h}} \sim \pi_{\hat{h}}(\cdot|s_{\hat{h}})$, $a'_{\hat{h}} \sim \pi'_{\hat{h}}(\cdot|s'_{\hat{h}})$, $s_{\hat{h}+1} \sim f(\cdot|s_{\hat{h}}, a_{\hat{h}})$, and $s'_{\hat{h}+1} \sim f(\cdot|s'_{\hat{h}}, a'_{\hat{h}})$.

Definition 2 (Q-function)**.** *We define the Q-function as*

$$Q_h^{\pi,\pi'}(s,a,s',a') = r_h(s,a,s',a') + \mathbb{E}\Big[\sum_{\hat{h}=h+1}^{H} r(s_{\hat{h}}, a_{\hat{h}}, s'_{\hat{h}}, a'_{\hat{h}})\Big],$$

where $s_{\hat{h}+1} \sim f(\cdot|s_{\hat{h}}, a_{\hat{h}})$ and $s'_{\hat{h}+1} \sim f(\cdot|s'_{\hat{h}}, a'_{\hat{h}})$.

We can define the MDP as a tuple $\mathcal{M} = (\mathcal{S}, \mathcal{A}, f, r, \nu_1, H)$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, H is the horizon (total steps). We will often denote $V_1^{\pi,\pi'}$ omitting the footnote, i.e. as $V^{\pi,\pi'}$. Moreover, notice that we consider potentially non stationary policies, i.e. they are indexed by h. We denote by π such non stationary policy and by π_h the distribution over actions at stage h corresponding to the non stationary policy π.

Definition 3. *A policy π is said an ϵ-approximate Nash equilibrium if it holds that*

$$\langle \nu_1, V^{\pi,\pi}\rangle - \min_{\bar{\pi}\in\Pi} \langle \nu_1, V^{\pi,\bar{\pi}}\rangle \le \epsilon, \qquad \textit{and} \qquad \max_{\bar{\pi}\in\Pi} \langle \nu_1, V^{\bar{\pi},\pi}\rangle - \langle \nu_1, V^{\pi,\pi}\rangle \le \epsilon.$$

Definition 4 (Occupancy measures)**.** *Given the policy π, the occupancy measure of π, is defined at stage h as $d_h^\pi(s,a) = \Pr(s_h = s, a_h = a)$ where $s_1 = x_1 \sim \nu_1, a_h \sim \pi_h(\cdot|s_h), s_h \sim f(\cdot|s_{h-1}, a_{h-1})$. We also define $d_h^\pi(s,a)|s_1 = \Pr(s_h = s, a_h = a|s_1 = s_1)$. In addition, given the policies $\pi, \bar{\pi}$, the occupancy measure of $(\pi, \bar{\pi})$ at stage h is defined as $d_h^{\pi,\bar{\pi}}(s,a,s',a') = \Pr(s_h = s, a_h = a, s'_h = s', a'_h = a')$, where $s_1 = s'_1 = x_1 \sim \nu_1, a_h \sim \pi(\cdot|s_h)$, $a'_h \sim \pi'(\cdot|s'_h)$, $s_h \sim f(\cdot|s_{h-1}, a_{h-1})$, and $s'_h \sim f(\cdot|s'_{h-1}, a'_{h-1})$.*

We additionally use a compact notation for representing the Bellman flow constraints. We denote by $E \in \mathbb{R}^{|\mathcal{S}|\times|\mathcal{A}||\mathcal{S}|}$ the matrix such that $(Ez)(s,a) = z(s)$ for all vectors $z \in \mathbb{R}^{|\mathcal{S}|}$. Additionally, we denote by F the matrix such that $(Fz)(s,a) = \sum_{s'} f(s'|s,a)z(s')$ for all vectors $z \in \mathbb{R}^{|\mathcal{S}|}$.

Remark: The value function can be represented as an inner product between the reward function and the occupancy measure, i.e., $V^{\pi,\bar{\pi}} = \sum_{h=1}^{H} \langle r_h, d_h^{\pi,\bar{\pi}}\rangle$. Given the structure of the game where the sequences of sentences and answers are generated independently by the two agents, we find that the joint occupancy measure at each step can be factorized as the product of the two agents occupancy measures. In particular, $d_h^{\pi,\bar{\pi}}(s,a,s',a') = d_h^\pi(s,a)d_h^{\bar{\pi}}(s',a')$ for all h, s, a, s', a'.

B Preliminary on Single-step RLHF and motivation of multi-step RLHF

In this section, we review the earlier methods in single-step RLHF. Classical RLHF methods [52, 31] assume that the preference oracle can be expressed by an underlying Bradley-Terry (BT) reward model [7], i.e.,

$$\mathbb{P}([x_1, a_1] \succ [x_1, a'_1]) = \sigma(r(x_1, a_1) - r(x_1, a'_1)).$$

Thus, one can first learn a reward model and optimize the policy based on the following KL-constrained RL objective with PPO:

$$\pi^\star = \arg\max_\pi \mathbb{E}_{x_1\sim\nu_1, a_1\sim\pi(\cdot|x_1)}(r(x_1,a_1) - \beta D(\pi(\cdot|x_1)||\pi_{\text{ref}}(\cdot|x_1))),$$

where β is a parameter controlling the deviation from the reference model π_{ref}. Another line of work, e.g., DPO [35] avoids explicit reward modeling and optimizes the following objective over pair-wise preference data (x_1, a_1^w, a_1^l).

$$\pi^\star = \arg\max_\pi \mathbb{E}_{(x_1,a_1^w,a_1^l)\sim\mathcal{D}}\left[\log\sigma\left(\beta\log\frac{\pi(a_1^w|x_1)}{\pi_1(a_1^w|x_1)} - \beta\log\frac{\pi(a_1^l|x_1)}{\pi_1(a_1^l|x_1)}\right)\right].$$

More recently, several studies [41, 27, 49, 50, 37] have circumvented the Bradley-Terry (BT) assumption by directly modeling the general oracle $\mathbb{P}$, avoiding the reliance on the reward model which is transitive. Specifically, the goal is to identify the Nash equilibrium (or von Neumann winner) of the following two-player constant-sum game:

$$(\pi^*, \pi^*) = \arg\max_\pi \min_{\pi'} \mathbb{E}_{x_1\sim\nu_1, a_1\sim\pi(\cdot|x_1), a'_1\sim\pi'(\cdot|x_1)}\mathbb{P}([x_1,a_1] \succ [x_1,a'_1]).$$

Our multi-step setting covers a number of alignment problems, and we list some examples below.

Example 4 (Single-step alignment)**.** *In single-step alignment, a language model receives one prompt and outputs one answer. Our framework covers the single-step alignment by dissecting the answer into single tokens. Specifically, we set x_1 as the prompt, $x_2, \ldots, x_{H+1}$ as empty sentences, and the answer a_h at each turn consists of only one token. Then the horizon H is the number of tokens in the answer. The transition between each state is deterministic.*

Example 5 (Chain-of-thought reasoning alignment)**.** *In the chain-of-thought reasoning, the horizon H denotes the reasoning step, where x_1 is the initial prompt and $x_2, \ldots, x_{H+1}$ are empty. Each a_h corresponds to a reasoning step. The transition between each state is deterministic.*

Example 6 (Mutli-turn conversation alignment)**.** *In the mutli-turn conversation, the horizon H denotes the total turn of conversation. In the h-th turn, x_h is the prompt, and a_h is the answer. The prompt in the terminal state x_{H+1} is the empty sentence. The transition between each state can be deterministic or stochastic.*

Here we make a few remarks on the benefit of incorporating human preferences at each step.

Remark 2. *Let s_1 denote the question particularly those framed with chain-of-though prompting, i.e., "answer this question step by step". If two answers of length $H+1$ (s_{H+1} and s'_{H+1}) are globally similar but differ in the early reasoning steps (e.g., s_1 and s_2 are better than s'_1 and s'_2), more credit should be assigned to s_{H+1}, encouraging the model to align with it. This follows the principle that humans typically master simpler tasks before progressing to more complex ones.*

Remark 3. *From a practical standpoint, including per-step preference data generates a richer dataset for training, helping the model learn which reasoning steps are correct or wrong. This incremental feedback can enhance overall performance by reinforcing the importance of foundational steps in reasoning.*

C Related work

RLHF under Bradley-Terry model. Over the years, significant strides have been made towards developing RLHF algorithms from various perspectives under the Bradley-Terry model [7]. Earlier RLHF pipelines usually included supervised fine-tuning, learning a reward model, and reinforcement learning optimization with PPO [52, 40, 5, 31]. Due to the instability and scaling issues of such a pipeline, direct alignment methods such as DPO have been proposed to bypass the training of the reward model [35]. Several follow-up methods, such as generalized preference optimization (GPO, **(author?)** 42), use offline preference data to directly optimize pairwise preferences against a fixed opponent. A number of works have proposed reference-model-free method [26, 17]. In [26], the impact of sequence length is mitigated by averaging the likelihood over the length of the sequence. In the multi-step scenario, several multi-step variants of DPO are introduced in the math reasoning task. [24] initiate from an intermediate step in a correct reasoning process and increase the temperature to produce a flawed reasoning path leading to an incorrect answer. Meanwhile, [21] leverage GPT-4 to detect the first incorrect step in a multi-step reasoning trajectory, then regenerate from that point to obtain the correct path. Together, these serve as the pair of samples for DPO.

RLHF under general preferences. The reward model in the Bradley-Terry model inherently implies transitivity in preferences. However, human preferences, especially the resulting averaged human preferences from populations, are usually nontransitive [44, 15]. To this end, [3] outline a general framework for RLHF starting from general preference optimization and shows that DPO is a special case with the assumption of Bradley-Terry model. They further proposed IPO without such an assumption. Subsequently, [27] try to solve the alignment of non-transitive general preferences using two-player nash learning in a bandit setting. In their work, preferences are regularized through KL divergence to a reference policy, and they prove the convergence of the last iterative. In [41], multi-step alignment is considered while preference signals are only applied at the final step. [41] do not demonstrate the effectiveness of this framework in large language models. [49] propose SPPO, studying bandit alignment under general preferences. They introduce a novel loss function that increases the log-likelihood of the selected response while decreasing that of the rejected response, in contrast to DPO. [37] start with the nash learning framework and propose Online DPO, which is an iterative version of DPO. [46] provide theoretical analysis on multi-step RLHF under general preference while practice application is not explored. In [46], the preference signal is given for the entire trajectory of an MDP while in this paper it is step-wise. [38] study multi-step alignment under general preferences. However, unlike their approach where only preferences at the final states are

considered, our work is built on a two-player Markov game which assumes that human preference is received at each step rather than only at the final step. Additionally, we leverage the optimistic online gradient descent to achieve a better convergence rate than [46, 38], and utilize Monte Carlo estimation with a small-scale pairwise reward model, avoiding the need for an additional function approximator for the critic network.

Two-player Markov game & optimistic online gradient descent. Two-player Markov games have been widely studied since the seminal work [39]. Particularly relevant to our work is the research line on policy gradient algorithms for two-player Markov games such as [12, 48, 2]. Our `OMPO` is strictly related to the idea of optimistic online gradient descent [33, 10, 36] originally proposed in online learning to achieve small regret in case of slow varying loss sequences. Our update that uses only one projection per update was proposed in [19]. The name of our method is due to a similar algorithm introduced in the context of variational inequalities by [25].

D Auxiliary Lemma

Lemma 1. *The value function and Q-value function satisfy the following Bellman equation for all* $h \in [H]$.

$$Q_h^{\pi,\pi'}(s,a,s',a') = r_h(s,a,s',a') + \mathbb{E}_{\hat{s}\sim f(\cdot|s,a),\bar{s}\sim f(\cdot|s',a')}[V_{h+1}^{\pi,\pi'}(\hat{s},\bar{s})]\,.$$

$$V_h^{\pi,\pi'}(s,s') = \mathbb{E}_{a\sim\pi_h(\cdot|s),a'\sim\pi'_h(\cdot|s')}Q_h^{\pi,\pi'}(s,a,s',a').$$

Lemma 2. *For a finite horizon MDP with initial distribution* ν_1 *it holds that:*

$$\left\langle \nu_1, V^{\pi,\bar{\pi}} - V^{\pi',\bar{\pi}} \right\rangle = \mathbb{E}_{s_1\sim\nu_1}\sum_{h=1}^{H}\mathbb{E}_{s\sim d_h^{\pi}|s_1}\left[\left\langle \mathbb{E}_{s',a'\sim d_h^{\bar{\pi}}|s_1}Q_h^{\pi',\bar{\pi}}(s,\cdot,s',a'), \pi_h(\cdot|s,s_1) - \pi'_h(\cdot|s,s_1)\right\rangle\right].$$

The lemma 2 is the extension of [20] to the multi-agent setting where the dynamics are controlled independently by each player but the reward depends on the joint-state action tuple.

E Proofs

E.1 Proof of Lemma 1

Proof. By the definition of the state action value function for the policy pair (π,π') we have that

$$Q_h^{\pi,\pi'}(s,a,s',a') = r(s,a,s',a') + \mathbb{E}\Big[\sum_{h'=h+1}^{H} r(s_{h'},a_{h'},s'_{h'},a'_{h'})\Big]\,.$$

Now, using tower property of the expectation we have that

$$\begin{aligned}
&Q_h^{\pi,\pi'}(s,a,s',a')\\
&= r(s,a,s',a') + \mathbb{E}_{s''\sim f(\cdot|s,a),\bar{s}\sim f(\cdot|s',a')}\Big[\mathbb{E}\Big[\sum_{h'=h+1}^{H} r(s_{h'},a_{h'},s'_{h'},a'_{h'})|s_{h+1}=s'',s'_{h+1}=\bar{s}\Big]\Big]\\
&= r(s,a,s',a') + \mathbb{E}_{s''\sim f(\cdot|s,a),\bar{s}\sim f(\cdot|s',a')}\Big[V^{\pi,\pi'}(s'',\bar{s})\Big],
\end{aligned}$$

where the last equality follows from the definition of the state value function. □

E.2 Proof of Lemma 2

Proof. Let us consider the Bellman equation in vectorial form for the policy pair $(\pi',\bar{\pi})$, that is

$$r_h + FV_{h+1}^{\pi',\bar{\pi}} = Q_h^{\pi',\bar{\pi}},$$

where F denoted the transition matrix induced by the transition function $f : \mathcal{S}^2 \times \mathcal{A} \to \Delta_{\mathcal{S}\times\mathcal{S}}$. Now, multiplying by the occupancy measure of the policy pair $(\pi, \bar{\pi})$ at stage h we obtain

$$\left\langle d_h^{\pi,\bar{\pi}}, r_h \right\rangle + \left\langle d_h^{\pi,\bar{\pi}}, FV_{h+1}^{\pi',\bar{\pi}} \right\rangle = \left\langle d_h^{\pi,\bar{\pi}}, Q_h^{\pi',\bar{\pi}} \right\rangle.$$

At this point, using the Bellman flow constraints [34], it holds that

$$F^T d_h^{\pi,\bar{\pi}} = E^T d_{h+1}^{\pi,\bar{\pi}},$$

where $E \in \mathbb{R}^{|\mathcal{S}|^2|\mathcal{A}|\times|\mathcal{S}|^2}$ such that $(E^T v)(s,a) = V(s)$ for all $V \in \mathbb{R}^{|\mathcal{S}|^2}$. Plugging this equality in the Bellman equation above we obtain

$$\left\langle d_h^{\pi,\bar{\pi}}, r_h \right\rangle + \left\langle d_{h+1}^{\pi,\bar{\pi}}, EV_{h+1}^{\pi',\bar{\pi}} \right\rangle = \left\langle d_h^{\pi,\bar{\pi}}, Q_h^{\pi',\bar{\pi}} \right\rangle.$$

Now, subtracting on both sides $\left\langle d_h^{\pi,\bar{\pi}}, EV_h^{\pi',\bar{\pi}} \right\rangle$ and rearranging, it holds that

$$\left\langle d_h^{\pi,\bar{\pi}}, r_h \right\rangle + \left\langle d_{h+1}^{\pi,\bar{\pi}}, EV_{h+1}^{\pi',\bar{\pi}} \right\rangle - \left\langle d_h^{\pi,\bar{\pi}}, EV_h^{\pi',\bar{\pi}} \right\rangle = \left\langle d_h^{\pi,\bar{\pi}}, Q_h^{\pi',\bar{\pi}} - EV_h^{\pi',\bar{\pi}} \right\rangle.$$

After this, taking sum from $h = 1$ to H and recognizing that for all policy pairs (π, π') it holds that $V_{H+1}^{\pi,\pi'} = 0$, it holds that

$$\sum_{h=1}^{H} \left\langle d_h^{\pi,\bar{\pi}}, r_h \right\rangle - \left\langle d_1^{\pi,\bar{\pi}}, EV_1^{\pi',\bar{\pi}} \right\rangle = \sum_{h=1}^{H} \left\langle d_h^{\pi,\bar{\pi}}, Q_h^{\pi',\bar{\pi}} - EV_h^{\pi',\bar{\pi}} \right\rangle.$$

Then, notice that for all policies $\pi, \bar{\pi}$ it holds that $\sum_{h=1}^{H} \left\langle d_h^{\pi,\bar{\pi}}, r_h \right\rangle = \langle \nu_1, V^{\pi,\bar{\pi}} \rangle$. Plugging in these observations, we get

$$\left\langle \nu_1, V^{\pi,\bar{\pi}} - V^{\pi',\bar{\pi}} \right\rangle = \sum_{h=1}^{H} \left\langle d_h^{\pi,\bar{\pi}}, Q_h^{\pi',\bar{\pi}} - EV_h^{\pi',\bar{\pi}} \right\rangle.$$

Therefore, expanding the expectation, and noticing that $d_h^{\pi,\bar{\pi}}(s,a,s',a'|s_1) = d_h^{\pi}(s,a|s_1) d_h^{\bar{\pi}}(s',a'|s_1)$ for all h, s, a, s', a' and conditioning s_1, we get that

$$\begin{aligned}
&\left\langle \nu_1, V^{\pi,\bar{\pi}} - V^{\pi',\bar{\pi}} \right\rangle \\
&= \mathbb{E}_{s_1\sim\nu_1} \sum_{h=1}^{H} \mathbb{E}_{s\sim d_h^{\pi}|s_1} \left[\left\langle \mathbb{E}_{s',a'\sim d_h^{\bar{\pi}}|s_1} Q_h^{\pi',\bar{\pi}}(s,\cdot,s',a'), \pi_h(\cdot|s,s_1) - \pi'_h(\cdot|s,s_1) \right\rangle \right].
\end{aligned}$$

□

E.3 Proof of Thm. 1

Proof. We set $\bar{\pi}_h^T(a_h|s_h) = \frac{\sum_{t=1}^{T} d_h^{\pi^t}(s_h,a_h)}{\sum_{t=1}^{T} d_h^{\pi^t}(s_h)}$, where $d(s)$ is the marginal distribution of $d(s,a)$ on state s, and $\bar{\pi}^T = (\bar{\pi}_h^T)_{h=1}^H$. We shows that $d_h^{\bar{\pi}^T} = \frac{1}{T}\sum_{t=1}^{T} d_h^{\pi^t}$ by induction. $h = 1$ holds by definition. Assuming on step h, the equation holds, we have

$$\begin{aligned}
d_{h+1}^{\bar{\pi}^T}(s_{h+1},a_{h+1}) &= d_{h+1}^{\bar{\pi}^T}(s_{h+1}) \bar{\pi}_{h+1}^T(a_{h+1}|s_{h+1}) \\
&= \sum_{s_h,a_h\sim\bar{\pi}_h^T(\cdot|s_h)} d_h^{\bar{\pi}^T}(s_h,a_h) f(s_{h+1}|s_h,a_h) \bar{\pi}_{h+1}^T(a_{h+1}|s_{h+1}) \\
&= \sum_{s_h,a_h\sim\bar{\pi}_h^T(\cdot|s_h)} \frac{1}{T}\sum_{t=1}^{T} d_h^{\pi^t}(s_h,a_h) f(s_{h+1}|s_h,a_h) \bar{\pi}_{h+1}^T(a_{h+1}|s_{h+1}) \\
&= \frac{1}{T}\sum_{t=1}^{T} d_{h+1}^{\pi^t}(s_{h+1}) \bar{\pi}_{h+1}^T(a_{h+1}|s_{h+1}) \\
&= \frac{1}{T}\sum_{t=1}^{T} d_{h+1}^{\pi^t}(s_{h+1},a_{h+1}),
\end{aligned}$$

where the last equation holds by definition of $\bar{\pi}^T_{h+1}$. Therefore, $h+1$ holds, and the $\bar{\pi}^T$ satisfy all equations for $h \in [H]$.

Using the value difference Lemma 2 we have that for any $\pi^\star \in \Pi$

$$\begin{aligned}
&\left\langle \nu_1, V^{\pi^\star,\pi^t} - V^{\pi^t,\pi^t} \right\rangle \\
&= \mathbb{E}_{s_1\sim\nu_1} \sum_{h=1}^{H} \mathbb{E}_{s\sim d_h^{\pi^\star}|s_1} \left[\left\langle \mathbb{E}_{s',a'\sim d_h^{\pi^t}|s_1} Q_h^{\pi^t,\pi^t}(s,\cdot,s',a'), \pi_h^\star(\cdot|s) - \pi_h^t(\cdot|s) \right\rangle \right].
\end{aligned}$$

Therefore, summing over t from $t=1$ to T we obtain

$$\begin{aligned}
&\sum_{t=1}^{T} \left\langle \nu_1, V^{\pi^\star,\pi^t} - V^{\pi^t,\pi^t} \right\rangle \\
&= \mathbb{E}_{s_1\sim\nu_1} \sum_{h=1}^{H} \mathbb{E}_{s\sim d_h^{\pi^\star}|s_1} \left[\sum_{t=1}^{T} \left\langle \mathbb{E}_{s',a'\sim d_h^{\pi^t}|s_1} Q_h^{\pi^t,\pi^t}(s,\cdot,s',a'), \pi_h^\star(\cdot|s) - \pi_h^t(\cdot|s) \right\rangle \right].
\end{aligned}$$

Therefore, we need to control the local regrets at each state s with loss $\ell_h^t(s,s_1) := \mathbb{E}_{s',a'\sim d_h^{\pi^t}|s_1} Q_h^{\pi^t,\pi^t}(s,\cdot,s',a')$. To this end, we can invoke a standard convergence result for online mirror descent [30, Theorem 6.10] we obtain that at each state we have

$$\sum_{t=1}^{T} \left\langle \ell_h^t(s,s_1), \pi^\star(\cdot|s) - \pi^t(\cdot|s) \right\rangle \le \frac{D(\pi^\star(\cdot|s), \pi^1(\cdot|s))}{\beta} + \beta \sum_{t=1}^{T} \|\ell_h^t(s,s_1)\|_\infty^2.$$

Now, noticing that we have $\|\ell_h^t(s,s_1)\|_\infty \le H$ it holds that

$$\sum_{t=1}^{T} \left\langle \ell_h^t(s), \pi_h^\star(\cdot|s) - \pi_h^t(\cdot|s) \right\rangle \le \frac{D(\pi_h^\star(\cdot|s), \pi_h^1(\cdot|s))}{\beta} + \beta T H^2.$$

Finally, using the assumption that $\pi^1(a|s) \ge \underline{\pi}$ for all $s,a \in \mathcal{S}\times\mathcal{A}$ it holds that $D(\pi^\star(\cdot|s), \pi^1(\cdot|s)) \le \log \underline{\pi}^{-1}$. Therefore, choosing $\beta = \sqrt{\frac{\log \underline{\pi}^{-1}}{TH^2}}$ it holds that

$$\sum_{t=1}^{T} \left\langle \ell_h^t(s,s_1), \pi^\star(\cdot|s) - \pi^t(\cdot|s) \right\rangle \le 2H\sqrt{T \log \underline{\pi}^{-1}}.$$

Thus, we conclude that

$$\sum_{t=1}^{T} \left\langle \nu_1, V^{\pi^\star,\pi^t} - V^{\pi^t,\pi^t} \right\rangle \le 2H^2\sqrt{T \log \underline{\pi}^{-1}}.$$

By the antisimmetry of the game, the same proof steps

$$\sum_{t=1}^{T} \left\langle \nu_1, V^{\pi^t,\pi^t} - V^{\pi^t,\bar{\pi}^\star} \right\rangle \le 2H^2\sqrt{T \log \underline{\pi}^{-1}}.$$

Therefore, it holds that for all $\pi^\star, \bar{\pi}^\star \in \Pi$

$$\sum_{t=1}^{T} \left\langle \nu_1, V^{\pi^\star,\pi^t} - V^{\pi^t,\pi^\star} \right\rangle \le 4H^2\sqrt{T \log \underline{\pi}^{-1}}.$$

Then, define $\bar{\pi}^T$ the trajectory level mixture policy as in [41], i.e. such that $d_h^{\bar{\pi}^T} = \frac{1}{T}\sum_{t=1}^{T} d_h^{\pi^t}$ for all stages $h \in [H]$. This implies that $V^{\bar{\pi}^T,\pi^\star} = \frac{1}{T}\sum_{t=1}^{T} V^{\pi^t,\pi^\star}$, and $V^{\pi^\star,\bar{\pi}^T} = \frac{1}{T}\sum_{t=1}^{T} V^{\pi^\star,\pi_t}$.

Therefore, we have that

$$\left\langle \nu_1, V^{\pi^\star,\bar{\pi}^T} - V^{\bar{\pi}^T,\bar{\pi}^\star} \right\rangle \le 4H^2\sqrt{\frac{\log \underline{\pi}^{-1}}{T}}.$$

Finally, selecting $\pi^\star = \left\langle \nu_1, \arg\max_{\pi\in\Pi} V^{\pi,\bar{\pi}^T} \right\rangle$ and $\bar{\pi}^\star = \left\langle \nu_1, \arg\min_{\pi\in\Pi} V^{\bar{\pi}^T,\pi} \right\rangle$, we obtain that

$$\max_{\pi\in\Pi} \left\langle \nu_1, V^{\pi,\bar{\pi}^T} \right\rangle - \min_{\pi\in\Pi} \left\langle \nu_1, V^{\bar{\pi}^T,\pi} \right\rangle \leq 4H^2 \sqrt{\frac{\log \underline{\pi}^{-1}}{T}}.$$

This implies that

$$\left\langle \nu_1, V^{\bar{\pi}^T,\bar{\pi}^T} \right\rangle - \min_{\pi\in\Pi} \left\langle \nu_1, V^{\bar{\pi}^T,\pi} \right\rangle \leq 4H^2 \sqrt{\frac{\log \underline{\pi}^{-1}}{T}},$$

and

$$\max_{\pi\in\Pi} \left\langle \nu_1, V^{\pi,\bar{\pi}^T} \right\rangle - \left\langle \nu_1, V^{\bar{\pi}^T,\bar{\pi}^T} \right\rangle \leq 4H^2 \sqrt{\frac{\log \underline{\pi}^{-1}}{T}},$$

Therefore, setting $T = \frac{16H^4 \log \underline{\pi}^{-1}}{\epsilon^2}$ we obtain an ϵ-approximate Nash equilibrium. □

E.4 Proof of Theorem 2

Proof. The optimization problem

$$\arg\max_{d\in\tilde{\mathcal{F}}} \min_{d'\in\tilde{\mathcal{F}}} \mathbb{E}_{s_1\sim\nu_1} \sum_{h=1}^{H} \sum_{s,a,s',a'} d_h(s,a|s_1) r(s,a,s',a') d'_h(s',a'|s_1)$$

can be carried out individually over possible initial states. That is for each $s_1 \in \mathrm{supp}(\nu_1)$ we aim at solving

$$\arg\max_{d\in\mathcal{F}_{s_1}} \min_{d'\in\mathcal{F}_{s_1}} \sum_{h=1}^{H} \sum_{s,a,s',a'} d_h(s,a|s_1) r(s,a,s',a') d'_h(s',a'|s_1)$$

To this end for any s_1, we consider $\phi_h^t \in \mathcal{F}$ and $\psi_h^t \in \mathcal{F}$ which are generated by the following updates

$$\phi_h^{t+1} = \arg\max_{\phi\in\mathcal{F}_{s_1}} \beta \left\langle \phi, 2\mathbb{E}_{s',a'\sim\psi^t} r_h(\cdot,\cdot,s',a') - \mathbb{E}_{s',a'\sim\psi^{t-1}} r_h(\cdot,\cdot,s',a') \right\rangle - \mathbb{D}(\phi,\phi_h^t),$$

and

$$\psi_h^{t+1} = \arg\min_{\psi\in\mathcal{F}_{s_1}} \beta \left\langle \psi, 2\mathbb{E}_{s',a'\sim\phi^t} r_h(s',a',\cdot,\cdot) - \mathbb{E}_{s',a'\sim\phi^{t-1}} r_h(s',a',\cdot,\cdot) \right\rangle + \mathbb{D}(\psi,\psi_h^t),$$

In order to prove convergence to an ϵ-approximate Nash equilibrium, we need to control the quantity

$$\mathrm{Gap}_{s_1} = \frac{1}{T} \sum_{h=1}^{H} \sum_{t=1}^{T} \left\langle \theta_h^t, \phi_h^t - \phi_h^\star \right\rangle + \frac{1}{T} \sum_{h=1}^{H} \sum_{t=1}^{T} \left\langle \zeta_h^t, \psi_h^t - \psi_h^\star \right\rangle,$$

for $\theta_h^t(s,a) = \sum_{s',a'} \psi_h^t(s',a') r_h(s,a,s',a')$ and $\zeta_h^t(s,a') = -\sum_{s,a} \phi_h^t(s,a) r_h(s,a,s',a')$. At this point, we bound the local regret term with the `OMPO` update. We have that for any $\phi_h \in \mathcal{F}$

$$\begin{aligned}
\beta \left\langle 2\theta_h^t - \theta_h^{t-1}, \phi_h - \phi_h^{t+1} \right\rangle &= \beta \left\langle \theta_h^t - \theta_h^{t+1}, \phi_h - \phi_h^{t+1} \right\rangle \\
&\quad + \beta \left\langle \theta_h^t + \theta_h^{t+1} - \theta_h^{t-1}, \phi_h - \phi_h^{t+1} \right\rangle \\
&= \beta \left\langle \theta_h^t - \theta_h^{t+1}, \phi_h - \phi_h^{t+1} \right\rangle \\
&\quad + \beta \left\langle \theta_h^t - \theta_h^{t-1}, \phi_h - \phi_h^t \right\rangle \\
&\quad + \beta \left\langle \theta_h^t - \theta_h^{t-1}, \phi_h^t - \phi_h^{t+1} \right\rangle \\
&\quad + \beta \left\langle \theta_h^{t+1}, \phi_h - \phi_h^{t+1} \right\rangle.
\end{aligned}$$

At this point, we work on the third summand above

$$\beta \left\langle \theta_h^t - \theta_h^{t-1}, \phi_h^t - \phi_h^{t+1} \right\rangle \leq \beta^2 \lambda \|\theta_h^t - \theta_h^{t-1}\|_\infty^2 + \frac{1}{4\lambda} \|\phi_h^t - \phi_h^{t+1}\|_1^2.$$

In addition, we have that$\|\theta_h^t - \theta_h^{t-1}\|_\infty \leq \|\psi_h^t - \psi_h^{t-1}\|_1$ and we can apply the $1/\lambda$ strong convexity of $\mathbb{D}$, we obtain

$$\beta \left\langle \theta_h^t - \theta_h^{t-1}, \phi_h^t - \phi_h^{t+1} \right\rangle \leq \lambda \beta^2 \|\psi_h^t - \psi_h^{t-1}\|_1^2 + \frac{1}{2}\mathbb{D}(\phi_h^{t+1}, \phi_h^t).$$

On the other hand, by the three point identity we have that for all $\phi \in \mathcal{F}$

$$\mathbb{D}(\phi_h, \phi_h^{t+1}) = \mathbb{D}(\phi_h, \phi_h^t) - \mathbb{D}(\phi_h^{t+1}, \phi_h^t) + \left\langle \nabla \mathbb{D}(\phi_h^{t+1}, \phi_h^t), \phi_h^{t+1} - \phi_h \right\rangle$$

Then, using the property of the update rule, we obtain that

$$\left\langle \nabla \mathbb{D}(\phi_h^{t+1}, \phi_h^t), \phi_h^{t+1} - \phi_h \right\rangle \leq \beta \left\langle 2\theta_h^t - \theta_h^{t-1}, \phi_h - \phi_h^{t+1} \right\rangle.$$

Putting all the pieces together we have that

$$\begin{aligned}
\mathbb{D}(\phi_h, \phi_h^{t+1}) &\leq \mathbb{D}(\phi_h, \phi_h^t) - \mathbb{D}(\phi_h^{t+1}, \phi_h^t) + \beta \left\langle 2\theta_h^t - \theta_h^{t-1}, \phi_h - \phi_h^{t+1}(\cdot|s) \right\rangle \\
&\leq \mathbb{D}(\phi_h, \phi_h^t) - \mathbb{D}(\phi_h^{t+1}, \phi_h^t) \\
&\quad + \beta \left\langle \theta_h^t - \theta_h^{t+1}, \phi_h - \phi_h^{t+1} \right\rangle \\
&\quad + \beta \left\langle \theta_h^t - \theta_h^{t-1}, \phi_h - \phi_h^t \right\rangle \\
&\quad + \beta^2 \|\psi_h^t - \psi_h^{t-1}\|_1^2 + \frac{1}{2}\mathbb{D}(\phi_h^{t+1}, \phi_h^t) \\
&\quad + \beta \left\langle \theta_h^{t+1}, \phi_h - \phi_h^{t+1} \right\rangle.
\end{aligned}$$

Now, rearranging the terms we get

$$\begin{aligned}
\beta \left\langle \theta_h^{t+1}, \phi_h - \phi_h^{t+1} \right\rangle &\leq \mathbb{D}(\phi_h, \phi_h^t) - \mathbb{D}(\phi_h, \phi_h^{t+1}) - \frac{1}{2}\mathbb{D}(\phi_h^{t+1}, \phi_h^t) \\
&\quad + \beta \left\langle \theta_h^t - \theta_h^{t+1}, \phi_h - \phi_h^{t+1} \right\rangle \\
&\quad + \beta \left\langle \theta_h^t - \theta_h^{t-1}, \phi_h - \phi_h^t \right\rangle \\
&\quad + \beta^2 \lambda \|\psi_h^t - \psi_h^{t-1}\|_1^2.
\end{aligned}$$

Now, denoting $\Phi_\phi^t := \mathbb{D}(\phi_h, \phi_h^t) + \beta \left\langle \theta_h^t - \theta_h^{t-1}, \phi_h - \phi_h^t \right\rangle$ and summing over t we obtain

$$\beta \sum_{t=1}^T \left\langle \theta_h^t, \phi_h - \phi_h^t \right\rangle \leq \sum_{t=1}^T \Phi_\phi^{t-1} - \Phi_\phi^t - \frac{1}{2}\sum_{t=1}^T \mathbb{D}(\phi_h^t, \phi_h^{t-1}) + \beta^2 \lambda \sum_{t=1}^T \|\psi_h^{t-1} - \psi_h^{t-2}\|_1^2.$$

Similarly we get

$$\beta \sum_{t=1}^T \left\langle \zeta^t(s, \cdot), \psi_h^t - \psi_h^t \right\rangle \leq \sum_{t=1}^T \Phi_\psi^{t-1} - \Phi_\psi^t - \frac{1}{2}\sum_{t=1}^T \mathbb{D}(\psi_h^t, \psi_h^{t-1}) + \beta^2 \lambda \sum_{t=1}^T \|\phi_h^{t-1} - \psi_h^{t-2}\|_1^2.$$

Now, using $1/\lambda$ strong convexity of $\mathbb{D}$ and summing the two terms we have that

$$\begin{aligned}
\beta T \mathrm{Gap}_{s_1,h} &\leq \Phi^0 - \Phi^{T-1} - \frac{1}{2}\sum_{t=1}^T (\mathbb{D}(\psi_h^t, \psi_h^{t-1}) + \mathbb{D}(\phi_h^t, \phi_h^{t-1})) \\
&\quad + 2\beta^2 \lambda \sum_{t=1}^T (\mathbb{D}(\psi_h^{t-1}, \psi_h^{t-2}) + \mathbb{D}(\phi_h^{t-1}, \phi_h^{t-2})),
\end{aligned}$$

with $\Phi^t = \Phi_\phi^t + \Phi_\psi^t$. At this point, setting $\beta \leq \frac{1}{\sqrt{2\lambda}}$, we obtain a telescopic sum

$$\begin{aligned}
&\beta T \mathrm{Gap}_{s_1,h} \\
&\leq \Phi^0 - \Phi^{T-1} - \frac{1}{2}\sum_{t=1}^T (\mathbb{D}(\psi_h^t, \psi_h^{t-1}) + \mathbb{D}(\phi_h^t, \phi_h^{t-1}) - \mathbb{D}(\psi_h^{t-1}, \psi_h^{t-2}) - \mathbb{D}(\phi_h^{t-1}, \phi_h^{t-2})) \\
&\leq \Phi^0 - \Phi^{T-1} + \frac{1}{2}\left(\mathbb{D}(\psi_h^1, \psi_h^0) + \mathbb{D}(\phi_h^1, \phi_h^0)\right).
\end{aligned}$$

Now recalling that by assumption the occupancy measure of the reference policy is lower bounded, i.e. $d^{\pi^1} \geq \underline{d}$, we can upper bound $\Phi^0 - \Phi^T \leq 2\log \underline{d}^{-1} + 8\beta$ that allows to conclude that for all $n \in [N]$ and setting $\psi_h^0 = \psi_h^1$ and $\phi_h^1 = \phi_h^0$,

$$\mathrm{Gap}_{s_1,h} \leq \frac{2\log \underline{d}^{-1} + 8\beta}{\beta T} \leq \frac{10 \log \underline{d}^{-1}}{\beta T}.$$

Now, notice that Gap can be rewritten as

$$\begin{aligned}
\mathrm{Gap}_{s_1} &= \sum_{h=1}^{H} \mathrm{Gap}_{s_1,h} \\
&= \frac{1}{T}\sum_{t=1}^{T}\sum_{h=1}^{H}\sum_{s,a,s',a'} \psi_h^\star(s',a') r_h(s,a,s',a') \phi_h^t(s,a) \\
&\qquad - \frac{1}{T}\sum_{t=1}^{T}\sum_{h=1}^{H}\sum_{s,a,s',a'} \psi_h^t(s',a') r_h(s,a,s',a') \phi_h^\star(s,a) \\
&= \sum_{h=1}^{H}\sum_{s,a,s',a'} \psi_h^\star(s',a') r_h(s,a,s',a') \frac{1}{T}\sum_{t=1}^{T} \phi_h^t(s,a) \\
&\qquad - \sum_{h=1}^{H}\sum_{s,a,s',a'} \frac{1}{T}\sum_{t=1}^{T} \psi_h^t(s',a') r_h(s,a,s',a') \phi_h^\star(s,a) \\
&= \sum_{h=1}^{H}\sum_{s,a,s',a'} \psi_h^\star(s',a') r_h(s,a,s',a') \bar{\phi}_h(s,a) - \sum_{h=1}^{H}\sum_{s,a,s',a'} \bar{\psi}_h(s',a') r_h(s,a,s',a') \phi_h^\star(s,a)\,.
\end{aligned}$$

At this point, let us define $\pi_\phi^{\mathrm{out}}(a|s) = \frac{\bar{\phi}(s,a)}{\sum_a \bar{\phi}(s,a)}$ and $\pi_\psi^{\mathrm{out}}(a|s) = \frac{\bar{\psi}(s,a)}{\sum_a \bar{\psi}(s,a)}$. For such policies and by appropriate choice for $\psi^\star$ and $\phi^\star$ it follows that

$$\mathrm{Gap}_{s_1} = \max_{\psi} V^{\pi_\phi^{\mathrm{out}},\psi}(s_1) - \min_{\phi} V^{\phi,\pi_\psi^{\mathrm{out}}}(s_1).$$

By the bound on Gap_{s_1} for each $s_1 \in \mathrm{supp}(\nu_1)$, it follows that

$$\left\langle \nu_1, \max_{\psi} V^{\pi_\phi^{\mathrm{out}},\psi} - \min_{\phi} V^{\phi,\pi_\psi^{\mathrm{out}}} \right\rangle = \mathbb{E}_{s_1\sim\nu_1} \mathrm{Gap}_{s_1} \leq \frac{10H \log \underline{d}^{-1}}{\beta T},$$

therefore $T \geq \frac{10H \log \underline{d}^{-1}}{\beta\epsilon}$. The proof is concluded invoking Thm. 3 that ensures that the policies π_ψ^{out} and π_ϕ^{out} coincide. □

E.5 Proof of Theorem 3

Proof. Let us consider two players performing the following updates

$$\phi_h^{t+1} = \arg\max_{\phi\in\mathcal{F}_{s_1}} \beta \left\langle \phi, 2\mathbb{E}_{s',a'\sim\psi^t} r_h(\cdot,\cdot,s',a') - \mathbb{E}_{s',a'\sim\psi^{t-1}} r_h(\cdot,\cdot,s',a') \right\rangle - \mathbb{D}(\phi,\phi_h^t),$$

and

$$\psi_h^{t+1} = \arg\min_{\psi\in\mathcal{F}_{s_1}} \beta \left\langle \psi, 2\mathbb{E}_{s',a'\sim\phi^t} r_h(s',a',\cdot,\cdot) - \mathbb{E}_{s',a'\sim\phi^{t-1}} r_h(s',a',\cdot,\cdot) \right\rangle + \mathbb{D}(\psi,\psi_h^t).$$

The goal is to proof that the iterates generated by the two updates are identical. We will prove this fact by induction. The base case holds by initialization which gives $\phi_h^0 = \psi_h^0$ for all $h \in [H]$. Then,

let us assume by the induction step that $\psi_h^t = \phi_h^t$ for all $h \in [H]$, then

$$
\begin{aligned}
&\phi_h^{t+1} \\
&= \arg\max_{\phi \in \mathcal{F}_{s_1}} \beta \left\langle \phi, 2\mathbb{E}_{s',a' \sim \psi^t} r_h(\cdot,\cdot,s',a') - \mathbb{E}_{s',a' \sim \psi^{t-1}} r_h(\cdot,\cdot,s',a') \right\rangle - \mathbb{D}(\phi, \phi_h^t) \\
&= \arg\max_{\phi \in \mathcal{F}_{s_1}} \beta \left\langle \phi, -2\mathbb{E}_{s',a' \sim \psi^t} r_h(s',a',\cdot,\cdot) + \mathbb{E}_{s',a' \sim \psi^{t-1}} r_h(s',a',\cdot,\cdot) \right\rangle - \mathbb{D}(\phi, \phi_h^t) + \beta \langle \phi, \mathbf{1} \rangle \\
&\text{(Antisymmetric Reward)} \\
&= \arg\max_{\phi \in \mathcal{F}_{s_1}} \beta \left\langle \phi, -2\mathbb{E}_{s',a' \sim \psi^t} r_h(s',a',\cdot,\cdot) + \mathbb{E}_{s',a' \sim \psi^{t-1}} r_h(s',a',\cdot,\cdot) \right\rangle - \mathbb{D}(\phi, \phi_h^t) + \beta \\
&\text{(Normalization of } \phi) \\
&= \arg\max_{\phi \in \mathcal{F}_{s_1}} \beta \left\langle \phi, -2\mathbb{E}_{s',a' \sim \psi^t} r_h(s',a',\cdot,\cdot) + \mathbb{E}_{s',a' \sim \psi^{t-1}} r_h(s',a',\cdot,\cdot) \right\rangle - \mathbb{D}(\phi, \phi_h^t) \\
&(\beta \text{ does not depend on } \phi) \\
&= \arg\max_{\phi \in \mathcal{F}_{s_1}} \beta \left\langle \phi, -2\mathbb{E}_{s',a' \sim \phi^t} r_h(s',a',\cdot,\cdot) + \mathbb{E}_{s',a' \sim \phi^{t-1}} r_h(s',a',\cdot,\cdot) \right\rangle - \mathbb{D}(\phi, \psi_h^t) \\
&\text{(Inductive Hypothesis)} \\
&= \arg\min_{\phi \in \mathcal{F}_{s_1}} \beta \left\langle \psi, 2\mathbb{E}_{s',a' \sim \phi^t} r_h(s',a',\cdot,\cdot) - \mathbb{E}_{s',a' \sim \phi^{t-1}} r_h(s',a',\cdot,\cdot) \right\rangle + \mathbb{D}(\psi, \psi_h^t) \\
&\text{(Renaming the optimization variable and } \arg\max_x f(x) = \arg\min_x -f(x)) \\
&= \psi_h^{t+1}.
\end{aligned}
$$

□

F Implementation of Algorithm 2 with updates over policies.

In this section, we explain how the update in Algorithm 2 for different choices of $\mathbb{D}$. In both cases, we will derive an update that can be summarized by following template. Let us define $r_h^t(s,a) = \mathbb{E}_{s',a' \sim d_h^t} r(s,a,s',a')$ and $r_h^{t-1}(s,a) = \mathbb{E}_{s',a' \sim d_h^{t-1}} r(s,a,s',a')$

- Compute the Q_h^t function corresponding to the reward function $2r_h^t - r_h^{t-1}$ minimizing a loss function that depends on the choice of $\mathbb{D}$.
- Update the policy as

$$\pi_h^{t+1}(a|s) \propto \pi_h^t(a|s) \exp\left(\beta Q_h^t(s,a)\right).$$

Finally, in Appx. F.3 we show that for $\mathbb{D}$ being the conditional relative entropy and for β small enough the value function Q_h^t is well approximated by the standard Bellman equations.

Remark 4. *Both choices of the Bregman divergence are* 1 *strongly convex so Thm. 2 applies with* $\lambda = 1$.

In the following we consider a generic reward function $\tilde{r}$. In our setting, we will apply the following results for $\tilde{r}_h = 2r_h^t - r_h^{t-1}$ in order to implement the updates of Alg. 2 for the different values of h and t.

F.1 $\mathbb{D}$ chosen as the sum of conditional and relative entropy

In this section, we explain how to implement the occupancy measure update in Algorithm 2 over policies. We use the machinery for single agent MDPs introduced in [6]. In particular, we consider the Bregman divergence given by the sum of the relative entropy $D(d,d') = \sum_{s,a} d(s,a) \log\left(\frac{d(s,a)}{d'(s,a)}\right)$ and of the conditional relative entropy given, i.e. $H(d,d') = \sum_{s,a} d(s,a) \log\left(\frac{\pi_d(a|s)}{\pi_{d'}(a|s)}\right)$ with $\pi_d(a|s) = d(s,a)/\sum_a d(s,a)$. Under this choice for $\mathbb{D}$, the update of Algorithm 2 for particular

values of h, t, s_1 corresponds to the solution of the following optimization program

$$d_h^{t+1} = \arg\max_{d \in \Delta^H} \sum_{h=1}^{H} \langle d_h, \tilde{r}_h \rangle - \frac{1}{\beta} D(d_h, d_h^t) - \frac{1}{\beta} H(d_h, d_h^t),$$
$$\text{s.t.} \quad E^T d_h = F^T d_{h-1} \quad \forall h \in [H]. \tag{Update I}$$

Theorem 7. *The policy π_h^{t+1} with occupancy measure d_h^{t+1} defined in Eq. (Update I) can be computed as follows*

$$\pi_h^{t+1}(a|s) \propto \pi_h^t(a|s) \exp\left(\beta Q_h^t(s,a)\right),$$

where Q_h^t is the minimizer of the following loss

$$\frac{1}{\beta} \sum_{h=1}^{H} \log \sum_{s,a} \mu_h^t(s,a) \exp\left(\beta(2\tilde{r}_h + PV_{h+1} - Q_h)(s,a)\right) + \langle \nu_1, V_1 \rangle,$$

while V_{h+1}^t is given by the following closed form.

$$V_{h+1}^t(s) = \frac{1}{\beta} \log \sum_a \pi_h^t(a|s) \exp(\beta Q_{h+1}^t(s,a)).$$

Proof. Let us introduce an auxiliary variable $\mu_h = d_h$ for all $h \in [H]$, then we can rewrite the optimization program as

$$\arg\max_{d \in \Delta^H} \max_{\mu \in \Delta^H} \sum_{h=1}^{H} \langle \mu_h, \tilde{r}_h \rangle - \frac{1}{\beta} D(\mu_h, \mu_h^t) - \frac{1}{\beta} H(d_h, d_h^t),$$
$$\text{s.t.} \quad E^T d_h = F^T \mu_{h-1} \quad \forall h \in [H],$$
$$\text{s.t.} \quad \mu_h = d_h \quad \forall h \in [H].$$

Then, by Lagrangian duality we have that

$$\begin{aligned}
&\max_{d \in \Delta^H} \max_{\mu \in \Delta^H} \min_{Q,V} \sum_{h=1}^{H} \langle \mu_h, \tilde{r} \rangle - \frac{1}{\beta} D(\mu_h, \mu_h^t) - \frac{1}{\beta} H(d_h, d_h^t) \\
&\quad + \left\langle -E^T d_h + F^T \mu_{h-1}, V_h \right\rangle + \langle Q_h, d_h - \mu_h \rangle \\
&\quad = \max_{d \in \Delta^H} \max_{\mu \in \Delta^H} \min_{Q,V} \sum_{h=1}^{H} \langle \mu_h, \tilde{r} + FV_{h+1} - Q_h \rangle + \langle d_h, Q_h - EV_h \rangle \\
&\quad - \frac{1}{\beta} D(\mu_h, \mu_h^t) - \frac{1}{\beta} H(d_h, d_h^t) \\
&\quad + \langle \nu_1, V_1 \rangle = \mathcal{L}^\star .
\end{aligned}$$

Then, by Lagrangian duality, we have that the objective is unchanged by swapping the min and max

$$\begin{aligned}
\mathcal{L}^\star = &\min_{Q,V} \max_{d \in \Delta^H} \max_{\mu \in \Delta^H} \sum_{h=1}^{H} \langle \mu_h, \tilde{r}_h + FV_{h+1} - Q_h \rangle + \langle d_h, Q_h - EV_h \rangle \\
&- \frac{1}{\beta} D(\mu_h, \mu_h^t) - \frac{1}{\beta} H(d_h, d_h^t) + \langle \nu_1, V_1 \rangle .
\end{aligned}$$

The inner maximization is solved by the following values

$$\mu_h^+(Q,V) \propto \mu_h^t \odot \exp\left(\beta(\tilde{r}_h + FV_{h+1} - Q_h)\right),$$
$$\pi_h^+(Q,V;s) \propto \pi_h^t(\cdot|s) \odot \exp\left(\beta(Q_h(s,\cdot) - V_h(s))\right),$$

where $\odot$ denotes the elementwise product between vectors. Then, replacing these values in the Lagrandian and parameterizing the functions V_h by the functions Q_h to ensure normalization of the policy, i.e. $V_h(s) = \frac{1}{\beta} \log \sum_a \pi_h^t(a|s) \exp(\beta Q_h(s,a))$ we have that

$$\mathcal{L}^\star = \min_Q \frac{1}{\beta} \sum_{h=1}^{H} \log \sum_{s,a} \mu_h^t(s,a) \exp\left(\beta(\tilde{r}_h + FV_{h+1} - Q_h)(s,a)\right) + \langle \nu_1, V_1 \rangle .$$

Therefore, denoting

$$Q_h^t = \arg\min_Q \frac{1}{\beta} \sum_{h=1}^{H} \log \sum_{s,a} \mu_h^t(s,a) \exp\left(\beta(\tilde{r}_h + FV_{h+1} - Q_h)(s,a)\right) + \langle \nu_1, V_1 \rangle ,$$

and $V_h^t = \frac{1}{\beta} \log \sum_a \pi_h^t(a|s) \exp(\beta Q_h^t(s,a))$, we have that the policy $\pi_h^{t+1}(\cdot|s) = \pi_h^+(Q^t, V^t; s)$ has occupancy measure equal to d_h^{t+1} for all $h \in [H]$. This is because by the constraints of the problem we have that d_h^{t+1} satisfies the Bellman flow constraints and that the policy π_h^{t+1} satisfies $\pi_h^{t+1}(a|s) = d_h^t(s,a) / \sum_a d_h^t(s,a)$. □

F.2 $\mathbb{D}$ chosen as conditional relative entropy [28]

In this section, we study the update considering $\mathbb{D}$ chosen as sum of the conditional relative entropy over the stages h' s.t. $1 \le h' \le h$, i.e. we study the following update.[4]

$$\begin{aligned} d^{t+1} = \arg\max_{d \in \Delta^H} & \sum_{h=1}^{H} \left(\langle d_h, \tilde{r}_h \rangle - \frac{1}{\beta} \sum_{h'=1}^{h} H(d_{h'}, d_{h'}^t) \right), \\ & \text{s.t.} \quad E^T d_h = F^T d_{h-1} \quad \forall h \in [H]. \end{aligned} \tag{2}$$

Theorem 8. *The policy π_h^{t+1} with occupancy measure d_h^{t+1} defined in Eq.* (2) *can be computed as follows*

$$\pi_h^{t+1}(a|s) \propto \pi_h^t(a|s) \exp\left(\frac{\beta}{H-h+1} (Q_h^t(s,a)) \right),$$

where Q_h^t and V_{h+1}^t satisfies the following recursion

$$\begin{aligned} Q_h^t &= \tilde{r}_h + FV_{h+1}^t \\ V_{h+1}^t(s) &= \frac{H-h+1}{\beta} \log \sum_a \pi_h^t(a|s) \exp\left(\frac{\beta}{H-h+1} Q_{h+1}^t(s,a) \right). \end{aligned}$$

Remark 5. *The above recurrencies are sometimes called soft Bellman equations [51, 14].*

Proof. Let us introduce an auxiliary variable $\mu_h = d_h$ for all $h \in [H]$, then we can rewrite the optimization program as

$$\begin{aligned} \arg\max_{d \in \Delta^H} \max_{\mu} & \sum_{h=1}^{H} \left(\langle \mu_h, \tilde{r}_h \rangle - \frac{1}{\beta} \sum_{h'=1}^{h} H(d_{h'}, d_{h'}^t) \right) \\ & \text{s.t.} \quad E^T d_h = F^T \mu_{h-1} \quad \forall h \in [H] \\ & \text{s.t.} \quad \mu_h = d_h \quad \forall h \in [H]. \end{aligned}$$

[4] The sum over previous stages is taken to ensure 1-strong convexity. Indeed, it holds that $\sum_{h'=1}^{h} H(d_{h'}, d'_{h'}) \ge D(d_h, d'_h) \ge \frac{1}{2} \|d_h - d'_h\|_1^2$. The first inequality is proven in [29, Lemma 7].

Notice that importantly, we do not constraint the variable μ. Then, by Lagrangian duality we have that

$$\begin{aligned}
&\max_{d\in\Delta^H}\max_{\mu}\min_{Q,V}\sum_{h=1}^{H}\langle\mu_h,\tilde{r}_h\rangle-\frac{1}{\beta}\sum_{h'=1}^{h}H(d_{h'},d^t_{h'})\\
&\quad+\left\langle -E^Td_h+F^T\mu_{h-1},V_h\right\rangle+\langle Q_h,d_h-\mu_h\rangle\\
&\quad=\max_{d\in\Delta^H}\max_{\mu}\min_{Q,V}\sum_{h=1}^{H}\langle\mu_h,\tilde{r}_h+FV_{h+1}-Q_h\rangle+\langle d_h,Q_h-EV_h\rangle\\
&\qquad-\frac{1}{\beta}\sum_{h'=1}^{h}H(d_{h'},d^t_{h'})+\langle\nu_1,V_1\rangle\\
&\quad=\min_{Q,V}\max_{d\in\Delta^H}\max_{\mu}\sum_{h=1}^{H}\langle\mu_h,\tilde{r}_h+FV_{h+1}-Q_h\rangle+\langle d_h,Q_h-EV_h\rangle\\
&\qquad-\frac{H-h+1}{\beta}H(d_h,d^t_h)+\langle\nu_1,V_1\rangle=\tilde{\mathcal{L}}^\star,
\end{aligned}$$

where the last equality holds by Lagrangian duality and by $\sum_{h=1}^{H}\sum_{h'=1}^{h}H(d_{h'},d^t_{h'})=\sum_{h=1}^{H}(H-h+1)H(d_{h'},d^t_{h'})$. Now since μ is unconstrained we have that $\max_\mu\sum_{h=1}^{H}\langle\mu_h,\tilde{r}_h+FV_{h+1}-Q_h\rangle$ is equivalent to impose the constraint $\tilde{r}_h+FV_{h+1}=Q_h$ for all $h\in[H]$. Moreover, as in the proof of Thm. 7 the optimal d_h needs to satisfies that $\pi_{d_h}(a|s)=d_h(s,a)/\sum_a d_h(s,a)$ is equal to $\pi^+_h(Q,V;s)=\pi^t_h(\cdot|s)\odot\exp\left(\frac{\beta}{H-h+1}(Q_h(s,\cdot)-V_h(s))\right)$ for $V_h(s)=\frac{H-h+1}{\beta}\log\sum_a\pi^t_h(a|s)\exp(\frac{\beta}{H-h+1}Q_h(s,a))$. Plugging in, these facts in the expression for $\tilde{\mathcal{L}}^\star$, we have that

$$\tilde{\mathcal{L}}^\star=\min_Q\langle\nu_1,V_1\rangle\quad\text{s.t.}\quad\tilde{r}_h+FV_{h+1}=Q_h\quad\forall h\in[H].$$

Since the above problem as only one feasible point, we have that the solution is the sequence Q^t_h satisfying the recursion $\tilde{r}_h+FV^t_{h+1}=Q^t_h$ with $V^t_h(s)=\frac{H-h+1}{\beta}\log\sum_a\pi^t_h(a|s)\exp(\frac{\beta}{H-h+1}Q^t_h(s,a))$. □

F.3 Approximating soft Bellman equations by standard Bellman equations.

Unfortunately, implementing the update for the V value as in Theorem 7 is often numerically instable. In this section, we show a practical approximation which is easy to implement and shown to be accurate for β sufficiently small.

Theorem 9. *Let us denote $\beta_h=\frac{\beta}{H-h+1}$ and let us assume that the values Q^t_h generated by the soft Bellman equations in Thm. 8 are uniformly upper bounded by $Q_{\max}$, and let us choose $\beta_h\le\frac{1}{Q_{\max}}$ for all $h\in[H]$. Then, it holds that*

$$\left\langle\pi^t_h(\cdot|s),Q^t_h(s,\cdot)\right\rangle\le\frac{1}{\beta_h}\log\sum_a\pi^t_h(a|s)\exp(\beta_hQ^t_h(s,a))\le\left\langle\pi^t_h(\cdot|s),Q^t_h(s,\cdot)\right\rangle+\beta_hQ^2_{\max}\,.$$

Proof.

$$\begin{aligned}
\frac{1}{\beta_h}\log\sum_a\pi^t_h(a|s)\exp(\beta_hQ^t_h(s,a))&\ge\frac{1}{\beta_h}\sum_a\pi^t_h(a|s)\log\exp(\beta_hQ^t_h(s,a))\\
&=\left\langle\pi^t_h(\cdot|s),Q^t_h(s,\cdot)\right\rangle,
\end{aligned}$$

Algorithm 3 MPO (Practical version)

input: reference policy π^1, preference oracle $\mathbb{P}$, learning rate β, number of generated samples K, horizon H, total iteration T.
for $t = 1, 2, \ldots, T$ **do**
  Generates response by sampling $s_1^1 \sim \nu_1$ and $a_h^1 \sim \pi^t(\cdot|s_h^1)$ for $h \in [H]$.
  Clear the dataset buffer $\mathcal{D}_t$.
  for $h = 1, 2, \ldots, H$ **do**
    Set $s_h^K =, \ldots, = s_h^2 = s_h^1$.
    Generate K conversations by sampling $a_{\hat{h}}^{1:K} \sim \pi^t(\cdot|s_{\hat{h}}^{1:K})$ for $\hat{h} \in [h, H]$.
    Estimate $\mathbb{E}_{a_h^{k'}} Q^{\pi^t,\pi^t}(s_h^1, a_h^k, s_h^1, a_h^{k'}), \forall k, k' \in [K]$ via Eq. (5) with query to $\mathbb{P}$.
    Form the data pair $\{(s_h^1, a_h^k, \mathbb{E}_{a_h^{k'}} Q^{\pi^t,\pi^t}(s_h^1, a_h^k, s_h^1, a_h^{k'})\}_{k\in[K]}$, add to $\mathcal{D}_t$.
  end for
  Optimize π_{t+1} over $\mathcal{D}_t$ according to

$$\pi^{t+1} \leftarrow \arg\min_{\pi} \mathbb{E}\bigg(\log\left(\frac{\pi(a_h^k|s_h^1)}{\pi^t(a_h^k|s_h^1)}\right) - \beta\bigg(\mathbb{E}_{a_h^{k'}} Q^{\pi^t,\pi^t}(s_h^1, a_h^k, s_h^1, a_h^{k'}) - \frac{H-h+1}{2}\bigg)\bigg)^2.$$

end for
output: π^{T+1}

where the above inequality holds for Jensen's. For the upper bound, we first use the inequality $e^x \le 1 + x + x^2$ for $x \le 1$ we have that

$$\begin{aligned}
&\frac{1}{\beta_h} \log \sum_a \pi_h^t \exp(\beta_h Q_h^t(s,a)) \\
&\le \frac{1}{\beta_h} \log \sum_a \pi_h^t (1 + \beta_h Q_h^t(s,a) + \beta_h^2 Q_{\max}^2) \quad \text{(Using } Q_h^t(s,a) \le Q_{\max}\text{)} \\
&= \frac{1}{\beta_h} \log (1 + \beta_h \sum_a \pi_h^t(a|s) Q_h^t(s,a) + \beta_h^2 Q_{\max}^2) \\
&\le \frac{1}{\beta_h} \left(\sum_a \pi_h^t(a|s) \beta_h Q_h^t(s,a) + \beta_h^2 Q_{\max}^2 \right) \quad \text{(Using } \log(1+x) \le x\text{)} \\
&\le \left\langle \pi_h^t(\cdot|s), Q_h^t(s,\cdot) \right\rangle + \beta_h Q_{\max}^2.
\end{aligned}$$

□

Remark 6. *Given this result, in the implementation for deep RL experiment, i.e. Algorithm 4 we compute the standard Q value satisfying the standard Bellman equations (given in Lemma 1) rather than the soft Bellman equation in Thm. 7. In virtue of Thm. 9, the approximation is good for β reasonably small.*

G Additional algorithms

Practical Relaxations of Alg. 1 According to Thm. 1, MPO requires the access of the Q function, which is unknown. Next, we are going to develop a practical algorithm to efficiently estimate the Q function and implement Alg. 1. Equivalently, the update in Alg. 1 can be written as

$$\pi_h^{t+1}(a|s) = \frac{\pi_h^t(a|s) \exp\{\beta \mathbb{E}_{s',a' \sim d_h^{\pi^t}|s_1(s)} Q_h^{\pi^t,\pi^t}(s,a,s',a')\}}{Z_h^t(s)}, \tag{3}$$

where $Z_h^t(s)$ is the partition function. Next, we express Eq. (3) as follows:

$$\log \frac{\pi_h^{t+1}(a|s)}{\pi_h^t(a|s)} = \beta \mathbb{E}_{s',a' \sim d_h^{\pi^t}|s_1(s)} Q_h^{\pi^t,\pi^t}(s,a,s',a') - \log Z_h^t(s). \tag{4}$$

Algorithm 4 OMPO (Practical version)

input: reference policy π^1, preference oracle $\mathbb{P}$, learning rate β, number of generated samples K, horizon H, total iteration T, tunable bias term τ.
for $t = 1, 2, \ldots, T$ **do**
  Generates response by sampling $s_1^1 \sim \nu_1$ and $a_h^1 \sim \pi^t(\cdot|s_h^1)$ for $h \in [H]$.
  Clear the dataset buffer $\mathcal{D}_t$.
  for $h = 1, 2, \ldots, H$ **do**
    Set $s_h^K = , \ldots, = s_h^2 = s_h^1$.
    Generate K conversations by sampling $a_{\hat{h}}^{1:K} \sim \pi^t(\cdot|s_{\hat{h}}^{1:K})$ for $\hat{h} \in [h, H]$.
    Estimate $\mathbb{E}_{a_h^{k'}} Q^{\pi^t,\pi^t}(s_h^1, a_h^k, s_h^1, a_h^{k'}) \forall k, k' \in [K]$ via Eq. (5).
    if $t > 1$ **then**
      Estimate $\mathbb{E}_{a_h^{k'}} Q^{\pi^t,\pi^{t-1}}(s_h^1, a_h^k, s_h^1, a_h^{k'}) \quad \forall k, k' \in [K]$ via Eq. (5).
      Add $\{(s_h^1, a_h^k, \mathbb{E}_{a_h^{k'}} Q^{\pi^t,\pi^t}(s_h^1, a_h^k, s_h^1, a_h^{k'}), \mathbb{E}_{a_h^{k'}} Q^{\pi^t,\pi^{t-1}}(s_h^1, a_h^k, s_h^1, a_h^{k'})\}_{k \in [K]}$ into $\mathcal{D}_t$.
    else
      Add $\{(s_h^1, a_h^k, \mathbb{E}_{a_h^{k'}} Q^{\pi^t,\pi^t}(s_h^1, a_h^k, s_h^1, a_h^{k'})\}$ into $\mathcal{D}_t$.
    end if
  end for
  if $t > 1$ **then**
    Optimize π_{t+1} over $\mathcal{D}_t$ according to

$$\pi^{t+1} \leftarrow \arg\min_{\pi} \mathbb{E}\left(\log\left(\frac{\pi(a_h^k|s_h^1)}{\pi^t(a_h^k|s_h^1)}\right) - \beta\left(2\mathbb{E}_{a_h^{k'}} Q^{\pi^t,\pi^t}(s_h^1, a_h^k, s_h^1, a_h^{k'}) - \mathbb{E}_{a_h^{k'}} Q^{\pi^t,\pi^{t-1}}(s_h^1, a_h^k, s_h^1, a_h^{k'}) - \tau\right)\right)^2.$$

  else
    Optimize π_{t+1} over $\mathcal{D}_t$ according to

$$\pi^{t+1} \leftarrow \arg\min_{\pi} \mathbb{E}\left(\log\left(\frac{\pi(a_h^k|s_h^1)}{\pi^t(a_h^k|s_h^1)}\right) - \beta\left(\mathbb{E}_{a_h^{k'}} Q^{\pi^t,\pi^t}(s_h^1, a_h^k, s_h^1, a_h^{k'}) - \frac{H-h+1}{2}\right)\right)^2.$$

  end if
end for
output: π^{T+1}

Next, we approximate Eq. (4) with an approximate solution of the following optimization program

$$\pi^{t+1} = \arg\min_{\pi} \sum_{h=1}^{H} \mathbb{E}_{\substack{s_1 \sim \nu_1 \\ (s_h, a_h) \sim d_h^{\pi^t}|s_1}} \left[\log \frac{\pi(a_h|s_h)}{\pi_h^t(a_h|s_h)} - (\mathbb{E}_{s', a' \sim d_h^{\pi^t}|s_1} Q_h^{\pi^t,\pi^t}(s_h, a_h, s', a') - \log Z_h^t(s_h)) \right]^2.$$

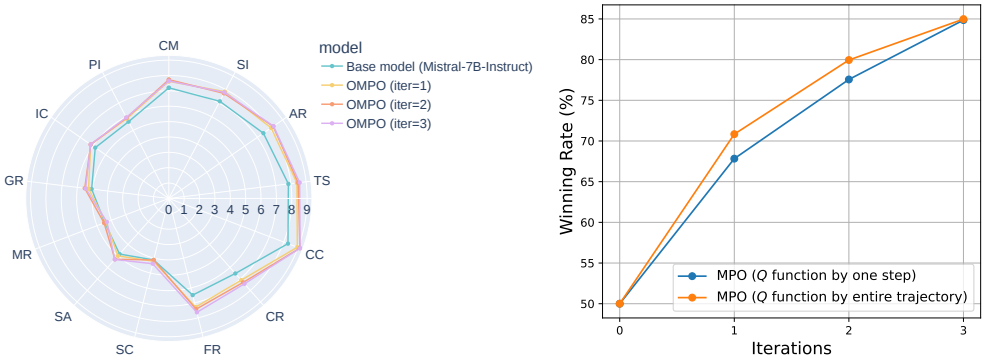
Unfortunately, solving the above minimization exactly is out of hope. The first difficulty is the efficient estimation of $\mathbb{E}_{s', a' \sim d_h^{\pi^t}|s_1} Q_h^{\pi^t,\pi^t}(s_h, a_h, s', a')$. In particular, since s' and s are sampled from the same distribution, we will sample a' from the state s_h and use the Monte Carlo estimator:

$$\mathbb{E}_{a' \sim \pi^t(\cdot|s_h)} Q_h^{\pi^t,\pi^t}(s_h, a_h, s_h, a') \approx \frac{1}{K} \sum_{k=1}^{K} \sum_{\hat{h}=h}^{H} \mathbb{P}([s_{\hat{h},k}, a_{\hat{h},k}], [s'_{\hat{h},k}, a'_{\hat{h},k}]), \tag{5}$$

where the sequences $\left\{(s_{\hat{h},k}, a_{\hat{h},k}, s'_{\hat{h},k}, a'_{\hat{h},k})\right\}_{\hat{h}=h}^{H}$ for $k \in [K]$ are generated by rollouts of the policies pair (π^t, π^t). The second difficulty is $Z_h^t(s)$, which is difficult to compute for large action spaces. In all states s, we replace $\log Z_h^t(s)$ with $\beta \frac{H-h+1}{2}$.

Remark 7. *The heuristics is motivated by the next observation. If the preference between a_h and a'_h in Eq.* (5) *results in a tie, then with such* $\log Z_h^t(s)$*, the solution of Eq.* (5) *is $\pi^{t+1} = \pi^t$, leaving the model unchanged.*

In summary, we provide a practical version of MPO in Alg. 3. In practice, we used a stationary policy that we find to be sufficient to obtain convincing results.



(a) Radar chart on different categories. (b) Winning rate against the base model.

Figure 1: (a): Radar chart result of OMPO on the MT-bench-101 dataset; (b) Winning rate against the base model when using different approximations for the Q functions.

H Additional experimental detail and result on MT-Bench101

The tasks in MT-Bench101 include Context Memory (CM), Anaphora Resolution (AR), Separate Input (SI), Topic Shift (TS), Content Confusion (CC), Content Rephrasing (CR), Format Rephrasing (FR), Self-correction (SC), Self-affirmation (SA), Mathematical Reasoning (MR), General Reasoning (GR), Instruction Clarification (IC), and Proactive Interaction (PI).

Given two conversations $[s_h, a_h]$ and $[s'_h, a'_h]$, PairRM will return a score that indicates the probability that $[s_h, a_h]$ is better than $[s'_h, a'_h]$, which can be used to considered as the preference oracle $\mathbb{P}$ defined in the previous section. For both DPO and SPPO, we sample $K = 5$ complete conversations starting from s_1, and estimate the winning rate $\mathbb{P}([s^k_{H+1}, a^k_{H+1}] \succ (s^{k'}_{H+1}, a^{k'}_{H+1}])\, \forall k, k' \in [K]$. Then we select both the best and worst conversations according to their winning rates against other answers, which is defined as $\frac{1}{K}\sum_{k'=1}^{K} \mathbb{P}([s^k_{H+1}, a^k_{H+1}] \succ [s^{k'}_{H+1}, a^{k'}_{H+1}])$ for the conversation $[s^k_{H+1}, a^k_{H+1}]$. Such a pair is used to train DPO while the winning rate is used to train SPPO. For both iterative DPO and iterative SPPO, we sample $K = 5$ complete conversations starting from s_1, and estimate the winning rate $\mathbb{P}([s^k_{H+1}, a^k_{H+1}] \succ (s^{k'}_{H+1}, a^{k'}_{H+1}])\, \forall k, k' \in [K]$. Then we select both the best and worst conversations according to their winning rates against others, which is defined as $\frac{1}{K}\sum_{k'=1}^{K} \mathbb{P}([s^k_{H+1}, a^k_{H+1}] \succ [s^{k'}_{H+1}, a^{k'}_{H+1}])$ for the conversation $[s^k_{H+1}, a^k_{H+1}]$. Such a pair is used to train DPO while the winning rate is used to train SPPO. For both Step-DPO, MPO, and OMPO, we do the same strategy with starting at s_h. In MPO, and OMPO, we estimate $Q(s_h, a_h, s_h, a'_h)$ by $\mathbb{P}([s_h, a_h], [s_h, a'_h])$ to enhance the efficiency. For OMPO, the $Q^{\pi^t, \pi^{t-1}}$ term is estimated by calculating the winning rate between two answers (the best and the worst) generated by the current policy π^t and the five answers previously generated by π^{t-1}, the τ is selected as zero. Each method is trained with epochs number selected from $\{1, 2\}$, learning rates from $\{5e\text{-}6, 5e\text{-}7\}$, and β values from $\{0.1, 0.01, 0.001\}$. The final model is chosen based on the highest winning rate against the base model, as determined by the PairRM model. We use full-parameter fine-tuning for all methods with bf16 precision. A batch size of 64 is used. The maximum output length and maximum prompt length during training are both set as 2048. We use AdamW optimizer [23] and cosine learning rate schedule [22] with a warmup ratio of 0.1.

Each round of dialogue is rated on a scale of 1 to 10 by GPT-4o mini, with the mean score reported for each dialogue. All methods are run for a total of 3 iterations. The results are summarized in Tab. 1, showing significant improvements over the baselines with the proposed MPO and OMPO approaches. In Fig. 1(a), we present the Radar chart on different categories and we can see that the proposed OMPO leads to improvements generally along the iterations. Fig. 1(b) shows that using the entire trajectory to estimate the Q function can lead to subtle improvement at the first two iterations while it finally achieves a similar winning rate when compared to the one that only use one step.

I Additional experiments on math reasoning

As discussed in Appx. B, our framework can also cover the alignment of chain-of-thought reasoning. In this section, we validate the proposed methods on math reasoning tasks. We select two widely used datasets: MATH [16] and GSM8K [11]. We use Qwen2-7B-Instruct as the base model and follow the same evaluation procedure as in [21]. For step-DPO, we use the checkpoint provided in [21]. For both `MPO` and `OMPO`, we perform full-parameter finetuning for 1 epoch with learning rate $5e^{-7}$ and β tuned in the range of $\{0.1, 0.01, 0.001\}$. For both `MPO` and `OMPO`, we select the Llama-3-based model as the preference oracle[5] and set the $\log z$ are set as 0.5. Unlike the MT-bench-101 benchmark, the final state with the answer is important in this task, so we use $_{H+1}$ in the calculation of the Q function and ignore comparisons with all previous states. We use AdamW optimizer [23] and cosine learning rate schedule [22] with a warmup ratio of 0.1. The result is provided in Appx. I, showing that the proposed methods achieve performance comparable to step-DPO. Notably, `MPO` and `OMPO` do not require the ground truth label of the dataset during fine-tuning while [21] require. Additionally, `MPO` and `OMPO` only need access to an oracle Llama-3 to compare two answers whereas step-DPO [21] requires GPT-4 to locate the identify the incorrect reasoning step in an answer.

Table 3: Performance of math reasoning on MATH and GSM8K dataset across various models.

Method	GSM8K	Math
Base (Qwen2-7B-Instruct)	0.8559	0.5538
Step-DPO [21]	0.8680	**0.5836**
`MPO` (iter=1)	0.8734	0.5734
`MPO` (iter=2)	0.8734	0.5786
`OMPO` (iter=1)	0.8734	0.5734
`OMPO` (iter=2)	**0.8779**	0.5786

J Limitation and future work

This work presents a novel framework to enhance the preference alignment of large language models in multi-step settings by casting the alignment process as a two-player Markov game. We introduce novel algorithms based on natural actor-critic and optimistic online gradient descent, supported by both theoretical analysis and empirical results. However, the limitations of this work include the finite-horizon assumption in our theoretical framework, which may not fully capture real-world conversations or reasoning processes that often span with different steps instead of a fixed step H. Additionally, our practical algorithm requires querying a preference oracle, which may limit its applicability in cases where such preference oracles are unavailable or when collecting human feedback is costly. Future work should explore extending the theoretical framework to infinite-horizon settings and finding more scalable methods for gathering preference feedback.

[5] https://huggingface.co/RLHFlow/pair-preference-model-LLaMA3-8B