Your Self-Play Algorithm is Secretly an Adversarial Imitator: Understanding LLM Self-Play through the Lens of Imitation Learning

Anonymous ACL submission

Abstract

Self-play post-training methods has emerged as an effective approach for finetuning large language models and turn the weak language model into strong language model without preference data. However, the theoretical foundations for self-play finetuning remain underexplored. In this work, we tackle this by connecting self-play finetuning with adversarial imitation learning by formulating finetuning procedure as a min–max game between the model and a regularized implicit reward player parameterized by the model itself. This perspective unifies self-play imitation and general preference alignment within a common framework. Under this formulation, we present a game-theoretic analysis showing that the self-play finetuning will converge to it's equilibrium. Guided by this theoretical formulation, we propose a new self-play imitation finetuning algorithm based on the χ^2-divergence variational objective with bounded rewards and improved stability. Experiments on various of language model finetuning tasks demonstrate consistent improvements over existing self-play methods and validate our theoretical insights.

1 Introduction

Large language models (LLMs) have demonstrated remarkable success across a wide range of applications that require complex reasoning or specialized domain knowledge. A major recent advance in LLM development is post-training alignment toward more desirable behaviors (Mishra et al., 2022; Thoppilan et al., 2022; Chung et al., 2024). Modern post-training pipelines typically combine supervised finetuning (SFT) (Ouyang et al., 2022) with a variety of reinforcement learning from human feedback (RLHF) methods (Bai et al., 2022; Rafailov et al., 2023; Guo et al., 2025).

Among these methods, Many reinforcement learning (RL)-based finetuning approaches (Rafailov et al., 2023; Wu et al., 2024; Zhang et al., 2024; Calandriello et al., 2024; Zhang et al., 2025) rely on a (general) preference oracle to label samples, which encourage the model to learn from preferred responses over the undesirable ones. To reduce reliance on human preference annotations, recent methods (Chen et al., 2024; Wang et al., 2026a,b) study a *self-play* regime and treat the ground-truth responses as positive samples and the model-generated responses as negative counterparts. Despite recent empirical advancement, the theoretical understanding of the self-play regime remains limited.

This gap in theory limits principled development of self-play algorithms. For example, considering a nontrivial subset of prompts in the dataset, the ground-truth response can be closely comparable in quality, or even partially worse than, the model-generated response; in such cases, the induced preference signal becomes ambiguous or misspecified, making the implicit reward model prone to overfitting these irregular comparisons and thus failing to provide a reliable learning signal. In such a cases, existing self-play formulations lack a clear theoretical mechanism to regularize the reward model and can introduce misspecified supervision and destabilize training. This raises the following question:

How can we theoretically characterize the (implicit) reward learning in self-play finetuning?

To answer this question, we connect the self-play finetuning with the adversarial imitation learning (AIL) framework. Serving long as a principled framework for imitation and inverse reinforcement learning (Ho and Ermon, 2016; Abbeel and Ng, 2004), AIL formulates imitation from expert demonstrations as a two-player game between a policy learner and a reward learner where the reward learner aims to distinguish expert behavior from learner behavior robustly, while the policy learner seeks to match the expert distribution by maximizing the reward model. This paradigm has

been successfully applied to a variety of robotics tasks (Rafailov et al., 2021; Ablett et al., 2023). Notably, a series of work Garg et al. (2021); Al-Hafez et al. (2023); Ren et al. (2024) has considered regularizing the reward learning with improved empirical performance for classical RL tasks.

In this work, we establish a conceptual and algorithmic connection between adversarial imitation learning and self-play finetuning in large language models. We formulate this alignment process as a min–max game in which the policy player corresponds to the language model itself, while the reward player can be reparameterized using the model and its previous snapshots. Based on this formulation, our contributions are threefold:

- We establish an adversarial imitation learning–based framework for self-play imitation finetuning of large language models, which naturally generalizes to self-play methods with general preference alignment.
- We provide a rigorous game-theoretic analysis of self-play language model finetuning within the adversarial imitation learning framework. Guided by this analysis, we propose a novel self-play imitation finetuning algorithm with theoretical advantages over existing approaches.
- We empirically evaluate our method on various families of language models, demonstrating consistent performance improvements over prior methods and validating our theoretical insights especially a more robust reward learner.

2 Related Works

In this section, we discuss the related works on the imitation learning and self-play finetuning of language models.

Imitation Learning. Imitation learning is an variation of reinforcement learning where the agent aims to *imitate* expert behavior leveraging the reward-free expert demonstration. Historically, imitation learning approaches can be broadly categorized into two major classes. The first category is usually referred to as *behavioral cloning* (Florence et al., 2022; Chi et al., 2024) which directly imitate the expert demonstrations in a supervised learning manner. Several recent works have established theoretical guarantees under these settings (Foster et al., 2024; Rohatgi et al., 2025).

The second category adopts a variational formulation casting imitation learning as a min-max optimization between a reward model differentiating agent's behavior from expert demonstration and a policy optimization trying to maximize the reward. This approach are usually referred to as adversarial imitation learning (AIL). Representative methods include GAIL (Ho and Ermon, 2016), IQ-Learn (Garg et al., 2021), and LS-IQ (Al-Hafez et al., 2023). This line of work has also been supported by rigorous theoretical analyses (Liu et al., 2021; Xu et al., 2024; Li et al., 2025).

Self-play with Preference Feedback. Self-play algorithms have been extensively studied in the context of large language model alignment. Many existing approaches focus on general preference alignment, where the model is iteratively updated using samples labeled by a preference oracle, often inspired by the DPO framework (Rafailov et al., 2023). This line of work is frequently described as RL with AI feedback, in which the preference oracle is typically instantiated by an LLM. For example, CPL (Hejna et al., 2023) optimizes policies from preference data via contrastive learning; iterative-DPO (Tu et al., 2025; Wu et al., 2024) repeatedly relabels model-generated responses with a preference oracle and applies DPO-style updates; and χPO (Huang et al., 2024) introduces χ^2-based regularization to stabilize policy optimization. In addition, SPPO (Wu et al., 2024) develops a preference-based self-play framework with a squared-loss objective and formulates the interaction as a constant-sum two-player game for general preference model. Closely related works (Calandriello et al., 2024; Zhang et al., 2024, 2025) further cast preference alignment as seeking a Nash equilibrium in two-player games, yielding both improved empirical performance and stronger theoretical guarantees.

Self-Play Finetuning with SFT Datasets. In parallel to self-play with a preference oracle, another line of work studies self-play finetuning using standard SFT datasets that contain expert (ground-truth) demonstrations. For instance, SPIN (Chen et al., 2024) introduces a DPO-style self-play objective that enables the model to iteratively imitate SFT data by competing against its own past instances. While SPIN does not require an explicit preference oracle, it relies on expert prompt–response pairs from SFT as the self-play targets. We refer to these methods as *self-play imitation*, as it directly imitates SFT behavior rather than indirectly aligning to a learned or external preference oracle. We provide an illustration of this setup in Figure 1. Follow-up works in a similar setting include SPACE (Wang

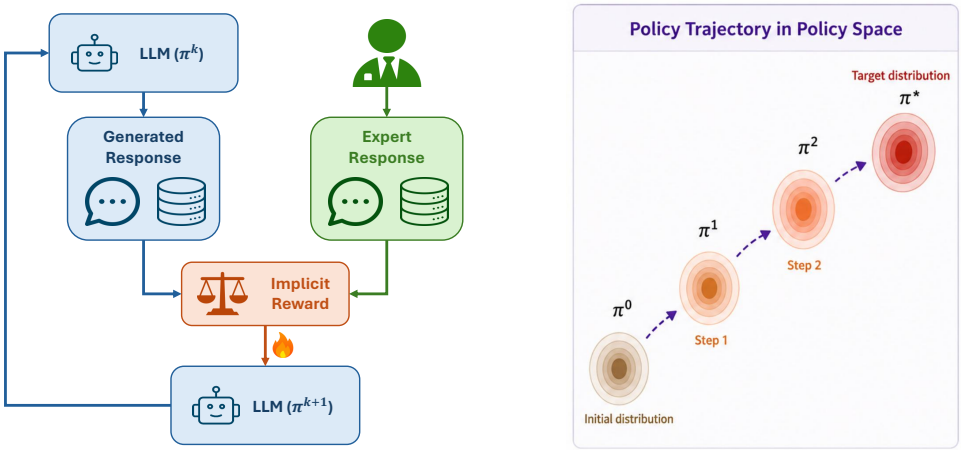

Figure 1: **Illustration of the LLM Self-Play Imitation Setup.** We illustrate the LLM self-play imitation setup studied in our paper, following prior works (Chen et al., 2024; Wang et al., 2026a,b). Given a batch of SFT data sampled from an expert policy $\pi^\star$, the language model is trained to align with $\pi^\star$ through iterative refinement. At each iteration, an implicit reward model is constructed by reparameterizing the LLM, which learns to discriminate between expert data and generated data from the previous policy π^k.

et al., 2026a) and T-SPIN (Wang et al., 2026b), which further improve upon SPIN.

Our work lies at the intersection of imitation learning and large language model self-play. We reinterpret self-play imitation finetuning methods, such as SPIN (Chen et al., 2024), through the lens of adversarial imitation learning, and provide the first unified game-theoretic analysis of this class of methods. We further show that the same perspective naturally extends to self-play algorithms for general preference, including SPPO (Wu et al., 2024) and INPO (Zhang et al., 2024). Finally, by instantiating our framework with a variational formulation based on the χ^2 divergence, in the spirit of IQ-Learn (Garg et al., 2021) and LS-IQ (Al-Hafez et al., 2023), we derive a self-play imitation finetuning algorithm that is both more stable and empirically more effective. Notably, Appendix A provides a unified and rigorous discussion of both self-play imitation and preference-based self-play finetuning under our AIL formulation.

3 Preliminaries

We formulate the language model generation process as a contextual bandit problem. For each round, the language model observes a context $x \in \mathcal{X}$ and generates a response $y \in \mathcal{Y}$. There exists a reward function $r(x, y)$ in a reward class $\mathcal{R}$ can be learned for each context and response pairs. We represent the language model as a policy $\pi(y|x)$ in a policy class Π. Since the model aims to align with a domain which an inaccessible generation policy $\pi^\star$ is preferred, the policy learning process can be formulated as an adversarial process $\max_r \min_{\pi\in\Pi} \mathbb{E}_{\pi^\star}[r] - \mathbb{E}_\pi[r]$ that jointly learn the reward and policy. This formulation is consistent with the standard adversarial imitation learning setup.

Self-play finetuning. Self-play finetuning (SPIN) (Chen et al., 2024) aims to align a language model with an SFT dataset $\mathcal{D}^\star$ generated by an expert policy $\pi^\star(y \mid x)$, which may represent human behavior. At each iteration, SPIN updates the model by maximizing the following objective:

$$\mathop{\mathbb{E}}_{\substack{(x,y)\sim\mathcal{D}^\star \\ y'\sim\pi^k(\cdot|x)}} \left[\sigma\left(\beta \log \frac{\pi(y|x)}{\pi^k(y|x)} - \beta \log \frac{\pi(y'|x)}{\pi^k(y'|x)}\right)\right],$$

where π^k denotes the model from the previous iteration, $\sigma(\cdot)$ is a monotonically non-decreasing link function. By iteratively optimizing this objective and resampling data, SPIN drives the policy π toward the expert policy $\pi^\star$.

Adversarial Imitation Learning. Instead of directly mimics the expert demonstrations using behavioral cloning, adversarial imitation learning (AIL) formulates the learning process as a two-player game, providing a variational characterization of distribution matching between the expert and behavioral policies. Formally, this involves jointly learning the reward and policy via the following optimization:

$$\max_r \min_\pi \mathop{\mathbb{E}}_{(s,a)\sim d^\star}[r(s,a)] - \mathop{\mathbb{E}}_{(s,a)\sim d^\pi}[r(s,a)] - \psi(r),$$

where $d^\pi = (1-\gamma)\sum_{t=0}^{\infty}\gamma^t \Pr_\pi(s_t = s,\, a_t = a)$ denotes the discounted occupancy measure induced by the behavioral policy π, and $d^\star$ is defined analogously for the expert policy $\pi^\star$. $\psi(r)$ is a convex regularizer associated with the choice of statistical distance between the expert and behavioral occupancy measures (Garg et al., 2021). Although we formulate adversarial imitation learning (AIL) within the Markov Decision Process (MDP) framework, in the LLM setting considered in this work the formulation naturally reduces to a contextual bandit problem, since no meaningful transition dynamics are present.

For the simplicity of the theoretical analysis, we assume that the optimal expert policy as well as the ground truth reward is realizable.

Assumption 3.1 (Realizability)**.** The ground-truth reward and optimal policy are realizable with the corresponding function classes, i.e., $r^\star \in \mathcal{R}$, $\pi^\star \in \Pi$.

4 Adversarial Imitation Learning View

4.1 A Single-Stage Formulation

In this section, we focus on formulating the self-play finetuning problem imitating an expert data distribution $\pi^\star$ from an initial policy distribution as an adversarial imitation learning process:

$$\max_r \min_\pi \mathbb{E}_x\Big[\sigma\big(\mathop{\mathbb{E}}_{y\sim\pi^\star}[r(x,y)] - \mathop{\mathbb{E}}_{y\sim\pi}[r(x,y)]\big) - \psi(r)\Big], \tag{1}$$

where σ denotes the monotonically non-decreasing link function such as $\sigma(t) = t$ or $\sigma(t) = -\log(1+\exp(-t))$, and $\psi(r)$ is a convex regularizer (Ho and Ermon, 2016). Notably, optimizing the objective in (1) is equivalent to minimizing a statistical distance between the expert and current policy distributions (Ho and Ermon, 2016; Garg et al., 2021), given a identical link function $\sigma(t)$.

In this work, we focus on the choice of the convex regularizer $\psi(r)$, corresponding to the original regularization term $\phi(r, r^{k-1})$ with the Bregman divergence component removed. Different choices of $\psi(r)$ induce different properties in the resulting self-play algorithms. We summarize the connection between the choice of the regularizer and the resulting self-play algorithm in Table 1. In particular, we study two specific regularizer choices that cast the min–max optimization in (1) as a statistical distance minimization:

Total Variation Distance. Consider a regularizer with $\psi(r) = 0$ for $\|r\|_\infty \le R_{\max}$ and $\psi(r) = \infty$ otherwise, and an identical link function with $\sigma(t) = t$. This formulation is equivalent to $\min_{\pi\in\Pi} R_{\max} D_{\mathrm{TV}}(\pi, \pi^\star)$ under a trust region constraint. Taking the closed form of the policy update, can recover the learning objective of (Linear) SPIN (Chen et al., 2024). However, original the reward reparameterization used in SPIN doesn't explicitly enforce the boundedness of the reward, thus $R_{\max}$ can be arbitrarily large.

KL Divergence. For $\psi(r) = \infty \cdot \mathbf{1}[|r|_\infty > R_{\max}]$, and a link function with $\sigma(t) = -\log(1+\exp(-t))$. This formulation can be seen as minimizing the KL divergence, which recovers SPIN (Chen et al., 2024) under logistic link function. We prove it under a multi-iteration two-stage surrogate, as used in SPIN, in Appendix I. Notably, $R_{\max}$ here is unbounded and can be arbitrary large.

Pearson χ^2 Divergence. Consider a regularizer with $\psi(r) = c\mathbb{E}_{\pi^\star}[(r(x,y))^2]$ and a link function $\sigma(t) = t$. This formulation is equivalent to $\min_{\pi\in\Pi} D_{\chi^2}(\pi\|\pi^\star)$ under a trust region constraint. This regularizer can further be generalized to $\psi(r) = c\alpha \cdot \mathbb{E}_{\pi^\star}[(r(x,y))^2] + c(1-\alpha)\cdot \mathbb{E}_\pi[(r(x,y))^2]$, which can characterize the data mixing strategy during self-play and equivalent to $\min_{\pi\in\Pi} D_{\chi^2}(\pi^\star\|\alpha\pi^\star + (1-\alpha)\pi)$ (Al-Hafez et al., 2023) under trust region constraints on policy and reward. Since SPIN with both identical and sigmoid link function cannot guarantee a bounded reward, the magnitude of the reward from SPIN may be arbitrarily large in their cases. In contrast, we can show that by applying the equally sampled mixed χ^2 regularization, i.e., $\psi(r) = (1/2)c\cdot\mathbb{E}_{\pi^\star}[(r(x,y))^2] + (1/2)c\cdot\mathbb{E}_\pi[(r(x,y))^2]$, can lead to a bounded divergence and a bounded reward, which leads to theoretical benefits in the following analysis in the two-stage optimization alternative. Notably, Al-Hafez et al. (2023) has derived a similar result within MDP setting with occupancy distribution matching. We'll adapt their result to contextual bandit setting in our case. We now introduce the following proposition showing the boundedness of the reward when using the mixed χ^2 convex regularizer:

Proposition 4.1 (Contextual bandit version of Proposition A.2 and A.3, Al-Hafez et al. 2023)**.** *The mixed Pearson χ^2 divergence between $\pi^\star$ and the mixture distribution $\frac{\pi+\pi^\star}{2}$ induced by the convex regularizer $\psi(r) = \frac{c}{2}\cdot\big(\mathbb{E}_{\pi^\star}[(r(x,y))^2] + \mathbb{E}_\pi[(r(x,y))^2]\big)$ is bounded by:*

$$0 \le 2D_{\chi^2}\left(\pi^\star \| \tfrac{\pi+\pi^\star}{2}\right) \le \tfrac{1}{c}.$$

Imitation Objective	Algorithm	$\sigma(t)$	$\psi(r)$	Distance
Query Response Pairs	SPIN (Chen et al., 2024)	$-\log(1+e^{-t})$	$\psi(r) = \infty \cdot \mathbf{1}[\lvert r\rvert_\infty > R_{\max}]$	D_{KL}
	(Linear) SPIN	t	$\psi(r) = \infty \cdot \mathbf{1}[\lvert r\rvert_\infty > R_{\max}]$	D_{TV}
	SPIF (Ours)	t	$\psi(r) = \mathbb{E}_{\text{mix}}[r^2]$	$D_{\chi^2,\text{mix}}$
Preference Oracle	Iter-DPO (Tu et al., 2025)	$-\log(1+e^{-t})$	$\psi(r) = \infty \cdot \mathbf{1}[\lvert r\rvert_\infty > R_{\max}]$	D_{KL}
	SPPO (Wu et al., 2024)	t	$\psi(r) = \mathbb{E}_{\text{mix}}[r^2]$	$D_{\chi^2,\text{mix}}$
	INPO (Zhang et al., 2025)	t	$\psi(r) = \mathbb{E}_{\text{mix}}[r^2]$	$D_{\chi^2,\text{mix}}$
Expert Trajectories (MDPs)	GAIL (Ho and Ermon, 2016)	t	$\psi(r) = \mathbb{E}[r - \log(2-e^{-r})]$	D_{JS}
	IQ-Learn (Garg et al., 2021)	t	$\psi(r) = \mathbb{E}[r^2]$	D_{χ^2}
	LS-IQ (Al-Hafez et al., 2023)	t	$\psi(r) = \mathbb{E}_{\text{mix}}[r^2]$	$D_{\chi^2,\text{mix}}$

Table 1: **Overview of the Algorithms.** This table summarizes AIL and LLM self-play algorithms under our unified framework, with different learning setting, imitation objective, choice of link function and convex regularizer, and the resulting statistical distance being minimized for each algorithm. $\mathbb{E}_{\text{mix}}$ denotes the mixed regularizer $\psi(r) = \frac{c}{2}\mathbb{E}_{\pi^\star}[r(x,y)^2] + \frac{c}{2}\mathbb{E}_{\pi}[r(x,y)^2]$, $D_{\chi^2,\text{mix}}$ denotes the mixed χ^2 divergence, defined between expert and model data for non-preference-based methods, and between positive and negative samples for preference-based methods. *(Linear) SPIN* is refer to as a variant of SPIN (Chen et al., 2024) using identical link function $\sigma(t) = t$. INPO is an KL-constrained optimization w.r.t π^{ref}, so it has an additional regularizer compared to other methods.

Furthermore, the optimal reward for solving the variational form of this Pearson χ^2 divergence:

$$
\begin{aligned}
&2D_{\chi^2}\left(\pi^\star \| \tfrac{\pi+\pi^\star}{2}\right) \\
&= \sup_r \mathbb{E}_{\pi^\star}[r(x,y)] - \mathbb{E}_{\pi}[r(x,y)] \\
&\quad - \frac{c}{2}\left(\mathbb{E}_{\pi^\star}[(r(x,y))^2] + \mathbb{E}_{\pi}[(r(x,y))^2]\right)
\end{aligned}
$$

is bounded within the interval $[-\frac{1}{c}, \frac{1}{c}]$.

4.2 A Two-Stage Iterative Alternative

Directly solving the min-max optimization problem in (1) may be hard in practice, prior work usually decompose it into a two-stage iterative optimization process (Ho and Ermon, 2016; Garg et al., 2021). This iterative process can be decomposed into a two-stage algorithm iteratively optimizing the reward r and policy π:

$$
\begin{aligned}
r^k &= \operatorname*{argmax}_r \mathbb{E}_\rho\big[\sigma(\mathbb{E}_{\pi^\star}[r(x,y)] - \mathbb{E}_{\pi^k}[r(x,y)]) \\
&\qquad - \phi(r, r^{k-1})\big], \\
\pi^{k+1} &= \operatorname*{argmax}_\pi \mathbb{E}_\rho\left[\mathbb{E}_\pi r^k(x,y) - \beta D_{\text{KL}}(\pi\|\pi^k)\right].
\end{aligned} \tag{2}
$$

To attain a more stable optimization process, we seek to constrain the optimization in (2) by regulating the optimization process with an additional KL constraint $D_{\text{KL}}(\pi\|\pi^k)$ on the policy for mirror descent and a one-step Bregman constraint for r, which leads to a new convex regularizer $\phi(r, r^{k-1}) = \psi(r) + \zeta D_f(r, r^{k-1})$. $D_f(r, r^{k-1})$ is a Bregman divergence defined by a convex function f. One thing worth mentioning is that the policy optimization objective has a closed-form solution $\pi^{k+1} \propto \pi^k \exp(\beta^{-1} r^k)$. SPIN (Chen et al., 2024) leveraged this closed-form to reparameterize the reward function, converting the two-stage algorithm into a single-stage iterative policy optimization.

Algorithm 1 Self-Play Imitation Finetuning (General)

1: **Input:** Number of iterations K, expert policy $\pi^\star$, reference policy π^{ref}.
2: Initialize $\pi^1 = \pi^{\text{ref}}$.
3: **for** $k = 1, 2, \ldots, K$ **do**
4: Obtain r^k by solving $\operatorname{argmax}_r \mathbb{E}_{\pi^\star}[r(x,y)] - \mathbb{E}_{\pi^k}[r(x,y')] - \phi(r, r^{k-1})$
5: Update policy π^{k+1} by solving $\operatorname{argmax}_\pi \mathbb{E}_\pi[r^k(x,y)] - \beta D_{\text{KL}}(\pi\|\pi^k)$
6: **end for**
7: **return** π^{K+1}

4.3 A Game-Theoretic Analysis

We first formulate (1) as a two-player game between policy π and reward r. Thus the following weak duality holds:

Proposition 4.2. *The problem defined in* (1) *has the following weak duality with a identical link function* $\sigma(\cdot)$*:*

$$
\min_\pi J(\pi, r^\star) \le J(\pi^\star, r^\star) \le \max_r J(\pi^\star, r),
$$

where $J(\pi, r) = \mathop{\mathbb{E}}_{x\sim\rho(x)}\Big[\mathop{\mathbb{E}}_{y\sim\pi^\star} r(x,y) - \mathop{\mathbb{E}}_{y\sim\pi} r(x,y)\Big]$.

Following Proposition 4.2, we further characterize the duality gap of the alternative two-stage adversarial algorithm:

Definition 4.3. For Algorithm 1, define the duality gap as

$$\text{DualGap} = \max_{r\in\mathcal{R}} J(\bar{\pi}, r) - \min_{\pi\in\Pi} J(\pi, \bar{r}),$$

where $\bar{r} = \frac{1}{K}\sum_{k=1}^{K} r^k$ and $\bar{\pi} = \frac{1}{K}\sum_{k=1}^{K} \pi^k$.

By Definition 4.3 and Proposition 4.2, we are ready to provide an upper bound for the duality gap of the self-play imitation algorithm:

Theorem 4.4. *Let $r \in \{\mathcal{X}\times\mathcal{Y}\to[\text{-}R_{\max}, R_{\max}]\}\cap\mathcal{R}$, and set constants $D = \max_{\pi\in\Pi} D_{KL}(\pi^\star\|\pi)$, $B = \max_{r\in\mathcal{R}} D_f(r^\star, r)/R_{\max}^2$, and parameter $\zeta = \frac{\sqrt{K}}{BR_{\max}^2}$, $\beta = \frac{\sqrt{K}}{D}$ in Algorithm 1. When an identical link function $\sigma(t)=t$ is applied, the duality gap is upper bounded by:*

$$\text{DualGap} \leq \mathcal{O}\left(\frac{(D+B)R_{\max}^2}{\sqrt{K}}\right).$$

Remark 4.5. Theorem 4.4 suggests the upper bounds for the iteration steps $K \leq \mathcal{O}(R_{\max}^4(D+B)^2\epsilon^{-2})$. Similar setting (adversarial imitation formulation with KL-constrained policy update) have been studied in OGAP (Liu et al., 2021) but with linear MDP and without estimation error. They achieve $\tilde{\mathcal{O}}(1/\sqrt{K})$ suboptimality gap upper bound, which matches our results although with different setting. Prior work considering the setting of Nash policy optimization with general preference also achieves the upper bound for K with similar order (Zhang et al., 2024).

Remark 4.6. From Proposition 4.1, we have that using the mixed Pearson χ^2 divergence as the choice for the convex regularizer leads to a bounded reward, where $R_{\max} = 1/c$. This suggests the theoretical benefit of leveraging χ^2 divergence as the regularization compared to using KL divergence or TV distance as in SPIN (Chen et al., 2024), since a bounded reward may result in small $R_{\max}$, and a tighter upper bound for the duality gap according to Theorem 4.4.

Remark 4.7. The reward space $\mathcal{R}$ in Theorem 4.4 is defined by strictly enforcing the regularizer $\psi(r)$, which can be seen as turning the Lagrangian dual form in (1) into a constrained optimization by turning $\psi(r)$ from to regularizer to a hard constraint on the reward space. By employing Assumption 3.1, we assume that the optimal reward is still in the constrained reward space.

Following Theorem 4.4, leveraging a mixed χ^2 regularizer $\psi(r) = (1/2)c\cdot\mathbb{E}_{\pi^\star}[(r(x,y))^2] + (1/2)c\cdot\mathbb{E}_{\pi}[(r(x,y))^2]$, by Proposition 4.1 we can have the bounded reward property, i.e., the reward that solves the variational form of Pearson χ^2 divergence is bounded by $[-1/c, 1/c]$. In this case, we can have a bounded $R_{\max}$, which leads to a tighter upper bound depicted in Theorem 4.4, compared to unbounded formulation in SPIN (Chen et al., 2024).

5 χ^2 Self-Play Imitation Finetuning

We consider a setting that given a finetuning dataset $\mathcal{D}^\star$ containing query-response pairs sampled from an oracle $\pi^\star$ similar with SPIN (Chen et al., 2024). Since Proposition 4.1 and Theorem 4.4 has shown theoretical benefit of leveraging χ^2 divergence as the convex regularizer in the self-play imitation setting. We aim to derive a practical algorithm under the scope of using χ^2 divergence by choosing a identical link function $\sigma(t) = t$ and a convex regularizer $\psi(r) = \alpha\cdot\mathbb{E}_{\pi^\star}[(r(x,y))^2] + (1-\alpha)\cdot\mathbb{E}_{\pi}[(r(x,y))^2]$ for the formulation in (2). The KL-regularized policy optimization (2) yields the following closed-form solution:

$$r(x,y) = \beta\log\left(\frac{\pi(y|x)}{\pi^k(y|x)}\right) + \beta\log Z(x).$$

To avoid estimating the partition function $Z(x)$, we establish a reparameterization of the reward by subtracting the partition function:

$$r(x,y) := \beta\log\left(\frac{\pi(y|x)}{\pi^k(y|x)}\right). \tag{3}$$

Using this reward reparameterization, we can turn the χ^2 regularized two-stage formulation in (2) into a singe stage least square optimization problem in the following proposition:

Proposition 5.1. *Updating via (2) under the mapped reward defined in (3) and the regularizer in the form of $\psi(r) = c\alpha\cdot\mathbb{E}_{\pi^\star}[(r(x,y))^2] + c(1-\alpha)\cdot\mathbb{E}_{\pi}[(r(x,y))^2]$ is equivalent to minimizing the following regularized objective:*

$$\pi^{k+1} = \text{argmin}_\pi\ \mathcal{L}(\pi) + \mathbb{E}_{\rho,\pi^\star,\pi^k}\ \zeta D_f(\pi,\pi^k),$$

where $\mathcal{L}(\pi)$ is the least square objective:

$$\mathcal{L}(\pi) := \alpha\,\mathbb{E}_{\rho,\pi^\star(x)}\left[\beta\log\frac{\pi(y|x)}{\pi^k(y|x)} - r_{\max}\right]^2 + (1-\alpha)\,\mathbb{E}_{\rho,\pi^k(x)}\left[\beta\log\frac{\pi(y|x)}{\pi^k(y|x)} - r_{\min}\right]^2,$$

in which $r_{\max} = 1/(2c\alpha)$, $r_{\min} = -1/[2c(1-\alpha)]$ and $D_f(\pi,\pi^k) = (1/2)\mathbb{E}_{\pi^\star(x)}[\beta\log(\pi(y|x)/\pi^k(y|x))]^2$.

Methods	Qwen3-4B				Mistral-7B				Qwen3-14B			
	Arc	MMLU	HS	WG	Arc	MMLU	HS	WG	Arc	MMLU	HS	WG
Base	51.62	68.33	67.57	65.43	53.33	53.18	74.47	71.67	58.70	77.97	81.34	73.64
SFT	54.46	68.58	69.68	66.80	54.24	54.08	76.11	72.77	59.02	77.76	82.54	73.82
SPIN Iter-1	53.84	68.10	67.98	66.06	**54.21**	54.11	75.52	72.93	59.24	77.36	82.11	73.18
SPACE Iter-1	54.62	68.45	68.36	66.37	53.86	54.43	**76.29**	72.94	59.43	77.82	81.93	**73.65**
SPIF Iter-1 (Ours)	**55.12**	**68.66**	**70.48**	**68.11**	54.05	**54.60**	75.92	**73.09**	**59.71**	**78.05**	**82.45**	73.52
SPIN Iter-2	54.58	67.90	68.10	67.13	**54.52**	54.40	75.39	73.14	59.43	77.84	82.70	73.78
SPACE Iter-2	55.37	68.24	69.86	67.05	54.18	54.81	76.47	73.21	59.74	77.65	82.93	73.80
SPIF Iter-2 (Ours)	**56.66**	**68.75**	**71.43**	**68.34**	54.41	**55.02**	**76.63**	**74.33**	**60.02**	**78.12**	**83.04**	**74.05**
SPIN Iter-3	55.12	68.06	68.32	67.61	54.60	54.38	75.47	73.42	59.80	77.96	82.62	73.02
SPACE Iter-3	56.84	68.16	70.34	67.45	54.16	55.02	76.60	73.45	59.93	77.81	83.08	73.64
SPIF Iter-3 (Ours)	**57.11**	**68.83**	**71.92**	**68.82**	**54.70**	**55.24**	**77.14**	**75.02**	**60.23**	**78.08**	**83.33**	**74.43**

Table 2: **Main Results.** We report results over three iterations and five random seeds on three language models, comparing our method against the supervised finetuning (SFT) baseline, SPIN (Chen et al., 2024), and SPACE (Wang et al., 2026a). Our approach consistently outperforms the existing baselines across most settings.

Remark 5.2. Similar formulations as in Proposition 5.1 has been previously explored in some literatures proposing stable versions of GANs (Mao et al., 2017) and adversarial imitation learning (Al-Hafez et al., 2023). The key difference of the formulation proposed in Proposition 5.1 compared to prior work is that we plugged the closed form solution given by the KL regularized policy optimization objective into the original least square reward learning objective using the reward mapping defined in (3).

Remark 5.3. As discussed in Sec. 4.1, leveraging the convex regularizer $\psi(r) = c\alpha \cdot \mathbb{E}_{\pi^\star}[(r(x,y))^2] + c(1-\alpha)\cdot \mathbb{E}_{\pi}[(r(x,y))^2]$ with coefficient α for reward learning objective is equivalent to measuring the mixed χ^2 divergence $D_{\chi^2}(\pi^\star \| \alpha\pi^\star + (1-\alpha)\pi^k)$. Therefore, α can be seen as a coefficient for mix-up ratio between oracle data generated from $\pi^\star$ and the data from the previous policy π^k. The data mix-up strategy has been applied in the practical implementation of SPIN (Chen et al., 2024). Usually, we set $\alpha = 0.5$ to consider the balanced sampling scenario, i.e., drawing the same amount of data from the oracle policy $\pi^\star$ and the previous step policy π^k.

By Proposition 5.1, we can retrieve the learning objective of our proposed χ^2 self-play imitation finetuning algorithm using finite dataset approximation and balanced sampling ($\alpha = 0.5$) for one iteration:

$$\pi^{k+1} = \underset{\pi}{\operatorname{argmin}}\ \widehat{\mathcal{L}}(\pi) + \frac{\zeta}{2} \underset{(x,y)\sim \mathcal{D}^\star \cup \mathcal{D}^k}{\mathbb{E}} \left[\log \frac{\pi(y|x)}{\pi^k(y|x)} \right]^2, \tag{4}$$

Algorithm 2 Self-Play Imitation Finetuning (Practical)

Require: Number of self-play iterations K, expert dataset $\mathcal{D}^\star$, reference policy π^{ref}, sample size M.

1: Initialize $\pi^1 = \pi^{\text{ref}}$.
2: **for** $k = 1, 2, \ldots, K$ **do**
3: Sample a dataset $\mathcal{D}^k$ using current policy π^k.
4: Update policy via (4).
5: **end for**
6: **return** π^{K+1}

where $\widehat{\mathcal{L}}(\pi)$ is the empirical loss with dataset $\mathcal{D}^\star$ and $\mathcal{D}^k$:

$$\widehat{\mathcal{L}}(\pi) := \frac{1}{2}\, \mathbb{E}_{(x,y)\sim\mathcal{D}^\star} \left[\beta \log \frac{\pi(y|x)}{\pi^k(y|x)} - r_{\max} \right]^2 + \frac{1}{2}\, \mathbb{E}_{(x,y)\sim\mathcal{D}^k} \left[\beta \log \frac{\pi(y|x)}{\pi^k(y|x)} - r_{\min} \right]^2,$$

with $r_{\max} = 1/c$ and $r_{\min} = -1/c$. Intuitively, $\widehat{\mathcal{L}}(\pi)$ corresponds to a least squares objective that pushes the rewards of expert samples in $\mathcal{D}^\star$ toward $r_{\max}$ while pulling the rewards of samples from the previous iteration dataset $\mathcal{D}^k$ toward $r_{\min}$, thereby creating a clear margin between the two classes of data for discrimination. The second term in (4) serves as a regularizer that mitigates over optimization and encourages the updated reward to remain close to the reward from the previous iteration, which aligns with the mirror descent update applied to the reward player in Algorithm 1.

6 Empirical Results

Here we present a detailed empirical analysis of our self-play imitation finetuning (SPIF) under the χ^2 divergence, along with a comparative evaluation against SPIN (Chen et al., 2024), SPACE (Wang et al., 2026a), and a standard supervised finetuning (SFT) baseline. We conduct experiments using Qwen3-4B and Qwen3-14B (Yang et al., 2025) and a instruction-following Mistral-7B model (Jiang et al., 2023) on 50k samples subsampled from the UltraChat SFT dataset (Ding et al., 2023).

For self-play–based methods (ours, non-linear version of SPIN (Chen et al., 2024) and SPACE (Wang et al., 2026a)), at each iteration k we first generate synthetic responses by sampling $y^k \sim \pi^k(\cdot \mid x)$ for each prompt x in the SFT dataset. The model is then trained following Algorithm 2 on data triples $(x, y^\star, y^k)$.

We evaluate the resulting models on a diverse suite of benchmarks, including Arc Challenge (Clark et al., 2018), MMLU (Hendrycks et al., 2020), HellaSwag (Zellers et al., 2019), and WinoGrande (Sakaguchi et al., 2019), to assess instruction-following capabilities. Performance results over three self-play iterations are reported in Table 2. Our method consistently outperforms the supervised finetuning (SFT) baseline, SPIN (Chen et al., 2024) and SPACE (Wang et al., 2026a) across most evaluation settings, demonstrating the effectiveness of the proposed algorithm. We provide detailed implementation descriptions and experimental settings in Appendix K, ablation studies in Appendix L, additional results on the mathematical capabilities in Appendix M and additional results with six self-play iterations in Appendix N.

Reward Dynamics Analysis. We analyze the reward dynamics during training for our SPIF approach with χ^2 divergence, and compare it against the reward behavior of SPIN (Chen et al., 2024). We observe that although our reward exhibits a significantly smaller magnitude, it still effectively discriminates between data sampled from the expert policy $\pi^\star$ and data generated by the previous-iteration model π^k. Figure 2 illustrates the reward distribution at the first self-play iteration for the Qwen-3-4B model. This empirical observation supports the theoretical analysis in Sec 4.3, which shows that SPIF with χ^2 regularization admits a bounded reward magnitude and consequently enjoys a tighter upper bound on the duality gap, as stated in Theorem 4.4, with a smaller $R_{\max}$.

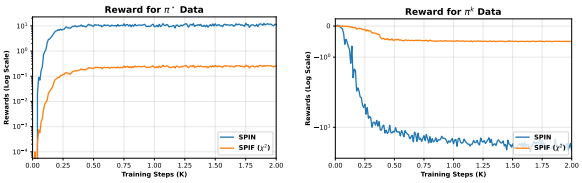

Figure 2: **Reward Dynamics Analysis.** We plot the reward curves (log-scaled) during training for our approach, SPIF with χ^2 regularization, and for SPIN (Chen et al., 2024). The results show that our method produces rewards with substantially smaller magnitude, which leads to more stable learning dynamics and is consistent with our theoretical analysis predicting a tighter duality gap.

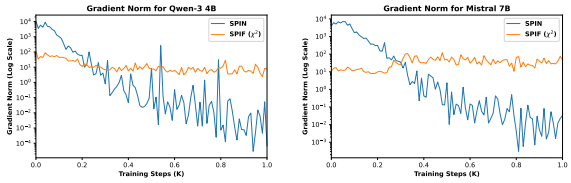

Figure 3: **Gradient Norm Analysis.** We plot the gradient norms (log-scaled) during training for our approach, SPIF with χ^2 regularization, and for SPIN (Chen et al., 2024). The results show that our method maintains significantly more stable gradient norms, indicating improved training stability compared to SPIN.

Gradient Norm Analysis. We report the gradient norm dynamics during training for our SPIF approach with χ^2 regularization and compare them against those of the original SPIN objective (Chen et al., 2024). Under the SPIN objective, the gradient norm is initially very large (near the order of 10^4) and then rapidly collapses to near zero (on the order of 10^{-4}), which can lead to unstable optimization behavior. In contrast, our χ^2-regularized approach maintains a relatively small and stable gradient norm, typically in the range of 10^1 to 10^2, resulting in more stable training dynamics, as show in in Figure 3.

7 Conclusion

We presented a unified theoretical view of self-play post-training for language model alignment by formulating it as adversarial imitation learning. With a game-theoretic analysis based on this perspective and clarifies the relationship underlying existing methods, we propose a self-play imitation finetuning algorithm based on a χ^2-divergence variational objective, which yields bounded rewards and improved training stability. Experiments on various of models demonstrate consistent improvements over prior self-play methods, validating both the theoretical insights and the practical effectiveness of the proposed approach.

Limitations

Our theoretical analysis relies on realizability assumptions. While such assumptions are standard in theoretical analyses, they may not always hold in practical LLM finetuning scenarios, where the reward and policy classes can be misspecified. In addition, our convergence result establishes an average-iterate guarantee, whereas practical self-play finetuning evaluates the last-iterate model produced by training. Establishing last-iterate convergence would require stronger assumptions on the game structure or optimization dynamics, which we leave for future work.

References

Pieter Abbeel and Andrew Y Ng. 2004. Apprenticeship learning via inverse reinforcement learning. In *Proceedings of the twenty-first international conference on Machine learning*, page 1.

Trevor Ablett, Bryan Chan, and Jonathan Kelly. 2023. Learning from guided play: Improving exploration for adversarial imitation learning with simple auxiliary tasks. *IEEE Robotics and Automation Letters*, 8(3):1263–1270.

Firas Al-Hafez, Davide Tateo, Oleg Arenz, Guoping Zhao, and Jan Peters. 2023. Ls-iq: Implicit reward regularization for inverse reinforcement learning. *arXiv preprint arXiv:2303.00599*.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, and 1 others. 2022. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*.

Qi Cai, Zhuoran Yang, Chi Jin, and Zhaoran Wang. 2020. Provably efficient exploration in policy optimization. In *International Conference on Machine Learning*, pages 1283–1294. PMLR.

Daniele Calandriello, Daniel Guo, Remi Munos, Mark Rowland, Yunhao Tang, Bernardo Avila Pires, Pierre Harvey Richemond, Charline Le Lan, Michal Valko, Tianqi Liu, and 1 others. 2024. Human alignment of large language models through online preference optimisation. *arXiv preprint arXiv:2403.08635*.

Zixiang Chen, Yihe Deng, Huizhuo Yuan, Kaixuan Ji, and Quanquan Gu. 2024. Self-play fine-tuning converts weak language models to strong language models. *arXiv preprint arXiv:2401.01335*.

Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. 2024. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, and 1 others. 2024. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv:1803.05457v1*.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Ning Ding, Yulin Chen, Bokai Xu, Yujia Qin, Zhi Zheng, Shengding Hu, Zhiyuan Liu, Maosong Sun, and Bowen Zhou. 2023. Enhancing chat language models by scaling high-quality instructional conversations. *ArXiv*, abs/2305.14233.

Pete Florence, Corey Lynch, Andy Zeng, Oscar A Ramirez, Ayzaan Wahid, Laura Downs, Adrian Wong, Johnny Lee, Igor Mordatch, and Jonathan Tompson. 2022. Implicit behavioral cloning. In *Conference on robot learning*, pages 158–168. PMLR.

Dylan J Foster, Adam Block, and Dipendra Misra. 2024. Is behavior cloning all you need? understanding horizon in imitation learning. *Advances in Neural Information Processing Systems*, 37:120602–120666.

Divyansh Garg, Shuvam Chakraborty, Chris Cundy, Jiaming Song, and Stefano Ermon. 2021. Iq-learn: Inverse soft-q learning for imitation. *Advances in Neural Information Processing Systems*, 34:4028–4039.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, and 1 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Joey Hejna, Rafael Rafailov, Harshit Sikchi, Chelsea Finn, Scott Niekum, W Bradley Knox, and Dorsa Sadigh. 2023. Contrastive preference learning: learning from human feedback without rl. *arXiv preprint arXiv:2310.13639*.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Xiaodong Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *ArXiv*, abs/2009.03300.

Jonathan Ho and Stefano Ermon. 2016. Generative adversarial imitation learning. *Advances in neural information processing systems*, 29.

Audrey Huang, Wenhao Zhan, Tengyang Xie, Jason D Lee, Wen Sun, Akshay Krishnamurthy, and Dylan J Foster. 2024. Correcting the mythos of kl-regularization: Direct alignment without overoptimization via chi-squared preference optimization. *arXiv preprint arXiv:2407.13399*.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *Preprint*, arXiv:2310.06825.

Shangzhe Li, Dongruo Zhou, and Weitong Zhang. 2025. Near-optimal second-order guarantees for model-based adversarial imitation learning. *arXiv preprint arXiv:2510.09487*.

Zhihan Liu, Yufeng Zhang, Zuyue Fu, Zhuoran Yang, and Zhaoran Wang. 2021. Provably efficient generative adversarial imitation learning for online and offline setting with linear function approximation. *arXiv preprint arXiv:2108.08765*.

Xudong Mao, Qing Li, Haoran Xie, Raymond YK Lau, Zhen Wang, and Stephen Paul Smolley. 2017. Least squares generative adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2794–2802.

Swaroop Mishra, Daniel Khashabi, Chitta Baral, and Hannaneh Hajishirzi. 2022. Cross-task generalization via natural language crowdsourcing instructions. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3470–3487.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, and 1 others. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741.

Rafael Rafailov, Tianhe Yu, Aravind Rajeswaran, and Chelsea Finn. 2021. Visual adversarial imitation learning using variational models. *Advances in Neural Information Processing Systems*, 34:3016–3028.

Juntao Ren, Gokul Swamy, Zhiwei Steven Wu, J Andrew Bagnell, and Sanjiban Choudhury. 2024. Hybrid inverse reinforcement learning. *arXiv preprint arXiv:2402.08848*.

Dhruv Rohatgi, Adam Block, Audrey Huang, Akshay Krishnamurthy, and Dylan J Foster. 2025. Computational-statistical tradeoffs at the next-token prediction barrier: Autoregressive and imitation learning under misspecification. *arXiv preprint arXiv:2502.12465*.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2019. Winogrande. *Communications of the ACM*, 64:99 – 106.

Romal Thoppilan, Daniel De Freitas, Jamie Hall, Noam Shazeer, Apoorv Kulshreshtha, Heng-Tze Cheng, Alicia Jin, Taylor Bos, Leslie Baker, Yu Du, and 1 others. 2022. Lamda: Language models for dialog applications. *arXiv preprint arXiv:2201.08239*.

Songjun Tu, Jiahao Lin, Xiangyu Tian, Qichao Zhang, Linjing Li, Yuqian Fu, Nan Xu, Wei He, Xiangyuan Lan, Dongmei Jiang, and 1 others. 2025. Enhancing llm reasoning with iterative dpo: A comprehensive empirical investigation. *arXiv preprint arXiv:2503.12854*.

Yibo Wang, Guangda Huzhang, Qingguo Chen, Zhao Xu, Weihua Luo, Kaifu Zhang, and Lijun Zhang. 2026a. Space: Noise contrastive estimation stabilizes self-play fine-tuning for large language models. *Advances in Neural Information Processing Systems*, 38:60042–60072.

Yibo Wang, Hai-Long Sun, Guangda Huzhang, Qingguo Chen, Zhao Xu, Weihua Luo, Kaifu Zhang, and Lijun Zhang. 2026b. Triplets better than pairs: Towards stable and effective self-play fine-tuning for llms. *Advances in Neural Information Processing Systems*, 38:40980–41009.

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

Tian Xu, Zhilong Zhang, Ruishuo Chen, Yihao Sun, and Yang Yu. 2024. Provably and practically efficient adversarial imitation learning with general function approximation. *Advances in Neural Information Processing Systems*, 37:66108–66146.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025. Qwen3 technical report. *ArXiv*, abs/2505.09388.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. Hellaswag: Can a machine really finish your sentence? In *Annual Meeting of the Association for Computational Linguistics*.

Yuheng Zhang, Dian Yu, Tao Ge, Linfeng Song, Zhichen Zeng, Haitao Mi, Nan Jiang, and Dong Yu. 2025. Improving llm general preference alignment via optimistic online mirror descent. *arXiv preprint arXiv:2502.16852*.

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

A Discussion

A.1 Adversarial Imitation Learning.

Existing adversarial imitation learning methods typically optimize a single-stage min–max objective as in (1), but are formulated under the full MDP setting rather than the contextual bandit setting considered in this paper. In particular, GAIL (Ho and Ermon, 2016) employs a regularization that makes the resulting objective equivalent to minimizing the Jensen–Shannon divergence between expert and behavioral distributions. IQ-Learn (Garg et al., 2021) uses the regularizer $\psi(r) = \mathbb{E}_{\rho^\star}[r^2]$, which is equivalent to minimizing the χ^2 divergence between the expert occupancy measure $\rho^\star$ and the learner occupancy measure ρ^π, where ρ denotes the policy occupancy measure. LS-IQ (Al-Hafez et al., 2023) further considers the regularizer $\psi(r) = \alpha\, \mathbb{E}_{\rho^\star}[r^2] + (1-\alpha)\, \mathbb{E}_{\rho^\pi}[r^2]$, which corresponds to minimizing the χ^2 divergence between $\rho^\star$ and a mixture of $\rho^\star$ and ρ^π with mixing coefficient α. Notably, the regularization used in LS-IQ is closely related to the formulation adopted in this paper.

A.2 Self-Play Imitation Finetuning

Self-play imitation finetuning refers to methods that leverage an SFT dataset to perform self-play, with the goal of imitating behaviors in the SFT data rather than optimizing external signals such as preferences. Both SPIN (Chen et al., 2024) and our proposed Algorithm 2 fall into this category. We show that both methods can be formulated within the standard adversarial imitation learning framework in (1).

In particular, the linear variant of SPIN (using idential link function $\sigma(t) = t$) directly minimizes the total variation distance between the expert policy $\pi^\star$ and the learned policy π, as proved in Appendix H. The nonlinear variant of SPIN ($\sigma(t) = -\log(1+\exp(-t))$) can be viewed as minimizing a KL divergence between $\pi^\star$ and π, as shown in Appendix I.

As self-play methods can be cast as imitation learning toward a prescribed objective (Table 1), they implicitly induce a capacity ceiling determined by the underlying imitation target. Consequently, the model capability attainable through such procedures is fundamentally bounded, implying that existing LLM self-play algorithms cannot achieve infinite capability gains via iterative self-improvement alone.

A.3 Self-Play for General Preference Alignment

In this sections, we extend our discussion to the general preference alignment with self-play without the need of Bradley-Terry preference model (Wu et al., 2024; Zhang et al., 2024, 2025). Similar with our proposed χ^2 self-play imitation finetuning, these methods often rely on the squared loss in different settings. Below, we show that this line of work admits an AIL interpretation with respect to a general preference oracle and exhibit a close connection to χ^2 divergence–regularized adversarial imitation learning.

SPPO. Given an estimated probability $P(y \succ \pi^k|x)$ using the general preference model, the objective in SPPO (Wu et al., 2024) for policy π^{k+1} can be written as:

$$\operatorname*{argmin}_{\pi} \mathop{\mathbb{E}}_{\rho,\pi^k}\left[\log\frac{\pi(y|x)}{\pi^k(y|x)} - \frac{1}{\beta}\Big(P(y \succ \pi^k|x) - \tfrac{1}{2}\Big)\right]^2.$$

This update rule shares similarities with our proposed objective in (4) by optimizing the χ^2 regularized AIL objective under a trust region constraint by slightly generalizing our proposed imitation framework in Sec. 4 to preference-based policy optimization using the following proposition:

Proposition A.1. *Given a preference model that outputs $w^k(x,y) := P(y \succ \pi^k \mid x)$, let $y \sim \pi^k(\cdot|x)$ and define the weighted expectations:*

$$\mathbb{E}_{\pi_+^k}[f(x,y)] := \mathbb{E}_{y\sim\pi^k(\cdot|x)}\Big[w^k(x,y)\, f(x,y)\Big],$$
$$\mathbb{E}_{\pi_-^k}[f(x,y)] := \mathbb{E}_{y\sim\pi^k(\cdot|x)}\Big[(1-w^k(x,y))\, f(x,y)\Big].$$

With $\sigma(t) = t$, mixed χ^2 regularizer and $r(x,y)$ being the reparameterized reward function in (3), the SPPO objective is equivalent to optimizing:

$$\arg\max_r \mathbb{E}_\rho\Big[\sigma\big(\mathbb{E}_{\pi_+^k}[r(x,y)] - \mathbb{E}_{\pi_-^k}[r(x,y)]\big) - \phi(r, r^{k-1})\Big],$$

Remark A.2. Proposition A.1 provides a new derivation of SPPO algorithm, differs from the original derivation from Wu et al. (2024). This new formulation shows that instead of imitating the expert policy $\pi^\star$ in original AIL formulation shown in (2), SPPO is adversarially imitating the general preference oracle $P(y \succ \pi^k \mid x)$.

Iterative DPO. Iterative DPO serves as a baseline in SPPO (Wu et al., 2024) and has also been applied to enhance LLM reasoning (Tu et al., 2025). In practice, it replaces the SPPO loss with a DPO-style objective while keeping the remaining components unchanged. In Appendix J, we show that this procedure implicitly minimizes the KL divergence between the model policy and an expert policy $\pi^\star$ induced by the preference oracle iteratively.

INPO. INPO (Zhang et al., 2024) formulates the general preference alignment problem as solving Nash policy with online mirror descent. For each iteration, it updates through the following update rule:

$$\pi^{k+1} = \operatorname*{argmin}_{\pi} \mathop{\mathbb{E}}_{\substack{y,y'\sim\pi^k(\cdot|x)\\ x\sim\rho(x)\\ y_w,y_l\sim\lambda_p(y,y')}} \left[\log\frac{\pi(y_w|x)}{\pi(y_l|x)} - \frac{\tau}{\eta}\log\frac{\pi_{\text{ref}}(y_w|x)}{\pi_{\text{ref}}(y_l|x)} - \frac{\eta-\tau}{\eta}\log\frac{\pi^k(y_w|x)}{\pi^k(y_l|x)} - \frac{1}{2\eta}\right]^2.$$

We show that this update rule is equivalent to a iterative AIL procedure operating on the preference oracle. Compared to SPPO (Wu et al., 2024), this equivalence arises under a different construction of π_+^k and π_-^k, as well as a distinct reparameterization of r.

Proposition A.3. *Given a preference model that outputs $w^k(x,y) := P(y \succ \pi^k \mid x)$, let $(y,y') \sim \pi^k(\cdot|x) \times \pi^k(\cdot|x)$ and define the weighted pairwise expectations:*

$$\mathbb{E}_{\pi_+^k}[f(x,y,y')] := \mathbb{E}_{(y,y')\sim\pi^k\times\pi^k}\Big[w^k(x,y)\, f(x,y,y')\Big],$$
$$\mathbb{E}_{\pi_-^k}[f(x,y,y')] := \mathbb{E}_{(y,y')\sim\pi^k\times\pi^k}\Big[w^k(x,y')\, f(x,y,y')\Big].$$

Then the INPO objective is equivalent to optimizing the reparameterized r with the mixed χ^2 regularizer for:

$$\arg\max_r \ \mathbb{E}_{x\sim\rho}\Big[\sigma\big(\mathbb{E}_{\pi_+^k}[r(x,y,y')] - \mathbb{E}_{\pi_-^k}[r(x,y,y')]\big) - \phi(r, r^{k-1})\Big],$$

where $\sigma(t) = t$, and $r(x,y,y')$ is defined in:

$$r(x,y,y') := \eta\log\frac{\pi(y|x)}{\pi(y'|x)} - \tau\log\frac{\pi_{\text{ref}}(y|x)}{\pi_{\text{ref}}(y'|x)} - (\eta-\tau)\log\frac{\pi^k(y|x)}{\pi^k(y'|x)}. \tag{5}$$

Remark A.4. The reward reparameterization defined in (5) does not directly arise in closed form from a direct mirror descent optimization as in (2). Instead, it has an additional KL regularizer $D_{\text{KL}}(\pi\|\pi_{\text{ref}})$, reflecting the fact that INPO (Zhang et al., 2024) formulates a constrained optimization problem with respect to the reference policy π_{ref}. The resulting formulation involves a paired response (y,y') for canceling the partition function. Nevertheless, the overall AIL interpretation remains consistent.

ONPO. ONPO (Zhang et al., 2025) extends INPO by replacing standard online mirror descent with optimistic online mirror descent. Under the assumption that $m_k = \mathbb{E}_{y'\sim\pi_{k-1}(\cdot|x)}[P(y \succ y')]$ is known, and by introducing an additional policy player, ONPO achieves an improved duality gap upper bound of $\mathcal{O}(1/K)$, in contrast to the standard $\mathcal{O}(1/\sqrt{K})$ rate of online mirror descent. By adopting the same assumption and augmenting Algorithm 1 with an additional policy player, our framework can similarly strengthen the result in Theorem 4.4 to an $\mathcal{O}(1/K)$ rate.

B Proof of Theorem 4.4

B.1 Key Lemmas

We first introduce the following lemmas:

Lemma B.1 (One-Step Descent, Cai et al. 2020)**.** *For two policy distributions $\pi^\star$ and π, and a reward function $r : \mathcal{X} \times \mathcal{Y} \to [-R_{\max}, R_{\max}]$, it holds for $\pi'(\cdot|x) \propto \pi(\cdot|x) \cdot \exp(\eta \cdot r(x, \cdot))$ that:*

$$\langle r(x,\cdot), \pi^\star(\cdot|x) - \pi(\cdot|x)\rangle \le \frac{\eta R_{\max}^2}{2} + \eta^{-1} \cdot \big(D_{KL}(\pi^\star(\cdot|x)\|\pi(\cdot|x)) - D_{KL}(\pi^\star(\cdot|x)\|\pi'(\cdot|x))\big)$$

Proof. For any function $r : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ and distributions $\pi(\cdot|x), \pi'(\cdot|x) \in \Delta(\mathcal{Y})$ that satisfy

$$\pi'(\cdot|x) \propto \pi(\cdot|x) \cdot \exp\big(\eta \cdot r(x, \cdot)\big),$$

we have

$$\begin{aligned}
\eta \cdot \langle r, \pi^\star - \pi'\rangle &= \langle Z + \log(\pi'/\pi), \pi^\star - \pi'\rangle \\
&= \langle Z, \pi^\star - \pi'\rangle + \langle \log(\pi^\star/\pi), \pi^\star\rangle + \langle \log(\pi'/\pi^\star), \pi^\star\rangle + \langle \log(\pi'/\pi), -\pi'\rangle \\
&= D_{\mathrm{KL}}(\pi^\star \,\|\, \pi) - D_{\mathrm{KL}}(\pi^\star \,\|\, \pi') - D_{\mathrm{KL}}(\pi' \,\|\, \pi). \qquad (6)
\end{aligned}$$

Here $Z : \mathcal{X} \to \mathbb{R}$ is a constant function defined by

$$Z(x) = \log\Big(\sum_{y\in\mathcal{Y}} \pi(y|x) \cdot \exp\big(\eta \cdot r(x, y)\big)\Big),$$

which implies that $\langle Z, \pi^\star - \pi'\rangle = 0$ in (6) as $\pi'(\cdot|x), \pi^\star(\cdot|x) \in \Delta(\mathcal{Y})$. Moreover, by (6) we have

$$\begin{aligned}
\eta \cdot \langle r(x,\cdot), \pi^\star(\cdot) - \pi(\cdot|x)\rangle &= \eta \cdot \langle r(x,\cdot), \pi^\star(\cdot) - \pi'(\cdot|x)\rangle - \eta \cdot \langle r(x,\cdot), \pi(\cdot|x) - \pi'(\cdot|x)\rangle \\
&\le D_{\mathrm{KL}}\big(\pi^\star(\cdot|x) \,\big\|\, \pi(\cdot|x)\big) - D_{\mathrm{KL}}\big(\pi^\star(\cdot|x) \,\big\|\, \pi'(\cdot|x)\big) - D_{\mathrm{KL}}\big(\pi'(\cdot|x) \,\big\|\, \pi(\cdot|x)\big) \\
&\quad + \eta \cdot \|r(x,\cdot)\|_\infty \cdot \|\pi(\cdot|x) - \pi'(\cdot|x)\|_1 \qquad (7)
\end{aligned}$$

for any state $x \in \mathcal{X}$. Meanwhile, by Pinsker's inequality, it holds that

$$D_{\mathrm{KL}}(\pi' \,\|\, \pi) \ge \|\pi - \pi'\|_1^2/2. \qquad (8)$$

Combining (7), (8), and the fact that $\|r(x,\cdot)\|_\infty \le R_{\max}$, with Lemma B.3, for any state $x \in \mathcal{X}$, we obtain

$$\begin{aligned}
\eta \cdot \langle r(x,\cdot), \pi^\star(\cdot) - \pi(\cdot|x)\rangle &\le D_{\mathrm{KL}}\big(\pi^\star(\cdot|x) \,\big\|\, \pi(\cdot|x)\big) - D_{\mathrm{KL}}\big(\pi^\star(\cdot|x) \,\big\|\, \pi'(\cdot|x)\big) - \|\pi(\cdot|x) - \pi'(\cdot|x)\|_1^2/2 \\
&\quad + \eta R_{\max} \cdot \|\pi(\cdot|x) - \pi'(\cdot|x)\|_1 \\
&\le D_{\mathrm{KL}}\big(\pi^\star(\cdot|x) \,\big\|\, \pi(\cdot|x)\big) - D_{\mathrm{KL}}\big(\pi^\star(\cdot|x) \,\big\|\, \pi'(\cdot|x)\big) + R_{\max}^2\eta^2/2,
\end{aligned}$$

which concludes the proof. □

Lemma B.2. *If the r-player optimizes r^k using Online Mirror Descent (OMD) against a π-player updated via Line 5 of Algorithm 1, and letting $BR_{\max}^2 = \max_{r\in\mathcal{R}} D_f(r^\star, r)$, the regret of the r-player is bounded by:*

$$\max_{r\in\mathcal{R}} \sum_{k=1}^{K} \langle r, \pi^\star - \pi^k\rangle - \sum_{k=1}^{K} \langle r^k, \pi^\star - \pi^k\rangle \le \mathcal{O}(R_{\max}^2 B\sqrt{K})$$

Proof. The r-player runs online mirror descent to maximize the sequence of objectives $\langle r, \pi^\star - \pi^k\rangle$. The gradients are $g^k := \nabla_r \langle r, \pi^\star - \pi^k\rangle = \pi^\star - \pi^k$. For a linear objective and bounded Bregman divergence $\max_{r\in\mathcal{R}} D_f(r^\star, r) = R_{\max}^2 B$, selecting the step size $\zeta = \sqrt{K}/(BR_{\max}^2)$ yields the standard OMD regret bound for a maximizer:

$$\max_{r\in\mathcal{R}} \sum_{k=1}^{K} \langle r, \pi^\star - \pi^k\rangle - \sum_{k=1}^{K} \langle r^k, \pi^\star - \pi^k\rangle \le \mathcal{O}(R_{\max}^2 B\sqrt{K}),$$

which concludes the proof. □

B.2 Supporting Lemmas

Lemma B.3. *Let $\pi(\cdot|x) \in \Delta(\mathcal{Y})$ be a probability distribution over a discrete action set $\mathcal{Y}$ conditioned on state x, and let $r : \mathcal{X} \times \mathcal{Y} \to [-R_{\max}, R_{\max}]$. Define the updated policy*

$$\pi'(y|x) \propto \pi(y|x) \exp(\eta r(x, y)), \quad \forall y \in \mathcal{Y}.$$

Then the KL divergence between π' and π satisfies

$$D_{KL}(\pi'(\cdot|x) \| \pi(\cdot|x)) \leq \frac{1}{2}\eta^2 R_{\max}^2.$$

Proof. Fix x and omit its notation for simplicity. Let

$$Z = \mathbb{E}_\pi[e^{\eta r}] = \sum_{y \in \mathcal{Y}} \pi(y|x) e^{\eta r(x,y)}, \quad f(\eta) = \log Z.$$

The updated distribution can be written as

$$\pi'(y|x) = \frac{\pi(y|x) e^{\eta r(x,y)}}{Z}.$$

The KL divergence can be expressed as

$$D_{\mathrm{KL}}(\pi' \| \pi) = \sum_{y \in \mathcal{Y}} \pi'(y|x) \log \frac{\pi'(y|x)}{\pi(y|x)} = \eta\, \mathbb{E}_{\pi'}[r] - \log Z.$$

Let $\mu = \mathbb{E}_\pi[r]$, and define the centered cumulant generating function

$$\psi(\eta) = f(\eta) - \eta\mu = \log \mathbb{E}_\pi\left[e^{\eta(r-\mu)}\right].$$

It follows that

$$\psi(0) = 0, \quad \psi'(\eta) = \mathbb{E}_{\pi'}[r] - \mu, \quad \psi''(\eta) = \mathrm{Var}_{\pi'}(r).$$

Thus,

$$D_{\mathrm{KL}}(\pi' \| \pi) = \eta\psi'(\eta) - \psi(\eta) = \int_0^\eta t\, \psi''(t)\, dt.$$

Since $\|r\|_\infty \leq R_{\max}$, we have $\psi''(t) = \mathrm{Var}_{\pi_t}(r) \leq R_{\max}^2$ for all t. Therefore,

$$D_{\mathrm{KL}}(\pi' \| \pi) \leq \int_0^\eta t R_{\max}^2\, dt = \frac{1}{2}\eta^2 R_{\max}^2.$$

□

B.3 Detailed Proof

Proof of Theorem 4.4. According to Definition 4.3, the duality gap is defined as:

$$\begin{aligned}\mathrm{DualGap} &= \max_{r \in \mathcal{R}} J(\bar{\pi}, r) - \min_{\pi \in \Pi} J(\pi, \bar{r}) \\ &= \max_{r \in \mathcal{R}} \langle r, \pi^\star - \bar{\pi} \rangle + \max_{\pi \in \Pi} \langle \bar{r}, \pi - \pi^\star \rangle.\end{aligned}$$

Multiplying by K and using the definitions $\bar{\pi} = \frac{1}{K}\sum_{k=1}^{K}\pi^k$ and $\bar{r} = \frac{1}{K}\sum_{k=1}^{K} r^k$, we can decompose $K \cdot \text{DualGap}$ by adding and subtracting $\sum_{k=1}^{K}\langle r^k, \pi^\star - \pi^k\rangle$:

$$
\begin{aligned}
K \cdot \text{DualGap} &= \max_{r\in\mathcal{R}} \sum_{k=1}^{K}\langle r, \pi^\star - \pi^k\rangle + \max_{\pi\in\Pi}\sum_{k=1}^{K}\langle r^k, \pi - \pi^\star\rangle \\
&= \underbrace{\left(\max_{r\in\mathcal{R}}\sum_{k=1}^{K}\langle r, \pi^\star - \pi^k\rangle - \sum_{k=1}^{K}\langle r^k, \pi^\star - \pi^k\rangle\right)}_{\text{Regret}_r} \\
&\quad + \underbrace{\left(\max_{\pi\in\Pi}\sum_{k=1}^{K}\langle r^k, \pi - \pi^\star\rangle + \sum_{k=1}^{K}\langle r^k, \pi^\star - \pi^k\rangle\right)}_{\text{Regret}_\pi}. \qquad (9)
\end{aligned}
$$

Notice that the two inner terms in the second bracket combine such that $\langle r^k, \pi - \pi^\star\rangle + \langle r^k, \pi^\star - \pi^k\rangle = \langle r^k, \pi - \pi^k\rangle$. Thus, it exactly simplifies to the π-player's regret:

$$
\text{Regret}_\pi = \max_{\pi\in\Pi}\sum_{k=1}^{K}\langle r^k, \pi - \pi^k\rangle.
$$

By Lemma B.2, the r-player's regret is bounded by $\mathcal{O}(R_{\max}^2 B\sqrt{K})$.

For the π-player, treating $\pi \in \Pi$ as the comparator in Lemma B.1 and taking $\eta = \beta^{-1}$, we sum the one-step descent bound over $k = 1, \ldots, K$:

$$
\sum_{k=1}^{K}\langle r^k, \pi - \pi^k\rangle \le \sum_{k=1}^{K}\frac{R_{\max}^2}{2\beta} + \beta\sum_{k=1}^{K}\left[D_{\text{KL}}(\pi(\cdot|x)\|\pi^k(\cdot|x)) - D_{\text{KL}}(\pi(\cdot|x)\|\pi^{k+1}(\cdot|x))\right].
$$

Taking the maximum over $\pi \in \Pi$, bounding the telescoping sum by $D = \max_{\pi\in\Pi} D_{\text{KL}}(\pi^\star\|\pi)$, and selecting $\beta = \sqrt{K}/D$, we obtain:

$$
\text{Regret}_\pi \le \frac{KR_{\max}^2}{2\beta} + \beta D = \mathcal{O}(D \cdot R_{\max}^2\sqrt{K}).
$$

Substituting the bounds for Regret_r and Regret_π back into (9) yields:

$$
K \cdot \text{DualGap} \le \mathcal{O}(R_{\max}^2 B\sqrt{K}) + \mathcal{O}(D \cdot R_{\max}^2\sqrt{K}) = \mathcal{O}\left((D+B)R_{\max}^2\sqrt{K}\right).
$$

Dividing both sides by K, we obtain the final bound:

$$
\text{DualGap} \le \mathcal{O}\left(\frac{(D+B)R_{\max}^2}{\sqrt{K}}\right),
$$

which concludes the proof. □

C Proof of Proposition 4.1

Proof of Proposition 4.1. We first proof the boundedness of the optimal reward. This proof follows the proof of Proposition A.2 in Al-Hafez et al. (2023). For the mixed χ^2 divergence, by the definition of the variational form of Pearson χ^2 divergence, we have:

$$
\begin{aligned}
&2D_{\chi^2}(\pi^\star\|(\pi+\pi^\star)/2) \\
&= \max_r\ 2\mathbb{E}_\rho\left(\mathbb{E}_{\pi^\star}[r(x,y)] - \mathbb{E}_\pi[r(x,y)] - \frac{c}{2}\mathbb{E}_{\pi^\star}[(r(x,y))^2] - \frac{c}{2}\mathbb{E}_\pi[(r(x,y))^2]\right) \\
&= \max_r \int_{\mathcal{X}}\int_{\mathcal{Y}} \pi^\star(y|x)\rho(x)\left(r(x,y) - \frac{c}{2}r(x,y)^2\right) - \pi(y|x)\rho(x)\left(r(x,y) + \frac{c}{2}r(x,y)^2\right)\mathrm{d}x\mathrm{d}y.
\end{aligned}
$$

For any $a, b \in \mathbb{R}^+/\{0\}$, the function $r \to a(r - \frac{c}{2}r^2) - b(r + \frac{c}{2}r^2)$ reaches its maximum at $\frac{1}{c}\frac{a-b}{a+b}$, which is bounded by $[-1/c, 1/c]$. By setting $a = \rho(x)\pi^\star(y|x)$ and $b = \rho(x)\pi(y|x)$, we can obtain the closed-form of the optimal reward under mixed χ^2 divergence matching:

$$r^\star(x, y) = \frac{1}{c}\frac{\pi^\star(y|x) - \pi(y|x)}{\pi^\star(y|x) + \pi(y|x)}\mathbb{I}\big(\pi^\star(y|x) \neq 0 \wedge \pi(y|x) \neq 0\big),$$

which is bounded by $[-1/c, 1/c]$. Furthermore, we prove the boundedness of the mixed Pearson χ^2 divergence $D_{\chi^2}(\pi^\star \| (\pi^\star + \pi)/2)$, i.e., $0 \leq 2D_{\chi^2}(\pi^\star \| (\pi^\star + \pi)/2) \leq \frac{1}{c}$. The proof of this result is adapted from the proof of Proposition A.3 in Al-Hafez et al. (2023). We consider the following algebraic transformation by plugging in the optimal reward formulation:

$$\begin{aligned}
&2D_{\chi^2}(\pi^\star \| (\pi + \pi^\star)/2) \\
&= \int_{\mathcal{X}}\int_{\mathcal{Y}} \pi^\star(y|x)\rho(x)\left(r^\star(x,y) - \frac{c}{2}r^\star(x,y)^2\right) - \pi(y|x)\rho(x)\left(r^\star(x,y) + \frac{c}{2}r^\star(x,y)^2\right)\mathrm{d}x\mathrm{d}y \\
&= \int_{\mathcal{X}}\int_{\mathcal{Y}} \pi^\star\rho\left(\frac{1}{c}\left(\frac{\pi^\star - \pi}{\pi^\star + \pi}\right) - \frac{1}{2c}\left(\frac{\pi^\star - \pi}{\pi^\star + \pi}\right)^2\right) \\
&\qquad - \pi\rho\left(\frac{1}{c}\left(\frac{\pi^\star - \pi}{\pi^\star + \pi}\right) + \frac{1}{2c}\left(\frac{\pi^\star - \pi}{\pi^\star + \pi}\right)^2\right)\mathrm{d}x\mathrm{d}y \\
&= \int_{\mathcal{X}}\int_{\mathcal{Y}} \frac{\rho(x)}{2c}\frac{(\pi^\star(y|x) - \pi(y|x))^2}{\pi^\star(y|x) + \pi(y|x)}\mathrm{d}x\mathrm{d}y \\
&= \frac{1}{2c}\mathbb{E}_\rho\left(\mathbb{E}_{\pi^\star}\left[\frac{\pi^\star}{\pi^\star + \pi}\right] + \mathbb{E}_\pi\left[\frac{\pi}{\pi^\star + \pi}\right] + \mathbb{E}_{\pi^\star}\left[\frac{-2\pi}{\pi^\star + \pi}\right]\right) \leq \frac{1}{c},
\end{aligned}$$

which concludes the proof. □

D Proof of Proposition 4.2

Proof. By Definition of the optimization problem in (1). □

E Proof of Proposition 5.1

Proof of Proposition 5.1. Consider the reward update rule:

$$(r)^k = \text{argmax}_r\, \mathcal{J}(r) \coloneqq \mathbb{E}_\rho\left[\sigma(\mathbb{E}_{\pi^\star}\, r(x,y) - \mathbb{E}_{\pi^k}\, r(x,y)) - \psi(r, (r)^{k-1})\right],$$

where $\psi(r, (r)^{k-1}) = \zeta D_f(r, (r)^{k-1}) + c\alpha \cdot \mathbb{E}_{\pi^\star}[(r(x,y))^2] + c(1-\alpha) \cdot \mathbb{E}_\pi[(r(x,y))^2]$ is the Bregman divergence constrained convex regularizer, and $\sigma(t) = t$ is the identical link function. By simple algebraic manipulation:

$$\begin{aligned}
&\mathcal{J}(r) \\
&= \mathbb{E}_\rho\left[\mathbb{E}_{\pi^\star}\, r(x,y) - \mathbb{E}_{\pi^k}\, r(x,y) - \psi(r, (r)^{k-1})\right] \\
&= \mathbb{E}_\rho\left[\mathbb{E}_{\pi^\star}\, r(x,y) - \mathbb{E}_{\pi^k}\, r(x,y) - c\alpha \cdot \mathbb{E}_{\pi^\star}[(r(x,y))^2] - c(1-\alpha) \cdot \mathbb{E}_{\pi^k}[(r(x,y))^2]\right] \\
&\qquad - \mathbb{E}_\rho \zeta D_f(r, (r)^{k-1}) \\
&= \mathbb{E}_\rho c\alpha\left[\frac{1}{c\alpha}\mathbb{E}_{\pi^\star}\, r(x,y) - \cdot\mathbb{E}_{\pi^\star}[(r(x,y))^2]\right] + \mathbb{E}_\rho c(1-\alpha)\left[-\mathbb{E}_{\pi^k}[(r(x,y))^2] - \frac{1}{c(1-\alpha)}\mathbb{E}_{\pi^k}\, r(x,y)\right] \\
&\qquad - \mathbb{E}_\rho \zeta D_f(r, (r)^{k-1}) \\
&= -\mathbb{E}_{\rho,\pi^\star}\, c\alpha\left[r(x,y) - \frac{1}{2c\alpha}\right]^2 + \frac{1}{4c\alpha} - \mathbb{E}_{\rho,\pi^k}\, c(1-\alpha)\left[r(x,y) + \frac{1}{2c(1-\alpha)}\right]^2 + \frac{1}{4c(1-\alpha)} \\
&\qquad - \mathbb{E}_\rho \zeta D_f(r, (r)^{k-1}).
\end{aligned}$$

Turning the maximization to minimization:

$$
\begin{aligned}
&\text{argmax}_r\ \mathcal{J}(r) \\
&= \text{argmin}_r\ \mathcal{L}(r) \\
&:= \mathbb{E}_{\rho,\pi^\star}\ c\alpha\ \left[r(x,y) - \frac{1}{2c\alpha}\right]^2 + \frac{1}{4c\alpha} + \mathbb{E}_{\rho,\pi^k}\ c(1-\alpha)\left[r(x,y) + \frac{1}{2c(1-\alpha)}\right]^2 + \frac{1}{4c(1-\alpha)} \\
&\quad + \mathbb{E}_\rho \zeta D_f(r,(r)^{k-1}),
\end{aligned}
$$

which is equivalent to:

$$
\text{argmin}_r \mathbb{E}_{\rho,\pi^\star}\ \alpha\ \left[r(x,y) - \frac{1}{2c\alpha}\right]^2 + \mathbb{E}_{\rho,\pi^k}\ (1-\alpha)\left[r(x,y) + \frac{1}{2c(1-\alpha)}\right]^2 + \mathbb{E}_\rho\ \zeta' D_f(r,(r)^{k-1}),
$$

where $\zeta' := \zeta/c$ and removing the constants that do not affect the learning objective. Taking $D_f(x,x') = \frac{1}{2}\|x-x'\|^2$ over both data generated by $\pi^\star$ and π^k and plugging in the reparameterization of r, i.e, $r = \beta\log(\pi/\pi^k)$ will lead to the final objective:

$$
\begin{aligned}
\text{argmin}_\pi \mathbb{E}_{\rho,\pi^\star}\ \alpha\ &\left[\beta\log\frac{\pi(y|x)}{\pi^k(y|x)} - \frac{1}{2c\alpha}\right]^2 + \mathbb{E}_{\rho,\pi^k}\ (1-\alpha)\left[\beta\log\frac{\pi(y|x)}{\pi^k(y|x)} + \frac{1}{2c(1-\alpha)}\right]^2 \\
&+ \mathbb{E}_{\rho,\pi^\star,\pi^k}\ \zeta''\left[\log\frac{\pi(y|x)}{\pi^k(y|x)}\right]^2.
\end{aligned}
$$

Taking $\zeta'' := \beta^2\zeta/c$ concludes the proof. □

F Proof of Proposition A.1

Proof. From Proposition A.1, we consider the general preference alignment formulation with the reparameterized reward r defined in (3). For each $x \sim \rho(x)$, we sample $y \sim \pi^k(\cdot|x)$ and denote $w^k(x,y) := P(y \succ \pi^k \mid x)$. The reward-player objective is

$$
\max_r\ \mathbb{E}_\rho\Big[\mathbb{E}_{\pi^k_+} r(x,y) - \mathbb{E}_{\pi^k_-} r(x,y) - \psi(r, r^{k-1})\Big]. \tag{10}
$$

Taking balanced sampling with $\alpha = 0.5$, the mixed χ^2 divergence regularizer without the Bregman divergence $D_f(r, r^{k-1})$ reduces to

$$
\phi(r, r^{k-1}) = \phi(r) = \frac{c}{2}\mathbb{E}_{\pi^k_+}\big[(r(x,y))^2\big] + \frac{c}{2}\mathbb{E}_{\pi^k_-}\big[(r(x,y))^2\big]. \tag{11}
$$

By simple algebraic transformation,

$$
\begin{aligned}
&\mathbb{E}_{\pi^k_+} r(x,y) - \mathbb{E}_{\pi^k_-} r(x,y) - \psi(r) \\
&= \mathbb{E}_{y\sim\pi^k(\cdot|x)}\big[w^k(x,y)r(x,y) - (1-w^k(x,y))r(x,y)\big] - c\mathbb{E}_{y\sim\pi^k(\cdot|x)}\big[(r(x,y))^2\big] \\
&= \mathbb{E}_{y\sim\pi^k(\cdot|x)}\big[(2w^k(x,y)-1)r(x,y) - c(r(x,y))^2\big].
\end{aligned}
$$

Discarding constant terms that do not affect the optimization, the above objective is equivalent to

$$
\min_r\ \mathbb{E}_{y\sim\pi^k(\cdot|x)}\left[r(x,y) - \frac{2w^k(x,y)-1}{2c}\right]^2. \tag{12}
$$

Substituting the reparameterized reward in (3),

$$
r(x,y) := \beta\log\frac{\pi(y|x)}{\pi^k(y|x)},
$$

the minimization in (12) can be written as

$$
\min_\pi\ \mathcal{L}(\pi) := \mathbb{E}_{y\sim\pi^k(\cdot|x)}\left[\log\frac{\pi(y|x)}{\pi^k(y|x)} - \frac{1}{c\beta}\Big(w^k(x,y) - \tfrac{1}{2}\Big)\right]^2. \tag{13}
$$

The original SPPO algorithm (Wu et al., 2024) sets $c = 1$ for the χ^2 regularization, which completes the proof. □

G Proof of Proposition A.3

Proof. From Proposition A.3, we consider the general preference alignment formulation with the reparameterized reward r defined in (5). For each $x \sim \rho(x)$, we sample $(y, y') \sim \pi^k(\cdot|x) \times \pi^k(\cdot|x)$ and define $w^k(x, y) := P(y \succ \pi^k \mid x)$. The reward-player objective is

$$\max_r \ \mathbb{E}_\rho\Big[\mathbb{E}_{(y,y')\sim\pi^k\times\pi^k}\big[(w^k(x,y) - w^k(x,y'))r(x,y,y')\big] - \psi(r, r^{k-1})\Big]. \tag{14}$$

Taking balanced sampling with $\alpha = 0.5$, the mixed χ^2 divergence regularizer without Bregman divergence $D_f(r, r^{k-1})$ reduces to

$$\psi(r, r^{k-1}) = \psi(r) = \frac{c}{2}\mathbb{E}_{(y,y')\sim\pi^k\times\pi^k}\big[r(x,y,y')^2\big]. \tag{15}$$

By simple algebraic transformation,

$$\begin{aligned}
&\mathbb{E}_{(y,y')\sim\pi^k\times\pi^k}\big[(w^k(x,y) - w^k(x,y'))r(x,y,y')\big] - \psi(r) \\
&= -\frac{c}{2}\mathbb{E}_{(y,y')\sim\pi^k\times\pi^k}\Big[r(x,y,y') - \frac{w^k(x,y) - w^k(x,y')}{c}\Big]^2 + \text{const.}
\end{aligned}$$

Discarding constant terms that do not affect the optimization and setting $c = 1$, the reward-player optimization is equivalent to

$$\min_r \ \mathcal{L}(r) := \mathbb{E}_{(y,y')\sim\pi^k\times\pi^k}\Big[r(x,y,y') - \big(w^k(x,y) - w^k(x,y')\big)\Big]^2. \tag{16}$$

We further substitute the reward reparameterization in (5),

$$r(x,y,y') = \eta \log\frac{\pi(y|x)}{\pi(y'|x)} - \tau\log\frac{\pi_{\text{ref}}(y|x)}{\pi_{\text{ref}}(y'|x)} - (\eta - \tau)\log\frac{\pi^k(y|x)}{\pi^k(y'|x)}.$$

The minimization in (16) can thus be written as

$$\min_\pi \ \mathcal{L}(\pi) := \mathbb{E}_{(y,y')\sim\pi^k\times\pi^k}\left[h^k(\pi,x,y,y') - \frac{w^k(x,y) - w^k(x,y')}{\eta}\right]^2, \tag{17}$$

where

$$h^k(\pi,x,y,y') = \log\frac{\pi(y|x)}{\pi(y'|x)} - \frac{\tau}{\eta}\log\frac{\pi_{\text{ref}}(y|x)}{\pi_{\text{ref}}(y'|x)} - \frac{\eta-\tau}{\eta}\log\frac{\pi^k(y|x)}{\pi^k(y'|x)}.$$

By Proposition 6 in Zhang et al. (2024), minimizing the above objective recovers the policy update used in INPO, which completes the proof. □

H Linear SPIN as TV Distance Minimization

In this section, we show that linear SPIN is equivalent to minimizing the total variation (TV) distance between the model distribution and the expert data distribution. Choosing $\psi(r) = 0$ if $|r| \le R_{\max}$ and $\psi(r) = \infty$ otherwise, setting $\sigma(t) = t$, (1) is equivalent to:

$$\max_{r\in\mathcal{R}}\min_{\pi\in\Pi}\ \mathop{\mathbb{E}}_{x\sim\rho(x)}\left(\mathop{\mathbb{E}}_{y\sim\pi^\star} r(x,y) - \mathop{\mathbb{E}}_{y\sim\pi} r(x,y)\right)\mathbb{I}(r \le R_{\max}),$$

where $R_{\max}$ can be arbitrarily large since original SPIN doesn't constrain the reward magnitude explicitly. By leveraging the reward reparameterization in (3):

$$r(x,y) = \beta\log\left(\frac{\pi(y|x)}{\pi^k(y|x)}\right),$$

it recovers the identical version of SPIN. By algebraic transformation:

$$\begin{aligned}\max_{r\in\mathcal{R}}\min_{\pi\in\Pi}\ \mathcal{J}(r,\pi) &:= \mathop{\mathbb{E}}_{x\sim\rho(x)}\left(\mathop{\mathbb{E}}_{y\sim\pi^\star} r(x,y) - \mathop{\mathbb{E}}_{y\sim\pi} r(x,y)\right)\mathbb{I}(r\le R_{\max})\\ &= \mathop{\mathbb{E}}_{x\sim\rho(x)}\langle r(x,\cdot),\pi^\star(\cdot|x)-\pi(\cdot|x)\rangle\mathbb{I}(r\le R_{\max})\\ &\le R_{\max}\mathbb{E}_\rho|\pi^\star(\cdot|x)-\pi(\cdot|x)|. \end{aligned}\tag{18}$$

Consider $r^\star(x,y)$ as the optimal reward solution for reward maximization part, by Assumption 3.1:

$$r^\star(x,y) = R_{\max}\cdot \mathrm{sgn}(\pi^\star(y|x)-\pi(y|x)).$$

Therefore, the inequality in (18) reaches equal when the optimal reward is reached. In this case,

$$\max_{r\in\mathcal{R}}\min_{\pi\in\Pi}\ \mathcal{J}(r,\pi) = \min_{\pi\in\Pi} 2R_{\max}\cdot\mathbb{E}_{x\sim\rho}\, D_{\mathrm{TV}}(\pi(\cdot|x),\pi^\star(\cdot|x))$$

holds because $D_{\mathrm{TV}}(\mu,\nu)=\frac{1}{2}\|\mu-\nu\|_1$. This is a Total Variation distance minimization.

I Non-Linear SPIN as KL Divergence Minimization

We begin by the following lemma for contraction:

Lemma I.1 (KL contraction toward $\pi^\star$)**.** *Let $\beta\ge 1$ and $\alpha := 1/\beta\in(0,1]$, and define the update*

$$\pi^{k+1}(y|x)\ \propto\ \pi^k(y|x)\Big(\frac{\pi^\star(y|x)}{\pi^k(y|x)}\Big)^\alpha\ =\ \pi^k(y|x)^{1-\alpha}\pi^\star(y|x)^\alpha. \tag{19}$$

Then the reverse KL to data contracts geometrically:

$$D_{KL}\big(\pi^\star\|\pi^{k+1}\big)\ \le\ (1-\alpha)\,D_{KL}\big(\pi^\star\|\pi^k\big)\ =\ \Big(1-\frac{1}{\beta}\Big)D_{KL}\big(\pi^\star\|\pi^k\big).$$

Consequently,

$$D_{KL}\big(\pi^\star\|\pi^k\big)\ \le\ \Big(1-\frac{1}{\beta}\Big)^k D_{KL}\big(\pi^\star\|\pi^{\mathrm{ref}}\big)\ \xrightarrow[k\to\infty]{}\ 0.$$

Proof. We begin by explicitly writing the normalized update rule. Let $Z(x)$ be the normalization constant (partition function) for the update in (19):

$$Z(x) = \sum_y \pi^k(y|x)^{1-\alpha}\pi^\star(y|x)^\alpha = \mathbb{E}_{y\sim\pi^k(\cdot|x)}\left[\left(\frac{\pi^\star(y|x)}{\pi^k(y|x)}\right)^\alpha\right].$$

The normalized policy is then given by:

$$\pi^{k+1}(y|x) = \frac{1}{Z(x)}\pi^k(y|x)^{1-\alpha}\pi^\star(y|x)^\alpha.$$

We now expand the KL divergence $D_{\mathrm{KL}}(\pi^\star\|\pi^{k+1})$:

$$\begin{aligned}D_{\mathrm{KL}}(\pi^\star\|\pi^{k+1}) &= \mathbb{E}_{y\sim\pi^\star}\left[\log\frac{\pi^\star(y|x)}{\pi^{k+1}(y|x)}\right]\\ &= \mathbb{E}_{y\sim\pi^\star}\left[\log\pi^\star(y|x) - \log\left(\frac{\pi^k(y|x)^{1-\alpha}\pi^\star(y|x)^\alpha}{Z(x)}\right)\right]\\ &= \mathbb{E}_{y\sim\pi^\star}\Big[\log\pi^\star(y|x) - (1-\alpha)\log\pi^k(y|x) - \alpha\log\pi^\star(y|x) + \log Z(x)\Big]\\ &= (1-\alpha)\mathbb{E}_{y\sim\pi^\star}\Big[\log\pi^\star(y|x)-\log\pi^k(y|x)\Big] + \log Z(x)\\ &= (1-\alpha)D_{\mathrm{KL}}(\pi^\star\|\pi^k) + \log Z(x). \end{aligned}\tag{20}$$

To bound the remainder term $\log Z(x)$, we invoke Jensen's inequality. Since $\beta \geq 1$, we have $\alpha = 1/\beta \in (0, 1]$. Consequently, the function $f(t) = t^\alpha$ is concave. Applying Jensen's inequality to the expectation of the likelihood ratio under π^k:

$$\begin{aligned} Z(x) &= \mathbb{E}_{y\sim\pi^k}\left[\left(\frac{\pi^\star(y|x)}{\pi^k(y|x)}\right)^\alpha\right] \\ &\leq \left(\mathbb{E}_{y\sim\pi^k}\left[\frac{\pi^\star(y|x)}{\pi^k(y|x)}\right]\right)^\alpha \\ &= \left(\sum_y \pi^k(y|x)\frac{\pi^\star(y|x)}{\pi^k(y|x)}\right)^\alpha \\ &= \left(\sum_y \pi^\star(y|x)\right)^\alpha = 1^\alpha = 1. \end{aligned}$$

Thus, $Z(x) \leq 1$, which implies $\log Z(x) \leq 0$. Substituting this inequality back into (20), we obtain:

$$D_{\mathrm{KL}}(\pi^\star\|\pi^{k+1}) \leq (1-\alpha)D_{\mathrm{KL}}(\pi^\star\|\pi^k).$$

Substituting $\alpha = 1/\beta$ yields the geometric contraction:

$$D_{\mathrm{KL}}(\pi^\star\|\pi^{k+1}) \leq \left(1 - \frac{1}{\beta}\right) D_{\mathrm{KL}}(\pi^\star\|\pi^k).$$

Applying this inequality recursively k times leads to the final convergence rate:

$$D_{\mathrm{KL}}(\pi^\star\|\pi^k) \leq \left(1 - \frac{1}{\beta}\right)^k D_{\mathrm{KL}}(\pi^\star\|\pi^{\mathrm{ref}}).$$

This completes the proof. □

Consider the choice of logistic link function $\sigma(t) = -\log(1 + \exp(-t))$ in (2), $\psi(r) = \infty \cdot \mathbf{1}[|r|_\infty > R_{\max}]$ without the Bregman constraint over r, it recovers the original SPIN (Chen et al., 2024) objective with non-identical link function. SPIN has proved the following lemma:

Lemma I.2 (Theorem 5.4 in Chen et al. 2024)**.** *Consider the choice of logistic link function in SPIN. Suppose π^{k+1} is the global minimum for the SPIN objective at iteration k, then the opponent player at iteration $k+1$ satisfies:*

$$\pi^{k+1}(y|x) \propto \pi^k(y|x)\left(\pi^\star(y|x)/\pi^k(y|x)\right)^{1/\beta}.$$

By first applying Lemma I.2 for SPIN objective then applying Lemma I.1, we can prove that SPIN with non-identical link function contracts under KL divergence between $\pi^\star$ and π^k for multiple iterations.

J Iterative DPO as KL Divergence Minimization

In this section, we analyze the theoretical properties of Iterative DPO (Tu et al., 2025; Wu et al., 2024). We demonstrate that iteratively solving the KL-regularized reinforcement learning objective—using the current policy as the reference—constitutes a contraction mapping that minimizes the KL divergence between the current policy and the optimal policy implied by the preference oracle. Let $\mathcal{X}$ be the input space and $\mathcal{Y}$ be the output space. We assume access to a preference oracle $P(y_1 \succ y_2|x)$ which adheres to the Bradley-Terry (BT) model. Under the BT model, the preference probability is determined by a latent reward function $r^\star(x, y)$:

$$P(y_1 \succ y_2|x) = \frac{\exp(r^\star(x, y_1))}{\exp(r^\star(x, y_1)) + \exp(r^\star(x, y_2))} = \sigma(r^\star(x, y_1) - r^\star(x, y_2)).$$

Let $\pi^\star$ denote the optimal policy that perfectly captures the underlying reward structure, i.e., $\pi^\star(y|x) \propto \exp(r^\star(x,y))$.

We consider an iterative setting where, at step k, we optimize a policy π against a reference policy π^k (the policy from the previous iteration). The objective is to maximize the expected reward subject to a KL-divergence constraint:

$$\max_{\pi} \mathcal{J}_k(\pi) = \mathbb{E}_{y\sim\pi(\cdot|x)}[r^\star(x,y)] - \beta D_{\text{KL}}(\pi(\cdot|x)||\pi^k(\cdot|x)). \tag{21}$$

Lemma J.1. *Given the reference policy π^k and the preference oracle P, the optimal policy π^{k+1} maximizing the objective in (21) is given by:*

$$\pi^{k+1}(y|x) \propto \pi^k(y|x)\left(\frac{P(y \succ y_{\text{ref}}|x)}{1-P(y \succ y_{\text{ref}}|x)}\right)^{1/\beta}$$

where preference estimates are taken relative to a baseline y_{ref} drawn from π^k.

Proof. The optimization problem in (21) has a well-known closed-form solution given by the Boltzmann distribution:

$$\pi^{k+1}(y|x) = \frac{1}{Z_k(x)}\pi^k(y|x)\exp\left(\frac{r^\star(x,y)}{\beta}\right). \tag{22}$$

To express the reward $r^\star(x,y)$ in terms of the oracle, we utilize the Bradley-Terry relationship. The odds of preferring y over a reference output y_{ref} are:

$$\frac{P(y \succ y_{\text{ref}}|x)}{P(y_{\text{ref}} \succ y|x)} = \frac{e^{r^\star(x,y)}}{e^{r^\star(x,y_{\text{ref}})}} = \exp(r^\star(x,y) - r^\star(x,y_{\text{ref}})).$$

Taking the logarithm yields the reward function (up to a constant shift $r^\star(x,y_{\text{ref}})$ which is absorbed into the partition function):

$$r^\star(x,y) = \log\left(\frac{P(y \succ y_{\text{ref}}|x)}{1-P(y \succ y_{\text{ref}}|x)}\right) + C.$$

Substituting this expression back into (22) yields the proposition:

$$\pi^{k+1}(y|x) \propto \pi^k(y|x)\left(\frac{P(y \succ y_{\text{ref}}|x)}{1-P(y \succ y_{\text{ref}}|x)}\right)^{1/\beta}.$$

□

We now show that this iterative process monotonically reduces the distance to the optimal policy $\pi^\star$.

Proposition J.2 (Contraction for Iterative DPO)**.** *Let $\pi^\star$ be the global optimal policy implied by the reward oracle. The sequence of policies $\{\pi^k\}_{k=0}^{\infty}$ generated by the update rule in Proposition J.1 satisfies the following contraction inequality:*

$$D_{KL}(\pi^\star||\pi^{k+1}) \le D_{KL}(\pi^\star||\pi^k) - D_{KL}(\pi^{k+1}||\pi^k).$$

Consequently, $D_{KL}(\pi^\star||\pi^{k+1}) < D_{KL}(\pi^\star||\pi^k)$ for all non-trivial updates ($\pi^{k+1} \neq \pi^k$).

Proof. We expand the KL divergence term $D_{\text{KL}}(\pi^\star||\pi^{k+1})$:

$$\begin{aligned}
D_{\text{KL}}(\pi^\star||\pi^{k+1}) &= \sum_y \pi^\star(y)\log\frac{\pi^\star(y)}{\pi^{k+1}(y)} \\
&= \sum_y \pi^\star(y)\left[\log\pi^\star(y) - \log\left(\frac{\pi^k(y)\exp(r^\star(y)/\beta)}{Z_k}\right)\right] \\
&= \sum_y \pi^\star(y)\log\frac{\pi^\star(y)}{\pi^k(y)} - \frac{1}{\beta}\mathbb{E}_{y\sim\pi^\star}[r^\star(y)] + \log Z_k \\
&= D_{\text{KL}}(\pi^\star||\pi^k) - \frac{1}{\beta}\mathbb{E}_{\pi^\star}[r^\star] + \log Z_k.
\end{aligned} \tag{23}$$

Hyperparameters	Value
β	$1e-3$
ζ	$1e-3$
α	0.5
c	2
Batch Size	32
Optimizer	AdamW
Learning Rate	$5e-7$
Learning Rate Scheduler	Linear
Warm-up Ratio	0.1
Epochs per Iteration	3
Generation Length	256

Table 3: **Hyperparameters.** This table lists the hyperparameters used for the algorithm, training, and generation.

Next, we relate the log-partition function $\log Z_k$ to the KL divergence between steps. By definition:

$$D_{\mathrm{KL}}(\pi^{k+1}||\pi^k) = \mathbb{E}_{\pi^{k+1}}\left[\log \frac{\pi^{k+1}}{\pi^k}\right] = \mathbb{E}_{\pi^{k+1}}\left[\frac{r^\star}{\beta} - \log Z_k\right] = \frac{1}{\beta}\mathbb{E}_{\pi^{k+1}}[r^\star] - \log Z_k.$$

Solving for $\log Z_k$:

$$\log Z_k = \frac{1}{\beta}\mathbb{E}_{\pi^{k+1}}[r^\star] - D_{\mathrm{KL}}(\pi^{k+1}||\pi^k). \tag{24}$$

Substituting (24) into (23):

$$D_{\mathrm{KL}}(\pi^\star||\pi^{k+1}) = D_{\mathrm{KL}}(\pi^\star||\pi^k) - D_{\mathrm{KL}}(\pi^{k+1}||\pi^k) - \frac{1}{\beta}\underbrace{\left(\mathbb{E}_{\pi^\star}[r^\star] - \mathbb{E}_{\pi^{k+1}}[r^\star]\right)}_{\Delta \geq 0}.$$

Since $\pi^\star$ maximizes the expected reward, the term Δ is non-negative. Since D_{KL} is non-negative, the inequality holds strictly, proving contraction towards the oracle distribution. □

K Experimental Settings and Implementation Details

K.1 Hyperparameters

In this section, we describe in detail the hyperparameters used in the practical implementation of our method. We adopt a single fixed set of hyperparameters across all self play iterations and model architectures. Table 3 summarizes the hyperparameters used in our algorithm, including the values of α, β, ζ, and c, as well as the training hyperparameters such as batch size, optimizer, learning rate, and scheduler, and the generation configuration including generation length.

K.2 Experimental Settings

Dataset Preparation For the first iteration, we use 50k examples subsampled from the UltraChat SFT dataset (Ding et al., 2023). For each prompt x, we construct (x, y, y') tuples, where y is the reference response from the dataset and y' is the response generated by the model. The pairs (x, y) are included in $\mathcal{D}^\star$, while (x, y') form $\mathcal{D}^0$ for training. Starting from the second iteration, we train using both our method and SPIN (Chen et al., 2024) on the datasets from the two most recent iterations, resulting in a combined dataset of 100k examples.

Baseline Methods For the SFT baseline, we reproduce supervised fine tuning by directly applying maximum likelihood estimation on $\mathcal{D}^\star$. For the SPIN baseline (Chen et al., 2024), we use the authors' official implementation with their recommended hyperparameter settings.

Scores (3 Iters)	w/ Regularizer	w/o Regularizer
Arc-Challenge	57.11	56.87
MMLU	68.83	68.79
HellaSwag	71.92	70.05
WinoGrande	68.82	68.43

Table 4: **Ablation on the Reward Constraint.** We conduct an ablation study on the reward regularizer that enforces mirror descent on the reward player. The results demonstrate that this regularization improves self-play performance. All results are reported after three self-play iterations on the Qwen-3-4B model.

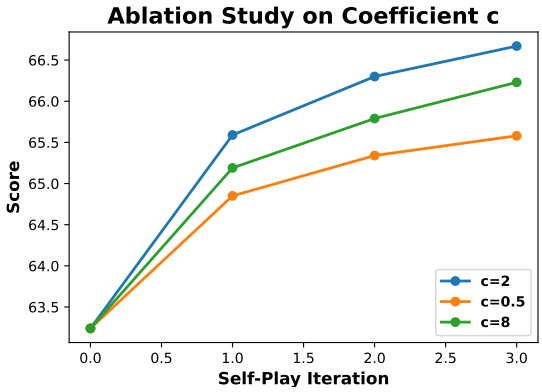

Figure 4: **Ablation on Hyperparameter c.** We evaluate the impact of the hyperparameter c by setting $c \in \{0.5, 2, 8\}$ and examining its effect on the self-play performance of our method. Performance is measured as the mean score across the four benchmarks used in the main experiments. We observe that both overly small and overly large values of c lead to performance degradation, highlighting the importance of an appropriate balance in reward scaling.

Evaluation Metrics We evaluate performance on Arc Challenge using a 25-shot setting with `acc_norm` as the evaluation metric. For MMLU, we use a 5-shot setting and report `acc`. For HellaSwag, we adopt a 10-shot evaluation with `acc_norm`. Finally, for WinoGrande, we use a 5-shot setting and report `acc`.

L Ablation Studies

Ablation on Hyperparameter c. We conduct an ablation study on the hyperparameter c, which controls the reward targets $r_{\max}$ and $r_{\min}$ in the least-squares regression objective. A larger value of c results in a smaller margin between $r_{\max}$ and $r_{\min}$, whereas a smaller value of c induces a larger margin and higher reward magnitude. We vary c to examine the effect of this trade-off.

In our main experiments, we set $c = 2$, corresponding to $r_{\max} = 0.5$ and $r_{\min} = -0.5$. When c is reduced to 0.5, the resulting larger reward magnitude leads to degraded performance, which is consistent with the theoretical predictions in Sec 4.3. Conversely, setting $c = 8$ yields a small reward magnitude and a narrow margin, potentially limiting the ability to discriminate between data generated by the expert policy $\pi^\star$ and the previous-iteration policy π^k.

Figure 4 summarizes the results for these settings, reporting the mean performance of the Qwen-3 4B model across four evaluation benchmarks (Arc-Challenge, MMLU, HellaSwag, WinoGrande) over three self-play iterations (with iteration 0 corresponding to the base model).

Ablation on $D_f(r, r^{k-1})$ Constraint on Reward. Our method includes an additional regularization term that constrains the deviation between successive policies π^{k+1} and π^k, inducing a mirror descent structure. We ablate this component by removing the final term in (4). The results show that omitting this regularizer leads to noticeable performance degradation, empirically demonstrating its effectiveness. We report results across four benchmarks in Table 4 after three self-play iterations on the Qwen-3-4B model.

Model	Base	SFT	SPIN (3 Iters)	SPACE (3 Iters)	SPIF (3 Iters)
Qwen3-4B	84.91	85.67	86.31	87.24	**88.55**
Mistral-7B	33.36	36.94	39.13	39.52	**40.44**

Table 5: **GSM8K Results.** We report GSM8K accuracy for the base model, SFT, SPIN, SPACE and SPIF methods after three self-play iterations. Results are reported with five random seeds.

M Additional Results on Mathematical Capabilities

To evaluate the mathematical reasoning capabilities of models aligned with our method, we conduct an additional evaluation on GSM8K (Cobbe et al., 2021). We compare the aligned Qwen3-4B and Mistral-7B models over three self-play iterations against SFT, SPIN (Chen et al., 2024) and SPACE (Wang et al., 2026a). Results are shown in Table 5.

N Results with Larger Number of Iterations

The main result in this paper conducts the self-play algorithm in three iterations. In this section, we demonstrate the result over six iterations on the average scores on four baselines (Arc Challenge (Clark et al., 2018), MMLU (Hendrycks et al., 2020), HellaSwag (Zellers et al., 2019), and WinoGrande (Sakaguchi et al., 2019)) with Qwen3 4B model. Results show that the performance generally saturates as the number of iteration increases during the self-play, showing that the self-play methods cannot bootstrap infinitely and has an upper limit, aligns with the theoretical results of our adversarial imitation learning theoretical framework.

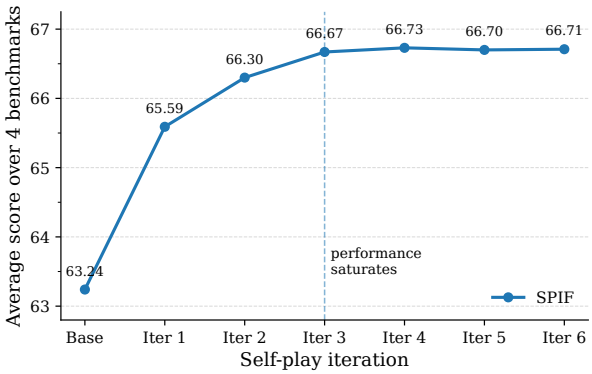

Figure 5: **Saturation of SPIF over self-play iterations.** We report the average score over four benchmarks for Qwen3-4B across six SPIF iterations. The performance improves substantially in the first few iterations and saturates after Iteration 3, consistent with the interpretation of SPIF as an imitation learning method.

O Limitations

Our theoretical analysis relies on realizability assumptions. While such assumptions are standard in theoretical analyses, they may not always hold in practical LLM finetuning scenarios, where the reward and policy classes can be misspecified. In addition, our convergence result establishes an average-iterate guarantee, whereas practical self-play finetuning evaluates the last-iterate model produced by training. Establishing last-iterate convergence would require stronger assumptions on the game structure or optimization dynamics, which we leave for future work.

P Generated Samples

Prompt	What is the best place to try traditional Indian sweets in Jaipur?
Ground Truth	One of the best places to try traditional Indian sweets in Jaipur is Laxmi Misthan Bhandar (LMB) in Johari Bazaar. They have been serving authentic Rajasthani sweets for over 300 years and are known for their decadent desserts like Ghewar, Rasgulla, and Mawa Kachori. Their ambiance and range of sweets make for a memorable experience.
Base Model	The best place to try traditional Indian sweets in Jaipur is Kishore Ganj. This bustling market area is famous for its wide variety of street food, including a wide range of traditional Indian sweets like Gajar ka Halwa, Jalebi, Kheer, Barfi, Ladoo, and Mithai. You can also find sweet shops (called Mithai Mandis) along the streets of Kishore Ganj, where local vendors offer a variety of sweet treats. For a more authentic experience, you can visit Kishore Ganj Bazaar, which is a popular spot for both locals and tourists to enjoy the local flavors and street food culture of Jaipur. Other nearby areas like Rajouri Garden and Anand Niketan also have good options for traditional Indian sweets, but Kishore Ganj is widely regarded as the best place to try them in Jaipur.
Iteration 1	The best place to try traditional Indian sweets in Jaipur is the famous Jaipur Sweet Shop, located on the corner of Bapu Road and Jai Singh Road. This shop is known for its wide variety of sweet dishes, including laddoos, jalebis, kheer, and much more. The shop has been in operation for over 100 years and is a favorite among locals and tourists alike.
Iteration 2	The best place to try traditional Indian sweets in Jaipur is the Chandni Chowk Bazaar.
Iteration 3	The best place to try traditional Indian sweets in Jaipur is Laxmi Mishtan Bhandar, located in the heart of the city.

Table 6: **Generated Samples.** We present generated samples from the Qwen-3 4B model across different self-play finetuning iterations for a single prompt with our proposed method.