# TOWARDS GENERAL PREFERENCE ALIGNMENT: NASH PREFERENCE OPTIMIZATION FOR DIFFUSION MODELS

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Reinforcement learning from human feedback (RLHF) has become a standard approach for adapting text-to-image (T2I) diffusion models to human feedback. Direct Preference Optimization (DPO) offers a computationally efficient approach that avoids explicit reward modeling and has been widely adopted in diffusion alignment. However, existing preference-based methods for diffusion alignment still rely on reward-induced preference signals and typically assume that human preferences can be adequately modeled by the Bradley–Terry–Luce (BTL) model, which may fail to capture the full complexity of human preferences. In this paper, we formulate diffusion alignment from a game-theoretic perspective. We propose Diffusion Nash Preference Optimization (Diff.-NPO), an intuitive general preference framework for diffusion alignment. Diff.-NPO encourages the current policy to play against itself to achieve self-improvement and lead to a better alignment. Empirically, we demonstrate the effectiveness of Diff.-NPO on the text-to-image generation task via various metrics. Diff.-NPO consistently outperforms existing preference-based diffusion alignment methods.

## 1 INTRODUCTION

Diffusion models have become a dominant framework for text-to-image generation due to their ability to produce high-fidelity images that are well aligned with textual descriptions. Representative systems such as Stable Diffusion (Podell et al., 2023; Rombach et al., 2022) and DALL-E 2 (Ramesh et al., 2022) demonstrate the strong generative capability of this class of models. Although large-scale pretraining on web data provides a powerful foundation, it does not by itself guarantee outputs that adequately reflect human preference. Drawing inspiration from alignment methods developed for large language models, recent work has adapted preference-based objectives such as DPO to diffusion models (Rafailov et al., 2024; Li et al., 2024; Yang et al., 2024b) , enabling learning from preference data without explicit reward modeling. By incorporating preference signals directly into the optimization process, these methods have demonstrated clear advantages over standard supervised fine-tuning (Wallace et al., 2024; Yang et al., 2024a).

Despite these advantages, most existing DPO-style diffusion alignment methods are inherited from the RLHF paradigm, where preference learning is often interpreted through an implicit reward induced by the Bradley–Terry–Luce (BTL) model (Bradley & Terry, 1952; Plackett, 1975). However, the BTL-based preference modeling fails to characterize the complexity of human preferences. More specifically, these approaches assign a scalar reward $r(\mathbf{x}, y)$ to each prompt–response pair $(\mathbf{x}, y)$. Given a prompt $\mathbf{x}$ and two responses $(y_1, y_2)$, they assume that the probability that $y_1$ is preferred over $y_2$ (denoted by $y_1 \succ y_2$) is $\sigma(r(\mathbf{x}, y_1) - r(\mathbf{x}, y_2))$, where $\sigma(u) = 1/(1 + \exp(-u))$ is the sigmoid function. The goal is then to find a policy $\pi(y \mid \mathbf{x})$ (e.g., LLM or an image generator) that maximizes the expected reward while staying close to a reference (often an instruction-tuned) model. However, this BTL preference model imposes a transitive ordering; writing $\mathbb{P}(y_1 \succ y_2 \mid \mathbf{x})$ for the probability that a randomly chosen individual prefers $y_1$ over $y_2$ given $\mathbf{x}$, the BTL model implies that

$$\mathbb{P}(y_1 \succ y_2 \mid \mathbf{x}) > \tfrac{1}{2} \quad \text{and} \quad \mathbb{P}(y_2 \succ y_3 \mid \mathbf{x}) > \tfrac{1}{2} \quad \Longrightarrow \quad \mathbb{P}(y_1 \succ y_3 \mid \mathbf{x}) > \tfrac{1}{2}.$$

On the other hand, studies in social choice and psychology have documented cyclic preferences: people may prefer $y_1$ over $y_2$ and $y_2$ over $y_3$, yet prefer $y_3$ over $y_1$ (May, 1954; Tversky, 1969;



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.

McGarvey, 1953; Loomes et al., 1991). Thus, the transitivity assumption may fail to capture observed human decision-making and preferences across diverse groups. In image generation, Bai et al. (2026) also report cyclic preference annotations in Pick-a-Pic and discard the corresponding groups when constructing consistent image rankings. These limitations motivate treating the pairwise preference probability $\mathbb{P}$ itself as the primitive object of alignment, rather than first reducing it to a scalar reward. This immediately raises a question: *when preferences need not admit a global ranking, what should an aligned policy optimize toward?*

Nash-learning-based alignment methods (Rosset et al., 2024; Zhang et al., 2024; Nan et al., 2026; Wu et al., 2024) provide a natural solution to this challenge by optimizing directly with respect to a general pairwise preference oracle $\mathbb{P}$. These methods formulate alignment as a min-max game between two policies. Given two policies $\pi_1$ and $\pi_2$, the expected preference, or win rate, of $\pi_1$ over $\pi_2$, denoted by $\mathbb{P}(\pi_1 \succ \pi_2)$, is the average of $\mathbb{P}(y_1 \succ y_2 \mid \mathbf{x})$ over prompts $\mathbf{x} \sim d_0$ and responses $y_i \sim \pi_i(\cdot \mid \mathbf{x})$, $i = 1, 2$. Here, $d_0$ is the prompt distribution, and the two responses are drawn independently given the prompt. The goal is to find a policy that maximizes its worst-case value against competing policies, while penalizing both players for moving away from an instruction-tuned reference model $\pi_{\text{sft}}$:

$$\pi^* = \arg\max_{\pi_1} \min_{\pi_2} \mathbb{P}(\pi_1 \succ \pi_2) - \lambda \mathrm{KL}(\pi_1 \mid \pi_{\text{sft}}) + \lambda \mathrm{KL}(\pi_2 \mid \pi_{\text{sft}}) \,.$$

As articulated in Munos et al. (2024), modeling pairwise preferences directly accommodates intransitivity and avoids the misspecification that can arise when these preferences are represented through a BTL reward model. Since its introduction, a growing body of work has developed NLHF algorithms for LLM alignment and studied their convergence properties, the limitations of scalar reward representations, and the diversity of the resulting policies (Zhou et al., 2025; Wu et al., 2026; Liu et al., 2026; Wang et al., 2025b; Calandriello et al., 2024; Swamy et al., 2024).

Although NLHF has been extensively studied for LLMs, its application to text-to-image diffusion models has received comparatively little attention. This paper aims to bridge this gap. However, transferring likelihood-based alignment methods to diffusion models is not straightforward: autoregressive LLMs allow response likelihoods to be computed from token probabilities, whereas computing an image likelihood under a diffusion model requires integrating over unobserved intermediate states. This difficulty also motivates Diffusion-DPO (Wallace et al., 2024), which reformulates the DPO objective of Rafailov et al. (2024) through variational approximations to obtain a tractable denoising loss. To bring Nash learning to diffusion alignment, we propose *Diffusion Nash Preference Optimization* (Diff.-NPO), an online method for learning from general pairwise preferences. We build on INPO (Zhang et al., 2024), which aligns LLMs iteratively: at round $s$, it trains a new policy to improve its expected win rate against the current policy $\pi_s$. Extending its mirror-descent formulation to diffusion trajectories, we derive a population update that balances this improvement with staying close to both $\pi_s$ and a fixed reference model. As in LLM alignment, we can compare diffusion policies by generating images $y \sim \pi_\theta(\cdot \mid \mathbf{x})$ and $y' \sim \pi_s(\cdot \mid \mathbf{x})$ for the same prompt $\mathbf{x}$ and asking which image is preferred. The difficulty lies in using these comparisons to update the model. INPO's loss requires a pairwise score containing $\log\left(\pi_\theta(y \mid \mathbf{x})/\pi_\theta(y' \mid \mathbf{x})\right)$, along with corresponding ratios under $\pi_s$ and the reference model. For autoregressive LLMs, these likelihoods can be computed from token probabilities. For diffusion models, evaluating them requires integrating over unobserved intermediate states. Thus, although we can generate and compare images, we cannot directly evaluate INPO's likelihood-based loss.

The main contribution of our work is a tractable diffusion alignment objective guided by this trajectory update. For a preference pair $(\mathbf{x}, y^+, y^-)$ (where $y^+ \succ y^-$), we add Gaussian noise at a sampled timestep and evaluate all three models (current, previous iterate and reference model) on the same noisy images. We combine their denoising errors into a score $S_\theta$ that measures how much the trainable model improves on $y^+$ relative to $y^-$, using the previous iterate and reference models as baselines. We train with the logistic loss $\ell(S_\theta) = -\log \sigma(S_\theta)$, with the scaling included in $S_\theta$. This loss encourages the score to favor the preferred image and gradually reduces the weight of the comparison as the score increases. Unlike INPO's squared loss, it does not require each sampled score to match a fixed numerical target. After updating the model, we generate fresh images, collect new comparisons, and repeat the procedure.

We evaluate Diff.-NPO on Stable Diffusion 1.5 and SDXL, using Pick-a-Pic prompts and automatic preference feedback for training. On Pick-a-Pic, Parti-Prompts, and HPSv2, Diff.-NPO achieves the highest average win rates across evaluators among the methods tested, with each method compared

against the original backbone. We also conduct ablation studies to examine how the logistic loss and reference weighting affect performance. Our main contributions are:

- We extend INPO's mirror-descent formulation to diffusion trajectories under general pairwise preferences. We derive a population update that balances expected win-rate improvement against the previous policy with regularization toward that policy and a fixed reference model.
- We develop Diff.-NPO, an online alignment algorithm guided by this update, with a tractable logistic surrogate based on denoising errors at a sampled timestep. The objective avoids evaluating image likelihoods and gradually reduces the weight of comparisons as their scores increase, without requiring each sampled score to match a fixed numerical target.
- We evaluate Diff.-NPO on Stable Diffusion 1.5 and SDXL across Pick-a-Pic, Parti-Prompts, and HPSv2. It achieves the highest average win rates across automatic evaluators among the methods tested, with each method compared against the original backbone. Ablation studies examine the effects of the logistic loss and reference weighting.

## 2 PRELIMINARIES

In this section, we introduce the basic background on diffusion models and Nash learning that will be used in our method, and defer a broader discussion of related work to Appendix B.

### 2.1 DIFFUSION MODELS

Let $\mathbf{y}_0 \sim p_{\text{data}}$ denote a clean image in the image space $\mathcal{Y}$. Diffusion models learn to generate images by reversing a process that gradually adds Gaussian noise to clean data. Given a noise schedule $\{\beta_t\}_{t=1}^T$, with $\beta_t \in (0,1)$, define $\alpha_t := 1 - \beta_t$ and $\bar{\alpha}_t := \prod_{s=1}^t \alpha_s$. The forward process Ho et al. (2020) is as follows:

$$q(\mathbf{y}_t \mid \mathbf{y}_{t-1}) = \mathcal{N}\big(\mathbf{y}_t; \sqrt{\alpha_t}\mathbf{y}_{t-1}, \beta_t I\big), \qquad q(\mathbf{y}_{1:T} \mid \mathbf{y}_0) = \prod_{t=1}^T q(\mathbf{y}_t \mid \mathbf{y}_{t-1}), \tag{1}$$

where $q$ is fixed throughout training. Equivalently, we can obtain a noisy image directly as $\mathbf{y}_t = \sqrt{\bar{\alpha}_t}\mathbf{y}_0 + \sqrt{1-\bar{\alpha}_t}\epsilon$, with $\epsilon \sim \mathcal{N}(0, I)$. The reverse process starts from Gaussian noise $p(\mathbf{y}_T) \sim \mathcal{N}(0, I)$ (for large $T$) and progressively removes noise through learned transitions $p_\theta(\mathbf{y}_{t-1} \mid \mathbf{y}_t)$. Together, these transitions define $p_\theta(\mathbf{y}_{0:T}) = p(\mathbf{y}_T)\prod_{t=1}^T p_\theta(\mathbf{y}_{t-1} \mid \mathbf{y}_t)$, whose marginal $p_\theta(\mathbf{y}_0)$ is the model's distribution over generated images. However, directly evaluating this marginal is difficult because it requires integrating over all intermediate states. Therefore we use the evidence lower bound (ELBO) (Ho et al., 2020) during training:

$$\log p_\theta(\mathbf{y}_0) \geq \mathbb{E}_{q(\mathbf{y}_{1:T}|\mathbf{y}_0)}\left[\log \tfrac{p_\theta(\mathbf{y}_{0:T})}{q(\mathbf{y}_{1:T}|\mathbf{y}_0)}\right]. \tag{2}$$

For $t \geq 2$, the ELBO encourages each reverse transition to match the tractable posterior $q(\mathbf{y}_{t-1} \mid \mathbf{y}_t, \mathbf{y}_0)$ (which is also Gaussian, as follows from Equation (1)) by minimizing their KL divergence. Under the standard Gaussian reverse model, this can be achieved by training a neural network $\epsilon_\theta(\mathbf{y}_t, t)$ to predict the noise $\epsilon$. With fixed reverse variance $\sigma_t^2 = \beta_t(1-\bar{\alpha}_{t-1})/(1-\bar{\alpha}_t)$, the denoising KL terms reduce to the weighted noise-prediction loss

$$\mathcal{L}_{\text{denoise}}(\theta) = \mathbb{E}_{t,\mathbf{y}_0,\epsilon}\left[\tfrac{\beta_t^2}{2\sigma_t^2\alpha_t(1-\bar{\alpha}_t)}\big\|\epsilon - \epsilon_\theta(\mathbf{y}_t, t)\big\|_2^2\right],$$

where $t \sim \text{Unif}\{2, \ldots, T\}$, $\mathbf{y}_0 \sim p_{\text{data}}$, and $\epsilon \sim \mathcal{N}(0, I)$ are sampled independently. The full ELBO also includes a term for reconstructing the clean image $\mathbf{y}_0$ from $\mathbf{y}_1$, and a term measuring how close $q(\mathbf{y}_T \mid \mathbf{y}_0)$ is to the Gaussian prior. The latter does not depend on $\theta$ and can be omitted during training.

### 2.2 DIFFUSION MODEL ALIGNMENT WITH HUMAN FEEDBACK

Direct preference optimization (DPO) aligns diffusion models with human preferences using comparisons between generated images. Let $r(\mathbf{x}, \mathbf{y}_0)$ denote the reward assigned to image $\mathbf{y}_0$ under prompt $\mathbf{x}$. The standard KL-regularized RLHF objective is

$$\min_\theta \mathbb{E}_{\mathbf{x}\sim d_0}\left[-\mathbb{E}_{\mathbf{y}_0\sim p_\theta(\cdot|\mathbf{x})} r(\mathbf{x}, \mathbf{y}_0) + \beta\,\text{KL}(p_\theta(\cdot \mid \mathbf{x}) \,\|\, p_{\text{ref}}(\cdot \mid \mathbf{x}))\right],$$

where $p_{\text{ref}}$ is a fixed reference model and $\beta > 0$ controls how strongly deviations from it are penalized. The Optimizer $p^*$ of the above objective over all prompt to image generator has the following closed form expression:

$$p^*(\mathbf{y}_0 \mid \mathbf{x}) \propto p_{\text{ref}}(\mathbf{y}_0 \mid \mathbf{x}) \exp\left(r(\mathbf{x}, \mathbf{y}_0)/\beta\right) \iff r(\mathbf{x}, \mathbf{y}_0) = \beta \log \frac{p^*(\mathbf{y}_0|\mathbf{x})}{p_{\text{ref}}(\mathbf{y}_0|\mathbf{x})} + \beta \log Z(\mathbf{x}),$$

where $Z(\mathbf{x})$ is a normalizing constant. Under the BTL model, given $(\mathbf{x}, \mathbf{y}_0, \mathbf{y}_0')$, the probability that $\mathbf{y}_0$ is preferred to $\mathbf{y}_0'$ is given by $\sigma(r(\mathbf{x}, \mathbf{y}_0) - r(\mathbf{x}, \mathbf{y}_0'))$, where $\sigma(u) = 1/(1 + e^{-u})$ is the sigmoid function. Substituting the above expression for $r$ expresses these probabilities in terms of $p^*$ and $p_{\text{ref}}$, since the prompt-dependent term $\beta \log Z(\mathbf{x})$ cancels while taking the reward difference. This is the key idea of DPO based approach for diffusion alignment that replaces the unknown $p^*$ with $p_\theta$ and learns $\theta$ by minimizing the negative log-likelihood of observed image comparisons, without separately estimating the reward function:

$$\mathcal{L}_{\text{DPO}}(\theta) = -\mathbb{E}_{(\mathbf{x}, \mathbf{y}_0^+, \mathbf{y}_0^-) \sim \mathcal{D}} \left[ \log \sigma \left( \beta \log \frac{p_\theta(\mathbf{y}_0^+|\mathbf{x})}{p_{\text{ref}}(\mathbf{y}_0^+|\mathbf{x})} - \beta \log \frac{p_\theta(\mathbf{y}_0^-|\mathbf{x})}{p_{\text{ref}}(\mathbf{y}_0^-|\mathbf{x})} \right) \right],$$

where $\mathcal{D}$ contains prompts paired with preferred images $\mathbf{y}_0^+$ and dispreferred images $\mathbf{y}_0^-$. For autoregressive LLMs, the likelihood of an observed response can be evaluated directly as a product of token probabilities. For diffusion models, however, the likelihood of a clean image requires integrating over unobserved intermediate diffusion states, making the likelihood ratios in the DPO objective generally intractable. Using a trajectory-based formulation, variational bounds, and approximations based on the forward noising process, Wallace et al. (2024) derive the tractable surrogate

$$\mathcal{L}_{\text{Diff-DPO}}(\theta) = -\mathbb{E}_{\mathcal{D}, t, \epsilon^+, \epsilon^-} \left[ \log \sigma(\beta T \omega_t [\Delta_\theta(t) - \Delta_{\text{ref}}(t)]) \right],$$

where $t \sim \text{Unif}\{1, \ldots, T\}$, $\omega_t \geq 0$ is a prescribed timestep weight, and for $a \in \{\theta, \text{ref}\}$,

$$\Delta_a(t) := \|\epsilon^- - \epsilon_a(\mathbf{y}_t^-, \mathbf{x}, t)\|_2^2 - \|\epsilon^+ - \epsilon_a(\mathbf{y}_t^+, \mathbf{x}, t)\|_2^2. \tag{3}$$

Here, $\mathbf{y}_t^\pm = \sqrt{\bar{\alpha}_t}\mathbf{y}_0^\pm + \sqrt{1 - \bar{\alpha}_t}\epsilon^\pm$ is obtained by adding Gaussian noise $\epsilon^\pm \sim \mathcal{N}(0, I)$ to each image, and $\epsilon_\theta$ and $\epsilon_{\text{ref}}$ are the prompt-conditioned denoisers of the current and reference models. Both models are evaluated on the same noisy images. Thus, $\Delta_a(t)$ measures the denoising advantage of the preferred image over the dispreferred image under model $a$. The loss encourages this advantage to be larger under the current model than under the reference model.

### 2.3 Nash Learning with Human Feedback

A key limitation of BTL-based alignment methods is their reliance on scalar rewards, which cannot capture cyclic preferences Munos et al. (2024). Nash learning instead uses pairwise preference probabilities directly. Let $\mathbb{P}(\mathbf{y}_0 \succ \mathbf{y}_0' \mid \mathbf{x}) \in [0, 1]$ denote the probability that image $\mathbf{y}_0$ is preferred to $\mathbf{y}_0'$ under the same prompt $\mathbf{x}$, where $\succ$ denotes the preference event. For a prompt $\mathbf{x} \sim d_0$, let policies $p_1$ and $p_2$ generate images $\mathbf{y}_0$ and $\mathbf{y}_0'$, respectively. NLHF formulates preference alignment as a two-player zero-sum game in which $p_1$ maximizes and $p_2$ minimizes the KL-regularized preference objective

$$\begin{aligned} J(p_1, p_2) = \mathbb{E}_{\mathbf{x} \sim d_0} \Big[ & \mathbb{E}_{\mathbf{y}_0 \sim p_1(\cdot|\mathbf{x}),\, \mathbf{y}_0' \sim p_2(\cdot|\mathbf{x})} \big[ \mathbb{P}(\mathbf{y}_0 \succ \mathbf{y}_0' \mid \mathbf{x}) \big] \\ & - \tau \, \text{KL}\big(p_1(\cdot \mid \mathbf{x}) \,\|\, p_{\text{ref}}(\cdot \mid \mathbf{x})\big) + \tau \, \text{KL}\big(p_2(\cdot \mid \mathbf{x}) \,\|\, p_{\text{ref}}(\cdot \mid \mathbf{x})\big) \Big]. \end{aligned} \tag{4}$$

Here $p_{\text{ref}}$ is a fixed reference policy and $\tau > 0$ controls the regularization strength. The first term measures $p_1$'s expected win rate against $p_2$. The opposite signs of the KL terms ensure that deviations from the reference are penalized for both the maximizing and minimizing players. We assume that $\mathbb{P}(\mathbf{y}_0 \succ \mathbf{y}_0' \mid \mathbf{x}) + \mathbb{P}(\mathbf{y}_0' \succ \mathbf{y}_0 \mid \mathbf{x}) = 1$. For a given policy class $\mathcal{P}$, we seek a Nash policy $p^* \in \mathcal{P}$ such that neither player can improve its objective by changing its policy alone. Specifically, $p^*$ satisfies

$$J(p_1, p^*) \leq J(p^*, p^*) \leq J(p^*, p_2), \qquad \forall p_1, p_2 \in \mathcal{P}.$$

The first inequality means that the maximizing player cannot gain by moving away from $p^*$, while the second means that the minimizing player cannot gain by doing so. This condition implies that $p^*$ maximizes the worst-case value of the KL-regularized preference objective $J$ against competing policies in $\mathcal{P}$.

## 3 NLHF FOR DIFFUSION MODELS

Having motivated the Nash policy through the minimax formulation of the KL-regularized preference game, we next use this game to construct an iterative learning direction for diffusion models. Rather than solving for the equilibrium directly, at iteration $s$ we optimize a regularized response against the current policy $p_s$. Following the online mirror-descent view of Nash learning, we derive the corresponding population policy update and then translate it into a tractable diffusion objective.

### 3.1 FROM MIRROR DESCENT TO DIFFUSION PREFERENCE OPTIMIZATION

We first derive an ideal population update against the current policy $p_s$, which is held fixed throughout this update. Let

$$U_s(\mathbf{x}, \mathbf{y}_0) := \mathbb{E}_{\mathbf{y}_0' \sim p_s(\cdot|\mathbf{x})}\left[\mathbb{P}(\mathbf{y}_0 \succ \mathbf{y}_0' \mid \mathbf{x})\right]$$

denote the expected win rate of image $\mathbf{y}_0$ against an image generated by $p_s$. The image-level loss for a candidate policy $p$ is

$$\ell_s(p) := \mathbb{E}_{\mathbf{x}\sim d_0}\left[-\mathbb{E}_{\mathbf{y}_0\sim p(\cdot|\mathbf{x})}[U_s(\mathbf{x}, \mathbf{y}_0)] + \tau\,\mathrm{KL}\left(p(\cdot \mid \mathbf{x}) \,\|\, p_{\mathrm{ref}}(\cdot \mid \mathbf{x})\right)\right].$$

Since the image-level KL divergence is generally intractable, we work with the following trajectory-level upper bound (derivation is provided in Appendix C.1):

$$\widetilde{\ell}_s(p) := \mathbb{E}_{\mathbf{x}\sim d_0}\left[-\mathbb{E}_{\mathbf{y}_{0:T}\sim p(\cdot|\mathbf{x})}[U_s(\mathbf{x}, \mathbf{y}_0)] + \tau\,\mathrm{KL}_{\mathbf{y}_{0:T}}\left(p(\cdot \mid \mathbf{x}) \,\|\, p_{\mathrm{ref}}(\cdot \mid \mathbf{x})\right)\right]. \tag{5}$$

Here preferences depend only on the final clean image, so the expected win rate is unchanged. An application of chain rule within KL divergence yields $\ell_s(p) \le \widetilde{\ell}_s(p)$ (See Appendix C.1 for details), which implies that the surrogate $\widetilde{\ell}_s$ is a majorization of $\ell_s$. However, directly minimizing $\widetilde{\ell}_s$ can produce unstable updates that move the policy far from $p_s$ (Zhang et al., 2024); such greedy updates are also known to incur large regret (Lattimore & Szepesvári, 2020). Following Zhang et al. (2024), we therefore approximate $\widetilde{\ell}_s(p)$ by its first-order expansion around the current policy $p_s$:

$$\widetilde{\ell}_s(p) \approx \widetilde{\ell}_s(p_s) + \langle \nabla\widetilde{\ell}_s(p_s),\, p - p_s\rangle,$$

where the inner product integrates over trajectories and averages over prompts $\mathbf{x} \sim d_0$. Writing $g_s := \nabla\widetilde{\ell}_s(p_s)$, we obtain

$$g_s(\mathbf{y}_{0:T}, \mathbf{x}) = -U_s(\mathbf{x}, \mathbf{y}_0) + \tau\left[\log \tfrac{p_s(\mathbf{y}_{0:T}|\mathbf{x})}{p_{\mathrm{ref}}(\mathbf{y}_{0:T}|\mathbf{x})} + 1\right].$$

We then minimize this linearized objective with a KL penalty that discourages large deviations from the current policy $p_s$. After omitting terms that do not depend on $p$, the resulting online mirror-descent update is:

$$p_{s+1}^{\mathrm{pop}} = \arg\min_p\ \mathbb{E}_{\mathbf{x}\sim d_0}\left[\mathbb{E}_{\mathbf{y}_{0:T}\sim p(\cdot|\mathbf{x})}\left[g_s(\mathbf{y}_{0:T}, \mathbf{x})\right] + \eta\,\mathrm{KL}_{\mathbf{y}_{0:T}}\left(p(\cdot \mid \mathbf{x}) \,\|\, p_s(\cdot \mid \mathbf{x})\right)\right]. \tag{6}$$

We take $\eta \ge \tau > 0$, with $1/\eta$ acting as the stepsize. This update admits a closed form expression:

$$p_{s+1}^{\mathrm{pop}}(\mathbf{y}_{0:T} \mid \mathbf{x}) \propto \exp(U_s(\mathbf{x}, \mathbf{y}_0)/\eta)\, p_{\mathrm{ref}}(\mathbf{y}_{0:T} \mid \mathbf{x})^{\gamma} p_s(\mathbf{y}_{0:T} \mid \mathbf{x})^{1-\gamma}, \qquad \gamma := \tau/\eta, \tag{7}$$

see Appendix C.1 for details. Compution of the normalizing constant in the definition of $p_{s+1}^{\mathrm{pop}}$ is often intractable, as it requires summing over the entire space of image trajectories. However, taking the log-ratio of two trajectories on both sides of Equation (7) removes the prompt-dependent normalizing constant (an observation, which is also the cornerstone of Rafailov et al. (2024) and Zhang et al. (2024)). Specifically, defining

$$h_s(p; \mathbf{x}, \mathbf{y}_{0:T}, \mathbf{y}_{0:T}') := \log \frac{p(\mathbf{y}_{0:T} \mid \mathbf{x})}{p(\mathbf{y}_{0:T}' \mid \mathbf{x})} - \gamma \log \frac{p_{\mathrm{ref}}(\mathbf{y}_{0:T} \mid \mathbf{x})}{p_{\mathrm{ref}}(\mathbf{y}_{0:T}' \mid \mathbf{x})} - (1-\gamma) \log \frac{p_s(\mathbf{y}_{0:T} \mid \mathbf{x})}{p_s(\mathbf{y}_{0:T}' \mid \mathbf{x})}, \tag{8}$$

the population update $p_{s+1}^{\mathrm{pop}}$ satisfies:

$$h_s(p_{s+1}^{\mathrm{pop}}; \mathbf{x}, \mathbf{y}_{0:T}, \mathbf{y}_{0:T}') = (U_s(\mathbf{x}, \mathbf{y}_0) - U_s(\mathbf{x}, \mathbf{y}_0'))/\eta. \tag{9}$$

This identity provides a pairwise characterization of the target update, although its right-hand side involves expected win rates that are not directly observed in individual comparison between $\mathbf{y}_0$ and $\mathbf{y}_0'$. In practice, we generate candidate images from $p_s$ and use a preference oracle to select preferred

and dispreferred images. Let $\mathcal{D}_s$ denote the resulting distribution of ordered triples $(\mathbf{x}, \mathbf{y}_0^+, \mathbf{y}_0^-)$, including the candidate-selection procedure (in our experiments, these are the highest- and lowest-ranked images among eight candidates, as described in Section 4). We augment each selected image with a trajectory from the forward process $q(\mathbf{y}_{1:T} \mid \mathbf{y}_0)$ and denote the resulting ordered trajectory-pair distribution by $\mathcal{D}_s^q$. Using $h_s$ as a pairwise score (Equation (9)), we adopt the logistic surrogate

$$\mathcal{L}_s^{\log}(p) := -\mathbb{E}_{(\mathbf{x}, \mathbf{y}_{0:T}^+, \mathbf{y}_{0:T}^-) \sim \mathcal{D}_s^q} \left[ \log \sigma\big(\beta h_s(p; \mathbf{x}, \mathbf{y}_{0:T}^+, \mathbf{y}_{0:T}^-)\big) \right], \tag{10}$$

where $\beta > 0$ scales the pairwise score. This is the main departure from INPO (Zhang et al., 2024): written in analogous notation, its population objective minimizes the expected squared difference between the pairwise score $h_s(p; \mathbf{x}, \mathbf{y}_{0:T}, \mathbf{y}'_{0:T})$ and the target $(U_s(\mathbf{x}, \mathbf{y}_0) - U_s(\mathbf{x}, \mathbf{y}'_0))/\eta$. When each two candidates are sampled independently from $p_s$ and ordered via pairwise comparision by the preference oracle, INPO shows that this population loss is equivalent (up to a policy-independent constant) to minimizing the expected squared distance between $h_s(p; \mathbf{x}, \mathbf{y}_{0:T}^+, \mathbf{y}_{0:T}^-)$ and a fixed constant $1/2\eta$. However, selecting the best and worst among the candidate set changes the pair distribution, so this equivalence does not directly apply to our training data. We use logistic loss to encourage larger values of $h_s$ on the selected preferred/dispreferred pairs. Squared loss pulls each score toward $1/(2\eta)$, including reducing scores above this target. On the other hand, logistic loss continues to reward larger scores, with diminishing gradient magnitude as the score increases (note that the derivative of the loss with respect to $h_s$ is $-\beta\sigma(-\beta h_s)$). This is precisely what motivates our practical ranking objective (Equation (10)), whose empirical advantage over squared loss is demonstrated in Appendix E.2.

To obtain a tractable diffusion training objective, we parameterize $p$ by $p_\theta$ and hold $p_s$ and $p_{\text{ref}}$ fixed during each update. We assume that the three models start denoising from $\mathcal{N}(0, I)$ and share same noise schedule, and fixed variances during the noise removal. In particular, the log-likelihood is

$$\log p_a(\mathbf{y}_{t-1} \mid \mathbf{y}_t, \mathbf{x}) = C_t - (\|\mathbf{y}_{t-1} - \mu_a(\mathbf{y}_t, \mathbf{x}, t)\|_2^2 / 2\sigma_t^2), \quad a \in \{\theta, s, \text{ref}\},$$

where $\mu_a$ is the predicted mean of the reversal process (i.e., we model $p_a(\mathbf{y}_{t-1} \mid \mathbf{y}_t, \mathbf{x}) \sim \mathcal{N}(\mu_a(\mathbf{y}_t, \mathbf{x}, t), \sigma_t^2 I)$) and $C_t$ does not depend on $a$. Thus, transition log-likelihood ratios are differences of squared prediction errors. Each model uses a neural network $\epsilon_a(\mathbf{y}_t, \mathbf{x}, t)$ to predict the Gaussian noise $\epsilon$ in $\mathbf{y}_t = \sqrt{\bar{\alpha}_t}\mathbf{y}_0 + \sqrt{1 - \bar{\alpha}_t}\epsilon$. The reverse mean is then computed from this prediction as

$$\mu_a(\mathbf{y}_t, \mathbf{x}, t) = \tfrac{1}{\sqrt{\alpha_t}} \left( \mathbf{y}_t - \tfrac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}} \epsilon_a(\mathbf{y}_t, \mathbf{x}, t) \right).$$

With this parameterization, averaging the negative transition log-likelihood over $q(\mathbf{y}_{t-1} \mid \mathbf{y}_t, \mathbf{y}_0)$ gives the weighted denoising error $\omega_t \|\epsilon - \epsilon_a(\mathbf{y}_t, \mathbf{x}, t)\|_2^2$ plus a constant independent of $a$. For each preference pair $(\mathbf{x}, \mathbf{y}_0^+, \mathbf{y}_0^-) \sim \mathcal{D}_s$, sample $t \sim \text{Unif}\{1, \ldots, T\}$ and Gaussian noises $\epsilon^\pm \sim \mathcal{N}(0, I)$, and construct

$$\mathbf{y}_t^\pm = \sqrt{\bar{\alpha}_t}\mathbf{y}_0^\pm + \sqrt{1 - \bar{\alpha}_t}\epsilon^\pm.$$

We evaluate all three denoisers on the same noisy image pair and timestep. Following Wallace et al. (2024), we combine these errors using the coefficients in $h_s$ and use the resulting single-timestep score inside the logistic loss:

$$\mathcal{L}_{\text{Diff-NPO}}^{(s)}(\theta) = \mathbb{E}_{\mathcal{D}_s, t, \epsilon^+, \epsilon^-} \left[ -\log \sigma(\beta T \omega_t \left[ \Delta_\theta(t) - \gamma \Delta_{\text{ref}}(t) - (1 - \gamma)\Delta_s(t) \right]) \right], \tag{11}$$

where $\gamma = \tau/\eta$, and $\Delta_a(t)$ is same as defined in Equation (3) for $a \in \{\theta, s, \text{ref}\}$. The factor $T$ accounts for uniformly sampling one of the $T$ timestep contributions. This practical surrogate encourages greater denoising improvement on preferred images than on dispreferred images, relative to the weighted errors of the reference and previous models. The approximation steps are detailed in Appendix C.2. The full algorithm is provided in Algorithm 1 Appendix A.

### 3.2 Gradient Interpretation of the Logistic Objective

We examine how the logistic loss weights the denoising comparisons used to update the model. For a sampled preference pair, timestep, and noise realizations, we write

$$E_\theta^\pm(t) := \|\epsilon^\pm - \epsilon_\theta(\mathbf{y}_t^\pm, \mathbf{x}, t)\|_2^2,$$

which implies $\Delta_\theta(t) = E_\theta^-(t) - E_\theta^+(t)$. The score used in our objective $\mathcal{L}_{\text{Diff-NPO}}^{(s)}(\theta)$ (Equation (11)) is

$$H_\theta(t) := T\omega_t \left[ \Delta_\theta(t) - \gamma \Delta_{\text{ref}}(t) - (1 - \gamma)\Delta_s(t) \right].$$

A positive $H_\theta(t)$ means that the trainable model's denoising advantage for the preferred image exceeds a weighted combination advantage of the reference and previous models. Since both baseline models and the sampled images are held fixed during the update, differentiation gives

$$\nabla_\theta[-\log\sigma(\beta H_\theta(t))] = w_\theta(t)\left[\nabla_\theta E_\theta^+(t) - \nabla_\theta E_\theta^-(t)\right],$$

where

$$w_\theta(t) := \beta T\omega_t\,\sigma(-\beta H_\theta(t)) \geq 0.$$

Thus, the loss assigns a nonnegative weight to the gradient that increases the denoising advantage of the preferred image. For a fixed timestep, this weight is larger when $H_\theta(t)$ is small or negative and decreases as $H_\theta(t)$ grows. The reference and previous models influence the update through this weight: together, they determine the baseline against which each comparison is assessed. For comparison, a squared loss $(H_\theta(t) - m)^2$ with target margin $m > 0$ gives

$$\nabla_\theta(H_\theta(t) - m)^2 = 2T\omega_t\big(m - H_\theta(t)\big)\left[\nabla_\theta E_\theta^+(t) - \nabla_\theta E_\theta^-(t)\right].$$

Its coefficient changes sign when $H_\theta(t) > m$, pulling the score back toward the prescribed target. The logistic loss continues to encourage a larger score, while gradually reducing its weight. This distinction is relevant to diffusion training: the same clean preference pair produces different scores across sampled timesteps and noise realizations. The logistic objective reinforces the observed ordering across these noisy comparisons without requiring every sampled score to approach the same numerical target. This characterizes the behavior of our practical surrogate, whose empirical performance relative to squared loss is evaluated in Appendix E.2.

**Remark 1 (Comparison with Diffusion-DPO)** *For the same preference pairs and timestep weights, $\mathcal{L}^{(s)}_{\text{Diff-NPO}}(\theta)$ reduces to $\mathcal{L}_{\text{Diff-DPO}}(\theta)$ when $\gamma = 1$. When $0 < \gamma < 1$, our Diff-NPO objective compares the denoising errors with those of both the reference and the previous policy. To understand why this can be useful, suppose $\Delta_s(t) > \Delta_{\text{ref}}(t)$. In that case, including $\Delta_s(t)$ decreases the score inside the sigmoid, which gives this comparison a larger gradient weight than in Diffusion-DPO. Thus, the pairs that are already well separated relative to the reference model receive greater weight, which encourages further improvement as the policy evolves. The reference model remains part of the comparison throughout training. Our experiments in Section 4.2 support this choice, showing higher average evaluator win rates than online Diffusion-DPO.*

# 4 Experiments

## 4.1 Experiment Setup

**Models, datasets, baselines, and evaluation metrics.** We fine-tune two text-to-image diffusion backbones, Stable Diffusion 1.5 (SD1.5) and SDXL. The prompts of training data are drawn from the Pick-a-Pic v1 preference dataset (Kirstain et al., 2023). We use prompts from the Pick-a-Pic test split (Kirstain et al., 2023), Parti-Prompts (Yu et al., 2022b), and HPSV2 (Wu et al., 2023) for testing. We compare against several representative baselines, including online SFT trained on preferred images, online Diffusion-DPO (Wallace et al., 2024), SPIN (Yuan et al., 2024) and SEPPO (Zhang et al., 2025a), together with the original pretrained SD1.5 and SDXL models. We use the public available checkpoints for SPIN[1] and SEPPO[2] for evaluation. For text-to-image alignment task, we adopt five widely used automatic metrics: PickScore (Kirstain et al., 2023), HPSV2 (Wu et al., 2023), CLIP (Radford et al., 2021), Aesthetic Score (Schuhmann, 2022), and ImageReward (Xu et al., 2023). We report both the raw metric scores and win rates. For each prompt and each automatic evaluator, the win rate of a method is the percentage of prompts for which its generated image receives a higher evaluator score than the image generated by the original backbone model, e.g., SD1.5 or SDXL. More details are provided in Appendix D.

**Online Generation and Preference Oracle.** During online generation, we use 10 inference steps for efficiency, following prior work showing that a small number of denoising steps has little impact on optimization performance (Zheng et al., 2025; Liu et al., 2025). For each prompt, we generate

[1] https://huggingface.co/UCLA-AGI/SPIN-Diffusion-iter3

[2] https://huggingface.co/DwanZhang/SePPO

Table 1: SD1.5 win-rate and metric results for Diff.-NPO compared with other baselines. Win rates are vs. original SD1.5. The yellow-highlighted rows indicate our method. For each column, the best result is shown in **bold**, and the second-best result is underlined.

| Dataset | Method | SD1.5 Winrate (%) ↑ | | | | | | SD1.5 Metrics ↑ | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | PS | HPS | CLIP | IM | AES | Avg | PS | HPS | CLIP | IM | AES |
| Pick-a-Pic | Original | – | – | – | – | – | – | 20.68 | 26.74 | 33.23 | 0.0495 | 5.5060 |
| | Online-SFT | 40.44 | 46.60 | 54.96 | 53.24 | 59.28 | 50.90 | 20.51 | 26.64 | 33.97 | 0.1555 | 5.5942 |
| | Online-DPO | 77.40 | 82.48 | 59.96 | 81.00 | 73.36 | 74.84 | 21.26 | 27.79 | **34.62** | 0.6861 | 5.7498 |
| | Seppo | **82.68** | 84.52 | 55.04 | 75.32 | 80.56 | 75.62 | 21.40 | 27.80 | 33.93 | 0.5381 | 5.8106 |
| | SPIN | 81.32 | 77.36 | 44.68 | 69.88 | **85.36** | 71.72 | 21.43 | 27.65 | 32.54 | 0.4461 | **5.9669** |
| | Diff.-NPO | 81.60 | **84.64** | **60.32** | **85.12** | 80.76 | **78.49** | **21.48** | **28.02** | 34.61 | **0.8896** | 5.8773 |
| Parti-Prompts | Original | – | – | – | – | – | – | 21.53 | 27.44 | 33.15 | 0.1926 | 5.3592 |
| | Online-SFT | 38.88 | 45.24 | 55.10 | 54.18 | 59.83 | 50.65 | 21.35 | 27.37 | 33.84 | 0.3366 | 5.4508 |
| | Online-DPO | 71.43 | 78.42 | **56.38** | 74.73 | 74.73 | 71.14 | 21.95 | 28.31 | **34.09** | 0.6699 | 5.5266 |
| | Seppo | 73.97 | 77.59 | 50.44 | 68.58 | 75.56 | 69.23 | 21.97 | 28.24 | 33.43 | 0.5143 | 5.5904 |
| | SPIN | 72.73 | 72.34 | 39.01 | 62.34 | **81.92** | 65.67 | 21.98 | 28.15 | 31.98 | 0.4014 | **5.7324** |
| | Diff.-NPO | **74.76** | **79.70** | 54.74 | **77.04** | 76.96 | **72.64** | **22.07** | **28.49** | 34.07 | **0.7966** | 5.6231 |
| HPSV2 | Original | – | – | – | – | – | – | 21.05 | 27.08 | 35.43 | 0.0727 | 5.5725 |
| | Online-SFT | 40.25 | 48.70 | 50.25 | 55.15 | 57.95 | 50.46 | 21.27 | 27.06 | 35.53 | 0.2053 | 5.6453 |
| | Online-DPO | 80.70 | 85.10 | 59.20 | 80.35 | 73.80 | 75.83 | 21.75 | 28.21 | 36.51 | 0.7092 | 5.8203 |
| | Seppo | **83.75** | 85.75 | 54.10 | 77.80 | 81.80 | 76.64 | 21.88 | 28.27 | 35.99 | 0.6121 | 5.9132 |
| | SPIN | 81.70 | 80.65 | 44.65 | 71.10 | **87.25** | 73.07 | 21.92 | 28.20 | 34.69 | 0.5027 | **6.0577** |
| | Diff.-NPO | 83.15 | **87.75** | **60.05** | **83.90** | 79.45 | **78.86** | **21.96** | **28.53** | **36.65** | **0.8761** | 5.8957 |

Table 2: SDXL win-rate and automatic metric results for Diff.-NPO compared with other baselines. Win rates are vs. original SDXL. Yellow rows indicate our method. Best is in **bold**; second-best is underlined.

| Dataset | Method | SDXL Winrate (%) ↑ | | | | | | SDXL Metrics ↑ | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | PS | HPS | CLIP | IM | AES | Avg | PS | HPS | CLIP | IM | AES |
| Pick-a-Pic | Original | – | – | – | – | – | – | 22.06 | 22.71 | 36.38 | 0.5562 | 6.0735 |
| | Online-SFT | 28.84 | 38.00 | **57.84** | 53.36 | 37.40 | 43.09 | 21.72 | 27.46 | **37.15** | 0.6539 | 5.9530 |
| | Online-DPO | 82.08 | 81.92 | 56.16 | 85.32 | 65.64 | 74.22 | 22.71 | 28.57 | 37.05 | 1.1247 | 6.2686 |
| | Diff.-NPO | **82.64** | **86.12** | 55.12 | **89.56** | **74.96** | **77.68** | **22.78** | **28.85** | 37.01 | **1.2599** | **6.4294** |
| Parti-Prompts | Original | – | – | – | – | – | – | 22.51 | 28.00 | 35.23 | 0.6003 | 5.7438 |
| | Online-SFT | 30.21 | 36.02 | **54.42** | 51.38 | 44.13 | 43.23 | 22.21 | 27.68 | **35.86** | 0.6976 | 5.6925 |
| | Online-DPO | 78.14 | 77.98 | 47.32 | 79.88 | 76.74 | 72.01 | 23.07 | 28.85 | 35.26 | 1.0427 | 6.0486 |
| | Diff.-NPO | **82.66** | **84.88** | 45.81 | **86.73** | **85.04** | **77.02** | **23.26** | **29.18** | 35.34 | **1.2505** | **6.2496** |
| HPSV2 | Original | – | – | – | – | – | – | 22.75 | 28.18 | 38.64 | 0.7526 | 6.1155 |
| | Online-SFT | 21.85 | 34.70 | 48.20 | 47.25 | 35.75 | 37.55 | 22.25 | 27.84 | 38.62 | 0.7715 | 6.0088 |
| | Online-DPO | **84.80** | 84.75 | **52.65** | 83.35 | 70.55 | 75.22 | 23.46 | 29.17 | **38.92** | 1.2205 | 6.3367 |
| | Diff.-NPO | 83.65 | **88.60** | 50.69 | **88.60** | **75.10** | **77.33** | **23.49** | **29.42** | 38.75 | **1.3319** | **6.4344** |

8 candidate images and use automatic preference oracles to construct preference pairs. Specifically, we evaluate each candidate using five scoring models: PickScore (Kirstain et al., 2023), HPSV2 (Wu et al., 2023), CLIP (Radford et al., 2021), Aesthetic Score (Schuhmann, 2022), and ImageReward (Xu et al., 2023). For each oracle, we rank the 8 candidates according to their scores, and then compute the average rank across the five oracles as the final preference score. The candidate with the best average rank is selected as the positive sample, while the candidate with the worst average rank is selected as the negative sample.

### 4.2 General Evaluation: Text-to-Image Alignment

We report text-to-image generation results on SD1.5 and SDXL in Tables 1 and 2. Diff.-NPO achieves the highest average win rate on every evaluation dataset across both SD1.5 and SDXL, showing consistent gains over the pairwise Online-Diff-DPO baseline. The improvement is particularly strong on SDXL, where Diff.-NPO improves the average win rate by 7.0% relatively on Parti-Prompts and remains the best method on Pick-a-Pic and HPSV2.

The advantage of Diff.-NPO is most evident on preference-oriented metrics. On SDXL Parti-Prompts, Diff.-NPO improves the ImageReward win rate from 79.88 to 86.73 and the AES win rate from 76.74 to 85.04, corresponding to relative improvements of 8.6% and 10.8% over Online-Diff-DPO, respectively. In contrast, SFT is unstable and often performs worse than the original SDXL model, with average win rates below 50%. This is likely because SFT is trained directly on images generated with only 10 inference steps, which are often low quality, whereas RL-based methods such as Diff.-NPO can leverage both positive and negative samples to improve even from poor initial generations. Figure 3 in Appendix F presents qualitative comparisons between Diff.-NPO and the baselines after fine-tuning SDXL. Across diverse prompt types, including objects, human faces, multiple people, and full-body portraits, Diff.-NPO consistently produces images with more coherent and appealing visual quality. In particular, for the prompt "full length portrait of a beautiful woman", Diff.-NPO better preserves anatomical details and generates clearer, more realistic full-body structures compared with the baselines. Additional qualitative results for SD1.5 and SDXL are provided in Figures 5 and 4, respectively.

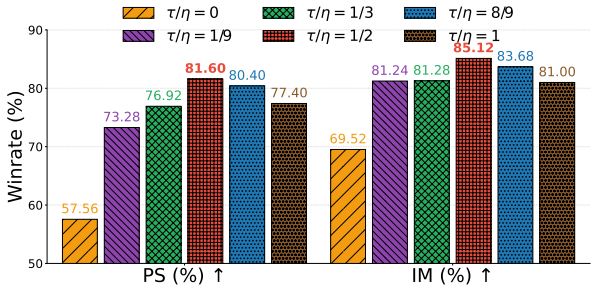



Figure 1: Win rates for different $\tau/\eta$ values, measured using PickScore and ImageReward. See Figure 2 in Appendix E.1 for all five metrics.

Table 3: Ablation study of different $\tau/\eta$ settings on Pick-a-Pic for SD1.5, evaluated using automatic metrics.

| $\tau/\eta$ | PS ↑ | HPS ↑ | CLIP ↑ | IM ↑ | AES ↑ |
|---|---|---|---|---|---|
| 1 | 21.26 | 27.79 | 34.62 | 0.6861 | 5.7498 |
| 8/9 | 21.40 | 27.92 | 34.59 | 0.8001 | 5.7854 |
| 1/2 | 21.48 | 28.02 | 34.61 | 0.8896 | 5.8773 |
| 1/3 | 21.34 | 27.76 | 33.97 | 0.7817 | 5.7916 |
| 1/9 | 21.29 | 27.89 | 34.12 | 0.8098 | 5.9041 |
| 0 | 20.87 | 27.41 | 33.78 | 0.4875 | 5.7541 |

### 4.3 Ablation Studies

**Impact of different $\tau/\eta$ regularization Coefficient.** We study the effect of the regularization ratio $\tau/\eta$, which controls the relative strength of the reference-policy and previous-policy regularization terms. When $\tau/\eta = 1$, the objective only regularizes toward the reference policy and reduces to the DPO-style setting; when $\tau/\eta = 0$, the objective only regularizes toward the previous policy, corresponding to a pure self-play update. Table 3 reports the automatic metric scores under different $\tau/\eta$ values, while Figure 1 visualizes the win-rate comparison on PickScore and ImageReward. The full win-rate comparison across all five metrics is provided in Figure 2 in Appendix E.1. We observe a clear increase-then-decrease trend as $\tau/\eta$ varies from 0 to 1, with the best overall performance achieved at an intermediate value. This suggests that relying solely on either the reference policy or the previous policy is suboptimal, and highlights the importance of jointly using both policies to provide stable and effective regularization.

## 5 Conclusion

In this paper, we proposed Diffusion Nash Preference Optimization (Diff.-NPO), a general preference optimization framework for diffusion models derived from a nash learning perspective. By directly optimizing relative preference probabilities through a self-play-style objective with reference-policy regularization, Diff.-NPO provides a simple and stable approach to online diffusion alignment. Experiments on Stable Diffusion 1.5 and SDXL across Pick-a-Pic, Parti-Prompts, and HPSV2 show that Diff.-NPO consistently improves alignment over strong preference-optimization baselines. These results suggest that Nash-style online learning is a promising alternative to reward-centric formulations for diffusion-model alignment. One limitation of the current study is that our average-rank preference construction relies on five scoring models, and incorporating a broader set of scoring models may provide a more comprehensive preference signal. In future work, we plan to extend this framework to stronger online data-collection strategies, and broader diffusion generation and editing tasks.

## AI USE STATEMENT

Generative AI tools were used solely for language polishing and writing assistance.

## ETHICS STATEMENT

This work uses existing public preference data from Pick-a-Pic and involves no new participant recruitment or manual preference annotation. Practical deployment should therefore retain appropriate content-safety safeguards.

## REPRODUCIBILITY STATEMENT

We provide complete mathematical derivations and algorithmic details in Appendices C and A. Dataset descriptions, baseline configurations, preference-pair construction, implementation details, and additional experimental results are provided in Appendix D.

## REFERENCES


Jiamu Bai, Xin Yu, Meilong Xu, Weitao Lu, Xin Pan, Kiwan Maeng, Daniel Kifer, Jian Wang, and Yu Wang. Towards better optimization for listwise preference in diffusion models. In *International Conference on Learning Representations*, volume 2026, pp. 133254–133282, 2026.

James Betker, Gabriel Goh, Li Jing, Tim Brooks, Jianfeng Wang, Linjie Li, Long Ouyang, Juntang Zhuang, Joyce Lee, Yufei Guo, et al. Improving image generation with better captions. *Computer Science. https://cdn. openai. com/papers/dall-e-3. pdf*, 2(3):8, 2023.

Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning. *arXiv preprint arXiv:2305.13301*, 2023.

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.

Daniele Calandriello, 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, 2024. URL https://arxiv.org/abs/2403.08635.

Kevin Clark, Paul Vicol, Kevin Swersky, and David J Fleet. Directly fine-tuning diffusion models on differentiable rewards. *arXiv preprint arXiv:2309.17400*, 2023.

Xiaoliang Dai, Ji Hou, Chih-Yao Ma, Sam Tsai, Jialiang Wang, Rui Wang, Peizhao Zhang, Simon Vandenhende, Xiaofang Wang, Abhimanyu Dubey, et al. Emu: Enhancing image generation models using photogenic needles in a haystack. *arXiv preprint arXiv:2309.15807*, 2023.

Ying Fan, Olivia Watkins, Yuqing Du, Hao Liu, Moonkyung Ryu, Craig Boutilier, Pieter Abbeel, Mohammad Ghavamzadeh, Kangwook Lee, and Kimin Lee. Dpok: Reinforcement learning for fine-tuning text-to-image diffusion models. *Advances in Neural Information Processing Systems*, 36:79858–79885, 2023.

Yi Gu, Zhendong Wang, Yueqin Yin, Yujia Xie, and Mingyuan Zhou. Diffusion-rpo: Aligning diffusion models through relative preference optimization. *arXiv preprint arXiv:2406.06382*, 2024.

Xiaoxuan He, Siming Fu, Wanli Li, Zhiyuan Li, Dacheng Yin, Kang Rong, Fengyun Rao, and Bo Zhang. Sail: Self-amplified iterative learning for diffusion model alignment with minimal human feedback. *arXiv preprint arXiv:2602.05380*, 2026.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models, 2020. URL https://arxiv.org/abs/2006.11239.

Luozhijie Jin, Zijie Qiu, Jie Liu, Zijie Diao, Lifeng Qiao, Ning Ding, Alex Lamb, and Xipeng Qiu. Inference-time alignment control for diffusion models with reinforcement learning guidance. *arXiv preprint arXiv:2508.21016*, 2025.

Sunwoo Kim, Minkyu Kim, and Dongmin Park. Test-time alignment of diffusion models without reward over-optimization, 2025. URL https://arxiv.org/abs/2501.05803.

Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiana, Joe Penna, and Omer Levy. Pick-a-pic: An open dataset of user preferences for text-to-image generation, 2023. URL https://arxiv.org/abs/2305.01569.

Tor Lattimore and Csaba Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020.

Kyungmin Lee, Xiahong Li, Qifei Wang, Junfeng He, Junjie Ke, Ming-Hsuan Yang, Irfan Essa, Jinwoo Shin, Feng Yang, and Yinxiao Li. Calibrated multi-preference optimization for aligning diffusion models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 18465–18475, 2025.

Shufan Li, Konstantinos Kallidromitis, Akash Gokul, Yusuke Kato, and Kazuki Kozuka. Aligning diffusion models by optimizing human utility. *Advances in Neural Information Processing Systems*, 37:24897–24925, 2024.

Yuming Li, Qingyu Li, Chengyu Bai, Xiangyang Luo, Zeyue Xue, Wenyu Qin, Meng Wang, Yikai Wang, and Shanghang Zhang. Aegpo: Adaptive entropy-guided policy optimization for diffusion models. *arXiv preprint arXiv:2602.06825*, 2026.

Zongjian Li, Zheyuan Liu, Qihui Zhang, Bin Lin, Feize Wu, Shenghai Yuan, Zhiyuan Yan, Yang Ye, Wangbo Yu, Yuwei Niu, et al. Uniworld-v2: Reinforce image editing with diffusion negative-aware finetuning and mllm implicit feedback. *arXiv preprint arXiv:2510.16888*, 2025.

Zhanhao Liang, Yuhui Yuan, Shuyang Gu, Bohan Chen, Tiankai Hang, Ji Li, and Liang Zheng. Step-aware preference optimization: Aligning preference with denoising performance at each step. *arXiv preprint arXiv:2406.04314*, 2(5):7, 2024.

Jie Liu, Gongye Liu, Jiajun Liang, Yangguang Li, Jiaheng Liu, Xintao Wang, Pengfei Wan, Di Zhang, and Wanli Ouyang. Flow-grpo: Training flow matching models via online rl. *arXiv preprint arXiv:2505.05470*, 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, 2026. URL https://arxiv.org/abs/2503.10990.

Graham Loomes, Chris Starmer, and Robert Sugden. Observing violations of transitivity by experimental methods. *Econometrica: Journal of the Econometric Society*, pp. 425–439, 1991.

Kenneth O May. Intransitivity, utility, and the aggregation of preference patterns. *Econometrica: Journal of the Econometric Society*, pp. 1–13, 1954.

David C McGarvey. A theorem on the construction of voting paradoxes. *Econometrica: Journal of the Econometric Society*, pp. 608–610, 1953.

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

Tianlong Nan, Xiaopeng Li, Christian Kroer, and Tianyi Lin. Efficient exploration for iterative nash preference optimization, 2026. URL https://arxiv.org/abs/2606.01382.

Robin L Plackett. The analysis of permutations. *Journal of the Royal Statistical Society Series C: Applied Statistics*, 24(2):193–202, 1975.

Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023.

Mihir Prabhudesai, Anirudh Goyal, Deepak Pathak, and Katerina Fragkiadaki. Aligning text-to-image diffusion models with reward backpropagation, 2024. URL https://arxiv.org/abs/2310.03739.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model, 2024. URL https://arxiv.org/abs/2305.18290.

Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1(2):3, 2022.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10684–10695, 2022.

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, 2024. URL https://arxiv.org/abs/2404.03715.

Christoph Schuhmann. Laion-aesthetics. https://laion.ai/blog/laion-aesthetics/, 2022.

Gokul Swamy, Christoph Dann, Rahul Kidambi, Zhiwei Steven Wu, and Alekh Agarwal. A minimaximalist approach to reinforcement learning from human feedback, 2024. URL https://arxiv.org/abs/2401.04056.

Zhiwei Tang, Jiangweizhi Peng, Jiasheng Tang, Mingyi Hong, Fan Wang, and Tsung-Hui Chang. Tuning-free alignment of diffusion models with direct noise optimization. In *ICML 2024 Workshop on Structured Probabilistic Inference {\&} Generative Modeling*, 2024.

Amos Tversky. Intransitivity of preferences. *Psychological Review*, 76(1):31–48, 1969. doi: 10.1037/h0026750.

Bram Wallace, Meihua Dang, Rafael Rafailov, Linqi Zhou, Aaron Lou, Senthil Purushwalkam, Stefano Ermon, Caiming Xiong, Shafiq Joty, and Nikhil Naik. Diffusion model alignment using direct preference optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8228–8238, 2024.

Fu-Yun Wang, Yunhao Shui, Jingtan Piao, Keqiang Sun, and Hongsheng Li. Diffusion-npo: Negative preference optimization for better preference aligned generation of diffusion models. *arXiv preprint arXiv:2505.11245*, 2025a.

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, 2025b. URL https://arxiv.org/abs/2410.16714.

Fang Wu, Xu Huang, Weihao Xuan, Zhiwei Zhang, Yijia Xiao, Guancheng Wan, Xiaomin Li, Bing Hu, Peng Xia, Jure Leskovec, and Yejin Choi. Multiplayer nash preference optimization, 2026. URL https://arxiv.org/abs/2509.23102.

Xiaoshi Wu, Yiming Hao, Keqiang Sun, Yixiong Chen, Feng Zhu, Rui Zhao, and Hongsheng Li. Human preference score v2: A solid benchmark for evaluating human preferences of text-to-image synthesis, 2023. URL https://arxiv.org/abs/2306.09341.

Yue Wu, Zhiqing Sun, Huizhuo Yuan, Kaixuan Ji, Yiming Yang, and Quanquan Gu. Self-play preference optimization for language model alignment, 2024. URL https://arxiv.org/abs/2405.00675.

Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. Imagereward: Learning and evaluating human preferences for text-to-image generation. *Advances in Neural Information Processing Systems*, 36:15903–15935, 2023.

Zeyue Xue, Jie Wu, Yu Gao, Fangyuan Kong, Lingting Zhu, Mengzhao Chen, Zhiheng Liu, Wei Liu, Qiushan Guo, Weilin Huang, et al. Dancegrpo: Unleashing grpo on visual generation. *arXiv preprint arXiv:2505.07818*, 2025.

Kai Yang, Jian Tao, Jiafei Lyu, Chunjiang Ge, Jiaxin Chen, Weihan Shen, Xiaolong Zhu, and Xiu Li. Using human feedback to fine-tune diffusion models without any reward model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8941–8951, 2024a.

Kai Yang, Jian Tao, Jiafei Lyu, Chunjiang Ge, Jiaxin Chen, Weihan Shen, Xiaolong Zhu, and Xiu Li. Using human feedback to fine-tune diffusion models without any reward model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8941–8951, 2024b.

Po-Hung Yeh, Kuang-Huei Lee, and Jun-Cheng Chen. Training-free diffusion model alignment with sampling demons. *arXiv preprint arXiv:2410.05760*, 2024.

Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, Ben Hutchinson, Wei Han, Zarana Parekh, Xin Li, Han Zhang, Jason Baldridge, and Yonghui Wu. Scaling autoregressive models for content-rich text-to-image generation, 2022a. URL https://arxiv.org/abs/2206.10789.

Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, et al. Scaling autoregressive models for content-rich text-to-image generation. *arXiv preprint arXiv:2206.10789*, 2(3):5, 2022b.

Huizhuo Yuan, Zixiang Chen, Kaixuan Ji, and Quanquan Gu. Self-play fine-tuning of diffusion models for text-to-image generation, 2024. URL https://arxiv.org/abs/2402.10210.

Kevin Zhai, Utsav Singh, Anirudh Thatipelli, Souradip Chakraborty, Anit Kumar Sahu, Furong Huang, Amrit Singh Bedi, and Mubarak Shah. Mira: Towards mitigating reward hacking in inference-time alignment of t2i diffusion models. *arXiv preprint arXiv:2510.01549*, 2025.

Daoan Zhang, Guangchen Lan, Dong-Jun Han, Wenlin Yao, Xiaoman Pan, Hongming Zhang, Mingxiao Li, Pengcheng Chen, Yu Dong, Christopher Brinton, and Jiebo Luo. Seppo: Semi-policy preference optimization for diffusion alignment, 2025a. URL https://arxiv.org/abs/2410.05255.

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

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. *arXiv preprint arXiv:2502.16852*, 2025b.

Kaiwen Zheng, Huayu Chen, Haotian Ye, Haoxiang Wang, Qinsheng Zhang, Kai Jiang, Hang Su, Stefano Ermon, Jun Zhu, and Ming-Yu Liu. Diffusionnft: Online diffusion reinforcement with forward process. *arXiv preprint arXiv:2509.16117*, 2025.

Runlong Zhou, Maryam Fazel, and Simon S Du. Extragradient preference optimization (egpo): Beyond last-iterate convergence for nash learning from human feedback. *arXiv preprint arXiv:2503.08942*, 2025.

Huaisheng Zhu, Teng Xiao, and Vasant Honavar. Dspo: Direct score preference optimization for diffusion model alignment. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 51112–51133, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/7f70331dbe58ad59d83941dfa7d975aa-Paper-Conference.pdf.

# A DIFF.-NPO PSEUDO-CODE

The Pseudo-code of Diff.-NPO is shown in Algorithm 1.

**Algorithm 1** Diffusion-NPO

**Input:** steps $S$, KL regularization $\tau$, OMD parameter $\eta$, reference policy $p_{\text{ref}}$, prompts $\mathbf{x} \sim d_0$
Initialize $p_1 \leftarrow p_{\text{ref}}$
**for** $s = 1, 2, \ldots, S$ **do**
  Sample a minibatch of prompts $\{\mathbf{x}^{(i)}\}_{i=1}^{n} \sim d_0$
  For each $\mathbf{x}^{(i)}$, draw a batch of size $k$, denoted by $(\mathbf{y}_1^{(i)}, \ldots, \mathbf{y}_k^{(i)}) \sim p_s(\cdot \mid \mathbf{x}^{(i)})$
  Query the average ranking oracle to obtain scores $(r_1^{(i)}, \ldots, r_k^{(i)})$ for $(\mathbf{y}_1^{(i)}, \ldots, \mathbf{y}_k^{(i)})$
  Rank the samples by their oracle scores and set
  $\mathbf{y}^{+,(i)} \leftarrow \arg\max_{\mathbf{y}_j^{(i)}} r_j^{(i)}$ and $\mathbf{y}^{-,(i)} \leftarrow \arg\min_{\mathbf{y}_j^{(i)}} r_j^{(i)}$
  Form the preference pairs $\{(\mathbf{x}^{(i)}, \mathbf{y}^{+,(i)}, \mathbf{y}^{-,(i)})\}_{i=1}^{n}$
  Update the diffusion policy by minimizing the loss with objective as equation 11:
  $p_{s+1} \leftarrow \arg\min_{p \in \mathcal{P}} L_{\text{Diff.-NPO}}$
  Implement soft update for the target policy:
  $p_{s+1} \leftarrow \lambda p_s + (1-\lambda) p_{s+1}$ where $\lambda = \min(0.001 * s, 0.5)$
**end for**
**Output:** $p_{S+1}$

# B RELATED WORK

## B.1 NASH LEARNING WITH HUMAN FEEDBACK

In the Nash learning with human feedback paradigm, a well-aligned policy is viewed as an optimal strategy that cannot be exploited by competing policies. NLHF(Munos et al., 2024) first introduced this game-theoretic perspective by formulating preference alignment as regularized Nash equilibrium learning under general, potentially non-transitive preferences. Iterative Nash Policy Optimization (INPO) (Zhang et al., 2024) develops a no-regret learning approach to general preference alignment. Direct Nash Optimization (DNO) (Rosset et al., 2024) uses a batched on-policy, regression-based objective to enable iterative self-improvement from general preference feedback. Other work explores optimistic mirror descent (Zhang et al., 2025b) and extragradient updates (Zhou et al., 2025), advancing theoretical guarantees and empirical stability. Nan et al. (2026) study explicit exploration in iterative NLHF, combining SFT-based regularization with adversarial policy exploration to obtain regret guarantees. More recently, Multiplayer Nash Preference Optimization (MNPO)(Wu et al., 2026) extends this line of work from two-player interactions to the multiplayer setting, enabling the modeling of richer preference structures.

## B.2 DIFFUSION MODEL ALIGNMENT

Diffusion models (Rombach et al., 2022; Podell et al., 2023) have achieved remarkable success in T2I generation. Recently, alignment methods inspired by RLHF have been increasingly adopted for diffusion models, both in direct fine-tuning (Dai et al., 2023; Betker et al., 2023; Clark et al., 2023; Prabhudesai et al., 2024; Fan et al., 2023; Black et al., 2023) and at inference time (Tang et al., 2024; Yeh et al., 2024; Kim et al., 2025; Zhai et al., 2025; Jin et al., 2025), with the goal of improving downstream reward-related objectives (Liu et al., 2025; Wallace et al., 2024; Xue et al., 2025). Existing diffusion alignment methods can be broadly grouped into three lines: (1) methods that optimize explicit reward functions, e.g., GRPO-based approaches (Xue et al., 2025; Li et al., 2025; 2026); (2) methods that optimize implicit rewards induced from pairwise preferences, such as Diffusion-DPO (Liang et al., 2024; Wang et al., 2025a; Lee et al., 2025; Gu et al., 2024; Yang et al., 2024a); and (3) likelihood-free methods that optimize forward scores and velocity fields directly (Zhu et al., 2025; Zheng et al., 2025). Despite their differences, most of these methods primarily focus on reward maximization, whereas only a few exploit on-policy self-play. In particular, Yang et al. (2024a) extends DPO to diffusion models by formulating denoising as a multi-step MDP

and directly optimizing human preferences over sampled image pairs. Yuan et al. (2024) introduces a self-play fine-tuning scheme in which the current model learns to outperform samples generated by earlier checkpoints. Zhang et al. (2025a) reduces reliance on human-labeled losing images by replacing them with reference samples generated from previous checkpoints. He et al. (2026) further constructs synthetic preference pairs by self-ranking on-policy generated images and then applies DPO-style optimization. Building on INPO (Zhang et al., 2024), we formulate a mirror-descent update over diffusion trajectories and use it to construct a tractable logistic training objective based on denoising errors.

# C Mathematical Analysis

## C.1 Derivation of the OMD Update for Nash Policy

The following derivation is adapted from Diffusion-DPO (Wallace et al., 2024). Given the preference probability $\mathbb{P}(\mathbf{y}_0 \succ \mathbf{y}_0' \mid \mathbf{x})$, define the loss at step $s$ as

$$\begin{aligned}
\ell_s(p) &:= -\mathbb{E}_{\mathbf{x}\sim d_0}\, \mathbb{E}_{\mathbf{y}_0\sim p(\mathbf{y}_0|\mathbf{x}),\, \mathbf{y}_0'\sim p_s(\mathbf{y}_0|\mathbf{x})}\big[\mathbb{P}(\mathbf{y}_0 \succ \mathbf{y}_0' \mid \mathbf{x})\big] + \tau\, \mathrm{KL}\big(p(\mathbf{y}_0 \mid \mathbf{x}) \,\|\, p_{\mathrm{ref}}(\mathbf{y}_0 \mid \mathbf{x})\big) \\
&\le -\mathbb{E}_{\mathbf{x}\sim d_0}\, \mathbb{E}_{\mathbf{y}_0\sim p(\mathbf{y}_0|\mathbf{x}),\, \mathbf{y}_0'\sim p_s(\mathbf{y}_0|\mathbf{x})}\big[\mathbb{P}(\mathbf{y}_0 \succ \mathbf{y}_0' \mid \mathbf{x})\big] + \tau\, \mathrm{KL}\big(p(\mathbf{y}_{0:T} \mid \mathbf{x}) \,\|\, p_{\mathrm{ref}}(\mathbf{y}_{0:T} \mid \mathbf{x})\big) \\
&= -\mathbb{E}_{\mathbf{x}\sim d_0}\, \mathbb{E}_{\mathbf{y}_{0:T}\sim p(\mathbf{y}_{0:T}|\mathbf{x}),\, \mathbf{y}_{0:T}'\sim p_s(\mathbf{y}_{0:T}|\mathbf{x})}\big[\mathbb{P}(\mathbf{y}_{0:T} \succ \mathbf{y}_{0:T}' \mid \mathbf{x})\big] \\
&\qquad\qquad + \tau\, \mathrm{KL}\big(p(\mathbf{y}_{0:T} \mid \mathbf{x}) \,\|\, p_{\mathrm{ref}}(\mathbf{y}_{0:T} \mid \mathbf{x})\big),
\end{aligned}$$

Since $\mathrm{KL}\big(p(\mathbf{y}_0 \mid \mathbf{x}) \,\|\, p_{\mathrm{ref}}(\mathbf{y}_0 \mid \mathbf{x})\big)$ is not directly tractable, we work on its upper bound $\mathrm{KL}\big(p(\mathbf{y}_{0:T} \mid \mathbf{x}) \,\|\, p_{\mathrm{ref}}(\mathbf{y}_{0:T} \mid \mathbf{x})\big)$, which lower-bounds the objective. Starting from the entropy-regularized online mirror descent update,

$$p_{s+1} = \arg\min_{p\in\mathcal{P}} \Big\langle \nabla \ell_s(p_s), p \Big\rangle + \eta\, \mathrm{KL}(p \| p_s),$$

The objective can be written (for a fixed prompt $\mathbf{x}$) as

$$\begin{aligned}
&\min_p \sum_{\mathbf{y}_{0:T}} p(\mathbf{y}_{0:T} \mid \mathbf{x})\, \nabla_{p_s(\mathbf{y}_{0:T})} \ell_s(p_s) + \eta \sum_{\mathbf{y}_{0:T}} p(\mathbf{y}_{0:T} \mid \mathbf{x}) \log \frac{p(\mathbf{y}_{0:T} \mid \mathbf{x})}{p_s(\mathbf{y}_{0:T} \mid \mathbf{x})}, \\
&\text{s.t.} \sum_{\mathbf{y}_{0:T}} p(\mathbf{y}_{0:T} \mid \mathbf{x}) = 1.
\end{aligned}$$

Introducing a Lagrange multiplier $\lambda$ for the normalization constraint, the Lagrangian is

$$\begin{aligned}
\mathcal{L}(p, \lambda) = \sum_{\mathbf{y}_{0:T}} p(\mathbf{y}_{0:T} \mid \mathbf{x})\, \nabla_{p_s(\mathbf{y}_{0:T})} \ell_s(p_s) + \eta \sum_{\mathbf{y}_{0:T}} p(\mathbf{y}_{0:T} \mid \mathbf{x}) \log \frac{p(\mathbf{y}_{0:T} \mid \mathbf{x})}{p_s(\mathbf{y}_{0:T} \mid \mathbf{x})} \\
+ \lambda\Big(\sum_{\mathbf{y}_{0:T}} p(\mathbf{y}_{0:T} \mid \mathbf{x}) - 1\Big).
\end{aligned}$$

When the derivative of $\mathcal{L}(p, \lambda)$ with respect to $p(\mathbf{y}_{0:T} \mid \mathbf{x})$ goes to zero, we get

$$\log \tfrac{p_{s+1}(\mathbf{y}_{0:T}|\mathbf{x})}{p_s(\mathbf{y}_{0:T}|\mathbf{x})} = -\tfrac{1}{\eta} \nabla_{p_s(\mathbf{y}_{0:T})} \ell_s(p_s) - 1 - \tfrac{\lambda}{\eta},$$

after exponentiating the left and right sides:

$$\begin{aligned}
p_{s+1}(\mathbf{y}_{0:T} \mid \mathbf{x}) &\propto p_s(\mathbf{y}_{0:T} \mid \mathbf{x}) \exp\Big(-\tfrac{1}{\eta} \nabla_{p_s(\mathbf{y}_{0:T})} \ell_s(p_s)\Big), \\
&\propto p_s(\mathbf{y}_{0:T} \mid \mathbf{x}) \exp\Big(\tfrac{1}{\eta}\, \mathbb{P}(\mathbf{y}_{0:T} \succ p_s \mid \mathbf{x})\Big) \Big(\tfrac{p_{\mathrm{ref}}(\mathbf{y}_{0:T}|\mathbf{x})}{p_s(\mathbf{y}_{0:T}|\mathbf{x})}\Big)^{\frac{\tau}{\eta}},
\end{aligned}$$

which can be collected as

$$p_{s+1}(\mathbf{y}_{0:T} \mid \mathbf{x}) \propto \exp\Big(\tfrac{1}{\eta}\, \mathbb{P}(\mathbf{y}_{0:T} \succ p_s \mid \mathbf{x})\Big)\, p_{\mathrm{ref}}(\mathbf{y}_{0:T} \mid \mathbf{x})^{\frac{\tau}{\eta}}\, p_s(\mathbf{y}_{0:T} \mid \mathbf{x})^{1-\frac{\tau}{\eta}}.$$

### C.2 Derivation of Final Objective (Equation 11)

To instantiate the full-chain policy objective in Equation 11 for diffusion models, we identify the generic policy notation with the corresponding diffusion reverse process. Specifically, the current policy $\pi$ is parameterized by the trainable diffusion model $p_\theta$, the previous policy $\pi_s$ is denoted by $p_{\text{prev}}$, and the reference policy $\pi_{\text{ref}}$ is denoted by $p_{\text{ref}}$. Under the Markov structure of the diffusion reverse process, the full-chain likelihood decomposes as

$$p_\theta(\mathbf{y}_{0:T} \mid \mathbf{x}) = p(\mathbf{y}_T) \prod_{t=1}^{T} p_\theta(\mathbf{y}_{t-1} \mid \mathbf{y}_t, \mathbf{x}),$$

and similarly for $p_{\text{ref}}$ and $p_{\text{prev}}$. Therefore, the log-ratio over full trajectories in Equation 11 can be decomposed into a sum of per-step reverse-transition log-ratios. Following the standard diffusion-DPO approximation, we sample a timestep $t \sim \mathcal{U}(0, T)$, replace the intractable reverse-chain sampling with the forward noising process $q(\mathbf{y}_t \mid \mathbf{y}_0)$, and optimize the resulting single-timestep objective with the factor $T$ accounting for uniform timestep sampling.

We start with Equation 11, the objective on reverse decomposition for current, reference, and previous policies,

$$\begin{aligned}
&\mathbb{E}_{\mathbf{y}^+_{0:T}, \mathbf{y}^-_{0:T} \sim \pi_t(\cdot)} \Big[ -\log \sigma \Big( \frac{\tau\beta}{\eta} \Big( \log \frac{\pi(\mathbf{y}^+_{0:T})}{\pi(\mathbf{y}^-_{0:T})} - \log \frac{\pi_{\text{ref}}(\mathbf{y}^+_{0:T})}{\pi_{\text{ref}}(\mathbf{y}^-_{0:T})} \Big) \\
&\qquad\qquad + \frac{(\eta-\tau)\beta}{\eta} \Big( \log \frac{\pi(\mathbf{y}^+_{0:T})}{\pi(\mathbf{y}^-_{0:T})} - \log \frac{\pi_t(\mathbf{y}^+_{0:T})}{\pi_t(\mathbf{y}^-_{0:T})} \Big) \Big) \Big] \\
=&\mathbb{E}_{\mathbf{y}^+_{1:T} \sim q(\mathbf{y}^+_{1:T} \mid \mathbf{y}^+_0), \mathbf{y}^-_{1:T} \sim q(\mathbf{y}^-_{1:T} \mid \mathbf{y}^-_0)} \Big[ -\log \sigma \Big( \frac{\tau\beta}{\eta} \Big( \log \frac{p_\theta(\mathbf{y}^+_{0:T})}{p_\theta(\mathbf{y}^-_{0:T})} - \log \frac{p_{\text{ref}}(\mathbf{y}^+_{0:T})}{p_{\text{ref}}(\mathbf{y}^-_{0:T})} \Big) \\
&\qquad\qquad + \frac{(\eta-\tau)\beta}{\eta} \Big( \log \frac{p_\theta(\mathbf{y}^+_{0:T})}{p_\theta(\mathbf{y}^-_{0:T})} - \log \frac{p_{\text{prev}}(\mathbf{y}^+_{0:T})}{p_{\text{prev}}(\mathbf{y}^-_{0:T})} \Big) \Big) \Big],
\end{aligned}$$

where we use $q(\mathbf{y}_{1:T} \mid \mathbf{y}_0)$ to approximate $p_{\text{prev}}(\mathbf{y}_{1:T} \mid \mathbf{y}_0)$ for efficient training. The objective hence becomes,

$$\begin{aligned}
L =&\mathbb{E}_{\substack{\mathbf{y}^+_{1:T} \sim q(\mathbf{y}^+_{1:T} \mid \mathbf{y}^+_0), \\ \mathbf{y}^-_{1:T} \sim q(\mathbf{y}^-_{1:T} \mid \mathbf{y}^-_0)}} \Big[ -\log \sigma \Big( \frac{\tau\beta}{\eta} \Big( \log \frac{p_\theta(\mathbf{y}^+_{0:T})}{p_\theta(\mathbf{y}^-_{0:T})} - \log \frac{p_{\text{ref}}(\mathbf{y}^+_{0:T})}{p_{\text{ref}}(\mathbf{y}^-_{0:T})} \Big) \\
&+ \frac{(\eta-\tau)\beta}{\eta} \Big( \log \frac{p_\theta(\mathbf{y}^+_{0:T})}{p_\theta(\mathbf{y}^-_{0:T})} - \log \frac{p_{\text{prev}}(\mathbf{y}^+_{0:T})}{p_{\text{prev}}(\mathbf{y}^-_{0:T})} \Big) \Big) \Big] \\
=&\mathbb{E}_{\substack{\mathbf{y}^+_{1:T} \sim q(\mathbf{y}^+_{1:T} \mid \mathbf{y}^+_0), \\ \mathbf{y}^-_{1:T} \sim q(\mathbf{y}^-_{1:T} \mid \mathbf{y}^-_0)}} \Big[ -\log \sigma \Big( \sum_{t=0}^{T-1} \frac{\tau\beta}{\eta} \Big( \log \frac{p_\theta(\mathbf{y}^+_{t-1} \mid \mathbf{y}^+_t)}{p_\theta(\mathbf{y}^-_{t-1} \mid \mathbf{y}^-_t)} - \log \frac{p_{\text{ref}}(\mathbf{y}^+_{t-1} \mid \mathbf{y}^+_t)}{p_{\text{ref}}(\mathbf{y}^-_{t-1} \mid \mathbf{y}^-_t)} \Big) \\
&+ \frac{(\eta-\tau)\beta}{\eta} \Big( \log \frac{p_\theta(\mathbf{y}^+_{t-1} \mid \mathbf{y}^+_t)}{p_\theta(\mathbf{y}^-_{t-1} \mid \mathbf{y}^-_t)} - \log \frac{p_{\text{prev}}(\mathbf{y}^+_{t-1} \mid \mathbf{y}^+_t)}{p_{\text{prev}}(\mathbf{y}^-_{t-1} \mid \mathbf{y}^-_t)} \Big) \Big) \Big] \\
=&\mathbb{E}_{\substack{\mathbf{y}^+_{1:T} \sim q(\mathbf{y}^+_{1:T} \mid \mathbf{y}^+_0), \\ \mathbf{y}^-_{1:T} \sim q(\mathbf{y}^-_{1:T} \mid \mathbf{y}^-_0)}} \Big[ -\log \sigma \Big( T\mathbb{E}_t \Big[ \frac{\tau\beta}{\eta} \Big( \log \frac{p_\theta(\mathbf{y}^+_{t-1} \mid \mathbf{y}^+_t)}{p_\theta(\mathbf{y}^-_{t-1} \mid \mathbf{y}^-_t)} - \log \frac{p_{\text{ref}}(\mathbf{y}^+_{t-1} \mid \mathbf{y}^+_t)}{p_{\text{ref}}(\mathbf{y}^-_{t-1} \mid \mathbf{y}^-_t)} \Big) \\
&+ \frac{(\eta-\tau)\beta}{\eta} \Big( \log \frac{p_\theta(\mathbf{y}^+_{t-1} \mid \mathbf{y}^+_t)}{p_\theta(\mathbf{y}^-_{t-1} \mid \mathbf{y}^-_t)} - \log \frac{p_{\text{prev}}(\mathbf{y}^+_{t-1} \mid \mathbf{y}^+_t)}{p_{\text{prev}}(\mathbf{y}^-_{t-1} \mid \mathbf{y}^-_t)} \Big) \Big] \Big) \Big] \\
=&\mathbb{E}_{\substack{\mathbf{y}^+_{t,t-1} \sim q(\mathbf{y}^+_{t,t-1} \mid \mathbf{y}^+_0), \\ \mathbf{y}^-_{t,t-1} \sim q(\mathbf{y}^-_{t,t-1} \mid \mathbf{y}^-_0)}} \Big[ -\log \sigma \Big( T\mathbb{E}_t \Big[ \frac{\tau\beta}{\eta} \Big( \log \frac{p_\theta(\mathbf{y}^+_{t-1} \mid \mathbf{y}^+_t)}{p_\theta(\mathbf{y}^-_{t-1} \mid \mathbf{y}^-_t)} \\
&- \log \frac{p_{\text{ref}}(\mathbf{y}^+_{t-1} \mid \mathbf{y}^+_t)}{p_{\text{ref}}(\mathbf{y}^-_{t-1} \mid \mathbf{y}^-_t)} \Big) \\
&+ \frac{(\eta-\tau)\beta}{\eta} \Big( \log \frac{p_\theta(\mathbf{y}^+_{t-1} \mid \mathbf{y}^+_t)}{p_\theta(\mathbf{y}^-_{t-1} \mid \mathbf{y}^-_t)} - \log \frac{p_{\text{prev}}(\mathbf{y}^+_{t-1} \mid \mathbf{y}^+_t)}{p_{\text{prev}}(\mathbf{y}^-_{t-1} \mid \mathbf{y}^-_t)} \Big) \Big] \Big) \Big].
\end{aligned}$$

By applying Jensen's inequality, we can lower bound the objective as

$$
\begin{aligned}
L \leq \mathbb{E}_{\substack{t,\, \mathbf{y}_t^+ \sim q(\mathbf{y}_t^+|\mathbf{y}_0^+),\\ \mathbf{y}_t^- \sim q(\mathbf{y}_t^-|\mathbf{y}_0^-)}} \Big[ -\log\sigma\Big( T\mathbb{E}_{\substack{\mathbf{y}_{t-1}^+ \sim q(\mathbf{y}_{t-1}^+|\mathbf{y}_t^+, \mathbf{y}_0^+),\\ \mathbf{y}_{t-1}^- \sim q(\mathbf{y}_{t-1}^-|\mathbf{y}_t^-, \mathbf{y}_0^-)}} \Big[ \frac{\tau\beta}{\eta}\Big( \log\frac{p_\theta(\mathbf{y}_{t-1}^+|\mathbf{y}_t^+)}{p_\theta(\mathbf{y}_{t-1}^-|\mathbf{y}_t^-)} \\
- \log\frac{p_{\text{ref}}(\mathbf{y}_{t-1}^+|\mathbf{y}_t^+)}{p_{\text{ref}}(\mathbf{y}_{t-1}^-|\mathbf{y}_t^-)}\Big) + \frac{(\eta-\tau)\beta}{\eta}\Big( \log\frac{p_\theta(\mathbf{y}_{t-1}^+|\mathbf{y}_t^+)}{p_\theta(\mathbf{y}_{t-1}^-|\mathbf{y}_t^-)} - \log\frac{p_{\text{prev}}(\mathbf{y}_{t-1}^+|\mathbf{y}_t^+)}{p_{\text{prev}}(\mathbf{y}_{t-1}^-|\mathbf{y}_t^-)}\Big)\Big]\Big)\Big] \\
= \mathbb{E}_{\substack{t,\, \mathbf{y}_t^+ \sim q(\mathbf{y}_t^+|\mathbf{y}_0^+),\\ \mathbf{y}_t^- \sim q(\mathbf{y}_t^-|\mathbf{y}_0^-)}} \Big[ -\log\sigma\Big( -T\Big[ \frac{\tau\beta}{\eta}\Big( \mathbb{D}_{\text{KL}}(q(\mathbf{y}_{t-1}^+|\mathbf{y}_{0,t}^+) \| \, p_\theta(\mathbf{y}_{t-1}^+|\mathbf{y}_t^+)) \\
- \mathbb{D}_{\text{KL}}(q(\mathbf{y}_{t-1}^-|\mathbf{y}_{0,t}^-) \| \, p_\theta(\mathbf{y}_{t-1}^-|\mathbf{y}_t^-)) \\
- \mathbb{D}_{\text{KL}}(q(\mathbf{y}_{t-1}^+|\mathbf{y}_{0,t}^+) \| \, p_{\text{ref}}(\mathbf{y}_{t-1}^+|\mathbf{y}_t^+)) + \mathbb{D}_{\text{KL}}(q(\mathbf{y}_{t-1}^-|\mathbf{y}_{0,t}^-) \| \, p_{\text{ref}}(\mathbf{y}_{t-1}^-|\mathbf{y}_t^-))\Big) \\
+ \frac{(\eta-\tau)\beta}{\eta}\Big( \mathbb{D}_{\text{KL}}(q(\mathbf{y}_{t-1}^+|\mathbf{y}_{0,t}^+) \| \, p_\theta(\mathbf{y}_{t-1}^+|\mathbf{y}_t^+)) - \mathbb{D}_{\text{KL}}(q(\mathbf{y}_{t-1}^-|\mathbf{y}_{0,t}^-) \| \, p_\theta(\mathbf{y}_{t-1}^-|\mathbf{y}_t^-)) \\
- \mathbb{D}_{\text{KL}}(q(\mathbf{y}_{t-1}^+|\mathbf{y}_{0,t}^+) \| \, p_{\text{prev}}(\mathbf{y}_{t-1}^+|\mathbf{y}_t^+)) + \mathbb{D}_{\text{KL}}(q(\mathbf{y}_{t-1}^-|\mathbf{y}_{0,t}^-) \| \, p_{\text{prev}}(\mathbf{y}_{t-1}^-|\mathbf{y}_t^-))\Big)\Big]\Big)\Big].
\end{aligned}
$$

Since the distribution of reverse step $t$ of diffusion models $p_\theta(\mathbf{y}_0)$ is

$$
p_\theta(\mathbf{y}_{t-1}|\mathbf{y}_t) = \mathcal{N}(\mathbf{y}_{t-1}; \sqrt{\frac{\alpha_{t-1}}{\alpha_t}}\left(\mathbf{y}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon(\mathbf{y}_t)\right), \sigma_t^2 I),
$$

we can parameterize the loss as

$$
\begin{aligned}
L \leq \mathbb{E}_{\substack{t,\, \mathbf{y}_t^+ \sim q(\mathbf{y}_t^+|\mathbf{y}_0^+),\\ \mathbf{y}_t^- \sim q(\mathbf{y}_t^-|\mathbf{y}_0^-)}} \Big[ -\log\sigma\Big( -T\Big[ \frac{\tau\beta}{\eta}\Big( \|\epsilon^+ - \epsilon_\theta(\mathbf{y}_t^+, t)\|^2 - \|\epsilon^- - \epsilon_\theta(\mathbf{y}_t^-, t)\|^2 \\
- \|\epsilon^+ - \epsilon_{\text{ref}}(\mathbf{y}_t^+, t)\|^2 + \|\epsilon^- - \epsilon_{\text{ref}}(\mathbf{y}_t^-, t)\|^2\Big) \\
+ \frac{(\eta-\tau)\beta}{\eta}\Big( \|\epsilon^+ - \epsilon_\theta(\mathbf{y}_t^+, t)\|^2 - \|\epsilon^- - \epsilon_\theta(\mathbf{y}_t^-, t)\|^2 \\
- \|\epsilon^+ - \epsilon_{\text{prev}}(\mathbf{y}_t^+, t)\|^2 + \|\epsilon^- - \epsilon_{\text{prev}}(\mathbf{y}_t^-, t)\|^2\Big)\Big]\Big)\Big]
\end{aligned}
$$

# D Additional Experimental Details

## D.1 Datasets

**Pick-a-Pic.** Pick-a-Pic is a large-scale public dataset of human preferences for text-to-image generation Kirstain et al. (2023) with over 500,000 examples. It is collected through a web interface in which users provide prompts, compare multiple generated images, and indicate which image they prefer, or whether the comparison results in a tie. Each example therefore consists of a prompt, two generated images, and a preference label. We use Pick-a-Pic v1 for training and its held-out test split for evaluation.

**PartiPrompts.** PartiPrompts is a curated benchmark designed to evaluate the compositional and semantic capabilities of text-to-image models Yu et al. (2022a). It contains prompts covering a broad range of visual concepts, including objects, attributes, styles, and complex relational descriptions. Owing to this diversity, PartiPrompts has become a standard benchmark for assessing prompt following, compositional generation, and overall image quality. In our experiments, we use PartiPrompts as a standardized evaluation suite for text-to-image generation.

**HPS v2.** We use prompts from the HPSv2 benchmark (Wu et al., 2023) to evaluate our aligned models across four visual styles: animation, concept art, painting, and photo. For each prompt, we generate images using the evaluated models and assess them with the five automatic evaluators described in Section 4.1.

### D.2 Baselines

We compare Diff.-NPO against several representative baselines. The first baseline is the original pretrained diffusion backbone, which provides the starting point for alignment on both SD1.5 and SDXL. We then consider online supervised fine-tuning (SFT) on preferred images, including both offline and online variants. We also compare with online-Diff-DPO in the main tables. In addition, we include two representative self-play preference optimization methods, SPIN (Yuan et al., 2024) and SEPPO (Zhang et al., 2025a), with detailed introductions in paragraphs below. Together, these baselines allow us to compare our method with reference-anchored preference optimization, pure self-play style updates, and simple supervised adaptation.

**SPIN.** SPIN-Diffusion iteratively fine-tunes a diffusion model using a dataset of prompt–image pairs. At each round, it contrasts images from this dataset with images generated by the previous model, using a self-play objective to bring the model closer to the target data distribution. The dataset images provide the positive examples, while the model-generated images provide the negative examples. This construction avoids the need for human-annotated losing images. In comparison, Diff.-NPO obtains both images from the current policy and uses a preference oracle to determine which is preferred. Its updates are guided by improving the expected win rate against the current policy, with regularization toward both that policy and a fixed reference model.

**SePPO.** SePPO combines winning images from an existing dataset with images generated by previous model checkpoints. These generated images replace the human-annotated losing images required by standard preference training. Since a generated image is not necessarily worse than the dataset image, SePPO uses an anchor-based criterion to assess whether it should be treated as a winning or losing example. It also updates the reference checkpoint during training. In comparison, Diff.-NPO generates both images from the current policy and uses a preference oracle to determine their ordering. Its training objective combines the previous policy with a separate, fixed reference model, guided by a Nash mirror-descent update that balances expected win-rate improvement with regularization toward both policies.

### D.3 Implementation Details

We adopt AdamW for training SD1.5 and Adafactor for training SDXL. The learning rate is set to $1 \times 10^{-8}$ with linear warmup and is scaled by the effective batch size. The global batch size is 2048: for pairwise methods, the effective batch contains 2048 pairs. For SD1.5 and SDXL, we set $\beta = 5000$ for Online-Diffusion-DPO and Diff.-NPO. For Diff.-NPO, we fix the ratio $\tau/\eta = 1/2$ in all experiments unless the ablation study. For SEPPO and SPIN, we directly evaluate the official checkpoints released by their authors. All experiments are conducted on eight NVIDIA A100 80GB GPUs.

## E Additional Ablation Results

### E.1 Ablation on $\tau/\eta$ Regularization

Figure 2 presents the complete win-rate comparison for different $\tau/\eta$ values across all five evaluation metrics. Consistent with the main-text results, the performance generally improves from the pure self-play setting and then decreases as the objective approaches the DPO-style setting. This trend suggests that intermediate regularization ratios better balance the stabilizing effect of the reference policy and the online self-improvement signal from the previous policy.

### E.2 Impact of the Objective Design

We further study the importance of the pairwise-logistic objective (PL), $\log \sigma(\cdot)$, by comparing it with the squared distance (SD) used in INPO-style objectives. As discussed in Section 3, the squared distance objective relies on a preference distribution assumption, whereas practical implementations often construct pairs by deterministically ranking generated samples with an oracle. Table 4 shows that the logistic objective achieves higher win rates than the squared-distance objective across all five evaluators, supporting our choice of training loss.

Table 4: Results on the objective design, comparing the squared distance objective (SD) with the pairwise-logistic objective (PL) used in Diff.-NPO.

| Win Rate (%) on Pick-a-Pic | | | | | |
|---|---|---|---|---|---|
| Obj. | PS ↑ | HPS ↑ | CLIP ↑ | IM ↑ | AES ↑ |
| SD | 60.80 | 74.60 | 54.80 | 70.20 | 71.60 |
| PL | **81.60** | **84.64** | **60.32** | **85.12** | **80.76** |

### E.3 Comparison with Offline SFT and DPO

We further compare Diff.-NPO with offline variants of SFT and DPO in Table 5. Offline SFT substantially outperforms online SFT, mainly because the preferred images in the offline Pick-a-Pic dataset are generally of higher quality than the images generated online with a small number of inference steps. Nevertheless, offline SFT still underperforms preference-optimization methods, suggesting that training only on preferred images cannot fully exploit the relative preference information contained in positive–negative pairs. We also observe that offline DPO is weaker than online DPO in this setting, indicating that online model-generated preference pairs better match the current policy distribution. Overall, Diff.-NPO achieves the best win rates across all five metrics, showing that the proposed online Nash-style objective provides stronger preference alignment than both supervised adaptation and standard DPO variants.

Table 5: Comparison between online and offline training variants on SD1.5 Pick-a-Pic, evaluated by win rate (%) against the original model.

| Win Rate (%) on Pick-a-Pic | | | | | |
|---|---|---|---|---|---|
| | PS ↑ | HPS ↑ | CLIP ↑ | IM ↑ | AES ↑ |
| SFT(online) | 40.4 | 46.6 | 55.0 | 53.2 | 59.3 |
| SFT(offline) | 73.4 | 80.7 | 56.3 | 74.9 | 71.7 |
| DPO(online) | 77.4 | 82.5 | 60.0 | 81.0 | 73.4 |
| DPO(offline) | 73.3 | 69.8 | 57.1 | 61.7 | 63.4 |
| Diff.-NPO | **81.6** | **84.6** | **60.3** | **85.1** | **80.8** |

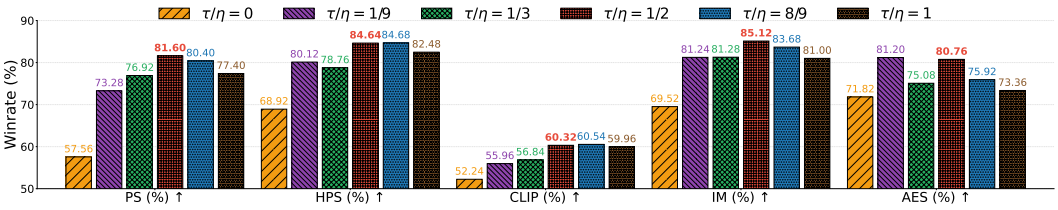



Figure 2: Full win-rate comparison for the ablation study of different $\tau/\eta$ values on Pick-a-Pic for SD1.5. We report win rates across all five evaluation metrics: PickScore, HPSV2, CLIP score, ImageReward, and Aesthetic Score.

## F More Qualitative Results

In this section, we provide qualitative comparisons for both SDXL and SD1.5. As shown in Figures 3, 4, and 5, Diff.-NPO consistently generates images with better semantic alignment, stronger visual coherence, and more appealing overall quality compared with the baselines. These results further support the quantitative improvements reported in the main text.




Figure 3: Qualitative comparison on SDXL across six representative prompts. Rows correspond to the original SDXL model, SFT, Online DPO, and Diff.-NPO. Compared with the baselines, Diff.-NPO produces images that are generally more realistic, semantically faithful, and visually coherent.

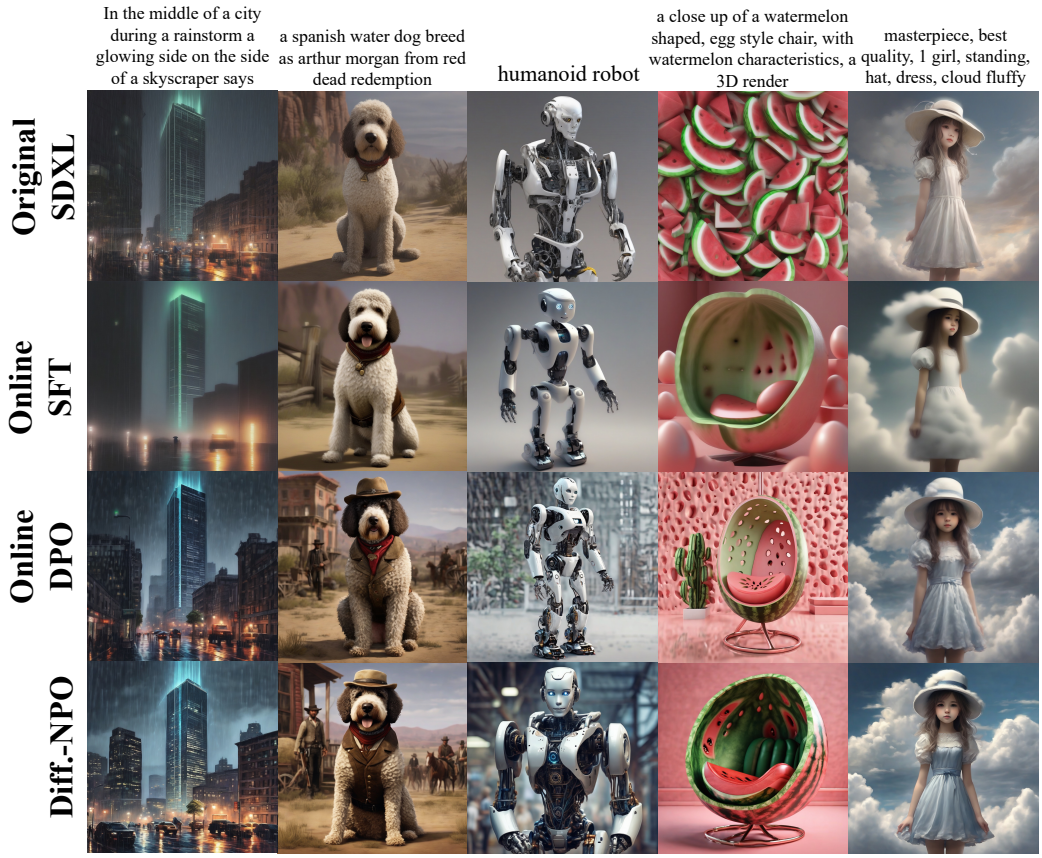



Figure 4: Additional qualitative comparison on SDXL. Compared with the baselines, Diff.-NPO produces images with stronger prompt alignment, more coherent global structure, and improved visual quality across diverse prompts.

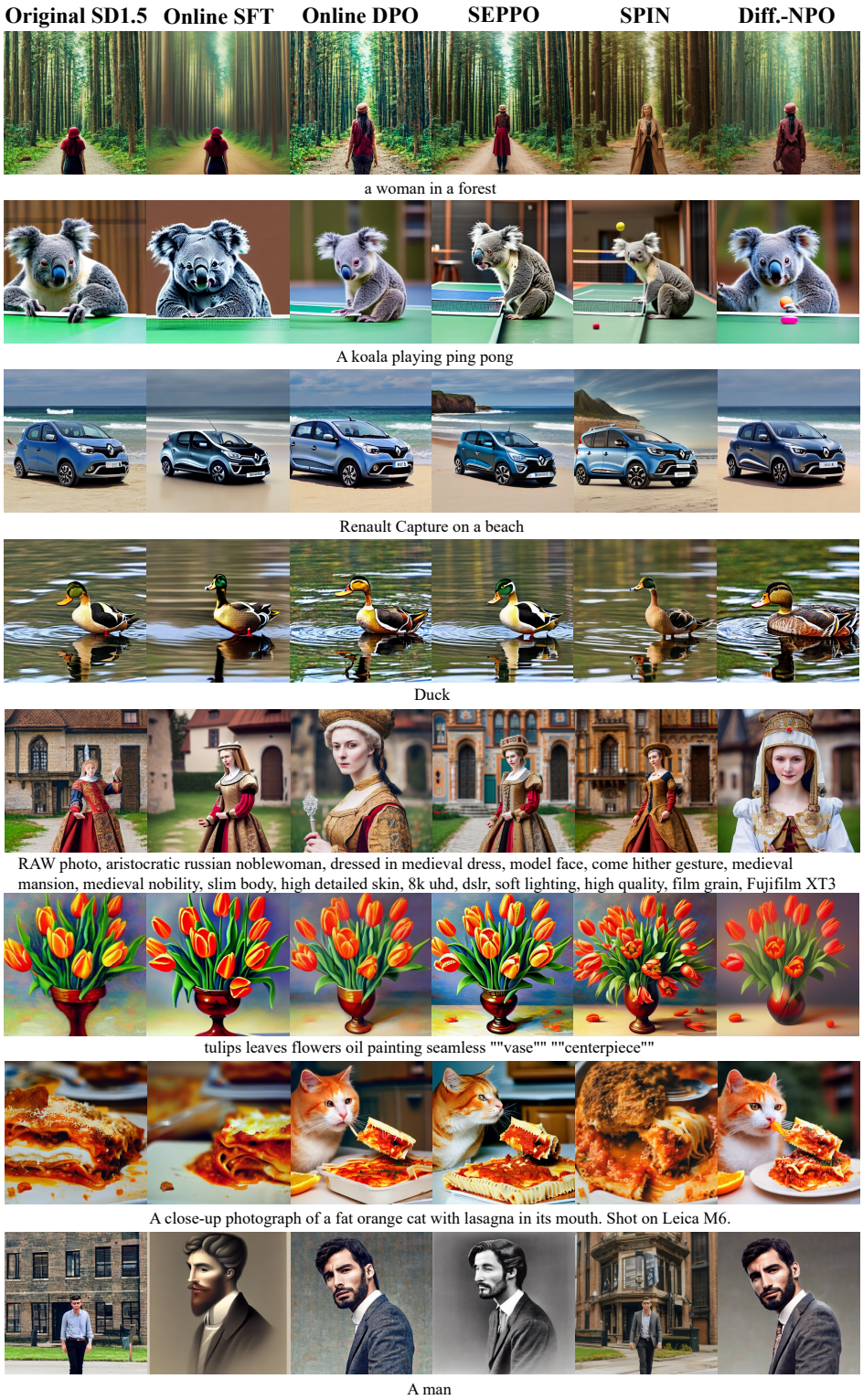



Figure 5: Qualitative comparison on SD1.5. Diff.-NPO consistently improves the realism, semantic faithfulness, and visual coherence of generated images, demonstrating that the benefits of our preference optimization method also transfer to the SD1.5 backbone.