Adaptive Divergence Regularized Policy Optimization for Fine-tuning Generative Models

Anonymous Author(s)
Affiliation
Address
email

Abstract

Balancing exploration and exploitation during reinforcement learning fine-tuning of generative models presents a critical challenge, as existing approaches rely on fixed divergence regularization that creates an inherent dilemma: strong regularization preserves model capabilities but limits reward optimization, while weak regularization enables greater alignment but risks instability or reward hacking. We introduce Adaptive Divergence Regularized Policy Optimization (ADRPO), which automatically adjusts regularization strength based on advantage estimates—reducing regularization for high-value samples while applying stronger regularization to poor samples, enabling policies to navigate between exploration and aggressive exploitation according to data quality. Our implementation with Wasserstein-2 regularization for flow matching generative models achieves remarkable results on text-to-image generation, achieving better semantics alignment and diversity than offline methods like DPO and online methods with fixed regularization like ORW-CFM-W2. ADRPO also enables 2B parameter SD3 model to surpass much larger models with 4.8B and 12B parameters in attribute binding, semantic consistency, artistic style transfer, and compositional control while maintaining generation diversity. ADRPO can also generalize to KL-regularized LLM fine-tuning, enhancing existing online RL methods like GRPO while requiring no additional networks or complex architectural changes. In LLM fine-tuning tasks, we observe that ADRPO even demonstrates an emergent ability to escape local optima by actively increasing exploration to discover superior policies, thus offering an effective, plug-and-play solution to the exploration-exploitation challenge in generative model fine-tuning.

1 Introduction

Reinforcement learning fine-tuning has emerged as a powerful paradigm for aligning generative models with human preferences, driving remarkable improvements in capabilities from text generation to image synthesis [22, 4, 34]. At the core of modern RLHF approaches lies a fundamental challenge: effectively balancing divergence regularization against reward maximization during policy optimization. This balance is critical as it determines whether models retain the beneficial properties of their pre-trained foundation while adapting to better satisfy human preferences [26, 44, 2].

The current standard practice employs divergence regularization with fixed coefficients to constrain policy updates - typically using Kullback-Leibler (KL) [31, 23] or Wasserstein-2 (W2) divergences [1, 13]. However, this approach creates an inherent dilemma that limits performance: strong regularization preserves model capabilities but hampers reward optimization, while weak regularization enables greater reward optimization but risks catastrophic forgetting, mode collapse, or reward hacking [18, 32, 33]. This trade-off is particularly pronounced in generative models where preserving diversity while improving quality represents a critical balance [3, 14]. Existing approaches like PPO

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

[30], GRPO [31], and DPO [26, 37] employ fixed regularization coefficients that treat all data points equally, regardless of whether the policy should prioritize exploitation (when rewards are reliable) or exploration (when falling into suboptimal solutions). This one-size-fits-all approach fails to adapt to the varying exploration needs across the complex landscapes of generative model policy optimization.

To address these limitations, we propose Adaptive Divergence Regularized Policy Optimization (ADRPO), a novel framework that dynamically adjusts regularization strength using advantage estimates [29, 35]. ADRPO employs advantage signals to fine-tune the exploration-exploitation trade-off: high-advantage samples reduce regularization for aggressive optimization, while low-advantage samples increase it for stability. This sample-level adaptation integrates seamlessly into training, providing an efficient and automated approach. By aligning regularization with sample quality, ADRPO overcomes the shortcomings of prior methods, delivering superior alignment and generative performance across diverse tasks, as demonstrated in our experiments with text-to-image alignment and language model fine-tuning. In summary, our approach makes several important contributions:

1. **General RL Framework with Adaptive Divergence Regularization.** We introduce ADRPO as a general-purpose framework that dynamically adjusts regularization based on advantage estimates, integrating with existing RL fine-tuning methods without architectural changes. Our proposed methods address the exploration-exploitation dilemma while preventing reward hacking and model collapse.
2. **Superior Text-to-Image Alignment with Smaller Model.** We first propose a novel online RL method based on ADRPO, combining advantage-based policy optimization and adaptive W2 regularization for fine-tuning flow matching models. Our experiments of fine-tuning SD3 demonstrate ADRPO's dominant Pareto frontier in the reward-diversity trade-off and reward-divergence trade-off compared to DPO [26] and fixed-regularization approaches [13] (See Figs. 2 and 3). Notably, our 2B parameter model outperforms larger 4.8B [38] and 12B [42] parameter models across attribute binding, compositional control, and semantic consistency (See Figs. 1 and Tab. 1).
3. **Emergent Exploration in LLMs.** We also apply our ADRPO to improve GRPO [31] for online fine-tuning of LLMs (See Figs. 4). ADRPO not only improves alignment but exhibits an emergent ability to escape local optima by actively increasing exploration when needed—a capability absent in fixed-regularization methods like GRPO.
4. **Cross-Domain Applicability.** ADRPO provides a unified solution for both continuous (flow matching with W2 regularization) and discrete (LLMs with KL divergence) generative paradigms, offering immediate practical benefits with minimal computational overhead.

2 Related Work

RL Fine-tuning for LLMs. Reinforcement learning has become the dominant approach for aligning large language models with human preferences. Pioneering work by [4] established the RLHF framework, which was later scaled by [22] to create models that better follow human instructions. The algorithmic landscape has evolved from PPO [30] to more efficient alternatives like GRPO [31] and offline approaches like DPO [26]. These methods have significantly improved the reasoning capabilities of models like DeepSeek-R1 [15] and improved their instruction-following abilities. Despite their success, these approaches typically rely on fixed regularization parameters that treat all samples equally, regardless of whether they represent promising directions for optimization or uncertainty-laden regions requiring more conservative updates (See Fig. 4).

RL Fine-tuning for Flow Matching Models. While RL fine-tuning is established for language models, its application to flow matching (FM) models [21] presents unique exploration-exploitation trade-off challenges due to their continuous-time nature and ODE-based sampling. Recent approaches like Online Reward-Weighted Fine-Tuning with Wasserstein regularization [13] and offline methods like diffusion-DPO [37] have made progress, but remain limited by fixed regularization schemes that cannot adapt to sample-specific characteristics. This fundamental limitation restricts their ability to optimally balance the critical exploration-exploitation trade-off necessary for effective fine-tuning of state-of-the-art image generation models like SD3 [12] (See Tab. 1 and Figs. 2).

Divergence Regularization in RL Fine-tuning. Divergence regularization plays a crucial role in RL fine-tuning by preventing the policy from deviating too far from the initial model, thus preserving

desirable properties while allowing for improvement. For language models, KL divergence serves as the standard metric in methods like PPO [30], GRPO [31], and DPO [26], while flow models benefit from Wasserstein distances [13] that better handle continuous distributions. Despite their importance, existing approaches typically employ fixed regularization coefficients that fail to handle the varying significance of regularization across different samples and different learning stages. This limitation can lead to suboptimal trade-offs between preserving model capabilities and maximizing rewards (e.g., GRPO in Fig. 4), risking model collapse [32, 16] or insufficient improvement. Our work addresses this gap through adaptive regularization based on advantage estimates, providing a novel approach to dynamically balancing exploration and exploitation during training.

3 Method

3.1 Problem Formulation

In this paper, we address the challenges of fine-tuning pre-trained generative models through online RL to improve their alignment with human preferences [4, 23]. Given a pre-trained reference policy π_{ref} and its fine-tuned counterpart π_θ parameterized by θ, our objective is to maximize the expected user-defined reward $\mathbb{E}_{x\sim\pi_\theta}[R(x,c)]$, where $R(x,c)$ quantifies human preference for generation x conditioned on context $c \sim p(c)$ (e.g., CLIP Score [24] for T2I tasks). This context may be a text prompt in LLMs [40, 41, 10] or an image description in text-to-image (T2I) models [12, 38, 42]. The standard approach in RL fine-tuning formulates this as a constrained optimization problem:

$$J(\theta) = \mathbb{E}_{x\sim\pi_\theta, c\sim p(c)}[R(x,c)] - \beta \cdot D(\pi_\theta, \pi_{\text{ref}}) \tag{1}$$

Here, $p(c)$ is the sample distribution of prompts (e.g., uniform sampling in our paper), $D(\pi_\theta, \pi_{\text{ref}})$ represents a divergence measure between the fine-tuned and reference policies—typically Kullback-Leibler divergence (KL) for discrete generative models [30, 31, 26] or Wasserstein distance for continuous distributions [36, 1, 13]. The coefficient β controls the trade-off between reward optimization and preservation of the pre-trained model's capabilities (e.g., diversity).

3.2 Adaptive Divergence Regularized Policy Optimization

Recent approaches to online RL fine-tuning of generative models have explored different divergence measures, including W2 regularization in flow matching models [13] and KL divergence in LLMs [31, 23]. However, these methods still rely on fixed regularization schemes that treat all samples equally, regardless of their potential for reward improvement or risk of degradation. This fundamental challenge of adaptive regularization—dynamically balancing exploration and exploitation (See Figs. 3 and 4) at the individual sample level—remains largely unaddressed in the literature.

3.2.1 Conventional RL Fine-tuning Approaches

The conventional RL objective in Equation (1) can be rewritten as a combination of two loss terms:

$$\mathcal{L}_{\text{RLHF}}(\theta) = \mathcal{L}_{\text{RL}}(\theta) + \beta \cdot \mathcal{L}_{\text{D}}(\theta) \tag{2}$$

where $\mathcal{L}_{\text{RL}}(\theta)$ is the policy optimization term such as policy gradient [30] or reward-weighting [13] and $\mathcal{L}_{\text{D}}(\theta) = D(\pi_\theta, \pi_{\text{ref}})$ is the divergence regularization term, such as KL divergence in LLMs [31] or W2 divergence in flow matching models [13]. In practice, this formulation has been instantiated in various ways. For example, Group Relative Policy Optimization (GRPO) [31] employs a KL-regularized policy gradient objective for LLMs:

$$\mathcal{L}_{\text{GRPO}}(\theta) = \mathcal{L}_{\text{PG}}(\theta) + \beta \cdot D_{\text{KL}}(\pi_\theta \| \pi_{\text{ref}}) \tag{3}$$

where $\mathcal{L}_{\text{PG}}(\theta)$ represents a clipped policy gradient loss based on group-level advantage estimation. Similarly, ORW-CFM-W2 [13] applies a W2 regularization term for flow matching models:

$$\mathcal{L}_{\text{ORW-CFM-W2}}(\theta) = \mathcal{L}_{\text{ORW}} + \beta \cdot \mathbb{E}_{c,t,x_t}[|\mathbf{v}_\theta(x_t,t,c) - \mathbf{v}_{\text{ref}}(x_t,t,c)|^2]$$

where $\mathcal{L}_{\text{ORW}} = \mathbb{E}_{c,x_1,t,x_t}[\omega(x_1,c) * |\mathbf{v}_\theta(x_t,t,c) - \mathbf{u}_t|^2]$ is the reward weighted loss and $\mathbf{v}_\theta$ and $\mathbf{v}_{\text{ref}}$ are the velocity fields of the fine-tuned and reference policies, respectively.

Critically, in all these approaches, the regularization strength β remains constant across all samples and training steps, failing to adapt to the varying quality of generated samples.

3.2.2 Our Approach: Advantage-Based Adaptive Regularization

We introduce Adaptive Divergence Regularized Policy Optimization (ADRPO), a principled framework that dynamically adjusts regularization strength based on the estimated advantages of individual samples. The key insight in ADRPO is that the regularization coefficient should not be static, but should vary inversely with the sample's estimated advantage. Formally, we propose:

$$\mathcal{L}_{\text{ADRPO}}(\theta) = \mathcal{L}_{\text{RL}}(\theta) + (\beta_0 - A) \cdot \mathcal{L}_{\text{D}}(\theta) \tag{4}$$

where A is an advantage estimate for the current sample and β_0 is a baseline regularization coefficient. This formulation creates an adaptive regularization coefficient $\beta_{\text{tot}} = \beta_0 - A$ that adapts based on the quality of each sample. This adaptive mechanism creates a natural balance: **1) Exploitation:** in regions where the policy generates high-quality samples (high advantage), ADRPO allows for efficient exploitation by reducing divergence penalties; **2) Exploration:** in uncertain or low-quality regions (low advantage), it enforces stronger regularization to maintain stability and preserve the model's original capabilities (See Figs. 3 and 4).

Based on Equ. (4), our ADRPO can be seamlessly integrated with various existing RL fine-tuning methods. For instance, when applied to GRPO for large language models (LLMs), the objective becomes $\mathcal{L}_{\text{ADRPO-GRPO}}(\theta) = \mathcal{L}_{\text{PG}}(\theta) + (\beta_0 - A_{\text{GRPO}}) \cdot D_{\text{KL}}(\pi_\theta \| \pi_{\text{ref}})$, where A_{GRPO} is the advantage estimate from GRPO's group-based estimation procedure [31, 15].

3.3 ADRPO for Flow Matching Generative Models

We now demonstrate how our ADRPO framework can be effectively applied to fine-tuning flow matching models [21, 36], particularly focusing on text-to-image generation models like SD3 [12].

3.3.1 Flow Matching Preliminaries

Flow matching (FM) models define a continuous-time transformation that maps a simple prior distribution $p(x_0)$ (e.g., Gaussian) to a complex target distribution via a probability flow p_t. An FM model learns a velocity field $\mathbf{v}_\theta(x_t, t, c)$ that approximates the true velocity field $\mathbf{u}_t(x_t|c)$. However, since $\mathbf{u}_t(x_t|c)$ is often intractable [21], Conditional Flow Matching (CFM) [36] proposes an equivalent yet tractable objective by conditioning the flow on target samples x_1 while learning a conditional target velocity field (e.g., $\mathbf{u}_t(x_t|x_1, c) = x_1 - x_0$ for linear interpolation path [21]):

$$\mathcal{L}_{\text{CFM}}(\theta) = \mathbb{E}_{c\sim p(c), t\sim U(0,1), x_1\sim p_{\text{data}}(x|c), x_t\sim p_t(x_t|x_1,c)}[|\mathbf{v}_\theta(x_t, t, c) - \mathbf{u}_t(x_t|x_1, c)|^2] \tag{5}$$

Given a pre-trained reference model like SD3 [12], flow matching fine-tuning aims to align generations with human preferences while preserving generative diversity. Traditional approaches, including supervised fine-tuning and offline RL methods like DPO [26, 37], sample target states $x_1 \sim p_{\text{data}}(x|c)$ from a fixed human-curated dataset—a stable but limiting approach that restricts exploration of potentially better policy regions (See Figs. 2 and 3). In contrast, our proposed ADRPO framework embraces an online RL paradigm, sampling target states from the fine-tuned policy itself: $x_1 \sim p_\theta^{n-1}(x|c)$, with p_θ^{n-1} representing the policy at the previous iteration. This online sampling strategy enables the model to continuously improve upon its own generations and explore the policy space more effectively but is prone to collapse [13], while our adaptive regularization mechanism specifically addresses the inherent instability and exploration-exploitation dilemma in online RL fine-tuning.

3.3.2 ADRPO with Wasserstein Regularization

A key observation across RL fine-tuning methods [31, 13, 23] is that effective policy optimization requires differentially weighting samples based on quality (e.g., upweighting probabilities of high-reward samples while downweighting poor ones). While traditional RL methods scale updates by advantage estimates [30, 29], this principle—strengthening high-quality trajectories while weakening low-quality ones based on advantage estimates [30, 35]—hasn't been fully leveraged in flow matching fine-tuning. Current approaches like reward-weighted flow matching [13] can only down-weight poor samples without actively discouraging them, significantly reducing efficiency in high-dimensional spaces (e.g., image generation) where undesirable regions vastly

outnumber desirable ones. We address this limitation by introducing advantage-based policy optimization for flow matching models, creating bidirectional learning signals through advantage estimates rather than non-negative reward weights to both enhance high-quality generations and actively suppress poor ones. Specifically, we propose an advantage-weighted flow matching objective: $\mathcal{L}_{\text{RL}}(\theta) = \mathbb{E}_{c\sim p(c),t\sim U(0,1),x_1\sim p_\theta^{n-1},x_t\sim p_t(x_t|x_1,c)}[A(x_1,c)\cdot|\mathbf{v}_\theta(x_t,t,c)-\mathbf{u}_t|^2]$, where p_θ^{n-1} is the current fine-tuned policy and $A(x_1,c)$ represents the estimated advantage for sample x_1 given context c, $\mathbf{v}_\theta(x_t,t,c)$ is the learned velocity field, $x_t=(1-t)x_0+tx_1$, and $\mathbf{u}_t=x_1-x_0$ is the target velocity for the straight-line interpolation in FM models [21].

This formulation creates a fundamentally different learning dynamic compared to reward-weighting approaches. For samples with positive advantage ($A>0$), the objective encourages matching the target velocity field, strengthening high-quality generations. Conversely, for samples with negative advantage ($A<0$), the sign inversion reverses the gradient direction, actively pushing the model away from poor generations rather than merely down-weighting them. Meanwhile, average-quality samples (where $A\approx 0$) contribute minimally to the gradient, naturally focusing computational resources on the most informative examples and facilitating efficient convergence (See Fig. 3).

Advantage Estimation. For FM models, we compute the advantage as the difference between the reward of a sample and the expected reward under the current policy as $A(x_1,c)=R(x_1,c)-V(c)$, where $R(x_1,c)$ is the human preference reward for the generated sample x_1 given context c, and $V(c)$ is a baseline value function estimated as the average reward over a batch of samples for the same context, which is computationally efficient.

Adaptive Regularization. Based on Equ. (4), we further propose to dynamically adjust the regularization strength based on the same advantage estimates. This creates a unified framework where the exploration-exploitation balance is automatically modulated at the individual sample level:

$$\begin{aligned}\mathcal{L}_{\text{ADRPO-FM}}(\theta) = \mathbb{E}_{c\sim p(c),t\sim U(0,1),x_1\sim p_\theta^{n-1},x_t\sim p_t(x_t|x_1,c)}[A(x_1,c)\cdot|\mathbf{v}_\theta(x_t,t,c)-\mathbf{u}_t|^2] \\ +(\beta_0-A(x_1,c))\cdot\mathbb{E}_{c,t,x_t}[|\mathbf{v}_\theta(x_t,t,c)-\mathbf{v}_{\text{ref}}(x_t,t,c)|^2]\end{aligned} \tag{6}$$

The adaptive regularization coefficient $\beta_{\text{tot}}=\beta_0-A(x_1,c)$ establishes a dynamic adaptation mechanism responsive to sample quality. For high-advantage samples ($A>0$), regularization decreases proportionally. For low-advantage samples ($A<0$), regularization strengthens proportionally, constraining updates to maintain proximity to the reference model. This bidirectional adaptation fundamentally transforms the exploration-exploitation landscape (Figure 3), replacing fixed regularization with sample-wise W2 regularization that continuously adapt to the evolving policies.

Stabilization, Efficient Learning. To ensure stable training with our adaptive advantage-based approach, we use advantage clipping that constrains advantages to a reasonable range $[A_{\min},A_{\max}]$ as $A_{\text{clipped}}(x_1,c)=\text{clip}(A(x_1,c),A_{\min},A_{\max})$. We also use LoRA [17] for efficient learning.

3.4 ADRPO for Fine-tuning LLMs

Applying our ADRPO framework to Large Language Models (LLMs) can address the limitation of static regularization in conventional online RL methods by dynamically controlling the penalty for deviating from the pre-trained policy based on sample advantage. High-advantage responses indicate promising directions warranting reduced regularization to encourage policy optimization, while low-advantage responses signal areas to avoid, requiring increased regularization to maintain proximity to the reliable pre-trained model and prevent undesirable outputs or instability. We integrate this principle with GRPO [31], modifying its objective by making the KL divergence regularization strength dependent on the advantage estimate (A_{GRPO}) for each sample. The objective becomes:

$$\mathcal{L}_{\text{ADRPO-GRPO}}(\theta)=\mathcal{L}_{\text{PG}}(\theta)+(\beta_0-A_{\text{GRPO}})\cdot D_{\text{KL}}\left(\pi_\theta\|\pi_{\text{ref}}\right) \tag{7}$$

Here, $\mathcal{L}_{\text{PG}}(\theta)$ is the clipped policy gradient term [31] (i.e., $-\min(A*\text{ratio}, A*\text{clip}(\text{ratio},1-\epsilon,1+\epsilon))$ and $\text{ratio}=\frac{\pi_\theta}{\pi_{\theta_{\text{old}}}}$), D_{KL} is the KL divergence, and β_0 is a baseline regularization. The term $(\beta_0-A_{\text{GRPO}})$ acts as an adaptive coefficient, decreasing for good samples ($\boldsymbol{A}_{\text{GRPO}}>0$) to promote exploitation and increasing for poor samples ($\boldsymbol{A}_{\text{GRPO}}<0$) to enforce conservative exploration, allowing ADRPO-GRPO to achieve a better exploration-exploitation trade-off (See Fig. 4).

Table 1: Comparison of text-to-image generation methods across different evaluation metrics. Best scores are highlighted in blue, second-best in green. We report standard errors estimated over 3 random seeds. ClipDiversity measures the mean pairwise distance of CLIP embeddings [24, 13].

Method	Task Metrics		Image Quality	Human Preference		
	ClipScore↑ [24]	ClipDiversity↑ [24]	Aesthetic↑ [39]	BLIPScore↑ [11]	ImageReward↑ [39]	PicScore↑ [19]
Base Model						
SD3 (2B) [12]	29.27±0.42	5.08±0.52	5.53±0.09	0.501±0.007	0.97±0.13	20.81±0.09
Other Flow Matching Models						
FLUX.1-Dev (12B) [42]	31.72±0.48	4.29±0.42	5.95±0.05	0.492±0.004	1.11±0.10	21.83±0.11
SANA-1.5 (4.8B) [38]	32.18±0.36	4.31±0.50	5.89±0.12	0.526±0.006	1.45±0.08	21.85±0.15
SD3 Fine-tuning Methods						
SD3+RAFT [8]	29.35±0.27	1.85±0.19	4.54±0.04	0.512±0.001	0.22±0.08	19.21±0.02
SD3+DPO [37]	31.30±0.52	4.78±0.46	5.82±0.05	0.509±0.005	1.48±0.10	21.31±0.10
SD3+ORW-CFM-W2 [13]	31.42±0.39	3.86±0.37	5.29±0.05	0.542±0.006	1.22±0.10	20.97±0.11
SD3+ADRPO (Ours)	32.97±0.46	5.13±0.47	6.27±0.06	0.567±0.004	1.61±0.05	22.78±0.15

4 Experiment

4.1 Experimental Setup

For our experiments, we evaluated ADRPO across two distinct domains: fine-tuning flow matching model and LLMs. **Fine-tuning FM Model.** We implemented ADRPO on SD3 (2B parameters) using a diverse range of prompts from DrawBench [28] that test various generative capabilities including color attribute binding, compositional reasoning, object counting, spatial relationships, and text rendering. We also incorporated complex prompts from RAFT [8] for artistic style transfer tasks (as shown in Figure 1). Our method employed the advantage-based ADRPO loss from Equation (6) with $\beta_0 = 1$ and $A_{max} = 1, A_{min} = -1$ for fine-tuning SD3 models while using CLIP score as rewards [24]. We conducted comprehensive comparisons against both offline methods like DPO [37] and online approaches with fixed regularization such as ORW-CFM-W2 [13]. Additionally, we benchmarked against substantially larger models including FLUX.1 Dev (12B) [42] and SANA-1.5 (4.8B) [38] to evaluate parameter efficiency. **Fine-tuning LLMs:** We fine-tuned Qwen2 [40] and Qwen3 [41] models using RM-Gemma-2B [27, 8] as the reward model on RLHFlow/test_generation_2k prompt dataset [8] (i.e., a mixture of UltraFeedback [5], Capybara [6], UltraInteract [43] and OpenOrca [20]). ADRPO was integrated with GRPO using KL-divergence regularization as described in Equation (7) with $\beta_0 = 0.04, A_{min} = -0.04, A_{max} = 0.04$, and compared against standard GRPO with fixed regularization ($\beta = 0.04$ [31]) to demonstrate our superior exploration-exploitation balance. Both experimental tracks employed advantage clipping techniques to ensure training stability. We chose β_0 equal to β in fixed regularization methods for fairness. See App. B and C for more details.

4.2 Main Results

Table 1 presents a comprehensive evaluation of text-to-image generation methods, demonstrating that our proposed ADRPO establishes a superior Pareto frontier in all metrics and achieves the best reward-diversity trade-off. Unlike competing approaches such as DPO and ORW-CFM-W2 that make significant compromises—improving semantic alignment at the cost of diversity or vice versa—ADRPO achieves state-of-the-art performance in both dimensions simultaneously through its dynamic regularization mechanism in Equ. (6). Our adaptive approach intelligently modulates regularization strength based on sample-specific advantage estimates, enabling aggressive exploitation in high-reward regions while maintaining exploration elsewhere (See Figures 3 and 4). Perhaps most remarkably, our method enables a relatively modest 2B parameter SD3 model to outperform substantially larger models including FLUX.1-Dev (12B) [42] and SANA-1.5 (4.8B) [38] across all evaluation metrics, particularly in human preferences. This quantitative superiority is visually evident in our qualitative results in Figures 1 and 2 where ADRPO-generated images demonstrate exceptional attribute binding, spatial understanding, text rendering, and artistic style transfer capabilities that even larger models struggle to match. Together, these findings suggest that adaptive regularization offers a more efficient path to performance improvement than simply scaling model parameters.

4.3 Qualitative Analysis

Figure 1: **Qualitative Comparison with Large FM Generative Models.** Our ADRPO demonstrates superior performance in Artistic Style Rendering, Attribute Binding, Coloring and Counting.

Comparison with SOTA Large FM Models. Figure 1 shows our ADRPO fine-tuned SD3 model (2B parameters) significantly outperforming much larger models like FLUX.1 Dev (12B) and SANA-1.5 (4.8B). This challenges the conventional wisdom that parameter scaling is the primary path to performance improvements. Our method excels in areas where larger models struggle: for artistic style transfer ("Jacques-Louis David style big ben"), complex compositions ("Van Gogh style astronaut"), and attribute binding ("green apple and black backpack"), ADRPO maintains both style accuracy and compositional integrity while larger models introduce visual artifacts despite their 2-6× parameter counts. These results demonstrate that adaptive regularization can enable smaller models to match or exceed much larger models' capabilities. See App. D for more results.

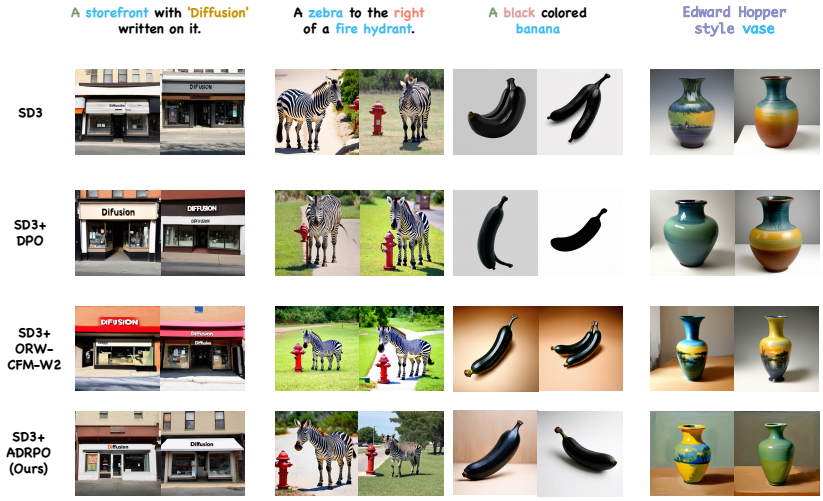

Figure 2: **Qualitative Comparison with Other RL Fine-tuning Methods.** Our ADRPO demonstrates superior performance in Artistic Style Rendering, Text Rendering, Attribute Binding, Coloring, Counting and Position. We use a similar DPO method as described in [7] to fine-tune SD3 models.

Comparison with other RL Fine-tuning Methods. Figure 2 demonstrates ADRPO's clear superiority over existing reinforcement learning fine-tuning approaches. While DPO [37] preserves diversity at the cost of semantic alignment and ORW-CFM-W2 [13] improves alignment but sacrifices diversity, ADRPO achieves excellence in both dimensions through advantage-guided regularization.

This is evident across text rendering ("Diffusion" storefront), attribute binding (zebra positioning), coloring (black banana), and style transfer tasks, where our method consistently delivers superior compositional accuracy. By dynamically modulating regularization strength—increasing constraints for uncertain samples while allowing greater divergence for reliable ones—ADRPO effectively resolves the exploration-exploitation dilemma that static approaches cannot address.

4.4 Visualizing Exploration-Exploitation Trade-off in Policy Optimization

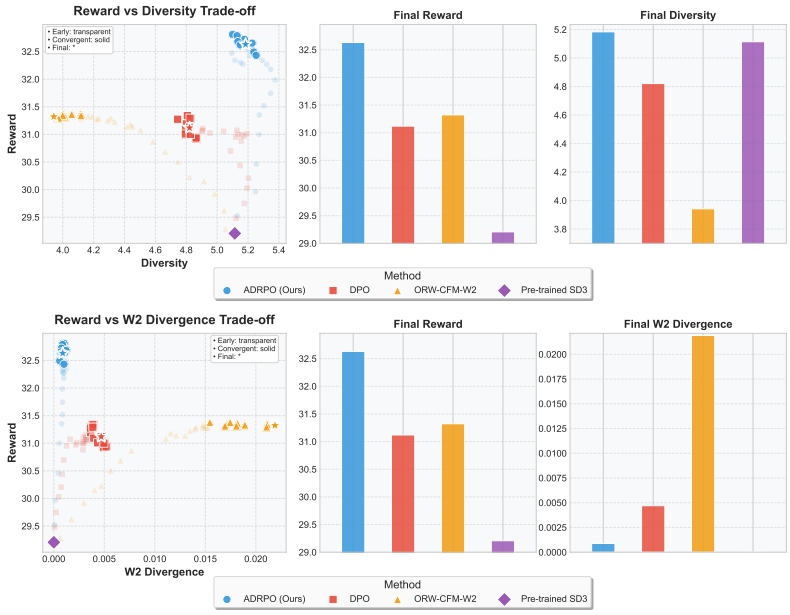

Figure 3: **Reward-Diversity/Divergence Trade-off.** Left: policy optimization trajectories (using a same seed) of different methods throughout training, with transparency indicating progression from early (transparent) to convergent (solid) to final (star) checkpoints. Each point is a learned policy from different iterations. Center and right: final reward and diversity/divergence across methods.

Reward Hacking Mitigation. Figs. 3 reveals distinct vulnerability patterns to reward hacking across methods. While DPO maintains moderate diversity but plateaus in reward optimization, ORW-CFM-W2 aggressively pursues reward optimization but exhibits significant diversity collapse (right panel), resulting in template-like generations (See Fig. 2). Our ADRPO, through advantage-guided regularization, achieves the highest reward without sacrificing diversity—a combination neither competing method attains. This translates to superior generations with precise attribute binding and high visual quality while maintaining creative flexibility. See App. D for whole learning curves.

Exploration-Exploitation Balance and Divergence Control. The trajectory visualization in Figs. 3 (left) captures each method's navigation of the exploration-exploitation landscape. The bottom plots further illustrate ADRPO's advantage in maintaining minimal W2 divergence while maximizing reward. While DPO makes modest improvements before plateauing and ORW-CFM-W2 follows an exploitation path that compromises diversity, ADRPO consistently expands the Pareto frontier. This superior balancing act stems from our adaptive mechanism making sample-specific regularization decisions, effectively resolving the dilemma that fixed-coefficient methods cannot address.

4.5 Application to Fine-tuning LLMs

To demonstrate ADRPO's versatility beyond FM models, we applied our approach to LLM fine-tuning with the Qwen2 [40] and Qwen3 [41] model families using RM-Gemma-2B [8, 9, 27] as the reward model, as shown in Figure 4. All methods are evaluated in RLHFlow/test_generation_2k dataset [8].

Superior Exploration-Exploitation Control. The policy optimization trajectories (left panels) in Figure 4 reveal distinct patterns between methods. While GRPO maintains high entropy throughout training but struggles to find high-reward regions—showing substantial horizontal movement with

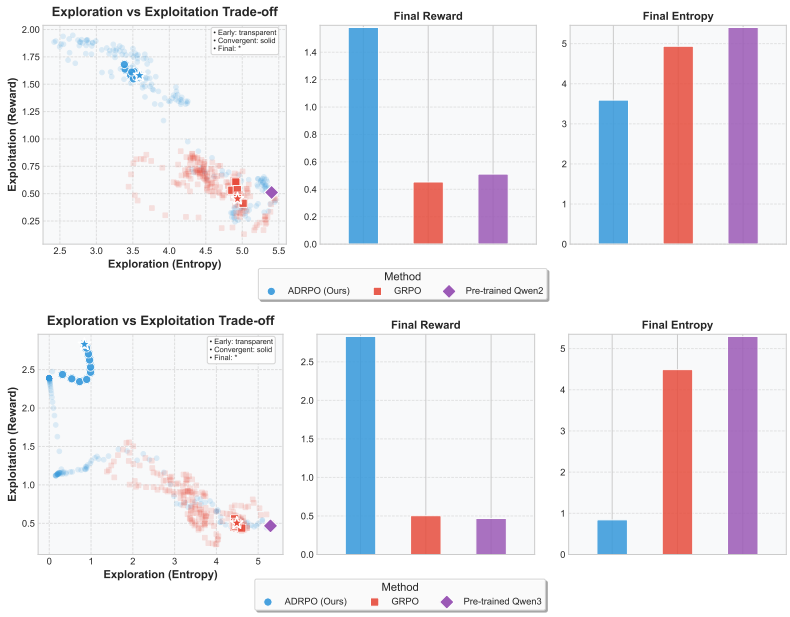

Figure 4: Ablation Studies and Exploration-Exploitation Trade-off in Fine-tuning LLMs. Left: policy optimization trajectories in reward-entropy space for ADRPO and GRPO (i.e., fixed β as [31]) across Qwen2 (0.5B) [40] (top) and Qwen3 (0.6B) [41] (bottom) models, with transparency indicating progression from early to final checkpoints. Center and right: Final performance of different methods.

limited vertical improvement—ADRPO implements a strategic exploration pattern that efficiently navigates the exploration-exploitation landscape. For Qwen3 (bottom left), ADRPO exhibits a remarkable ability to first explore lower entropy regions, then actively increase entropy to escape local optima, before converging to a final checkpoint with 5× higher reward than GRPO.

Preventing Model Collapse. ADRPO demonstrates superior resistance to model collapse during extended training. GRPO's performance tends to plateau or deteriorate as training progresses (also see Fig. D.2.1), with later checkpoints (darker red points) often showing lower rewards than earlier ones—a common failure mode of fixed-regularization methods. In contrast, ADRPO shows consistent improvement throughout training by dynamically adjusting regularization strength based on advantage estimates, eliminating the need for careful early stopping to prevent performance regression.

Cross-Architecture Generalizability. The consistent superior performance across both flow matching models and different LLM architectures confirms that ADRPO addresses fundamental limitations in reinforcement learning fine-tuning. By adapting regularization strength to sample-specific advantage estimates, our method provides a generalizable solution to the exploration-exploitation dilemma that effectively transfers between domains. See App. D for whole learning curves.

5 Conclusion

The exploration-exploitation dilemma represents a critical challenge in generative model RL fine-tuning that fixed regularization approaches fail to address. To tackle this, we propose Adaptive Divergence Regularized Policy Optimization (ADRPO), which dynamically adjusts regularization strength based on sample-specific advantage estimates—reducing constraints for high-value samples while strengthening them for poor ones. Our experiments demonstrate ADRPO's effectiveness across domains: in text-to-image generation, it outperforms other methods in alignment, quality, and diversity, enabling our 2B parameter SD3 model to surpass much larger models (4.8B and 12B) in various tasks; in LLM fine-tuning, it exhibits an emergent ability to escape local optima by actively increasing exploration. ADRPO establishes a superior Pareto frontier in the reward-diversity trade-off, confirming that sample-adaptive regularization offers a plug-and-play solution that generalizes across generative domains with minimal computational overhead. See App. A for more discussion.

References

[1] Martín Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 214–223. PMLR, 2017.

[2] 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 B. Brown, Jack Clark, Sam McCandlish, Chris Olah, Benjamin Mann, and Jared Kaplan. Training a helpful and harmless assistant with reinforcement learning from human feedback. *CoRR*, abs/2204.05862, 2022.

[3] Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024.

[4] Paul F. Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 4299–4307, 2017.

[5] Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Wei Zhu, Yuan Ni, Guotong Xie, Zhiyuan Liu, and Maosong Sun. Ultrafeedback: Boosting language models with high-quality feedback, 2023.

[6] Luigi Daniele and Suphavadeeprasit. Amplify-instruct: Synthetically generated diverse multi-turn conversations for efficient llm training. *arXiv preprint arXiv:(coming soon)*, 2023.

[7] Carles Domingo-Enrich, Michal Drozdzal, Brian Karrer, and Ricky T. Q. Chen. Adjoint matching: Fine-tuning flow and diffusion generative models with memoryless stochastic optimal control. In *The Thirteenth International Conference on Learning Representations*, 2025.

[8] Hanze Dong, Wei Xiong, Deepanshu Goyal, Yihan Zhang, Winnie Chow, Rui Pan, Shizhe Diao, Jipeng Zhang, Kashun Shum, and Tong Zhang. Raft: Reward ranked finetuning for generative foundation model alignment. *arXiv preprint arXiv:2304.06767*, 2023.

[9] Hanze Dong, Wei Xiong, Bo Pang, Haoxiang Wang, Han Zhao, Yingbo Zhou, Nan Jiang, Doyen Sahoo, Caiming Xiong, and Tong Zhang. Rlhf workflow: From reward modeling to online rlhf. *arXiv preprint arXiv:2405.07863*, 2024.

[10] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurélien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Rozière, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Graeme Nail, Grégoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel M. Kloumann, Ishan Misra, Ivan Evtimov, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, and et al. The llama 3 herd of models. *CoRR*, abs/2407.21783, 2024.

[11] Maksim Dzabraev, Alexander Kunitsyn, and Andrei Ivaniuta. VLRM: vision-language models act as reward models for image captioning. *CoRR*, abs/2404.01911, 2024.

[12] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net, 2024.

[13] Jiajun Fan, Shuaike Shen, Chaoran Cheng, Yuxin Chen, Chumeng Liang, and Ge Liu. Online reward-weighted fine-tuning of flow matching with wasserstein regularization. In *The Thirteenth International Conference on Learning Representations*, 2025.

[14] 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. *CoRR*, abs/2305.16381, 2023.

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

[16] Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. The curious case of neural text degeneration. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020.

[17] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022.

[18] James Kirkpatrick, Razvan Pascanu, Neil C. Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. Overcoming catastrophic forgetting in neural networks. *CoRR*, abs/1612.00796, 2016.

[19] 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. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023.

[20] Wing Lian, Bleys Goodson, Eugene Pentland, Austin Cook, Chanvichet Vong, and "Teknium". Openorca: An open dataset of gpt augmented flan reasoning traces. `https://https://huggingface.co/datasets/Open-Orca/OpenOrca`, 2023.

[21] Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023.

[22] 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 F. Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022.

[23] 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 F. Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human

feedback. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022.

[24] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR, 2021.

[25] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR, 2021.

[26] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023.

[27] Morgane Rivière, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, Johan Ferret, Peter Liu, Pouya Tafti, Abe Friesen, Michelle Casbon, Sabela Ramos, Ravin Kumar, Charline Le Lan, Sammy Jerome, Anton Tsitsulin, Nino Vieillard, Piotr Stanczyk, Sertan Girgin, Nikola Momchev, Matt Hoffman, Shantanu Thakoor, Jean-Bastien Grill, Behnam Neyshabur, Olivier Bachem, Alanna Walton, Aliaksei Severyn, Alicia Parrish, Aliya Ahmad, Allen Hutchison, Alvin Abdagic, Amanda Carl, Amy Shen, Andy Brock, Andy Coenen, Anthony Laforge, Antonia Paterson, Ben Bastian, Bilal Piot, Bo Wu, Brandon Royal, Charlie Chen, Chintu Kumar, Chris Perry, Chris Welty, Christopher A. Choquette-Choo, Danila Sinopalnikov, David Weinberger, Dimple Vijaykumar, Dominika Rogozinska, Dustin Herbison, Elisa Bandy, Emma Wang, Eric Noland, Erica Moreira, Evan Senter, Evgenii Eltyshev, Francesco Visin, Gabriel Rasskin, Gary Wei, Glenn Cameron, Gus Martins, Hadi Hashemi, Hanna Klimczak-Plucinska, Harleen Batra, Harsh Dhand, Ivan Nardini, Jacinda Mein, Jack Zhou, James Svensson, Jeff Stanway, Jetha Chan, Jin Peng Zhou, Joana Carrasqueira, Joana Iljazi, Jocelyn Becker, Joe Fernandez, Joost van Amersfoort, Josh Gordon, Josh Lipschultz, Josh Newlan, Ju-yeong Ji, Kareem Mohamed, Kartikeya Badola, Kat Black, Katie Millican, Keelin McDonell, Kelvin Nguyen, Kiranbir Sodhia, Kish Greene, Lars Lowe Sjösund, Lauren Usui, Laurent Sifre, Lena Heuermann, Leticia Lago, and Lilly McNealus. Gemma 2: Improving open language models at a practical size. *CoRR*, abs/2408.00118, 2024.

[28] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L. Denton, Seyed Kamyar Seyed Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, Jonathan Ho, David J. Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022.

[29] John Schulman, Philipp Moritz, Sergey Levine, Michael I. Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. In Yoshua Bengio and Yann LeCun, editors, *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, 2016.

[30] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *CoRR*, abs/1707.06347, 2017.

[31] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *CoRR*, abs/2402.03300, 2024.

[32] Ilia Shumailov, Zakhar Shumaylov, Yiren Zhao, Nicolas Papernot, Ross Anderson, and Yarin Gal. Ai models collapse when trained on recursively generated data. *Nature*, 631(8022):755–759, 2024.

[33] Ilia Shumailov, Zakhar Shumaylov, Yiren Zhao, Nicolas Papernot, Ross J. Anderson, and Yarin Gal. AI models collapse when trained on recursively generated data. *Nat.*, 631(8022):755–759, 2024.

[34] Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel M. Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F. Christiano. Learning to summarize with human feedback. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

[35] Richard S. Sutton and Andrew G. Barto. Reinforcement learning: An introduction. *IEEE Trans. Neural Networks*, 9(5):1054–1054, 1998.

[36] Alexander Tong, Kilian Fatras, Nikolay Malkin, Guillaume Huguet, Yanlei Zhang, Jarrid Rector-Brooks, Guy Wolf, and Yoshua Bengio. Improving and generalizing flow-based generative models with minibatch optimal transport. *Trans. Mach. Learn. Res.*, 2024, 2024.

[37] 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 *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pages 8228–8238. IEEE, 2024.

[38] Enze Xie, Junsong Chen, Yuyang Zhao, Jincheng Yu, Ligeng Zhu, Yujun Lin, Zhekai Zhang, Muyang Li, Junyu Chen, Han Cai, Bingchen Liu, Daquan Zhou, and Song Han. SANA 1.5: Efficient scaling of training-time and inference-time compute in linear diffusion transformer. *CoRR*, abs/2501.18427, 2025.

[39] 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. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023.

[40] An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jianxin Yang, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Xuejing Liu, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, Zhifang Guo, and Zhihao Fan. Qwen2 technical report. *CoRR*, abs/2407.10671, 2024.

[41] An Yang, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoyan Huang, Jiandong Jiang, Jianhong Tu, Jianwei Zhang, Jingren Zhou, Junyang Lin, Kai Dang, Kexin Yang, Le Yu, Mei Li, Minmin Sun, Qin Zhu, Rui Men, Tao He, Weijia Xu, Wenbiao Yin, Wenyuan Yu, Xiafei Qiu, Xingzhang Ren, Xinlong Yang, Yong Li, Zhiying Xu, and Zipeng Zhang. Qwen2.5-1m technical report. *CoRR*, abs/2501.15383, 2025.

[42] Chenglin Yang, Celong Liu, Xueqing Deng, Dongwon Kim, Xing Mei, Xiaohui Shen, and Liang-Chieh Chen. 1.58-bit FLUX. *CoRR*, abs/2412.18653, 2024.

[43] Lifan Yuan, Ganqu Cui, Hanbin Wang, Ning Ding, Xingyao Wang, Jia Deng, Boji Shan, Huimin Chen, Ruobing Xie, Yankai Lin, Zhenghao Liu, Bowen Zhou, Hao Peng, Zhiyuan Liu, and Maosong Sun. Advancing llm reasoning generalists with preference trees, 2024.

[44] Daniel M. Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B. Brown, Alec Radford, Dario Amodei, Paul F. Christiano, and Geoffrey Irving. Fine-tuning language models from human preferences. *CoRR*, abs/1909.08593, 2019.

A Discussion

In reinforcement learning fine-tuning of generative models, the exploration-exploitation trade-off represents a critical challenge: too much exploitation leads to reward hacking and diversity collapse, while excessive exploration prevents effective alignment. This dilemma is particularly pronounced in online RL fine-tuning where models continuously learn from their own generations. Existing methods rely on fixed regularization coefficients that treat all samples equally, regardless of their reward potential or uncertainty, creating an inherent tension between preserving model capabilities and optimizing for reward.

> **Key Insight 1:** ADRPO solves the fundamental exploration-exploitation dilemma in generative model fine-tuning through a principled adaptive mechanism where regularization strength automatically scales with sample-specific advantage estimates.

In this paper, we introduced Adaptive Divergence Regularized Policy Optimization (ADRPO), a novel approach that fundamentally reimagines how divergence regularization is applied during generative model fine-tuning. Unlike existing approaches that employ fixed regularization coefficients, ADRPO dynamically modulates regularization strength based on sample-specific advantage estimates, effectively turning the static trade-off parameter into an adaptive function. Through comprehensive experimentation, we have demonstrated that this simple yet powerful modification successfully overcomes the inherent limitations of fixed-regularization methods across both text-to-image generation and language model domains, establishing a new paradigm for reinforcement learning fine-tuning of generative models.

> **Key Finding 1:** ADRPO establishes a dominant Pareto frontier in the reward-diversity trade-off, achieving state-of-the-art performance in both dimensions simultaneously where previous methods could only optimize one at the expense of the other (Figure 3).

Our experimental results reveal critical insights about the nature of reinforcement learning fine-tuning. The reward-diversity trade-off, long considered an unavoidable compromise in generative model alignment, can be effectively navigated through our adaptive regularization framework. As shown in Figure 3 (left), DPO preserves moderate diversity but plateaus in reward optimization, while ORW-CFM-W2 improves alignment but suffers significant diversity collapse. In contrast, ADRPO achieves dominant performance in both dimensions simultaneously (Figure 3, center and right panels), establishing a new state-of-the-art that was previously considered unattainable. This is achieved through our bidirectional adaptation mechanism, where regularization strength decreases for high-advantage samples to enable aggressive optimization while increasing for low-advantage samples to maintain stability and diversity.

> **Key Finding 2:** ADRPO enables remarkable parameter efficiency, allowing a 2B parameter model to consistently outperform substantially larger models (4.8B and 12B), demonstrating that optimization strategy can be more consequential than model scale (Table 1).

The parameter efficiency enabled by our approach challenges fundamental assumptions about scaling laws in generative AI. As demonstrated in Table 1, our 2B parameter SD3 model fine-tuned with ADRPO consistently outperformed substantially larger models including FLUX.1-Dev (12B) and SANA-1.5 (4.8B) across all evaluation metrics—from ClipScore to human preference metrics like ImageReward and PicScore. This finding suggests that optimization strategy can be more consequential than raw parameter count for generative quality, with profound implications for both research focus and practical deployment. By enabling smaller models to match or exceed the capabilities of models 2-6× their size, ADRPO offers a path to personal access to high-quality generative AI while significantly reducing computational and environmental costs. The qualitative results in Figure 1 further support this finding, showing superior attribute binding and style transfer capabilities compared to larger models.

> **Key Finding 3:** ADRPO exhibits an emergent ability to intelligently navigate the exploration-exploitation landscape, actively increasing exploration to escape local optima—a sophisticated capability that emerges naturally from our advantage-guided mechanism (Figure 4).

Perhaps the most surprising property of ADRPO is its emergent ability to escape local optima through strategic exploration. As visualized in Figure 4, our LLM experiments with Qwen3 revealed that

ADRPO implements a sophisticated optimization trajectory absent in fixed-regularization methods. While GRPO maintained high entropy throughout training but struggled to find high-reward regions (Figure 4, bottom left, red points), ADRPO exhibited a remarkable three-phase pattern: first exploring lower entropy regions, then actively increasing entropy to escape local optima, before finally converging to a high-reward solution (Figure 4, bottom left, blue trajectory). This behavior—reminiscent of sophisticated simulated annealing schedules—emerged organically from our advantage-guided adaptive mechanism without explicit programming, resulting in final checkpoints with 5× higher reward than GRPO (Figure 4, bottom center). This finding suggests that advantage-guided regularization may unlock entirely new regions of policy space previously inaccessible to fixed-regularization methods.

A.1 Limitations

While ADRPO demonstrates significant improvements over existing approaches, several limitations should be acknowledged. Our experiments, though comprehensive, were primarily conducted on models of moderate scale (SD3 2B, Qwen2-0.5B, and Qwen3-0.6B) due to computational constraints. An important avenue for future work is extending these findings to much larger foundation models, where the exploration-exploitation dilemma becomes even more critical. Particularly, models with parameters in the hundreds of billions are known to be more susceptible to training collapse during online RL fine-tuning, potentially making adaptive regularization even more crucial at scale.

The advantage-based formulation introduces a slight computational overhead compared to purely reward-weighted methods like ORW-CFM-W2 [13] and RAFT [8]. Though we mitigate this through efficient batch-based normalization techniques similar to those used in GRPO [31], further optimization could reduce this overhead. Our current implementation of advantage estimation using batch statistics works well in practice but could be improved with more sophisticated value approximation methods, especially for complex reward landscapes with high variance.

A.2 Broader Impact

In general, ADRPO represents a significant advance in reinforcement learning fine-tuning methodology with potential impacts extending beyond the specific models tested. The ability to efficiently navigate the exploration-exploitation trade-off through adaptive regularization addresses a fundamental challenge in the field, potentially influencing how the broader research community approaches model alignment.

Our finding that a relatively small model (2B parameters) can outperform substantially larger models (4.8B and 12B parameters) when fine-tuned with ADRPO has important implications for personal access to high-quality generative AI. This parameter efficiency could significantly reduce the computational resources required for state-of-the-art performance, making advanced generative capabilities more accessible to researchers and organizations with limited resources while reducing the environmental footprint of both training and deploying such systems.

The emergent ability to escape local optima demonstrated in our LLM experiments suggests that advantage-guided adaptive regularization may unlock previously unattainable regions of policy space. This capability could inspire new approaches to optimization in high-dimensional spaces where fixed regularization schemes tend to converge prematurely to a sub-optimal.

Our work also introduces a unified framework that bridges continuous and discrete generative paradigms, offering a consistent solution to the exploration-exploitation dilemma across domains. This cross-domain applicability could foster greater knowledge transfer between previously disparate research communities working on different types of generative models.

B Experimental Details

In this section, we provide comprehensive details of our experimental setup for both text-to-image alignment and language model fine-tuning tasks. To maintain clarity, we present these domains separately.

B.1 Flow Matching Model Fine-tuning Tasks

B.1.1 Baseline Methods

Base Model We adopt Stable Diffusion 3 (SD3) [12] as our base model—a 2B parameter architecture that combines a Multimodal Diffusion Transformer (MMDiT) backbone with a rectified flow training objective. SD3 represents the latest evolution of latent diffusion models, introducing a joint text-image Transformer that enables rich bidirectional attention between prompt tokens and image latents. Unlike earlier score-based approaches, SD3 is trained via conditional flow matching under a rectified trajectory, where the model learns to predict direct velocity fields between noise and data, improving sample efficiency and semantic alignment. It leverages multiple frozen text encoders (CLIP and T5) and improved autoencoding for high-resolution image synthesis, while achieving state-of-the-art performance on prompt fidelity, compositional reasoning, and text rendering. This combination of high quality, controllability, and architectural flexibility makes SD3 a robust and representative base model for studying the effects of reinforcement learning fine-tuning, such as ADRPO.

Larger-Scale Flow Matching Models To comprehensively evaluate ADRPO against parameter scaling approaches, we selected two state-of-the-art text-to-image diffusion models that leverage flow-matching training objectives and significantly larger parameter counts than our base model. These models serve as strong upper baselines in terms of both capacity and generation quality:

1. **FLUX.1-Dev** [42], with 12B parameters, represents the high-performance frontier of open-source flow-matching architectures. It employs a rectified flow training objective based on Wasserstein-2 optimal transport, enabling more stable and efficient training compared to traditional score-matching methods. FLUX integrates a multimodal diffusion transformer (MMDiT) with powerful prompt conditioning mechanisms, achieving near-photorealistic output, superior compositional fidelity, and high stylistic diversity. It is widely regarded as one of the most capable open models in terms of prompt adherence, fine-grained semantic alignment, and artistic control.
2. **SANA-1.5** [38], with 4.8B parameters, serves as a strong intermediate-scale baseline. It introduces an efficient diffusion transformer architecture that combines linear attention mechanisms and a highly compressed autoencoder (32x downsampling), enabling high-resolution generation at lower computational cost. SANA adopts a decoder-style language model for text conditioning and achieves state-of-the-art results on the GenEval benchmark for prompt-image alignment. Despite its moderate size, SANA.1.5 offers a competitive trade-off between generation quality, efficiency, and controllability.

Both models exemplify the benefits of scaling parameter count and architectural sophistication within the flow-matching paradigm. By comparing against them, we isolate the advantages of ADRPO in improving alignment and control without relying solely on larger models. This allows us to highlight ADRPO's efficiency and generalization capabilities, especially when applied to smaller models such as our 2B-parameter SD3 baseline.

RL Baseline Methods For fine-tuning method comparisons, we included approaches representing the most representative spectrum of current techniques:

RAFT [8] implements a reward-ranked fine-tuning approach that selects high-quality outputs based on reward scores, providing a online RL baseline. This approach has demonstrated considerable effectiveness in improving generative models but lacks the adaptive divergence regularization mechanisms essential for preserving model capabilities during policy optimization.

DPO [37] adapts the Direct Preference Optimization method to flow matching models, providing an established offline RL baseline. We apply diffusion-DPO to flow matching following methodologies

established in recent literature [7]. This approach offers stable optimization and effective diversity preservation through its implicit regularization properties, though it may be limited in its ability to explore the full policy space due to its offline nature. Given DPO's widespread adoption and demonstrated success across various fine-tuning tasks, it serves as our primary offline RL fine-tuning baseline.

ORW-CFM-W2 [13] represents the current state-of-the-art in online RL fine-tuning for flow matching models, employing fixed Wasserstein-2 regularization combined with reward weighting. As the first online RL fine-tuning method developed specifically for flow matching models, it achieves leading performance in this domain through its W2 regularized online RL framework. This method provides a crucial benchmark against which to evaluate our ADRPO approach, as it represents the online SOTA method with fixed Wasserstein-2 regularization, allowing us to directly highlight the effectiveness of our proposed adaptive divergence regularization mechanism.

B.1.2 Reward Models and Evaluation

For text-to-image generation, we implemented a comprehensive reward system combining multiple complementary models to evaluate different aspects of generation quality:

1. **Reward Model.** We used CLIP Score [25] to compute cosine similarity between text prompts and generated images (ClipScore) as our reward model for all text-to-image alignment task.
2. **Quality Assessment.** We employed a aesthetic predictor to evaluate visual appeal from ImageReward [39].
3. **Human Preference Models.** We incorporated ImageReward [39], trained on direct human judgments, and PickScore [19], developed through large-scale pick-one-from-four preference data, to align our generations with human aesthetic preferences. We complemented this with BLIP-based [11] evaluation to mitigate architecture-specific biases.
4. **Diversity Evaluation.** For diversity evaluation, we developed ClipDiversity, which measures the average pairwise distance between CLIP embeddings of multiple generated images of current FM model.

B.1.3 Prompt Datasets

Our text-to-image experiments utilized a diverse collection of prompts selected to evaluate different capabilities. DrawBench [28] provided our primary test set, covering attribute binding, spatial relationships, counting accuracy, and text rendering. We extended this with artistic style prompts from RAFT [8] (e.g., "Van Gogh style astronaut") and custom compositional prompts testing multi-object relationships (e.g., "A green apple and a black backpack").

B.2 LLM Fine-tuning Tasks

B.2.1 Baseline Methods

Base Models For our language model experiments, we employed Qwen2 [40] and Qwen3 [41] models as base architectures, representing recent advancements in autoregressive LLM models. These models demonstrate strong foundational capabilities and serve as robust pre-trained reference model for fine-tuning experiments. Specifically, Qwen2 and Qwen3 incorporate key architectural enhancements such as Grouped Query Attention (GQA), RoPE positional encoding, and long-context support (up to 128K for Qwen3), which contribute to efficient inference and robust context handling. Moreover, despite their relatively small parameter sizes (0.5B and 0.6B respectively), these models exhibit competitive performance across a range of reasoning, code generation, and language understanding benchmarks. Their strong pretraining on diverse multilingual and domain-specific corpora—including high-quality instructional data, code, and math—ensures excellent generalization. Qwen3 further introduces a hybrid prompting paradigm that enables dynamic switching between direct answering and step-by-step reasoning, significantly enhancing the model's flexibility and interpretability during instruction-following tasks. These strengths make Qwen2 and Qwen3 especially well-suited for fine-tuning via reinforcement learning from human feedback (RLHF), where high-quality priors and reasoning ability are essential for aligning model behavior with human preferences.

RL Baseline Methods For RL fine-tuning comparison, we selected GRPO [31] with fixed KL regularization ($\beta = 0.04$ [31]) as it represents the current state-of-the-art in online RL fine-tuning for LLMs. GRPO improves upon earlier methods like PPO [30] through group-level advantage estimation and more efficient policy optimization. However, it crucially still relies on static regularization that treats all samples equally regardless of their quality or uncertainty. This limitation makes GRPO an ideal candidate for demonstrating the advantages of our adaptive regularization approach, as both methods share the same underlying optimization framework but differ specifically in their treatment of regularization (also can be served as our ablation studies).

B.2.2 Reward Model and Evaluation

For LLM fine-tuning, we used RM-Gemma-2B [27, 8], a reward model built upon the Gemma-2B language model and fine-tuned using a diverse collection of human preference datasets. RM-Gemma-2B maps input completions to scalar reward values, which serve as proxy signals for alignment with human preferences. The model is trained using pairwise comparison data spanning a wide range of tasks—including helpfulness, harmlessness, factuality, and reasoning—through a Bradley-Terry style objective that encourages higher scores for preferred responses. This formulation enables the reward model to capture nuanced quality differences across candidate outputs. To support more stable and informed policy updates, we further incorporated entropy-based regularization to evaluate and balance the exploration-exploitation dynamics of the fine-tuned policies. This combined approach ensures that the optimization process not only aligns outputs with human values but also maintains diversity and adaptability in model behavior.

B.2.3 Prompt Datasets

For the large language model fine-tuning, we have used the RLHFlow/test_generation_2k dataset [9], containing 2,000 diverse prompts compiled from high-quality instruction-following datasets, and we randomly choose 10% as test prompts. This diverse prompt set allowed comprehensive evaluation across multiple dimensions, including factual accuracy, reasoning capabilities, and response quality. Specifically, the prompts were drawn from a combination of several representative and complementary sources: **UltraFeedback** [5], **Capybara** [6], **UltraInteract** [43], and **OpenOrca** [20].

- **UltraFeedback** provides high-quality single-turn instruction-response pairs with rich feedback annotations generated by GPT-4, including multi-dimensional numerical scores (e.g., helpfulness, correctness, conciseness) and textual critiques. These annotations support fine-grained evaluation and reward modeling.
- **Capybara** contributes multi-turn dialogues generated through the Amplify-Instruct pipeline, which enriches single-turn seed prompts into deep, logically consistent conversations. It emphasizes diverse topics, natural phrasing, and contextual reasoning, making it valuable for evaluating sustained dialogue coherence.
- **UltraInteract** focuses on complex tasks involving step-by-step reasoning, such as math, coding, and logic problems. Each example includes multi-step trajectories with intermediate model outputs, environment feedback, and correctness signals, enabling assessment of models' planning and iterative refinement abilities.
- **OpenOrca** offers a large-scale collection of instruction-response pairs distilled from GPT-4 and GPT-3.5 using the FLAN dataset collection. Its responses often include chain-of-thought style rationales, making it a useful benchmark for evaluating models' reasoning depth and informativeness.

By combining prompts from these datasets, the test set enables comprehensive evaluation of a model's capabilities across a wide range of real-world tasks and dialogue scenarios, from single-turn factual queries to multi-turn, multi-step reasoning challenges.

B.3 Computation Resources

All experiments were conducted on NVIDIA A6000 (48GB) GPUs. For SD3 fine-tuning tasks [12], we employed parameter-efficient LoRA [17] adaptation to reduce memory requirements and training time, while still achieving excellent results. In contrast, for the relatively smaller Qwen2-0.5B [40]

and Qwen3-0.6B [41] language models, we performed direct full-parameter fine-tuning without LoRA.

Our experimental setup utilized publicly available open-source reward/evaluation models and datasets across all domains, ensuring reproducibility and alignment with established benchmarks. The computation requirements varied significantly between tasks: LLM fine-tuning experiments were relatively efficient, typically completing within 12-24 hours per model configuration, while SD3 fine-tuning tasks were more computationally intensive, requiring approximately 2-3 days.

C Algorithm Pseudocode

We first detail our algorithm pseudocode in Algorithm 1 for fine-tuning flow matching models (we use linear interpolation path as an example). Noting that, we can sample from current learned velocity field $\mathbf{v}_\theta(x_t, t, c)$ via solving: $x_1 = x_0 + \int_0^1 \mathbf{v}_\theta(x_t, t, c)dt$, wherein $x_0 \sim p(x_0)$ and $p(x_0)$ is a standard gaussian distribution [21, 12]. As for our method for fine-tuning LLM models, we can simply add an extra advantage-weighted KL divergence into the original GRPO training loss as Equ. (7), therefore it is easy to be implemented.

Algorithm 1 Adaptive Divergence Regularized Policy Optimization (ADRPO) for SD3 Fine-tuning

Require: Pre-trained flow matching model π_{ref} (SD3), baseline regularization coefficient β_0, advantage clipping range $[A_{\min}, A_{\max}]$, learning rate η

Initialize fine-tuned policy π_θ^0 with pre-trained parameters (or LoRA adaptation)
for training iteration $n = 1, 2, \ldots$ **do**
    Sample a batch of text prompts $\{c_i\}_{i=1}^B \sim p(c)$
    Sample target states $\{x_1^i\}_{i=1}^B \sim \pi_\theta^{n-1}(x|c_i)$ from current policy ▷ Online sampling strategy
    for each prompt c_i and its generated image x_1^i **do**
        Compute reward $R(x_1^i, c_i)$ using CLIP Score
        Sample intermediate time step $t_i \sim \mathcal{U}(0, 1)$
        Compute intermediate state $x_t^i = (1 - t_i)x_0^i + t_i x_1^i$ ▷ Straight-line interpolation
        Compute target velocity $u_t^i = x_1^i - x_0^i$
    end for
    Compute baseline value $V(c_i) = \frac{1}{B}\sum_{i=1}^B R(x_1^i, c_i)$ for each context
    Compute advantage $A(x_1^i, c_i) = R(x_1^i, c_i) - V(c_i)$
    Apply advantage clipping: $A_{\text{clipped}}(x_1^i, c_i) = \text{clip}(A(x_1^i, c_i), A_{\min}, A_{\max})$
    Compute adaptive regularization coefficient $\beta_{\text{tot}} = \beta_0 - A_{\text{clipped}}(x_1^i, c_i)$
    Update model parameters using the ADRPO loss $\mathcal{L}_{\text{ADRPO-FM}}(\theta)$ from Equation (6):
    $\theta \leftarrow \theta - \eta \nabla_\theta \mathcal{L}_{\text{ADRPO-FM}}(\theta)$
end for
return Fine-tuned policy π_θ

D Additional Experimental Results

D.1 Flow Matching Model Fine-tuning Tasks

D.1.1 Additional Qualitative Results

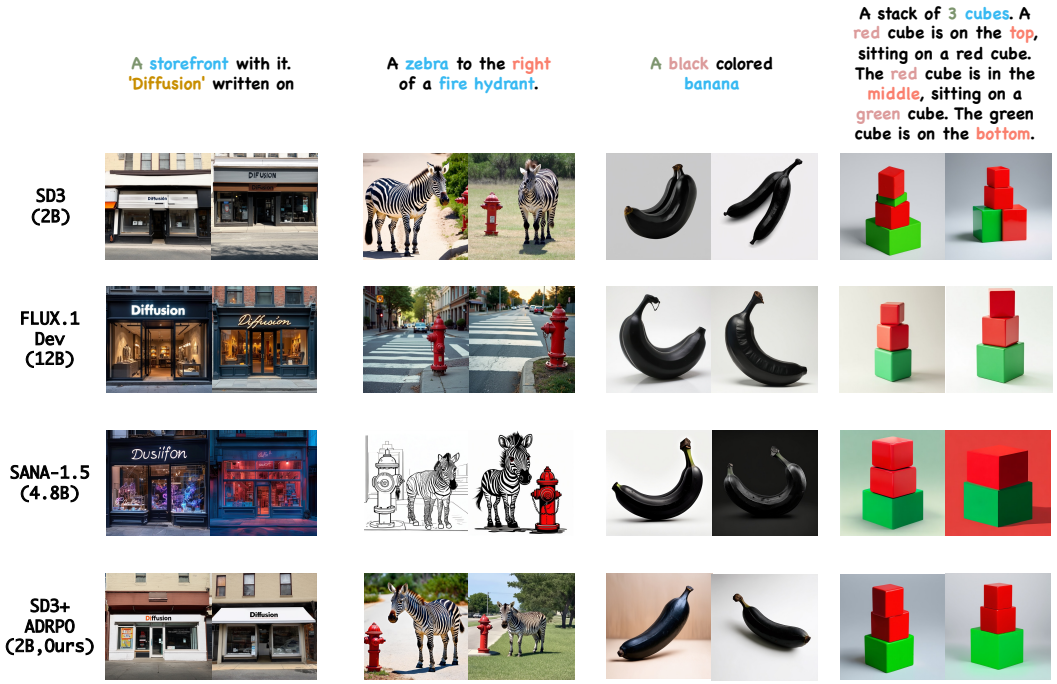

Figure 5: **Additional Qualitative Comparison with Large FM Generative Models.** Our ADRPO demonstrates superior performance in Attribute Binding, Coloring, Counting, Text Rendering and Position.

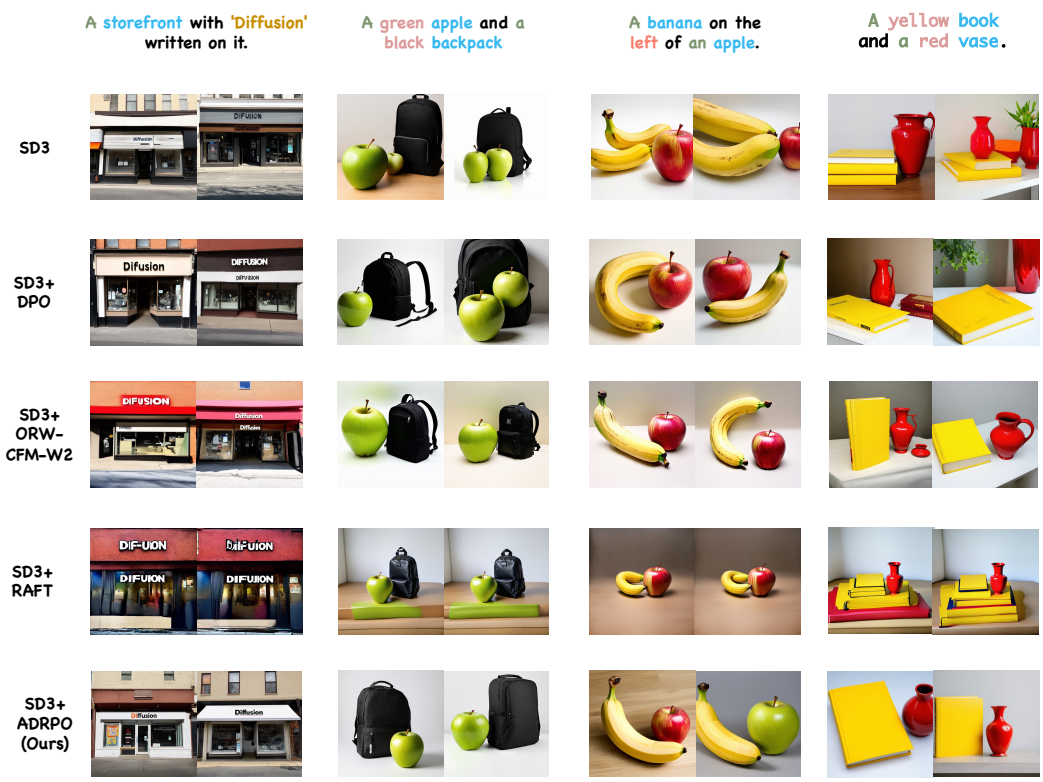

Figure 6: **Additional Qualitative Comparison with Other RL Fine-tuning Methods.** Our ADRPO demonstrates superior performance in Text Rendering, Attribute Binding, Coloring, Counting and Position.

D.1.2 Learning Curves

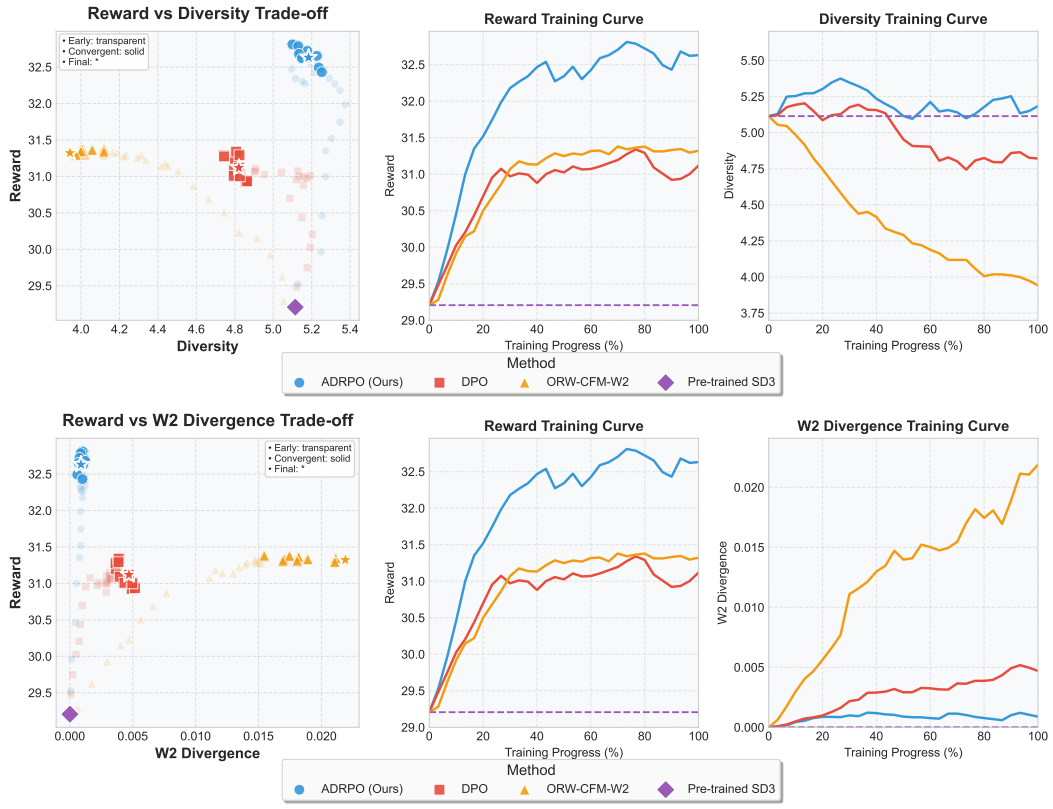

Figure 7: Learning Curves of Fine-tuning SD3. Left: Complete policy optimization trajectories across three different methods throughout training using a same seed (for fairness). Transparency indicates progression from early (transparent) stages through convergent (solid) to final (star) checkpoints, with each point representing a learned policy from different iterations. Center and right: Learning curves of RL agents.

D.2 LLM Fine-tuning Tasks

D.2.1 Learning Curves

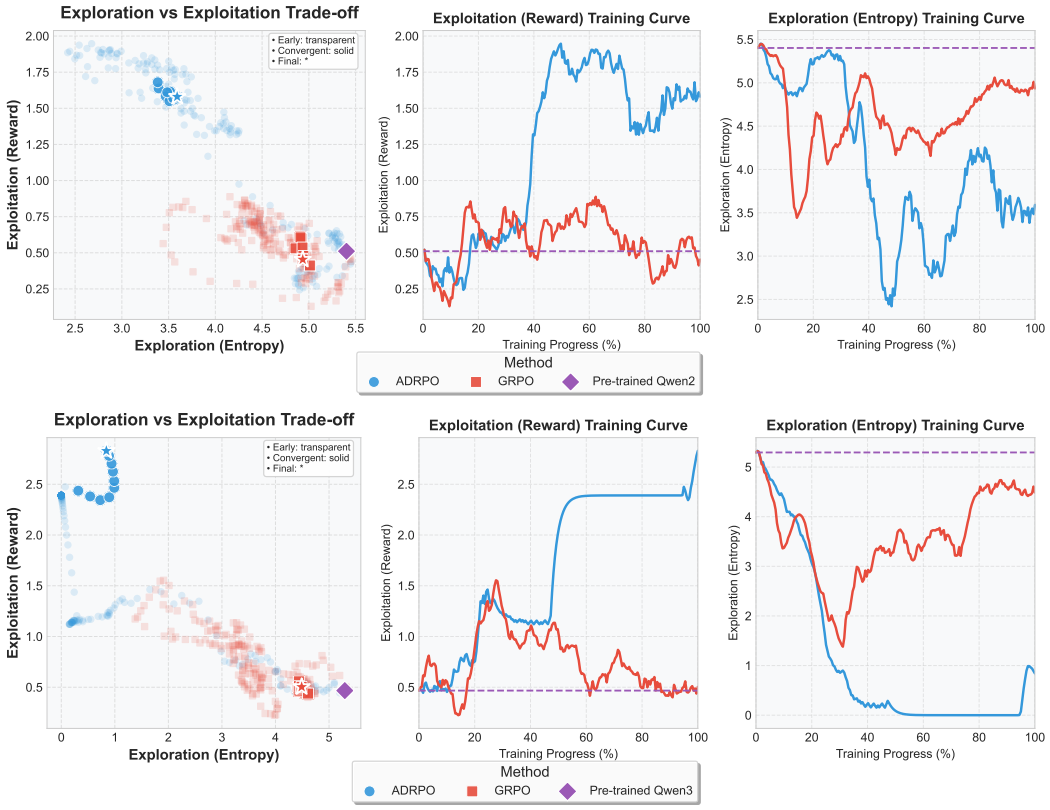

Figure 8: Learning Curves of LLM Fine-tuning Experiments (100 iterations, no early stop).

D.2.2 Reward and KL Divergence Trade-off

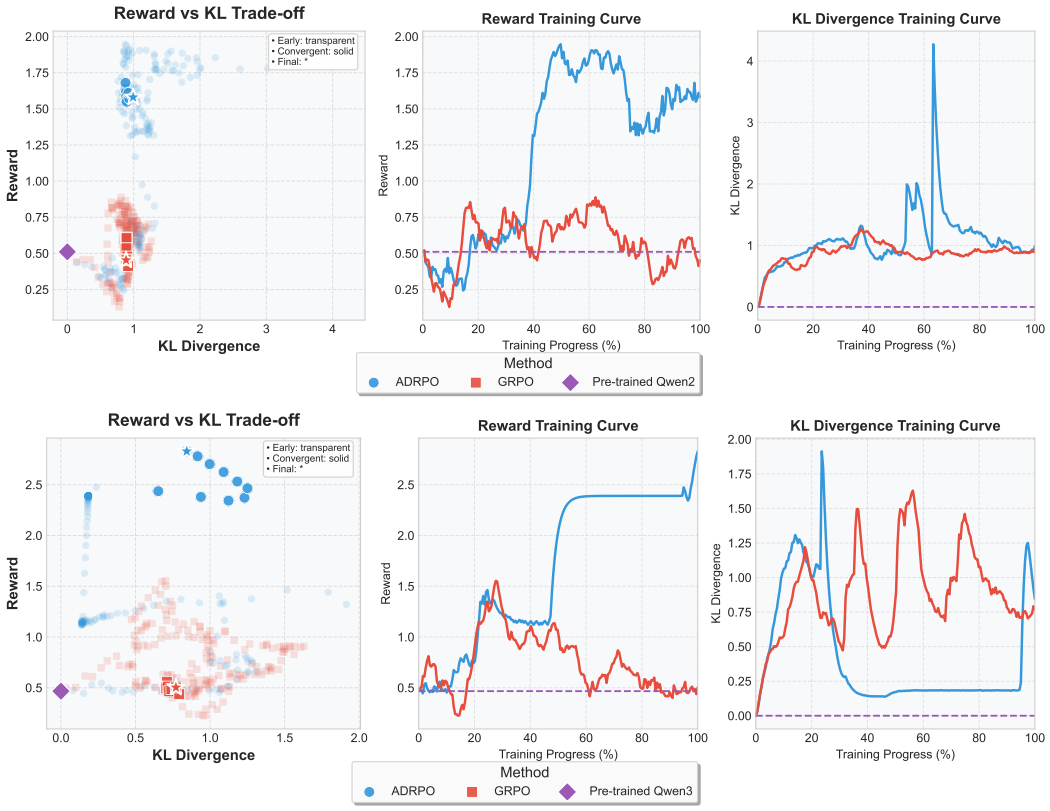

Figure 9: Reward Divergence Trade-off of LLM Fine-tuning Experiments (100 iterations, no early stop).

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Our abstract and introduction clearly articulate the key contributions, specifically: (1) introducing ADRPO as a general framework for adjusting regularization based on advantage estimates in Sec. 3, (2) demonstrating superior text-to-image alignment with ADRPO enabling a 2B parameter model to outperform larger models and other RL methods (See Fig. 1 and Fig. 2), (3) showing emergent exploration behavior in LLMs (See Fig. 4), and (4) establishing cross-domain applicability across flow matching models and LLMs. These claims are fully supported by the experimental results in Sec. 4 and additional results in App. D.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We include a detailed limitations section in App. A.1 that discusses computational overhead, scaling to larger models, and potential improvements to advantage estimation techniques. We acknowledge that our experiments were primarily conducted on models of moderate scale (SD3-2B [12], Qwen2-0.5B [40], and Qwen3-0.6B [41]) due to computational constraints and discuss how the approach might behave on much larger foundation models.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.

- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

   Justification: Our paper focuses on an empirical approach to reinforcement learning fine-tuning rather than providing formal theoretical results or proofs. We present algorithm formulations (see Equs. (4), (6), and (7)) and empirical validations of the proposed ADRPO method through extensive experiments in Sec. 4.

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We provide comprehensive details on the experimental setup in Sec. 4.1 and App. B, including model architectures, datasets, reward models, and evaluation metrics. Our algorithm pseudocode in App. C (Algorithm 1) further enhances reproducibility by detailing the implementation of ADRPO for SD3 fine-tuning. We also discuss LLM/FM fine-tuning details in App. C and App. B.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example

(a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
(b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We use publicly available models (SD3 [12], Qwen2 [40], Qwen3 [41]) and datasets (DrawBench [28], RLHFlow [8]) as noted in App. B. Our implementation details and algorithm pseudocode in App. C and App. B provide sufficient information for reproduction, and we plan to release our codes upon publication.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We specify training and test details in Section 4.1 and Appendix B, including prompt datasets (DrawBench for text-to-image, RLHFlow for LLMs), hyperparameters, and optimization approaches (LoRA adaptation for SD3, full parameter fine-tuning for Qwen).

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.

- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Table 1 reports standard errors estimated over 3 runs with different random seeds for all evaluation metrics. This is clearly stated in the table caption ("We report standard errors estimated over 3 runs of different random seeds") and the results consistently show ADRPO outperforming other methods beyond statistical error margins.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
   - If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: In App. B, we specify that all experiments were conducted on NVIDIA A6000 (48GB) GPUs, with LoRA [17] used for SD3 fine-tuning to reduce memory requirements. We also note the approximate time requirements: LLM experiments completed within 12-24 hours per model configuration, while SD3 fine-tuning required 2-3 days.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

Answer: [Yes]

Justification: Our research conforms to the NeurIPS Code of Ethics. We have ensured transparency in our methodology, cited all sources appropriately in our bibliography, and discussed both benefits and potential limitations of our approach in App. A and App. A.1.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: In App. A.2, we discuss both positive impacts (parameter efficiency leading to reduced computational resources and costs, making advanced generative capabilities more accessible) and potential implications of our work. We highlight that "our finding that a relatively small model (2B parameters) can outperform substantially larger models (4.8B and 12B parameters) when fine-tuned with ADRPO has important implications for personal access to high-quality generative AI."

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: Our paper doesn't explicitly release models or datasets that pose high risk for misuse. We focus on improving fine-tuning methodology for existing models (Sec. 3) rather than releasing new assets that would require specific safeguards.

Guidelines:

- The answer NA means that the paper poses no such risks.

- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We properly cite the original sources for all models and datasets used, including SD3 [12], Qwen2 [40], Qwen3 [41], DrawBench [28], SANA-1.5 [38], FLUX.1-Dev [42], and RLHFlow [8] as referenced throughout our paper (see Sections 1, 2, 4, and App. B).

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer:[NA]

    Justification: Our paper doesn't introduce new datasets, code packages, or model releases; it presents a new methodology (ADRPO) for fine-tuning existing models.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: Our research doesn't involve crowdsourcing or direct human subjects. We use existing public datasets and evaluation metrics rather than collecting new human preference data or conducting human evaluations.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: Our paper doesn't involve research with human subjects that would require IRB approval or equivalent.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: While we fine-tune LLMs (Qwen2, Qwen3), LLMs are not used as components in our research methodology itself; they are the subject of study rather than tools used to develop the core method.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.