NASH REJECTION SAMPLING: FROM MAX–MIN GAMES TO ON-POLICY OPTIMIZATION FOR NASH LEARNING FROM HUMAN FEEDBACK

Anonymous authors
Paper under double-blind review

ABSTRACT

Reinforcement learning from human feedback (RLHF) is successful for large language models alignment, but the approaches commonly assume that human preferences admit a transitive Bradley–Terry representation. This assumption may fail for non-transitive preferences. Nash learning from human feedback (NLHF) addresses such general preferences by seeking a Nash equilibrium of the two-player constant-sum game induced by the preference relation. Existing approaches typically either require costly alternating updates, or adopt identity preference optimization (IPO)-style objectives. We introduce Nash Rejection Sampling (Nash-RS), an on-policy optimization framework that reduces NLHF to a standard RLHF-style policy-gradient problem. At each iteration, Nash-RS uses current-policy responses and preference feedback to rejection-sample opponents from a reference policy, yielding a distribution that provide implicit scalar rewards for the same on-policy rollouts. Consequently, Nash-RS can directly reuse existing on-policy RLHF algorithms, including PPO, while avoiding a max–min optimizer and its associated nested game updates. Compared with IPO-style approaches, Nash-RS performs on-policy updates. Theoretically, we establish linear last-iterate convergence of the exact update to the regularized Nash equilibrium, while the practical implementation approximates this update using finite rejection samples and neural policy optimization. Experiments across multiple preference datasets and language-model families compare Nash-RS with method-native NLHF baselines, demonstrating competitive performance.

1 INTRODUCTION

Large language models (LLMs), such as GPT-6 Astra (OpenAI, 2026) and Claude Fable 5.1 (Anthropic, 2026), have exhibited remarkable capabilities across diverse domains, including code generation, data analysis, mathematics, and reasoning (Chowdhery et al., 2023; Touvron et al., 2023; Ji et al., 2025b). These models are increasingly integrated into decision-making processes previously considered unlikely to be automated in the near future (Bubeck et al., 2023; Eloundou et al., 2024).

Reinforcement learning from human feedback (RLHF) has become a standard approach for aligning large language models with human preferences (Ouyang et al., 2022). In the conventional pipeline, pairwise comparisons are first used to learn a scalar reward under the Bradley–Terry–Luce model (Bradley & Terry, 1952; Luce, 2012); the language model is then fine-tuned to maximize this reward, commonly with proximal policy optimization (PPO) (Schulman et al., 2017). This reduction from comparisons to scalar rewards is attractive because it turns alignment into a familiar reinforcement learning problem. It nevertheless imposes a strong structural assumption: all pairwise preferences must be explained by a single transitive ranking of responses. Aggregated human preferences can instead contain heterogeneous or cyclic comparisons, for which no scalar reward can represent all pairwise probabilities (Liu et al., 2026a). Standard RLHF may also concentrate on one mode of a diverse preference distribution, a phenomenon studied as preference collapse (Xiao et al., 2025).

Nash learning from human feedback (NLHF) avoids the scalar-reward assumption by retaining the pairwise preference model. Given a prompt x, two policies play a constant-sum game whose payoff is the probability that a response from the first policy is preferred to one from the second. The

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

desired policy is the Nash equilibrium of this game, regularized toward a reference policy. Munos et al. (2024) introduced this formulation together with Nash mirror descent, which enjoys last-iterate convergence in the tabular setting. Related methods optimize general preferences through self-play, online IPO objectives, or other game-theoretic updates (Azar et al., 2024; Calandriello et al., 2024; Swamy et al., 2024; Shi et al., 2025). These approaches make general preferences tractable, but solving or approximating the two-player problem can require specialized updates, mixture policies, or alternating optimization. At the same time, recent work emphasizes that preference fine-tuning benefits from feedback on responses generated by the current policy, rather than relying only on a fixed offline dataset (Tajwar et al., 2024; Tang et al., 2024).

This work asks whether the generality of NLHF can be retained while recovering the simplicity of an on-policy RLHF update. We answer this question through a reduction of the regularized max–min game. For a fixed policy, the inner player has a closed-form Gibbs best response obtained by exponentially tilting the reference policy according to its loss against the current policy. Substituting this response into the game yields a single concave maximization problem. Its objective has the same outer form as an RLHF objective: a policy reward minus a KL penalty. The policy reward is distribution-dependent, but its gradient admits an ordinary score-function representation. In particular, the response-level signal is the expected preference against the Gibbs opponent, which acts as an implicit scalar reward for samples from the current policy.

The remaining difficulty is to sample this Gibbs opponent without training or maintaining a second policy. We introduce *Nash Rejection Sampling* (Nash-RS), which proposes responses from the fixed reference policy and accepts them with a probability determined by their expected preference loss against current-policy responses. With the exact acceptance score, the accepted responses follow the Gibbs best response by the classical rejection-sampling identity (Forsythe, 1972). They can therefore be paired with the same on-policy responses to estimate the implicit reward. Nash-RS reduces each iteration to an RLHF-style policy update and can reuse standard on-policy optimizers, including PPO, without explicitly alternating between two learned players. Unlike IPO-style methods, its policy update is driven by responses sampled from the current policy.

Our theoretical analysis separates the exact policy-space algorithm from its neural approximation. For the exact Gibbs reward, an entropic mirror step on the reduced objective converges linearly in the last iterate to the unique regularized Nash equilibrium. The proof uses the relative strong convexity and relative smoothness of the reduced loss in KL geometry (Bauschke et al., 2017; Lu et al., 2018). Our approach is a single policy updated against exact samples of the Gibbs best response, with no second learned player, which converges linearly.

We compare Nash-RS with method-native implementations on two tasks with general preference oracles. The first is safety alignment on PKU-SafeRLHF, judged by a direct pairwise preference model. The second is on UltraFeedback, judged by a mixture of two reward models. In both settings Nash-RS is the only policy that is preferred to every other policy, including the initial model, in head-to-head comparison, and hence the only one with zero empirical exploitability. On a held-out PKU-SafeRLHF test set, its average win rate against the other policies is 0.519, compared with at most 0.503 for the baselines.

Contributions. Our main contributions are as follows:

- We reduce regularized NLHF to a single concave policy-reward maximization problem and derive its policy gradient as an expectation of an implicit scalar reward against a Gibbs best response.
- We introduce Nash-RS, an on-policy rejection-sampling procedure that samples this Gibbs opponent from the reference policy and turns the reduced objective into a standard RLHF-style optimization problem.
- We establish linear last-iterate convergence of the exact policy-space update to the unique regularized Nash equilibrium.
- We empirically compare Nash-RS with method-native NLHF baselines across preference datasets and language-model families.

1.1 Related Work

In addition to NLHF, recent research has explored the formulation of two-player constant-sum games for aligning human preferences (Swamy et al., 2024; Chen et al., 2024). These studies employ mirror descent (Beck & Teboulle, 2003; Beck, 2017) to learn the Nash equilibrium. Another line of work focuses on directly learning the Nash equilibrium from a preference dataset (Wu et al., 2025; Rosset et al., 2024; Calandriello et al., 2024; Zhang et al., 2025b). Furthermore, the work of (Wang et al., 2025; Liu et al., 2026b) analyzed the convergence of NLHF to the Nash equilibrium in the last iterate. The Nash solution of a general preference, the von Neumann winner, originates in dueling bandits (Dudík et al., 2015) and has been analyzed for RLHF by Wang et al. (2023). More recent algorithms accelerate or stabilize convergence to the regularized equilibrium with mirror-prox (Tiapkin et al., 2025), optimistic mirror-descent (Zhang et al., 2025a), and extragradient (Zhou et al., 2025) updates, study reference-policy regularization in self-play (Tang et al., 2025), and extend NLHF to multi-turn interactions (Shani et al., 2024). In game theory, last-iterate linear convergence has been established for entropy-regularized two-player zero-sum games (Perolat et al., 2021; Cen et al., 2021; Sokota et al., 2023; Tiapkin et al., 2025).

Nash-RS is an on-policy method. Offline preference optimization, such as direct preference optimization (DPO) (Rafailov et al., 2023), avoids training a reward model, but has been found less effective than on-policy reward-based RLHF, partly because on-policy sampling mitigates the distribution shift between the training data and the policy's own responses (Li et al., 2023; Xu et al., 2024; Tajwar et al., 2024). We discuss these and other related lines of work in Appendix F.

Rejection sampling scheme has been used heuristically in many RLHF techniques (Bai et al., 2022; Dong et al., 2023; Kim et al., 2023; Nakano et al., 2021; Tesauro & Janssens, 1988; Touvron et al., 2023). Specifically, these works use rejection sampling to select responses which have relatively high reward for RLHF fine-tuning. In contrast, Nash-RS uses rejection sampling to draw the Gibbs best-response opponent from the reference policy, with an acceptance rule derived from the NLHF objective, and uses it to construct an implicit reward for on-policy responses. Our work therefore provides a principled justification for rejection sampling in LLM alignment.

2 Policy Reward and Linear Convergence

Preliminaries. Let $x \sim \rho$ denote a prompt and let $\mathcal{Y}$ be the response space. A policy $\pi(\cdot \mid x)$ is defined as a distribution over $\mathcal{Y}$, and π_{ref} denotes a fixed reference policy with full support. Human preferences are represented by a pairwise preference model $\mathcal{P}(y \succ y' \mid x) \in [0,1]$, where $\mathcal{P}(y \succ y' \mid x) + \mathcal{P}(y' \succ y \mid x) = 1$.

We extend the notation $\mathcal{P}$ to policy arguments by averaging over the corresponding responses:

$$\mathcal{P}(\pi \succ \pi' \mid x) := \mathbb{E}_{y\sim\pi(\cdot|x),\, y'\sim\pi'(\cdot|x)}\big[\mathcal{P}(y \succ y' \mid x)\big],$$

When either policy is replaced by a single response, the corresponding expectation is omitted, and we define $\mathcal{P}(y \succ \pi' \mid x)$ and $\mathcal{P}(\pi \succ y' \mid x)$. We write $\mathcal{P}(\pi \succ \pi') := \mathbb{E}_{x\sim\rho}\left[\mathcal{P}(\pi \succ \pi' \mid x)\right]$ to account for the expectation over the prompt $x \sim \rho$, and use the same convention for response-policy comparisons. In arguments where x is fixed, we suppress the conditioning on x for simplicity.

NLHF seeks a policy that is robust against every competing policy. In practice, we also add an additional KL regularization term, yielding

$$\max_{\pi} \min_{\pi'} \mathcal{P}_\tau\left(\pi \succ \pi'\right), \tag{2.1}$$

where $\mathcal{P}_\tau$ is the regularized preference defined by

$$\mathcal{P}_\tau\left(\pi \succ \pi'\right) := \mathcal{P}\left(\pi \succ \pi'\right) - \tau\mathbb{E}_{x\sim\rho}\,\mathrm{KL}(\pi(\cdot \mid x)\|\pi_{\mathrm{ref}}(\cdot \mid x)) + \tau\mathbb{E}_{x\sim\rho}\,\mathrm{KL}(\pi'(\cdot \mid x)\|\pi_{\mathrm{ref}}(\cdot \mid x)),$$

and $\tau > 0$ is a parameter controlling the deviation from the base reference policy π_{ref}. Unlike standard RLHF, this formulation directly uses pairwise preferences and does not require them to admit a scalar-reward or Bradley–Terry representation.

2.1 Policy Reward

In this section, we show that the problem of NLHF can be reformulated as a maximization problem. For the NLHF max-min problem, the inner sub-problem admits a closed-form solution using

techniques similar to those discussed in (Rafailov et al., 2023). Thus, the max-min problem can be simplified to the following maximization problem.

Theorem 2.1 (Equivalent Maximization Problem of NLHF)**.** *The problem* (2.1) *is equivalent to the following problem:*

$$\max_{\pi} \mathbb{E}_{x\sim\rho}[\mathcal{Z}(x,\pi) - \mathrm{KL}(\pi(\cdot \mid x)\|\pi_{\mathrm{ref}}(\cdot \mid x))], \tag{2.2}$$

where

$$\mathcal{Z}(x,\pi) := -\log \mathbb{E}_{y'\sim\pi_{\mathrm{ref}}(\cdot|x)}\left[\exp\left(-\frac{1}{\tau}\cdot\mathcal{P}(\pi \succ y' \mid x)\right)\right]. \tag{2.3}$$

Here, (2.2) has a similar form to the standard RLHF loss function, with the expected reward $\mathbb{E}\left[r(x,y)\right]$ replaced by the term $\mathcal{Z}(x,\pi)$ in (2.3), which we refer to as the *policy reward*.

Remark 2.1. The primary distinction between $\mathcal{Z}(x,\pi)$ and the standard reward function in reinforcement learning is that $\mathcal{Z}(x,\pi)$ is defined based on the policy π rather than the response y. However, (2.3) does not take the form of an expected reward, and thus (2.2) cannot be reduced to a standard RL problem.

The reformulated problem (2.2) exhibits particularly favorable properties. Most notably, its objective forms a concave functional with respect to the policy π, which ensures strong theoretical guarantees for optimization and convergence behavior.

Theorem 2.2 (Concavity)**.** *The maximization problem* (2.2) *is concave w.r.t.* π.

An immediate consequence of this theorem is that the regularized NLHF problem has a unique Nash equilibrium $\pi_\star$, which reproduces Proposition 1 in Munos et al. (2024).

Corollary 2.2 (Proposition 1 in Munos et al. (2024))**.** *There exists a unique Nash equilibrium of the regularized preference model* (2.1)*.*

2.2 Linear Convergence

In this section, we analyze an update in policy space for the equivalent problem (2.2). For a fixed prompt x and any policy $\pi(\cdot \mid x)$, define the Gibbs opponent:

$$\pi^{\mathrm{G}}(y') = \pi_{\mathrm{ref}}(y')\exp\left(-\frac{\mathcal{P}(\pi \succ y')}{\tau} + \mathcal{Z}(\pi)\right),$$

where $\mathcal{Z}(\pi)$ is the normalization constant defined by

$$\mathcal{Z}(\pi) = -\log\sum_{y'}\pi_{\mathrm{ref}}(y')\exp\left(-\frac{\mathcal{P}(\pi \succ y')}{\tau}\right).$$

Section 2.1 hence shows that the regularized Nash policy $\pi_\star$ of NLHF is the unique maximizer of $F(\pi) := \mathcal{Z}(\pi) - \mathrm{KL}(\pi\|\pi_{\mathrm{ref}})$. Starting from a full-support policy $\pi_0 := \pi_{\mathrm{ref}}$, we then apply entropic mirror ascent for $F(\pi)$, with step-size $0 < \gamma_t < 1$,

$$\pi_{t+1} = \operatorname*{argmax}_{\pi\in\Delta(\mathcal{Y})}\left\{\gamma_t\left(\mathbb{E}_{y\sim\pi}\left[\frac{\mathcal{P}\left(y \succ \pi_t^{\mathrm{G}}\right)}{\tau}\right) - \mathrm{KL}(\pi\|\pi_{\mathrm{ref}})\right] - (1-\gamma_t)\,\mathrm{KL}(\pi\|\pi_t)\right\},$$

where $\Delta(\mathcal{Y})$ denote the probability simplex. This update admits the closed-form solution

$$\pi_{t+1}(y) = \frac{\pi_t(y)^{1-\gamma_t}\pi_{\mathrm{ref}}(y)^{\gamma_t}\exp\left(\tau^{-1}\gamma_t\mathcal{P}\left(y \succ \pi_t^{\mathrm{G}}\right)\right)}{\sum_{z\in\mathcal{Y}}\pi_t(z)^{1-\gamma_t}\pi_{\mathrm{ref}}(z)^{\gamma_t}\exp\left(\tau^{-1}\gamma_t\mathcal{P}\left(z \succ \pi_t^{\mathrm{G}}\right)\right)}. \tag{2.4}$$

Intuitively, the KL regularization w.r.t π_{ref} makes $F(\cdot)$ strongly concave, while the bounded preferences ensure that $F(\cdot)$ is smooth, both relative to negative entropy. Together, these properties imply the linear last-iterate convergence of this policy update (2.4). We formalize this intuition in Theorem 2.3, and provide the complete proof in Appendix B.3.

Theorem 2.3 (Linear last-iterate convergence of Nash-RS)**.** *Let* $L := 1 + \tau^{-2}$*. If* $0 < \gamma_t \le L^{-1}$*, then* (2.4) *satisfies*

$$\mathrm{KL}(\pi_\star\|\pi_{t+1}) + \gamma_t\big(F(\pi_\star) - F(\pi_{t+1})\big) \le (1-\gamma_t)\,\mathrm{KL}(\pi_\star\|\pi_t). \tag{2.5}$$

In particular, for $T \ge 1$*,* $\gamma_t = L^{-1}$ *and* $\kappa = 1 - L^{-1} = (1+\tau^2)^{-1}$*, we have*

$$\mathrm{KL}(\pi_\star\|\pi_T) \le \kappa^T\,\mathrm{KL}(\pi_\star\|\pi_0)\,,\ F(\pi_\star) - F(\pi_T) \le L\kappa^T\,\mathrm{KL}(\pi_\star\|\pi_0). \tag{2.6}$$

For prompts $x \sim \rho$, the same linear convergence rate holds with $F(\cdot)$ and $\mathrm{KL}(\cdot)$ replaced by their expectations over x, provided the update is applied separately at every prompt and the initial expected KL divergence is finite.

The theorem establishes linear last-iterate convergence without averaging. Since $\pi_0 = \pi_{\mathrm{ref}}$, the equilibrium identity $F(\pi_\star) = 1/(2\tau)$ and $\mathcal{Z}(\pi_\star) \le 1/\tau$ give $\mathrm{KL}(\pi_\star \| \pi_0) \le 1/(2\tau)$. For $0 < \varepsilon < 1/(2\tau)$, it therefore suffices to take $T \ge L \log(1/(2\tau\varepsilon))$ to ensure $\mathrm{KL}(\pi_\star \| \pi_T) \le \varepsilon$ with $\gamma_t = L^{-1}$. This iteration complexity is independent of the size of the response space and the smallest reference probability.

Theorem 2.3 has a game-theoretic interpretation. Let $\mathcal{E}_\tau(\pi) := \frac{1}{2} - \min_{\pi'} \mathcal{P}_\tau(\pi \succ \pi')$, then

$$\mathcal{E}_\tau(\pi_T) = \tau \left(F(\pi_\star) - F(\pi) \right) \le \tau L \kappa^T \, \mathrm{KL}(\pi_\star \| \pi_0) \le \frac{L\kappa^T}{2}. \tag{2.7}$$

Moreover, $F(\pi_\star) - F(\pi) = \mathrm{KL}(\pi \| \pi_\star) + \mathrm{KL}(\pi_\star \| \pi^{\mathrm{G}})$. Theorem 2.3 thus implies both $\pi_T \to \pi_\star$ and $\pi_T^{\mathrm{G}} \to \pi_\star$ as $T \to \infty$.

3 Nash Rejection Sampling

For deep learning architectures, with a specific focus on its applicability in the context of LLMs, we consider policies π_ϕ parameterized by ϕ, where ϕ denotes the weights of the LLM. In this case, the maximization problem (2.2) becomes

$$\max_{\phi} \mathbb{E}_{x \sim \rho}[\mathcal{Z}(x, \pi_\phi) - \mathrm{KL}(\pi_\phi(\cdot \mid x) \| \pi_{\mathrm{ref}}(\cdot \mid x))]. \tag{3.1}$$

Problem (3.1) is no longer convex as ϕ is parameterized by neural networks. Inspired by the similarity between problem (3.1) and the standard RLHF problem, we apply PPO to solve the problem. However, due to the non-linearity of the $\mathbb{E}_{\pi_\phi}(\cdot)$ term appearing in $\mathcal{Z}(x, \pi_\phi)$, directly using it as the advantage function in PPO can lead to a biased gradient estimate. Therefore, we need to examine the policy gradient of $\mathcal{Z}(x, \pi_\phi)$ carefully.

Theorem 3.1 (Policy Gradient of $\mathcal{Z}(x, \pi_\phi)$)**.** *The policy gradient of $\mathcal{Z}(x, \pi_\phi)$ is*

$$\mathbb{E}_{y \sim \pi_\phi(\cdot|x)} \Big[\nabla_\phi \log \pi_\phi(y \mid x) \cdot r'(x, y, \phi) \Big],$$

where $r'(x, y, \phi) := \frac{\mathcal{P}(y \succ \pi_\phi^{\mathrm{G}} | x)}{\tau}$ is referred to as to the implicit reward, and $\pi_\phi^{\mathrm{G}}(y' \mid x)$ is the Gibbs distribution defined by

$$\pi_\phi^{\mathrm{G}}(y' \mid x) := \exp\left(\log \pi_{\mathrm{ref}}(y' \mid x) - \frac{\mathcal{P}(\pi_\phi \succ y' \mid x)}{\tau} + \mathcal{Z}(x, \pi_\phi) \right). \tag{3.2}$$

Notably, the policy gradient of the reformulated NLHF problem replaces the reward in the policy gradient of standard RLHF, $\mathbb{E}_{y \sim \pi_\phi(\cdot|x)}[\nabla_\phi \log \pi_\phi(y \mid x) \cdot r(x, y)]$, with the *implicit reward* $r'(x, y, \phi)$. Consequently, once the implicit reward of NLHF is obtained, all existing RLHF algorithms can be directly applied to NLHF.

Solving the equivalent NLHF problem therefore reduces to estimating the implicit reward, which, in turn, depends on efficiently sampling from the Gibbs distribution (3.2). To address this, we introduce *Nash Rejection Sampling (Nash-RS)*, a novel and practical algorithm designed for solving the equivalent NLHF problem.

We apply the rejection sampling technique to sample from a probability distribution $f(\cdot)$. Suppose there exists a constant $c > 0$ and another probability distribution $g(\cdot)$, which is called the proposal distribution, satisfying $f \leqslant c \cdot g$, then the following rejection sampling procedure (Forsythe, 1972) produces unbiased samples from $f(\cdot)$: 1) Sample $X \sim g(\cdot)$. 2) Sample $U \sim \mathrm{Unif}[0, 1]$. 3) Accept X if $\frac{f(X)}{c \cdot g(X)} \geqslant U$. 4) Otherwise, go to step 1.

To sample from (3.2), in our setting, the target distribution is $f(\cdot) = \pi_\phi^{\mathrm{G}}(\cdot \mid x)$. We choose the proposal distribution to be $g(\cdot) = \pi_{\mathrm{ref}}(\cdot \mid x)$ and the constant to be $c = \exp\left(\mathcal{Z}(x, \pi_\phi)\right)$. The algorithm of rejection sampling from π_ϕ^{G} is provided in Algorithm 1.

Algorithm 1 : Rejection Sampling from π_ϕ^{G} in (3.2)

Require: prompt x, reference model π_{ref}, policy π_ϕ, batch size B_1, B_2
for $1 \leqslant i \leqslant B_1$ **do**
  Sample response $y_i \sim \pi_\phi(\cdot \mid x)$
end for
$j = 0$
while $j < B_2$ **do**
  Sample response $y' \sim \pi_{\mathrm{ref}}(\cdot \mid x)$
  Sample $u \sim \mathrm{Unif}[0, 1]$
  if $\frac{1}{B_1} \sum_{i=1}^{B_1} \mathcal{P}(y_i \succ y' \mid x) \leqslant -\tau \log u$ **then**
    $j \leftarrow j + 1$; $y'_j \leftarrow y'$
  end if
end while
return B_2 samples from π_ϕ^{G}: $\{y'_j | 1 \leqslant j \leqslant B_2\}$

Algorithm 1 obtain samples drawn from (3.2) by proposing $y' \sim \pi_{\mathrm{ref}}(\cdot \mid x)$ and accepting with probability $e^{-\mathcal{P}(\pi_\phi \succ y' | x)/\tau}$. Its acceptance probability is contained in $[e^{-1/\tau}, 1]$. Thus, increasing τ reduces the cost of rejection sampling but also changes the target equilibrium by strengthening regularization toward the reference policy.

After obtaining a batch of samples from π_ϕ^{G}, we can calculate the implicit reward $r'(x, y, \phi)$ via Algorithm 2.

Algorithm 2 : Computing Implicit Reward

Require: prompt x, reference model π_{ref}, policy π_ϕ, response y from policy π_ϕ, batch size B_1, B_2
Run Algorithm 1 with input prompt x, reference model π_{ref}, policy π_ϕ, batch size B_1 and B_2
$r'(x, y, \phi) \leftarrow \frac{1}{\tau} \frac{1}{B_2} \sum_{j=1}^{B_2} \mathcal{P}(y \succ y'_j \mid x)$
return $r'(x, y, \phi)$

4 Comparison with Preference Optimization Algorithms

We compare Nash-RS with existing methods along two perspectives: the policy update and the convergence guarantee. Table 1 summarizes the distinctions.

4.1 Comparison with Policy-Gradient Methods

Against an opponent q held fixed during differentiation, the KL-regularized policy gradient is

$$G_q(\phi) = \mathbb{E}_{y \sim \pi} \left[\nabla_\phi \log \pi(y) \left(\mathcal{P}(y \succ q) - \tau \log \frac{\pi(y)}{\pi_{\mathrm{ref}}(y)} \right) \right], \tag{4.1}$$

where $\mathcal{P}(y \succ q) = \mathbb{E}_{y' \sim q}[\mathcal{P}(y \succ y')]$. Nash-RS uses the Gibbs best response $\pi^{\mathrm{G}}(y') \propto \pi_{\mathrm{ref}}(y') \exp\{-\mathcal{P}(\pi \succ y')/\tau\}$. By the envelope theorem, $G_{\pi^{\mathrm{G}}}$ is the gradient of the reduced max–min objective $\tau F(\pi_\phi)$: differentiating through the opponent is unnecessary. Rejection sampling constructs this opponent and provides implicit scalar rewards for an RLHF-style update, including PPO (Schulman et al., 2017).

Nash-MD-PG instead uses a geometric mixture $q \propto \pi^{1-\beta} \pi_{\mathrm{ref}}^{\beta}$, while Nash-EMA uses an opponent whose parameters average past iterates (Munos et al., 2024). SPO uses self-play rewards against the current policy (Swamy et al., 2024). These choices define different update directions away from equilibrium. Nash-RS specifically follows the gradient of the objective obtained by analytically eliminating the opponent. Its advantage is therefore this unbiased reduction and compatibility with scalar-reward policy optimization; Nash-MD has a single-step tabular update and need not solve a nested optimization problem.

Table 1: Representative NLHF methods. *Opponent*: the policy whose responses the actor is compared against in the update. *Target*: *KL-reg.* is the Nash equilibrium of the preference game with a KL penalty $\tau\,\mathrm{KL}(\pi \,\|\, \pi_{\mathrm{ref}})$ of fixed strength τ, which depends on τ and π_{ref}; *Unreg.* is the Nash equilibrium (von Neumann winner) of the original, unregularized game. *Guarantee*: the result proved for the idealized (exact or tabular) update in each paper, not for its neural implementation. Shading groups methods whose policy update is a pairwise regression loss (blue) or reinforcement learning on a scalar reward (orange).

Method	Opponent	Training signal	Target	Guarantee
Online-IPO	Current policy	IPO loss	KL-reg.	Fixed-point characterization; no rate
IPO-MD	Geometric mixture of π and π_{ref}	IPO loss	KL-reg.[a]	Fixed-point characterization; no rate
EGPO	Extragradient half step	Online-IPO loss	KL-reg.[b]	Last-iterate linear
SPPO	Current policy	Square loss on win rate	Unreg.	Average-iterate $O(1/\sqrt{T})$ duality gap
DNO	Current policy	DPO-style loss	Unreg.	Per-iteration regression error[c]
INPO	Current policy	IPO-style loss	KL-reg.	Last-iterate $O(1/T)$ KL
ONPO	Optimistic prediction from previous iterate	IPO-style loss	Unreg.	Average-iterate $O(1/T)$ duality gap
COMAL	Current policy; π_{ref} reset to each regularized equilibrium	INPO loss (inner solver)	Unreg.	Last-iterate, asymptotic
SPO	Current policy (self-play)	RL on win-rate reward	Unreg.	Average-iterate $O(1/\sqrt{T})$
Nash-MD	Geometric mixture of π and π_{ref}	Policy gradient	KL-reg.	Last-iterate $O(1/T)$ KL
MPO	Magnet policy, periodically reset	REINFORCE (ReMax)	Unreg.	Last-iterate, asymptotic[d]
Nash-RS	**Gibbs best response, rejection-sampled from $\boldsymbol{\pi_{\mathrm{ref}}}$**	**PPO on implicit scalar reward**	**KL-reg.**	**Last-iterate linear KL (exact); $\boldsymbol{O(1/T)}$ expected KL (stochastic)**

[a]The mixture policy is the equilibrium of a KL-regularized game with strength $\tau/(1-\beta)$. [b]Also an ε-Nash equilibrium of the unregularized game after $\widetilde{O}(1/\varepsilon)$ exact iterations, with $\tau \propto \varepsilon$ and a uniform reference policy. [c]Convergence of the averaged policy is inherited from SPO. [d]Linear within each KL-regularized subproblem.

4.2 Convergence Guarantees and On-Policy Training

Online-IPO samples current-policy responses but fits a pairwise IPO-style loss, whose expected update matches the self-play policy gradient up to scaling (Calandriello et al., 2024); IPO-MD samples from a geometric mixture rather than the actor. In Nash-RS, the actor response in (4.1) is on-policy and the opponent is sampled separately from π_{ref}. Reusing a rollout batch after the actor has changed requires a surrogate or a correction, as in PPO.

Convergence also depends on the target game and the update being analyzed. Nash-MD and INPO have $O(1/T)$ last-iterate KL bounds for regularized games (Munos et al., 2024; Zhang et al., 2025b). SPO, SPPO, and idealized DNO provide averaged-policy guarantees (Swamy et al., 2024; Wu et al., 2025; Rosset et al., 2024). COMAL and MPO reach an unregularized equilibrium by updating the reference policy across regularized subproblems; MPO's linear rate concerns each fixed subproblem (Liu et al., 2026b; Wang et al., 2025). EGPO also gives linear convergence for regularized games, and an ε-Nash guarantee for the unregularized game with accuracy-dependent regularization (Zhou et al., 2025).

For Nash-RS, Theorem 2.3 establishes linear convergence of the exact policy-space mirror update to the fixed-τ regularized equilibrium. Proposition B.3 gives $O(1/T)$ expected KL error with conditionally unbiased gradient estimates, bounded second moments, and decreasing step sizes. Theorem 2.3 reaches the fixed-τ regularized Nash equilibrium in $O((1+\tau^{-2})\log(1/\varepsilon))$ iterations. EGPO, which is an IPO-style training loss, instead of a reinforcement learning objective, also converge linearly. Our analysis exploits a different structure. The KL penalty makes the opponent's best response available in closed form, so the game reduces to the concave objective F, whose

curvature relative to negative entropy is bounded by $L = 1 + \tau^{-2}$ (Lemma B.2); the τ^{-2} factor reflects how strongly the Gibbs response reacts to the current policy.

5 EXPERIMENTS

In this section, we compare Nash-RS with several NLHF baselines on two tasks with different preference oracles. On one task we also include standard RLHF and self-play as reference points.

5.1 SETUP

Baselines. We take one representative from each family in Table 1. Nash-MD (Munos et al., 2024) is the algorithm introduced in the original work of NLHF. EGPO (Zhou et al., 2025) represents the methods that regress an IPO-type loss. It improves on Online-IPO and IPO-MD with an extragradient update, so we do not run the IPO variants separately. MPO (Wang et al., 2025) represents the methods that approach the unregularized equilibrium through a magnet policy, such as COMAL. For the self-play family, we use self-play against the current policy, which is Nash-MD without mixing. Reward-PPO, which runs PPO on the scalar mixture of the two UltraFeedback reward models used by the oracle, represents standard RLHF. Each baseline keeps its native update rule and implementation: Nash-MD uses the token-level geometric-mixture trainer of TRL (von Werra et al., 2020), EGPO uses the authors' extragradient trainer, and MPO uses its periodic magnet and proximal updates. Nash-RS draws proposals from the reference policy, never forces acceptance, and trains on the implicit reward with PPO in OpenRLHF (Hu et al., 2024), using vLLM (Kwon et al., 2023) for generation. Every method is reported from a single training pass.

General preferences: Qwen2.5-3B on UltraFeedback. We train all parameters of Qwen2.5-3B-Instruct (Qwen Team, 2024) on the UltraFeedback dataset (Cui et al., 2024). The oracle is an equal-weight mixture of two independently trained reward models (Köpf et al., 2023; Liu et al., 2025). On a diagnostic set, this mixture is not representable by a single Bradley–Terry model.

Safety alignment: Gemma-2-2B on PKU-SafeRLHF. Gemma-2-2B (Gemma Team, 2024) is the main model in the EGPO and MPO papers. We therefore start from the SFT checkpoint and use the pairwise preference model released by the EGPO authors, so that these baselines run in the setting for which they were developed. The oracle is used as a general preference and is never converted to a scalar reward. Every method makes one pass over all PKU-SafeRLHF training prompts (Ji et al., 2025a; Dai et al., 2024) with rank-16 LoRA adapters, a global batch of 64 prompts, and a 512-token response limit. Nash-RS uses $K = 8$ proposals per batch, with an average acceptance rate of 0.54–0.57. Because of limited compute, we compare only the NLHF algorithms on this task and omit Reward-PPO and self-play. Further details are given in Appendix D.

Evaluation. For each task, we evaluate the initial policy and all trained policies jointly under the training oracle with greedy decoding. W_{ij} is the probability that policy i is preferred to policy j. We report the full matrix W, the average win rate $\frac{1}{n-1}\sum_{j\neq i} W_{ij}$ as mean $\pm$ the half-width of a 95% prompt-bootstrap interval, and the worst-case win rate $\min_{j\neq i} W_{ij}$. A policy whose worst-case win rate exceeds 0.5 is not beaten by any other policy in the population, so its empirical exploitability is zero. We use the 1,024 held-out UltraFeedback test prompts and all 7,491 held-out PKU-SafeRLHF test prompts.

5.2 RESULTS

Nash-RS wins every head-to-head comparison. In both tasks, the Nash-RS row of the pairwise matrix is above 0.5 in every column (Tables 2 and 3). Nash-RS is therefore the only policy whose worst-case win rate exceeds 0.5, and the only one with zero empirical exploitability.

On UltraFeedback (Table 2), Nash-RS beats every policy with probabilities 0.504–0.530. The closest competitor is Reward-PPO, which optimizes the scalar mixture directly and reaches an average of 0.518, so the oracle does reward genuine improvement. Even so, Reward-PPO remains exploitable by Nash-RS (0.504). Nash-MD, EGPO, MPO and self-play stay within 0.004 of the initialization's average win rate.

Table 2: Pairwise win probabilities W_{ij} (row i vs. column j) for Qwen2.5-3B on UltraFeedback. Avg.: average win rate ± half-width of the 95% prompt-bootstrap interval. Worst: worst-case win rate. Bold: best in column.

	Init.	**Nash-RS**	Nash-MD	EGPO	MPO	Rew.-PPO	Self-play	Avg.	Worst
Initial	—	0.473	0.497	0.497	0.503	0.477	0.503	0.492 ± 0.004	0.473
Nash-RS	**0.527**	—	**0.524**	**0.523**	**0.529**	**0.504**	**0.530**	$\mathbf{0.523} \pm 0.005$	**0.504**
Nash-MD (Ours)	0.503	0.476	—	0.499	0.506	0.480	0.506	0.495 ± 0.003	0.476
EGPO	0.503	0.477	0.501	—	0.506	0.480	0.507	0.496 ± 0.003	0.477
MPO	0.497	0.471	0.494	0.494	—	0.475	0.500	0.488 ± 0.004	0.471
Reward-PPO	0.523	0.496	0.520	0.520	0.525	—	0.526	0.518 ± 0.005	0.496
Self-play	0.497	0.470	0.494	0.493	0.500	0.474	—	0.488 ± 0.003	0.470

Table 3: Pairwise win probabilities W_{ij} (row i vs. column j) for Gemma-2-2B on PKU-SafeRLHF. Columns as in Table 2.

	Init.	**Nash-RS**	Nash-MD	EGPO	MPO	Avg.	Worst
Initial SFT	—	0.477	0.492	0.496	0.499	0.491 ± 0.003	0.477
Nash-RS (Ours)	**0.523**	—	**0.511**	**0.520**	**0.524**	$\mathbf{0.519} \pm 0.003$	**0.511**
Nash-MD	0.508	0.489	—	0.505	0.510	0.503 ± 0.003	0.489
EGPO	0.504	0.480	0.495	—	0.506	0.496 ± 0.003	0.480
MPO	0.501	0.476	0.490	0.494	—	0.490 ± 0.003	0.476

On PKU-SafeRLHF (Table 3), Nash-RS beats the three Nash baselines with probabilities 0.511–0.524 and the SFT initialization with 0.523. Its average win rate, 0.519 ± 0.003, is separated from every other policy's by more than the two half-widths combined. The strongest baseline is Nash-MD, with an average of 0.503; it improves on the initialization but loses to Nash-RS with probability 0.489. EGPO and MPO remain close to parity with the initialization (0.504 and 0.501).

Response lengths do not explain the gaps. On UltraFeedback, Nash-RS averages 355 tokens against 337–365 for the other policies, and truncation rates are 43–46% for every policy. On PKU-SafeRLHF, mean response lengths are 156–165 tokens for every policy, with no truncation.

Limitations. The bootstrap intervals capture evaluation uncertainty, not training variance. Both evaluations use the training oracle. On PKU-SafeRLHF this oracle was itself trained on PKU preferences, so our results measure optimization of a general preference, not independently judged safety. The PKU evaluations were run on A800 GPUs and the models were trained on H200s. Re-evaluating the same checkpoints on all 7,491 test prompts with H200 GPUs gives the same conclusion: Nash-RS reaches 0.518 ± 0.004 against at most 0.504 for the other policies, and again wins every pairwise comparison.

6 Conclusion

We showed that the KL-regularized NLHF game reduces to a single concave policy objective. Its gradient is an on-policy policy gradient with an implicit scalar reward, computed against a Gibbs best-response opponent. Nash-RS draws this opponent from the reference policy by rejection sampling, so regularized NLHF can be trained with standard RLHF optimizers such as PPO and without a second learned player. The exact policy-space update converges linearly in the last iterate to the regularized Nash equilibrium. With Qwen2.5-3B on UltraFeedback and Gemma-2-2B on PKU-SafeRLHF, Nash-RS was the only policy that beat every other policy in head-to-head comparison.

These results have limitations that suggest future work. The guarantee covers the exact update, which finite rejection samples and PPO only approximate. Rejection sampling replaces policy updates with preference queries, and this cost grows as τ decreases, which limits how closely Nash-RS can approach the unregularized equilibrium. Combining it with reference-policy updates, as in COMAL and MPO, is a natural extension. Finally, independent judges and scale-specific tuning are needed to establish how far the gains extend.

AI USE STATEMENT

We used generative AI tools as follows. **Writing.** AI tools were used to polish the text for readability, searched for related literature; the authors checked the content. **Theory.** The research question, the formulation, and the main results were developed by the authors. AI tools were used to check the correctness of the proofs and to polish their presentation. **Experiments.** AI tools were used to write and run the code; the authors checked the codes the results. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

All theoretical results state their assumptions in the main text, and complete proofs are given in the appendix. For the experiments, all the training details are provided in the appendix.

REFERENCES

Anthropic. Claude Fable 5.1 and Claude Mythos 5.1 System Card. Technical report, Anthropic, September 2026. URL https://www.anthropic.com/claude-fable-5-1-mythos-5-1-system-card.

Mohammad Gheshlaghi Azar, Zhaohan Daniel Guo, Bilal Piot, Remi Munos, Mark Rowland, Michal Valko, and Daniele Calandriello. A general theoretical paradigm to understand learning from human preferences. In *International Conference on Artificial Intelligence and Statistics*, pp. 4447–4455. PMLR, 2024.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, Nicholas Joseph, Saurav Kadavath, Jackson Kernion, Tom Conerly, Sheer El-Showk, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Tristan Hume, Scott Johnston, Shauna Kravec, Liane Lovitt, Neel Nanda, Catherine Olsson, Dario Amodei, Tom Brown, Jack Clark, Sam McCandlish, Chris Olah, Ben Mann, and Jared Kaplan. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022. URL https://arxiv.org/abs/2204.05862.

Heinz H. Bauschke, Jérôme Bolte, and Marc Teboulle. A descent lemma beyond Lipschitz gradient continuity: First-order methods revisited and applications. *Mathematics of Operations Research*, 42(2):330–348, 2017.

Amir Beck. *First-order methods in optimization*. SIAM, 2017.

Amir Beck and Marc Teboulle. Mirror descent and nonlinear projected subgradient methods for convex optimization. *Operations Research Letters*, 31(3):167–175, 2003.

Emily M Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, pp. 610–623, 2021.

Ralph Allan Bradley and Milton E. Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952.

Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, Harsha Nori, Hamid Palangi, Marco Tulio Ribeiro, and Yi Zhang. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*, 2023.

Daniele Calandriello, Zhaohan Daniel Guo, Remi Munos, Mark Rowland, Yunhao Tang, Bernardo Avila Pires, Pierre Harvey Richemond, Charline Le Lan, Michal Valko, Tianqi Liu, Rishabh Joshi, Zeyu Zheng, and Bilal Piot. Human alignment of large language models through online preference optimisation. In *Forty-first International Conference on Machine Learning*, 2024.

Shicong Cen, Yuting Wei, and Yuejie Chi. Fast policy extragradient methods for competitive games with entropy regularization. In *Advances in Neural Information Processing Systems*, 2021.

Jonathan D Chang, Wenhao Shan, Owen Oertell, Kianté Brantley, Dipendra Misra, Jason D Lee, and Wen Sun. Dataset reset policy optimization for rlhf. *arXiv preprint arXiv:2404.08495*, 2024.

Zixiang Chen, Yihe Deng, Huizhuo Yuan, Kaixuan Ji, and Quanquan Gu. Self-play fine-tuning converts weak language models to strong language models. In *International Conference on Machine Learning*, pp. 6621–6642. PMLR, 2024.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113, 2023.

Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Bingxiang He, Wei Zhu, Yuan Ni, Guotong Xie, Ruobing Xie, Yankai Lin, Zhiyuan Liu, and Maosong Sun. UltraFeedback: Boosting language models with scaled AI feedback. In *International Conference on Machine Learning*, 2024.

Josef Dai, Xuehai Pan, Ruiyang Sun, Jiaming Ji, Xinbo Xu, Mickel Liu, Yizhou Wang, and Yaodong Yang. Safe RLHF: Safe reinforcement learning from human feedback. In *International Conference on Learning Representations*, 2024.

Hanze Dong, Wei Xiong, Deepanshu Goyal, Yihan Zhang, Winnie Chow, Rui Pan, Shizhe Diao, Jipeng Zhang, SHUM KaShun, and Tong Zhang. Raft: Reward ranked finetuning for generative foundation model alignment. *Transactions on Machine Learning Research*, 2023.

Miroslav Dudík, Katja Hofmann, Robert E. Schapire, Aleksandrs Slivkins, and Masrour Zoghi. Contextual dueling bandits. In *Conference on Learning Theory*, pp. 563–587, 2015.

Tyna Eloundou, Sam Manning, Pamela Mishkin, and Daniel Rock. GPTs are GPTs: Labor market impact potential of LLMs. *Science*, 384(6702):1306–1308, 2024.

Shangbin Feng, Chan Young Park, Yuhan Liu, and Yulia Tsvetkov. From pretraining data to language models to downstream tasks: Tracking the trails of political biases leading to unfair nlp models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 11737–11762, 2023.

George E Forsythe. Von neumann's comparison method for random sampling from the normal and other distributions. *Mathematics of Computation*, 26(120):817–826, 1972.

Gemma Team. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*, 2024.

Alexey Gorbatovski, Boris Shaposhnikov, Alexey Malakhov, Nikita Surnachev, Yaroslav Aksenov, Ian Maksimov, Nikita Balagansky, and Daniil Gavrilov. Learn your reference model for real good alignment. *arXiv preprint arXiv:2404.09656*, 2024.

Jian Hu, Xibin Wu, Zilin Zhu, Xianyu, Weixun Wang, Dehao Zhang, and Yu Cao. Openrlhf: An easy-to-use, scalable and high-performance rlhf framework. *arXiv preprint arXiv:2405.11143*, 2024.

Jiaming Ji et al. PKU-SafeRLHF: Towards multi-level safety alignment for LLMs with human preference. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics*, 2025a.

Wenlong Ji, Weizhe Yuan, Emily Getzen, Kyunghyun Cho, Michael I Jordan, Song Mei, Jason E Weston, Weijie J Su, Jing Xu, and Linjun Zhang. An overview of large language models for statisticians. *arXiv preprint arXiv:2502.17814*, 2025b.

Sungdong Kim, Sanghwan Bae, Jamin Shin, Soyoung Kang, Donghyun Kwak, Kang Min Yoo, and Minjoon Seo. Aligning large language models through synthetic feedback. In *The 2023 Conference on Empirical Methods in Natural Language Processing*, 2023.

Andreas Köpf et al. OpenAssistant conversations – democratizing large language model alignment. In *Advances in Neural Information Processing Systems, Datasets and Benchmarks Track*, 2023.

Hadas Kotek, Rikker Dockum, and David Sun. Gender bias and stereotypes in large language models. In *Proceedings of the ACM collective intelligence conference*, pp. 12–24, 2023.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with PagedAttention. In *Proceedings of the 29th Symposium on Operating Systems Principles*, 2023.

Ziniu Li, Tian Xu, and Yang Yu. Policy optimization in rlhf: The impact of out-of-preference data. *arXiv preprint arXiv:2312.10584*, 2023.

Ziniu Li, Congliang Chen, Tian Xu, Zeyu Qin, Jiancong Xiao, Ruoyu Sun, and Zhi-Quan Luo. Entropic distribution matching for supervised fine-tuning of LLMs: Less overfitting and better diversity. In *NeurIPS 2024 Workshop on Fine-Tuning in Modern Machine Learning: Principles and Scalability*, 2024a. URL `https://openreview.net/forum?id=dulz3WVhMR`.

Ziniu Li, Tian Xu, Yushun Zhang, Zhihang Lin, Yang Yu, Ruoyu Sun, and Zhi-Quan Luo. Remax: a simple, effective, and efficient reinforcement learning method for aligning large language models. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 29128–29163, 2024b.

Chris Yuhao Liu et al. Skywork-Reward-V2: Scaling preference data curation via human-AI synergy. *arXiv preprint arXiv:2507.01352*, 2025.

Kaizhao Liu, Qi Long, Zhekun Shi, Weijie J. Su, and Jiancong Xiao. Statistical impossibility and possibility of aligning llms with human preferences: From condorcet paradox to nash equilibrium. *The Annals of Statistics*, 54(4):2110–2135, 2026a. ISSN 0090-5364, 2168-8966. doi: 10.1214/26-AOS2643.

Tianqi Liu, Yao Zhao, Rishabh Joshi, Misha Khalman, Mohammad Saleh, Peter J Liu, and Jialu Liu. Statistical rejection sampling improves preference optimization. In *The Twelfth International Conference on Learning Representations*, 2024.

Yixin Liu, Argyris Oikonomou, Weiqiang Zheng, Yang Cai, and Arman Cohan. COMAL: A convergent meta-algorithm for aligning LLMs with general preferences. In *International Conference on Learning Representations*, 2026b.

Haihao Lu, Robert M. Freund, and Yurii Nesterov. Relatively smooth convex optimization by first-order methods, and applications. *SIAM Journal on Optimization*, 28(1):333–354, 2018.

R Duncan Luce. *Individual choice behavior: A theoretical analysis*. Courier Corporation, 2012.

Queenie Luo, Michael J Puett, and Michael D Smith. A perspectival mirror of the elephant. *Communications of the ACM*, 67(8):98–105, 2024.

Remi Munos, Michal Valko, Daniele Calandriello, Mohammad Gheshlaghi Azar, Mark Rowland, Zhaohan Daniel Guo, Yunhao Tang, Matthieu Geist, Thomas Mesnard, Côme Fiegel, Andrea Michi, Marco Selvi, Sertan Girgin, Nikola Momchev, Olivier Bachem, Daniel J Mankowitz, Doina Precup, and Bilal Piot. Nash learning from human feedback. In *Forty-first International Conference on Machine Learning*, 2024.

Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, Xu Jiang, Karl Cobbe, Tyna Eloundou, Gretchen Krueger, Kevin Button, Matthew Knight, Benjamin Chess, and John Schulman. Webgpt: Browser-assisted question-answering with human feedback. *arXiv preprint arXiv:2112.09332*, 2021.

OpenAI. GPT-6 Astra System Card. Technical report, OpenAI, September 2026. URL `https://deploymentsafety.openai.com/gpt-6-astra`.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744, 2022.

Julien Perolat, Rémi Munos, Jean-Baptiste Lespiau, Shayegan Omidshafiei, Mark Rowland, Pedro Ortega, Neil Burch, Thomas Anthony, David Balduzzi, Bart De Vylder, et al. From Poincaré recurrence to convergence in imperfect information games: Finding equilibrium via regularization. In *International Conference on Machine Learning*, pp. 8525–8535, 2021.

Qwen Team. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

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

Rafael Rafailov, Joey Hejna, Ryan Park, and Chelsea Finn. From r to Q^*: Your language model is secretly a Q-function. In *First Conference on Language Modeling*, 2024. URL `https://openreview.net/forum?id=kEVcNxtqXk`.

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

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Lior Shani, Aviv Rosenberg, Asaf Cassel, Oran Lang, Daniele Calandriello, Avital Zipori, Hila Noga, Orgad Keller, Bilal Piot, Idan Szpektor, Avinatan Hassidim, Yossi Matias, and Rémi Munos. Multi-turn reinforcement learning with preference human feedback. In *Advances in Neural Information Processing Systems*, 2024.

Zhekun Shi, Kaizhao Liu, Qi Long, Weijie J Su, and Jiancong Xiao. Fundamental limits of game-theoretic llm alignment: Smith consistency and preference matching. *arXiv preprint arXiv:2505.20627*, 2025.

Samuel Sokota, Ryan D'Orazio, J. Zico Kolter, Nicolas Loizou, Marc Lanctot, Ioannis Mitliagkas, Noam Brown, and Christian Kroer. A unified approach to reinforcement learning, quantal response equilibria, and two-player zero-sum games. In *International Conference on Learning Representations*, 2023.

Gokul Swamy, Christoph Dann, Rahul Kidambi, Steven Wu, and Alekh Agarwal. A minimaximalist approach to reinforcement learning from human feedback. In *International Conference on Machine Learning*, pp. 47345–47377. PMLR, 2024.

Fahim Tajwar, Anikait Singh, Archit Sharma, Rafael Rafailov, Jeff Schneider, Tengyang Xie, Stefano Ermon, Chelsea Finn, and Aviral Kumar. Preference fine-tuning of llms should leverage suboptimal, on-policy data. In *International Conference on Machine Learning*, pp. 47441–47474. PMLR, 2024.

Xiaohang Tang, Sangwoong Yoon, Seongho Son, Huizhuo Yuan, Quanquan Gu, and Ilija Bogunovic. RSPO: Regularized self-play alignment of large language models. *arXiv preprint arXiv:2503.00030*, 2025.

Yunhao Tang, Daniel Zhaohan Guo, Zeyu Zheng, Daniele Calandriello, Yuan Cao, Eugene Tarassov, Rémi Munos, Bernardo Ávila Pires, Michal Valko, Yong Cheng, and Will Dabney. Understanding the performance gap between online and offline alignment algorithms. *arXiv preprint arXiv:2405.08448*, 2024.

Gerald Tesauro and Bob Janssens. Scaling relationships in back-propagation learning. *Complex Systems*, 2(1):39–44, 1988.

Daniil Tiapkin, Daniele Calandriello, Denis Belomestny, Eric Moulines, Alexey Naumov, Kashif Rasul, Michal Valko, and Pierre Ménard. Accelerating nash learning from human feedback via mirror prox. *arXiv preprint arXiv:2505.19731v1*, 2025.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

Leandro von Werra, Younes Belkada, Lewis Tunstall, Edward Beeching, Tristan Thrush, Nathan Lambert, Shengyi Huang, Kashif Rasul, and Quentin Gallouédec. TRL: Transformer reinforcement learning. `https://github.com/huggingface/trl`, 2020.

Martin J Wainwright. *High-dimensional statistics: A non-asymptotic viewpoint*, volume 48. Cambridge university press, 2019.

Mingzhi Wang, Chengdong Ma, Qizhi Chen, Linjian Meng, Yang Han, Jiancong Xiao, Zhaowei Zhang, Jing Huo, Weijie J Su, and Yaodong Yang. Magnetic preference optimization: Achieving last-iterate convergence for language model alignment. In *The Thirteenth International Conference on Learning Representations*, 2025. URL `https://openreview.net/forum?id=PDnEDS244P`.

Yuanhao Wang, Qinghua Liu, and Chi Jin. Is RLHF more difficult than standard RL? A theoretical perspective. In *Advances in Neural Information Processing Systems*, 2023.

Yue Wu, Zhiqing Sun, Rina Hughes, Kaixuan Ji, Yiming Yang, and Quanquan Gu. Self-play preference optimization for language model alignment. In *International Conference on Learning Representations*, 2025.

Jiancong Xiao, Ziniu Li, Xingyu Xie, Emily Getzen, Cong Fang, Qi Long, and Weijie J Su. On the algorithmic bias of aligning large language models with rlhf: Preference collapse and matching regularization. *Journal of the American Statistical Association*, 120(552):2154–2164, 2025.

Shusheng Xu, Wei Fu, Jiaxuan Gao, Wenjie Ye, Weilin Liu, Zhiyu Mei, Guangju Wang, Chao Yu, and Yi Wu. Is dpo superior to ppo for llm alignment? a comprehensive study. In *International Conference on Machine Learning*, pp. 54983–54998. PMLR, 2024.

Joy Qiping Yang, Salman Salamatian, Ziteng Sun, Ananda Theertha Suresh, and Ahmad Beirami. Asymptotics of language model alignment. In *2024 IEEE International Symposium on Information Theory (ISIT)*, pp. 2027–2032. IEEE, 2024.

Chenlu Ye, Wei Xiong, Yuheng Zhang, Hanze Dong, Nan Jiang, and Tong Zhang. Online iterative reinforcement learning from human feedback with general preference model. *Advances in Neural Information Processing Systems*, 37:81773–81807, 2024.

Yuheng Zhang, Dian Yu, Tao Ge, Linfeng Song, Zhichen Zeng, Haitao Mi, Nan Jiang, and Dong Yu. Improving LLM general preference alignment via optimistic online mirror descent. In *Advances in Neural Information Processing Systems*, 2025a.

Yuheng Zhang, Dian Yu, Baolin Peng, Linfeng Song, Ye Tian, Mingyue Huo, Nan Jiang, Haitao Mi, and Dong Yu. Iterative Nash policy optimization: Aligning LLMs with general preferences via no-regret learning. In *International Conference on Learning Representations*, 2025b.

Runlong Zhou, Maryam Fazel, and Simon S. Du. Extragradient preference optimization (EGPO): Beyond last-iterate convergence for Nash learning from human feedback. In *Conference on Language Modeling*, 2025.

A Additional Background

A.1 Preference Optimization Algorithms

ΨPO The work of (Azar et al., 2024) introduced a general paradigm called Ψ-preference optimization (ΨPO) for learning from human preference. Given a general non-decreasing function $\Psi : [0, 1] \to \mathbb{R}$, the goal is to maximize a nonlinear transformation of the preference relative to a fixed reference policy π_{ref}:

$$\max_{\pi} \mathbb{E}_{x\sim\rho}[\mathbb{E}_{y\sim\pi(\cdot|x)}\mathbb{E}_{y'\sim\pi_{\text{ref}}(\cdot|x)}\Psi\left(\mathcal{P}(y \succ y' \mid x)\right) - \tau\,\mathrm{KL}(\pi(\cdot \mid x)\|\pi_{\text{ref}}(\cdot \mid x))] \tag{A.1}$$

where $\tau > 0$ is a parameter controlling the deviation from the base reference policy π_{ref}. Above, the second term is the regularization term and the Kullback—Leibler (KL) divergence between $\pi(y \mid x)$ and $\pi_{\text{ref}}(y \mid x)$ is defined as

$$\mathrm{KL}(\pi(\cdot \mid x)\|\pi_{\text{ref}}(\cdot \mid x)) = \mathbb{E}_{y\sim\pi(\cdot|x)}\big[\log(\pi(y \mid x)) - \log(\pi_{\text{ref}}(y \mid x))\big].$$

When Ψ is the identity mapping, the corresponding objective is

$$\max_{\pi} \mathcal{P}(\pi \succ \pi_{\text{ref}}) - \tau\mathbb{E}_{x\sim\rho}\,\mathrm{KL}(\pi(\cdot \mid x)\|\pi_{\text{ref}}(\cdot \mid x)), \tag{A.2}$$

which is called identity preference optimization (IPO) (Azar et al., 2024).

In practice, we also add an additional KL regularization term, yielding

$$\max_{\pi}\min_{\pi'} \mathcal{P}_\tau\left(\pi \succ \pi'\right) \tag{A.3}$$

where $\mathcal{P}_\tau$ is the regularized preference defined by

$$\mathcal{P}_\tau\left(\pi \succ \pi'\right) := \mathcal{P}\left(\pi \succ \pi'\right) - \tau\mathbb{E}_{x\sim\rho}\,\mathrm{KL}(\pi(\cdot \mid x)\|\pi_{\text{ref}}(\cdot \mid x)) + \tau\mathbb{E}_{x\sim\rho}\,\mathrm{KL}(\pi'(\cdot \mid x)\|\pi_{\text{ref}}(\cdot \mid x)), \tag{A.4}$$

and $\tau > 0$ is a a parameter controlling the deviation from the base reference policy π_{ref}.

In the last subsection, we review several preference optimization algorithms.

Online-IPO Proposed in Calandriello et al. (2024), Online-IPO, also known as *self-play*, is defined by the following iteration:

$$\pi_{t+1}(\cdot \mid x) = \arg\max_{\pi}\big(\mathcal{P}(\pi \succ \pi_t \mid x) - \tau\mathrm{KL}(\pi(\cdot \mid x)\|\pi_{\text{ref}}(\cdot \mid x))\big). \tag{A.5}$$

The fixed point of Online-IPO iteration is the Nash equilibrium of (A.3), as shown in Proposition A.1.

Proposition A.1 (Proposition 4.1 in Calandriello et al. (2024))**.** *The solution of problem* (A.3) *satisfies the following fixed point equation:*

$$\pi(y \mid x) \propto \pi_{\text{ref}}(y \mid x)\exp\left(\frac{1}{\tau}\cdot\mathcal{P}(y \succ \pi \mid x)\right).$$

Nash-MD Nash-MD is a step of mirror descent defined in (Munos et al., 2024) relative to the regularized policy π_t^{ref}

$$\pi_{t+1}(\cdot \mid x) = \arg\max_{\pi}\left(\eta_t\mathcal{P}(\pi \succ \pi_t^{\text{ref}} \mid x) - \mathrm{KL}(\pi(\cdot \mid x)\|\pi_t^{\text{ref}}(\cdot \mid x))\right), \tag{A.6}$$

where η_t is the learning rate and the regularized policy π_t^{ref} is a geometric mixture between the current policy π_t and the reference policy π_{ref}:

$$\pi_t^{\text{ref}}(y \mid x) \propto \pi_t(y \mid x)^{1-\eta_t\tau}\pi_{\text{ref}}(y \mid x)^{\eta_t\tau}.$$

However, the fixed point of Nash-MD iteration (A.6) is not the original Nash equilibrium (Calandriello et al., 2024).

It is also worth mentioning that there are some offline preference optimization algorithms such as IPO (Azar et al., 2024) which seek to optimize an offline version of (A.2), and DPO (Rafailov et al., 2023). However, they are not the focus of this work.

A.2 Proof of Proposition A.1

Proof of Proposition A.1. Note that

$$
\begin{aligned}
&\max_{\pi}\min_{\pi'}\mathcal{P}_\tau\left(\pi\succ\pi'\right)\\
&=\max_{\pi}\min_{\pi'}\mathbb{E}_{x\sim\rho,y\sim\pi(\cdot|x),y'\sim\pi'(\cdot|x)}\left[\mathcal{P}\left(y\succ y'\mid x\right)-\tau\log\frac{\pi(y\mid x)}{\pi_{\mathrm{ref}}(y\mid x)}+\tau\log\frac{\pi'(y'\mid x)}{\pi_{\mathrm{ref}}(y'\mid x)}\right]\\
&=1-\min_{\pi}\max_{\pi'}\mathbb{E}_{x\sim\rho,y\sim\pi(\cdot|x),y'\sim\pi'(\cdot|x)}\left[\mathcal{P}\left(y'\succ y\mid x\right)+\tau\log\frac{\pi(y\mid x)}{\pi_{\mathrm{ref}}(y\mid x)}-\tau\log\frac{\pi'(y'\mid x)}{\pi_{\mathrm{ref}}(y'\mid x)}\right]\\
&=1-\min_{\pi'}\max_{\pi}\mathbb{E}_{x\sim\rho,y\sim\pi(\cdot|x),y'\sim\pi'(\cdot|x)}\left[\mathcal{P}\left(y\succ y'\mid x\right)+\tau\log\frac{\pi'(y'\mid x)}{\pi_{\mathrm{ref}}(y'\mid x)}-\tau\log\frac{\pi(y\mid x)}{\pi_{\mathrm{ref}}(y\mid x)}\right]\\
&=1-\min_{\pi'}\max_{\pi}\mathcal{P}_\tau(\pi\succ\pi')=1-\max_{\pi}\min_{\pi'}\mathcal{P}_\tau\left(\pi\succ\pi'\right).
\end{aligned}
$$

Thus, $\max_{\pi}\min_{\pi'}\mathcal{P}_\tau\left(\pi\succ\pi'\right)=\frac{1}{2}$. Recall that in Theorem 2.1, $\arg\min_{\pi'}\mathcal{P}_\tau(\pi\succ\pi')=\pi^{\mathrm{G}}(\cdot\mid x)$, where

$$
\pi^{\mathrm{G}}(y\mid x)\propto\pi_{\mathrm{ref}}(y\mid x)\exp\left(-\frac{\mathcal{P}(\pi\succ y\mid x)}{\tau}\right)\propto\pi_{\mathrm{ref}}(y\mid x)\exp\left(\frac{\mathcal{P}(y\succ\pi\mid x)}{\tau}\right). \tag{A.7}
$$

Therefore, problem (A.3) can be rewritten as

$$
\arg\max_{\pi}\left\{\mathcal{P}\left(\pi\succ\pi^{\mathrm{G}}\right)-\tau\cdot\mathbb{E}_{x\sim\rho}\,\mathrm{KL}\left(\pi(\cdot\mid x)\|\pi_{\mathrm{ref}}(\cdot\mid x)\right)+\tau\cdot\mathbb{E}_{x\sim\rho}\,\mathrm{KL}\left(\pi^{\mathrm{G}}(\cdot\mid x)\|\pi_{\mathrm{ref}}(\cdot\mid x)\right)\right\}. \tag{A.8}
$$

Taking $\pi=\pi^{\mathrm{G}}$ in (A.8), we obtain the optimal value $\frac{1}{2}$, since

$$
\mathcal{P}(\pi\succ\pi)=1-\mathcal{P}(\pi\succ\pi)\Longrightarrow\mathcal{P}(\pi\succ\pi)=\frac{1}{2}.
$$

Therefore, due to the uniqueness of the solution, the solution is given by (A.7). □

B Proofs and Additional Results for Section 2

B.1 Proof of Theorem 2.1

Proof of Theorem 2.1. The proof relies on solving the inner minimization problem in problem (2.1) explicitly and plugging the solution into the original problem (2.1). Consider the inner minimization problem in problem (2.1) first:

$$
\min_{\pi'}\mathcal{P}_\tau\left(\pi\succ\pi'\right). \tag{B.1}
$$

Notice that

$$
\begin{aligned}
\mathcal{P}_\tau\left(\pi\succ\pi'\right)&=\mathbb{E}_{x\sim\rho,\ y\sim\pi(\cdot|x),\ y'\sim\pi'(\cdot|x)}\left[\mathcal{P}(y\succ y'|x)-\tau\log\frac{\pi(y|x)}{\pi_{\mathrm{ref}}(y|x)}+\tau\log\frac{\pi'(y'|x)}{\pi_{\mathrm{ref}}(y'|x)}\right]\\
&=\mathcal{P}(\pi\succ\pi')-\mathbb{E}_{x\sim\rho,\ y\sim\pi(\cdot|x)}\left[\tau\log\frac{\pi(y|x)}{\pi_{\mathrm{ref}}(y|x)}\right]+\mathbb{E}_{x\sim\rho,\ y'\sim\pi'(\cdot|x)}\left[\tau\log\frac{\pi'(y'|x)}{\pi_{\mathrm{ref}}(y'|x)}\right].
\end{aligned}
$$

The second term does not depend on π'. Thus we can drop the second term in the minimization problem (B.1) and consider the following equivalent minimization problem:

$$
\min_{\pi'}\left\{\mathcal{P}(\pi\succ\pi')+\mathbb{E}_{x\sim\rho,\ y'\sim\pi'(\cdot|x)}\left[\tau\log\frac{\pi'(y'|x)}{\pi_{\mathrm{ref}}(y'|x)}\right]\right\}. \tag{B.2}
$$

For this problem, we have

$$
\begin{aligned}
&\mathcal{P}(\pi \succ \pi') + \mathbb{E}_{x\sim\rho,\, y'\sim\pi'(\cdot|x)}\left[\tau \log \frac{\pi'(y'|x)}{\pi_{\mathrm{ref}}(y'|x)}\right] \\
&= \mathbb{E}_{x\sim\rho,\, y'\sim\pi'(\cdot|x)}\Big[\mathcal{P}(\pi \succ y' \mid x)\Big] + \mathbb{E}_{x\sim\rho,\, y'\sim\pi'(\cdot|x)}\left[\tau \log \frac{\pi'(y'|x)}{\pi_{\mathrm{ref}}(y'|x)}\right] \\
&= \mathbb{E}_{x\sim\rho,\, y'\sim\pi'(\cdot|x)}\left[\mathcal{P}(\pi \succ y' \mid x) + \tau \log \frac{\pi'(y'|x)}{\pi_{\mathrm{ref}}(y'|x)}\right] \\
&= \tau\left\{\mathbb{E}_{x\sim\rho,\, y'\sim\pi'(\cdot|x)}\left[\log \pi'(y'|x) - \left(\log \pi_{\mathrm{ref}}(y'|x) - \frac{1}{\tau}\cdot\mathcal{P}(\pi \succ y' \mid x)\right)\right]\right\} \\
&= \tau\left\{\mathbb{E}_{x\sim\rho}\Big[\mathrm{KL}\left(\pi'(\cdot|x)\|\pi^{\mathrm{G}}(\cdot|x)\right) + \mathcal{Z}(x,\pi)\Big]\right\}
\end{aligned}
$$

where

$$
\pi^{\mathrm{G}}(y'|x) := \exp\left(\log \pi_{\mathrm{ref}}(y'|x) - \frac{1}{\tau}\cdot\mathcal{P}(\pi \succ y' \mid x) + \mathcal{Z}(x,\pi)\right) \tag{B.3}
$$

is the Gibbs distribution and $\mathcal{Z}(x,\pi)$ is defined in (2.3), which does not depend on π'. Therefore, the minimum of problem (B.2) is attained at π^{G}, and the minimum value is $\tau\mathbb{E}_{x\sim\rho}\mathcal{Z}(x,\pi)$. Accordingly, the minimum value of problem (B.1) is given by

$$
\min_{\pi'} \mathcal{P}_\tau\left(\pi \succ \pi'\right) = \tau\mathbb{E}_{x\sim\rho}\mathcal{Z}(x,\pi) - \mathbb{E}_{x\sim\rho,\, y\sim\pi(\cdot|x)}\left[\tau \log \frac{\pi(y|x)}{\pi_{\mathrm{ref}}(y|x)}.\right]
$$

Plugging the above minimum value into problem (2.1), we yield that problem (2.1) is equivalent to

$$
\max_{\pi} \tau\mathbb{E}_{x\sim\rho}\mathcal{Z}(x,\pi) - \tau\mathbb{E}_{x\sim\rho}\,\mathrm{KL}(\pi(\cdot|x)\|\pi_{\mathrm{ref}}(\cdot|x)),
$$

which is equivalent to problem (2.2) as $\tau > 0$. □

B.2 Proof of Theorem 2.2

Proof of Theorem 2.2. The convexity of KL divergence is straightforward (Wainwright, 2019). Now we consider the concavity of $\mathcal{Z}(x,\pi)$. Recall that $\mathcal{Z}(x,\pi)$ is defined in (2.3) as

$$
\mathcal{Z}(x,\pi) = -\log \mathbb{E}_{y'\sim\pi_{\mathrm{ref}}(\cdot|x)}\left[\exp\left(-\frac{1}{\tau}\cdot\mathcal{P}(\pi \succ y' \mid x)\right)\right].
$$

In this proof, we abbreviate $\mathbb{E}_{y'\sim\pi_{\mathrm{ref}}}$ by $\mathbb{E}$ for simplicity. Note that for $0 \leqslant t \leqslant 1$,

$$
\begin{aligned}
&\mathcal{Z}(x, t\pi_1 + (1-t)\pi_2) = -\log \mathbb{E}\left[\exp\left(-\frac{1}{\tau}\cdot\mathcal{P}(t\pi_1 + (1-t)\pi_2 \succ y' \mid x)\right)\right] \\
&= -\log \mathbb{E}\left[\exp\left(-\frac{t}{\tau}\cdot\mathcal{P}(\pi_1 \succ y' \mid x) - \frac{1-t}{\tau}\cdot\mathcal{P}(\pi_2 \succ y' \mid x)\right)\right] \\
&= -\log \mathbb{E}\left[\left(\exp\left(-\frac{1}{\tau}\cdot\mathcal{P}(\pi_1 \succ y' \mid x)\right)\right)^t \cdot \left(\exp\left(-\frac{1}{\tau}\cdot\mathcal{P}(\pi_2 \succ y' \mid x)\right)\right)^{1-t}\right] \\
&\geq -\log\left(\left(\mathbb{E}\left[\exp\left(-\frac{1}{\tau}\cdot\mathcal{P}(\pi_1 \succ y' \mid x)\right)\right]\right)^t \cdot \left(\mathbb{E}\left[\exp\left(-\frac{1}{\tau}\cdot\mathcal{P}(\pi_2 \succ y' \mid x)\right)\right]\right)^{1-t}\right) \\
&= -t\cdot\log \mathbb{E}\left[\exp\left(-\frac{1}{\tau}\cdot\mathcal{P}(\pi_1 \succ y' \mid x)\right)\right] - (1-t)\cdot\log \mathbb{E}\left[\exp\left(-\frac{1}{\tau}\cdot\mathcal{P}(\pi_2 \succ y' \mid x)\right)\right] \\
&= t\mathcal{Z}(x,\pi_1) + (1-t)\mathcal{Z}(x,\pi_2),
\end{aligned}
$$

where we use the Hölder inequality, $\mathbb{E}|X|^t|Y|^{1-t} \leqslant \left(\mathbb{E}|X|\right)^t\left(\mathbb{E}|Y|\right)^{1-t}$. By definition, this shows that $\mathcal{Z}(x,\pi)$ is concave with respect to π. □

B.3 Convergence analysis

We give the proof of Theorem 2.3 in this subsection. Throughout this subsection, we fix a prompt x and suppress the conditioning on x for simplicity. We write $\mathcal{Y} = \{1, \dots, m\}$ and $\pi_{\mathrm{ref}}(i) > 0$ for any $i \in [m]$. Thus a policy is identified with a vector $\pi \in \Delta_m$. We also write $f(\pi) := -F(\pi) = \mathrm{KL}(\pi \| \pi_{\mathrm{ref}}) - \mathcal{Z}(\pi)$.

The following lemma provides some basic properties of the unique regularized Nash equilibrium $\pi_\star$.

Lemma B.1. *The regularized game $\mathcal{P}_\tau(\pi \succ \pi')$ has a unique Nash equilibrium $(\pi_\star, \pi_\star)$, with full support and value $1/2$. Moreover,*

$$\min_{\pi'} \mathcal{P}_\tau(\pi \succ \pi') = -\tau f(\pi), \tag{B.4}$$

where $f(\pi_\star) = -\frac{1}{2\tau}$ and $\pi_\star^{\mathrm{G}} = \pi_\star$.

Proof of Lemma B.1. Expanding the definition of π^{G} gives, for every $\pi' \in \Delta_m$,

$$\mathcal{P}(\pi \succ \pi') + \tau\,\mathrm{KL}(\pi' \| \pi_{\mathrm{ref}}) = \tau\,\mathrm{KL}(\pi' \| \pi^{\mathrm{G}}) + \tau \mathcal{Z}(\pi).$$

This proves (B.4). By Corollary 2.2, the regularized game has a unique saddle point. Then (B.4) shows that the unique minimizing response to $\pi_\star$ is $\pi_\star^{\mathrm{G}}$, so this saddle point is $(\pi_\star, \pi_\star^{\mathrm{G}})$.

Notice that $\mathcal{P}(i \succ j) + \mathcal{P}(j \succ i) = 1$ implies $\mathcal{P}_\tau(\pi \succ \pi') = 1 - \mathcal{P}_\tau(\pi' \succ \pi)$. Consequently, if $(\pi_\star, \pi_\star^{\mathrm{G}})$ is a saddle point, then $(\pi_\star^{\mathrm{G}}, \pi_\star)$ is also a saddle point. Uniqueness gives $\pi_\star^{\mathrm{G}} = \pi_\star$. The Gibbs formula then shows that this policy has full support as the reference policy π_{ref} has full support. Finally, $\mathcal{P}_\tau(\pi_\star \succ \pi_\star) = 1/2$, proving all claims. □

Let $D_f(\widetilde{\pi}, \bar{\pi}) = f(\widetilde{\pi}) - f(\bar{\pi}) - \langle \nabla f(\bar{\pi}), \widetilde{\pi} - \bar{\pi} \rangle$ be the Bregman divergence generated by f, and let $H(\pi) = \sum_i \pi_i \log \pi_i$ be the entropy. The Bregman divergence generated by H on the simplex is $D_H(\widetilde{\pi}, \bar{\pi}) = \mathrm{KL}(\widetilde{\pi} \| \bar{\pi})$.

Lemma B.2. *For $\widetilde{\pi} \in \Delta_m$ and interior policy $\bar{\pi} \in \Delta_m$, let $L = 1 + \tau^{-2}$, then*

$$\mathrm{KL}(\widetilde{\pi} \| \bar{\pi}) \le D_f(\widetilde{\pi}, \bar{\pi}) \le L\,\mathrm{KL}(\widetilde{\pi} \| \bar{\pi}), \tag{B.5}$$

Moreover, $D_f(\widetilde{\pi}, \bar{\pi}) = \mathrm{KL}(\widetilde{\pi} \| \bar{\pi}) + \mathrm{KL}\big(\bar{\pi}^{\mathrm{G}} \| \widetilde{\pi}^{\mathrm{G}}\big)$.

Proof of Lemma B.2. Differentiating $f(\pi)$ gives, for any $i \in [m]$ and any $v \in \mathbb{R}^m$,

$$\begin{aligned} [\nabla f(\pi)]_i &= \log \frac{\pi_i}{\pi_{\mathrm{ref}}(i)} + 1 - \frac{\mathcal{P}\big(i \succ \pi^{\mathrm{G}}\big)}{\tau}, \\ v^\top \nabla^2 f(\pi) v &= \sum_i \frac{v_i^2}{\pi_i} + \frac{1}{\tau^2} \operatorname{Var}_{i' \sim \pi^{\mathrm{G}}} \left(\sum_i v_i \mathcal{P}(i \succ i') \right). \end{aligned} \tag{B.6}$$

The variance term is nonnegative, and for any $v \in \mathbb{R}^m$,

$$\operatorname{Var}_{i' \sim \pi^{\mathrm{G}}} \left(\sum_i v_i \mathcal{P}(i \succ i') \right) \le \mathbb{E}_{i' \sim \pi^{\mathrm{G}}} \left(\sum_i v_i \mathcal{P}(i \succ i') \right)^2 \le \sum_i \frac{v_i^2}{\pi_i}.$$

The last inequality follows from Cauchy–Schwarz and $\sum_i \pi_i \mathcal{P}(i \succ i')^2 \le 1$. Therefore $\nabla^2 H(\pi) \preceq \nabla^2 f(\pi) \preceq L \nabla^2 H(\pi)$. Integrating these inequalities along the segment between $\bar{\pi}$ and $\widetilde{\pi}$ proves (B.5).

For the identity, the definition of π^{G} gives

$$\mathrm{KL}\big(\bar{\pi}^{\mathrm{G}} \| \widetilde{\pi}^{\mathrm{G}}\big) = \mathcal{Z}(\bar{\pi}) - \mathcal{Z}(\widetilde{\pi}) + \tau^{-1} \sum_i (\widetilde{\pi}_i - \bar{\pi}_i) \mathcal{P}\big(i \succ \bar{\pi}^{\mathrm{G}}\big).$$

This is the Bregman divergence of $-\mathcal{Z}$. Adding the KL contribution proves the claimed identity. The same identities extend to boundary $\widetilde{\pi}$ by continuity, with $\bar{\pi}$ kept in the interior. □

Lemma B.2 and $\pi_\star^{\mathrm{G}} = \pi_\star$ give the exact identity

$$f(\pi) - f(\pi_\star) = \mathrm{KL}(\pi\|\pi_\star) + \mathrm{KL}(\pi_\star\|\pi^{\mathrm{G}}). \tag{B.7}$$

By Lemma B.1, we have $\mathcal{E}_\tau(\pi) = \frac{1}{2} - \min_{\pi'} \mathcal{P}_\tau(\pi \succ \pi') = \tau\big(f(\pi) - f(\pi_\star)\big)$. We next give the proof of Theorem 2.3.

Proof of Theorem 2.3. For a full-support iterate π_t, the policy update is given by

$$\pi_{t+1} = \arg\min_{\widetilde{\pi}\in\Delta_m} \left\{ \langle \nabla f(\pi_t), \widetilde{\pi} - \pi_t\rangle + \gamma_t^{-1}\,\mathrm{KL}(\widetilde{\pi}\|\pi_t) \right\}. \tag{B.8}$$

Its first-order condition gives $\pi_{t+1,i} \propto \pi_{t,i}\exp(-\gamma_t \nabla f(\pi_t)_i)$. Substitution of (B.6) thus yields (2.4), so every iterate has full support. For any $\widetilde{\pi} \in \Delta_m$, the three-point inequality is

$$\gamma_t\langle \nabla f(\pi_t), \pi_{t+1} - \widetilde{\pi}\rangle \le \mathrm{KL}(\widetilde{\pi}\|\pi_t) - \mathrm{KL}(\widetilde{\pi}\|\pi_{t+1}) - \mathrm{KL}(\pi_{t+1}\|\pi_t). \tag{B.9}$$

This follows by taking the inner product of the first-order condition with $\pi_{t+1} - \widetilde{\pi}$ and expanding the three KL terms. Lemma B.2 gives

$$f(\pi_{t+1}) \le f(\pi_t) + \langle \nabla f(\pi_t), \pi_{t+1} - \pi_t\rangle + L\,\mathrm{KL}(\pi_{t+1}\|\pi_t),$$
$$f(\pi_\star) \ge f(\pi_t) + \langle \nabla f(\pi_t), \pi_\star - \pi_t\rangle + \mathrm{KL}(\pi_\star\|\pi_t).$$

Subtracting and applying (B.9) with $\widetilde{\pi} = \pi_\star$ yields

$$\gamma_t\big(f(\pi_{t+1}) - f(\pi_\star)\big) \le (1-\gamma_t)\,\mathrm{KL}(\pi_\star\|\pi_t) - \mathrm{KL}(\pi_\star\|\pi_{t+1}) - (1-\gamma_t L)\,\mathrm{KL}(\pi_{t+1}\|\pi_t). \tag{B.10}$$

Since $\gamma_t L \le 1$, dropping the final non-positive term gives (2.5). Since $f(\pi_{t+1}) - f(\pi_\star) \ge 0$, then $\mathrm{KL}(\pi_\star\|\pi_{t+1}) \le (1-\gamma_t)\,\mathrm{KL}(\pi_\star\|\pi_t)$. For $\gamma_t = L^{-1}$, induction gives $\mathrm{KL}(\pi_\star\|\pi_T) \le \kappa^T\,\mathrm{KL}(\pi_\star\|\pi_0)$. For $T \ge 1$, the same inequality gives

$$\begin{aligned} f(\pi_T) - f(\pi_\star) &\le (L-1)\,\mathrm{KL}(\pi_\star\|\pi_{T-1}) - L\,\mathrm{KL}(\pi_\star\|\pi_T) \\ &\le (L-1)\kappa^{T-1}\,\mathrm{KL}(\pi_\star\|\pi_0) = L\kappa^T\,\mathrm{KL}(\pi_\star\|\pi_0). \end{aligned}$$

These are precisely (2.6). □

The objective also decreases monotonically in the minimization convention. Comparing (B.8) with $\widetilde{\pi} = \pi_t$ and using relative smoothness gives

$$f(\pi_{t+1}) \le f(\pi_t) - (\gamma_t^{-1} - L)\,\mathrm{KL}(\pi_{t+1}\|\pi_t) \le f(\pi_t).$$

The following result shows how gradient errors affect the argument, without specifying batch sizes or a particular estimator.

Proposition B.3. *Replace $\nabla f(\pi_t)$ in* (B.8) *by $\nabla f(\pi_t) + e_t$. Let $\mathcal{F}_t$ contain the history before drawing the iteration-t gradient estimate, and assume $\|\mathbb{E}[e_t \mid \mathcal{F}_t]\|_\infty \le b_t$ and $\mathbb{E}[\|e_t\|_\infty^2 \mid \mathcal{F}_t] \le V_t$. For $0 < \gamma_t \le (2L)^{-1}$,*

$$\mathbb{E}[\mathrm{KL}(\pi_\star\|\pi_{t+1}) \mid \mathcal{F}_t] \le (1-\gamma_t)\,\mathrm{KL}(\pi_\star\|\pi_t) + 2\gamma_t b_t + \gamma_t^2 V_t. \tag{B.11}$$

If $b_t = 0$, $V_t \le V$, and $\gamma_t = 2/(t+c)$ with $c \ge 4L$, then

$$\mathbb{E}\,\mathrm{KL}(\pi_\star\|\pi_T) \le \frac{\max\{(c-1)\,\mathrm{KL}(\pi_\star\|\pi_0), 4V\}}{T+c-1}. \tag{B.12}$$

Proof of Proposition B.3. Repeating (B.10) with the gradient $\nabla f(\pi_t)$ replaced by $\nabla f(\pi_t) + e_t$ and dropping the nonnegative objective gap gives

$$\mathrm{KL}(\pi_\star\|\pi_{t+1}) \le (1-\gamma_t)\,\mathrm{KL}(\pi_\star\|\pi_t) - (1-\gamma_t L)\,\mathrm{KL}(\pi_{t+1}\|\pi_t) - \gamma_t\langle e_t, \pi_{t+1} - \pi_\star\rangle.$$

Split $\pi_{t+1} - \pi_\star = (\pi_t - \pi_\star) + (\pi_{t+1} - \pi_t)$. The conditional expectation of the first error term is at most $2\gamma_t b_t$. Since $1 - \gamma_t L \ge 1/2$, Pinsker's inequality and $ab \le a^2 + b^2/4$ imply

$$-\tfrac{1}{2}\,\mathrm{KL}(\pi_{t+1}\|\pi_t) + \gamma_t\|e_t\|_\infty\|\pi_{t+1} - \pi_t\|_1 \le \gamma_t^2\|e_t\|_\infty^2.$$

Taking conditional expectations proves (B.11). Let $M = \max\{(c-1)\,\mathrm{KL}(\pi_\star\|\pi_0), 4V\}$, and let $s = t + c$. If $\mathbb{E}\,\mathrm{KL}(\pi_\star\|\pi_t) \le M/(t+c-1)$, then

$$\mathbb{E}\,\mathrm{KL}(\pi_\star\|\pi_{t+1}) \le \left(1 - \frac{2}{s}\right)\frac{M}{s-1} + \frac{4V}{s^2} \le \frac{M}{s},$$

because $4V \le M$ and $s > 1$. The claim therefore follows by induction. □

For a constant step $\gamma \le (2L)^{-1}$ and uniform bounds $b_t \le b$, $V_t \le V$, the same recursion gives

$$\mathbb{E}\,\mathrm{KL}(\pi_\star\|\pi_T) \le (1-\gamma)^T\,\mathrm{KL}(\pi_\star\|\pi_0) + (2b + \gamma V)\left(1 - (1-\gamma)^T\right).$$

C Proofs and Additional Results for Section 3

C.1 Proof of Theorem 3.1

Proof of Theorem 3.1. The policy gradient $\nabla_\phi \mathcal{Z}(x, \pi_\phi)$ of $\mathcal{Z}(x, \pi_\phi)$ is given by

$$\frac{1}{\tau} \cdot \frac{\mathbb{E}_{y' \sim \pi_{\mathrm{ref}}(\cdot|x)} \left[\mathbb{E}_{y \sim \pi_\phi(\cdot|x)} \left[\nabla_\phi \log \pi_\phi(y|x) \cdot \mathcal{P}(y \succ y'|x) \right] \exp\left(-\frac{1}{\tau} \cdot \mathcal{P}(\pi_\phi \succ y' \mid x) \right) \right]}{\mathbb{E}_{y' \sim \pi_{\mathrm{ref}}(\cdot|x)} \left[\exp\left(-\frac{1}{\tau} \cdot \mathcal{P}(\pi_\phi \succ y' \mid x) \right) \right]}$$
$$= \frac{1}{\tau} \cdot \mathbb{E}_{y \sim \pi_\phi(\cdot|x), y' \sim \pi_\phi^{\mathrm{G}}(\cdot|x)} \Big[\nabla_\phi \log \pi_\phi(y|x) \cdot \mathcal{P}\left(y \succ y'|x\right) \Big],$$

where π_ϕ^{G} is the Gibbs distribution defined in (3.2). Then the desired result directly follows. □

D Experimental Setup

Implementations. Each baseline is run with its native update rule: Nash-MD with the token-level geometric-mixture trainer of TRL (von Werra et al., 2020), EGPO with the authors' extragradient trainer, and MPO with its periodic magnet and proximal updates. Nash-RS draws proposals from the reference policy, accepts one with the rejection rule $u \leq \exp(-\widehat{g}/\tau)$, and then trains on the resulting implicit reward with PPO in OpenRLHF (Hu et al., 2024), using vLLM (Kwon et al., 2023) for generation. We use $B_1 = B_2 = 2$, never force acceptance.

General preferences: Qwen2.5-3B on UltraFeedback. We train all parameters of Qwen2.5-3B-Instruct (Qwen Team, 2024) on the UltraFeedback (Cui et al., 2024) dataset. The oracle is an equal-weight mixture of two independently trained reward models, the OpenAssistant (Köpf et al., 2023) DeBERTa-v3-large reward model[1] and Skywork-Reward-V2-Qwen3-0.6B (Liu et al., 2025).[2] On a diagnostic set, this mixture is not representable by a single Bradley–Terry model. Here Nash-RS uses $\tau = 0.5$, which was chosen on a separate 0.5B temperature sweep (Appendix E), proposal batches of 4, and a cap of 64. We also report Reward-PPO and self-play (the Nash-MD endpoint without mixing) as reference points.

Safety alignment: Gemma-2-2B on PKU-SafeRLHF. The initial policy is the Gemma-2-2B (Gemma Team, 2024) SFT checkpoint released by the EGPO authors (Zhou et al., 2025).[3] The oracle is their direct pairwise preference classifier, trained on a preference mixture that includes PKU-SafeRLHF, and we query it with a randomized answer order.[4] We use it as a general preference and never convert it to a scalar reward. From all three PKU-SafeRLHF (Ji et al., 2025a; Dai et al., 2024) subsets, we deduplicate prompts, remove any prompt that also appears in the test split, and keep prompts that fit in 128 chat-template tokens. This leaves 32,223 training prompts. Every method makes one pass over all of them with rank-16 LoRA adapters, a global batch of 64 prompts, and a 512-token response limit. Nash-RS uses $\tau = 0.5$, proposal batches of $K = 8$, and a 4,096-proposal cap that is never reached. Its average acceptance rate is 0.54–0.57.

Evaluation. Greedy decoding uses at most 512 new tokens, and the bootstrap intervals use 10,000 prompt resamples. The UltraFeedback test split is frozen and disjoint from the training and selection prompts. The PKU evaluation uses all 7,491 eligible held-out test prompts. The full test set shrinks the bootstrap half-width to about 0.003.

E Rejection-Temperature Ablation

We select the rejection temperature used in Section 5 on a small model. Starting from `Qwen/Qwen2.5-0.5B-Instruct`, we train on the first 512 of the 2,048 UltraFeedback training prompts with the same general-preference mixture oracle, for 256 optimizer steps and a 512-token response limit. We then evaluate each Nash-RS variant jointly with the initial policy and six baselines

[1] https://huggingface.co/OpenAssistant/reward-model-deberta-v3-large-v2

[2] https://huggingface.co/Skywork/Skywork-Reward-V2-Qwen3-0.6B

[3] https://huggingface.co/vectorzhou/gemma-2-2b-it-alpaca-cleaned-SFT

[4] https://huggingface.co/vectorzhou/gemma-2-2b-it-preference_dataset_mixture2_and_safe_pku-Preference

Table 4: Nash-RS temperature ablation at 0.5B (seed 47). All rows use 512 prompts and 256 optimizer steps. Win rates are computed within the 13-policy evaluation population.

τ	Acceptance	PM calls	Tokens (k)	H200-hours	Avg. win	Worst win	Exploit.
0.25	0.179	12,492	1,513.7	2.851	0.4884	0.4644	0.0356
0.50	0.415	5,964	943.5	1.736	**0.5271**	**0.5122**	**0.0000**
0.75	0.578	4,568	790.6	1.421	0.5137	0.4878	0.0122
1.00	0.665	4,102	762.3	1.384	0.4936	0.4702	0.0298
2.00	0.808	3,560	720.3	1.309	0.5118	0.4863	0.0137
4.00	0.893	3,318	729.8	1.330	0.5134	0.4872	0.0128

(Reward-PPO, self-play, Nash-MD, MPO, EGPO and COMAL), 13 policies in total, on the 256 held-out prompts using greedy decoding.

We retrain Nash-RS at $\tau \in \{0.25, 0.5, 0.75, 1, 2, 4\}$ with seed 47 and the same budget. Table 4 shows that acceptance and compute are strongly related, whereas policy quality is non-monotone in τ. The acceptance rate rises from 17.9% at $\tau = 0.25$ to 89.3% at $\tau = 4$, but $\tau = 0.5$ has the best average win rate, worst-case win rate, and empirical exploitability. We therefore use $\tau = 0.5$ in all main-text experiments. The low-temperature $\tau = 0.25$ run uses a maximum-proposal cap of 1,024 rather than 64 so that it can complete the same accepted-sample budget; this change affects cost but not the target acceptance rule.

With $\pm$ the half-width of the 95% prompt-bootstrap interval, the average win rate of $\tau = 0.5$ is 0.527 ± 0.013. The corresponding values for $\tau = 0.75$, 2, and 4 are 0.514 ± 0.014, 0.512 ± 0.013, and 0.513 ± 0.012. These intervals quantify prompt uncertainty for one training seed and must not be interpreted as across-training-seed confidence intervals.

F Other Related Work

Bias and preference diversity in LLM alignment. Concerns have been raised about LLMs regarding fairness and potential biases, which have been extensively discussed in recent literature (Bender et al., 2021). These issues include gender bias (Kotek et al., 2023), political bias (Feng et al., 2023), and language bias (Luo et al., 2024). One contributing factor to these biases arises from RLHF—Xiao et al. (2025) observed that conventional RLHF methods often converge to a single preference type, a phenomenon known as preference collapse. Additionally, a recent study (Li et al., 2024a) analyzed diversity in the supervised fine-tuning phase, further highlighting the challenges in maintaining preference diversity.

Preference fine-tuning for RLHF. Several preference fine-tuning methods have been proposed for RLHF. The work of Li et al. (2024b) showed that proximal policy optimization (PPO) (Schulman et al., 2017) does not fully leverage the advantages of RLHF in aligning LLMs with human preferences. The work of Tang et al. (2024) examined the performance gap between online and offline algorithms for alignment tasks, while Ye et al. (2024) introduced an online iterative RLHF algorithm that incorporates a general preference model.

Direct preference optimization. A popular alternative to reward-based RLHF is direct preference optimization (DPO) (Rafailov et al., 2023). DPO fine-tunes LLMs directly on human preference data, eliminating the need to train a reward model, which makes it computationally more efficient. Several notable variants of DPO have been developed (Liu et al., 2024; Azar et al., 2024; Chang et al., 2024; Gorbatovski et al., 2024; Rafailov et al., 2024; Yang et al., 2024). However, recent studies (Li et al., 2023; Xu et al., 2024; Tajwar et al., 2024) suggest that DPO is less effective than reward-based RLHF for aligning LLMs. Both Li et al. (2023) and Xu et al. (2024) argued that this inferiority stems from representation misspecification in DPO, which limits its ability to achieve robust alignment compared to reinforcement learning approaches like PPO. Moreover, the on-policy nature of reward-based fine-tuning enhances LLM performance by mitigating distribution shifts between the training dataset and online responses (Tajwar et al., 2024).