FLOWCPO: A UNIFIED DIVERGENCE VIEW OF PREFERENCE ALIGNMENT FOR FLOW MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Preference alignment for flow and diffusion models now spans online reinforcement learning and offline preference optimization, but the relation between these methods remains unclear. In particular, existing forward-process alignment methods require fresh samples from the current model, while offline methods based on fixed preference pairs rely primarily on positive-only fine-tuning or DPO-style likelihood-ratio surrogates. We organize these approaches through a divergence-based framework and introduce **FLOWCPO**, an offline forward-KL objective that uses both preferred and dispreferred samples without online rollouts. For linear interpolation, we show under explicit regularity conditions that the forward-KL objective is bounded by a contrastive flow matching loss, yielding a tractable surrogate on fixed data. We further show that this loss is nonnegative, whereas the signed regression loss of simplified FlowDPO can be unbounded below. In the in-domain setting, FLOWCPO achieves higher mean GenEval and OCR scores than the evaluated baselines, reaching 0.84 and 0.87 versus 0.81 and 0.74 for FlowDPO at CFG 3.0. In the out-of-domain setting, the results are mixed, with the best GenEval result but lower reward scores than RFT on several metrics.

1 INTRODUCTION

Recent advancements in flow matching and diffusion-based generative modeling (Ho et al., 2020; Lipman et al., 2022; Albergo & Vanden-Eijnden, 2022; Sohl-Dickstein et al., 2015; Luo, 2022; Holderrieth & Erives, 2025) have facilitated the development of practical preference alignment algorithms for continuous generative spaces (Liu et al., 2025a; Zheng et al., 2025; Liu et al., 2025b; Xue et al., 2025; Bergmeister et al., 2026). These efforts have yielded diverse alignment paradigms, ranging from online reinforcement learning frameworks like FlowGRPO (Liu et al., 2025a), DiffusionNFT (Zheng et al., 2025), AWM (Xue et al., 2025) and RAM (Bergmeister et al., 2026) to offline preference optimization methods such as FlowDPO (Liu et al., 2025b). In contrast to alignment in discrete autoregressive models, these continuous-time methods operate directly on sampling trajectories and velocity fields rather than token-level probabilities. While empirical results are promising, the relations among these methods remain difficult to compare.

Existing flow-based alignment methods can be organized along two axes (Tab. 1): the *sampling regime* (online versus offline) and *the direction of the idealized KL objective*. Under this view, FlowGRPO (Liu et al., 2025a) and FlowDPO (Liu et al., 2025b) represent online and offline instances of the reverse-KL branch, whereas DiffusionNFT (Zheng et al., 2025) provides a forward-process objective but refreshes its training samples online. The unresolved problem is therefore the offline forward-KL objective using both preferred and dispreferred target samples: a forward-KL expectation can use fixed target samples, but its log-likelihood is intractable for flow matching. An offline method needs a justified bridge from this distribution-level objective to velocity-field regression on fixed preferred and dispreferred data.

Table 1: **Objective-level view of flow-based preference alignment.** The two axes separate the sampling regime from the direction of the idealized KL objective.

Divergence	Online	Offline
Reverse-KL	FlowGRPO	FlowDPO
Forward-KL	DiffusionNFT	**FLOWCPO (ours)**

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

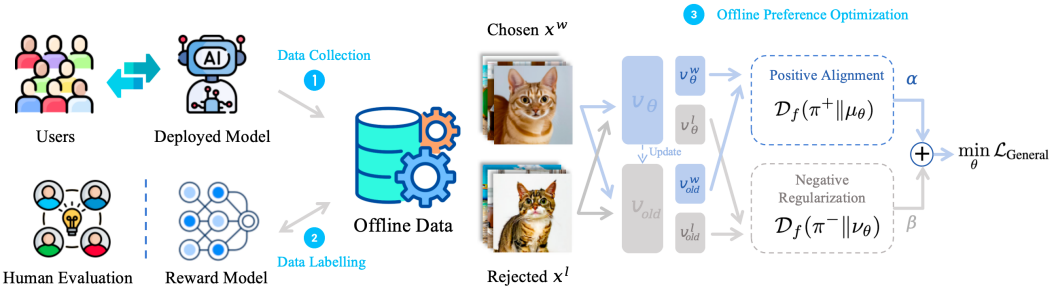


Figure 1: **FLOWCPO uses a fixed preference dataset to train two coupled flow branches.** A frozen generator supplies preferred/dispreferred pairs $\mathcal{D} = \{(c, \mathbf{x}_0^w, \mathbf{x}_0^l)\}$; the positive branch matches preferred samples, while a mirrored branch matches dispreferred samples. Training uses only noised versions of these fixed endpoints and requires no online rollout.

We address this gap by establishing a unified divergence-based framework for flow-based preference alignment and deriving **Flow Contrastive Preference Optimization (FLOWCPO)**. Starting from forward-KL terms over preferred and dispreferred target distributions, we use a flow matching bound for linear interpolation to obtain a contrastive regression surrogate. The resulting loss can be estimated from a fixed preference dataset and therefore requires no sampling from the model during training.

We also connect FLOWCPO to simplified FlowDPO (App. B.5). Simplified FlowDPO subtracts the regression error on dispreferred samples and can be unbounded below, whereas FLOWCPO adds two nonnegative regression errors. With equal branch weights ($\lambda = 1$), our loss decomposition identifies corrections determined by the difference between the current model and its exponential moving average (EMA).

Empirically, we evaluate our approach under a strictly offline protocol. In the in-domain setting, where preference pairs are generated by the reference model, FLOWCPO attains higher mean GenEval and OCR scores than RFT and FlowDPO while remaining strong on general-preference metrics. In the out-of-domain setting, the result is more mixed: FLOWCPO gives the best GenEval score but RFT is stronger on several reward metrics. The overall offline optimization pipeline is illustrated in Fig. 1. **Our contributions are threefold:**

- We provide a unified divergence-based framework that separates the online/offline sampling regime from the direction of the idealized KL objective, clarifying the assumptions behind connections among existing flow-based alignment methods.
- We derive FLOWCPO, a nonnegative contrastive surrogate for offline forward-KL alignment. Under stated regularity conditions for linear interpolation, its regression objective bounds the branch-wise forward-KL objective and is estimable from fixed preference data. We also derive its loss decomposition relative to simplified FlowDPO.
- We evaluate FLOWCPO with two types of offline data, showing higher GenEval and OCR scores than FlowDPO in the in-domain setting and mixed results in the out-of-domain setting.

2 RELATED WORK

Online alignment with policy-generated samples. Online methods improve a generator using samples drawn from its current or recent policy. ReFL (Xu et al., 2023) uses reward feedback, DDPO (Black et al., 2024) and DPOK (Fan et al., 2023) estimate trajectory policy gradients, and DRaFT (Clark et al., 2024) and AlignProp (Prabhudesai et al., 2023) differentiate through the sampler. FlowGRPO (Liu et al., 2025a) optimizes stochasticized flow trajectories, with subsequent refinements to temporal allocation and trajectory reuse (He et al., 2025; Li et al., 2025b;c; Ding & Ye, 2025). SPO (Liang et al., 2025) generates candidates at each denoising step and selects preference pairs with a step-aware model. DiffusionNFT (Zheng et al., 2025), AWM (Xue et al., 2025), and RAM (Bergmeister et al., 2026) use regression-style objectives on noised on-policy samples.

Offline alignment with fixed preference data. Fixed-data methods differ in how they use preferred and dispreferred samples. D3PO (Yang et al., 2024a) extends pairwise learning to diffusion trajec-

tories; DiffusionDPO (Wallace et al., 2024) uses reference-relative denoising errors, with temporal weighting in Dense Reward (Yang et al., 2024b) and a flow-model extension in FlowDPO (Liu et al., 2025b). DMPO (Li et al., 2025a) develops a reverse-KL objective, whereas MaPO (Hong et al., 2026) learns preference margins without a reference model. RFT (Xiong et al., 2025; Chen et al., 2025) fits only preferred samples. FLOWCPO instead uses both sides of fixed preference pairs in a coupled, two-branch forward-KL objective, with a flow-matching upper-bound surrogate under stated conditions.

Divergence-based alignment in language models. LLM alignment also treats divergence and utility as design choices. f-DPG (Go et al., 2023) generalizes distribution matching, while f-DPO (Wang et al., 2024) varies the divergence constraint for preference learning. GPO (Tang et al., 2024) relates DPO (Rafailov et al., 2023), IPO (Azar et al., 2024), and SLiC (Zhao et al., 2023) through convex pairwise losses. f-PO (Han et al., 2025) unifies divergence-minimization formulations including DPO and EXO (Ji et al., 2024).

3 PRELIMINARIES

We review linear flow matching, preference optimization, and the two KL directions used in our framework. Additional background is provided in App. F.

3.1 DIFFUSION MODELS AND FLOW MATCHING

We first briefly review the two continuous-time generative paradigms used throughout the paper. Both start from a simple latent variable, typically Gaussian noise $\epsilon \sim \mathcal{N}(0, I)$, and define intermediate states $\mathbf{x}_t$ for $t \in [0, 1]$ that connect data and noise.

In diffusion models (Ho et al., 2020; Sohl-Dickstein et al., 2015), one specifies a forward noising process $q(\mathbf{x}_t \mid \mathbf{x}_0)$ and trains a network $\epsilon_\theta(\mathbf{x}_t, c, t)$, or equivalently a score model, to predict the noise added to the clean sample $\mathbf{x}_0$. Generation then approximately reverses this process from noise to data through a reverse-time SDE or the associated probability flow ODE. Flow matching (Lipman et al., 2022; Albergo & Vanden-Eijnden, 2022) instead directly learns the vector field of a prescribed probability path. In this paper we use the linear interpolation

$$\mathbf{x}_t = (1 - t) \cdot \mathbf{x}_0 + t \cdot \epsilon, \qquad \epsilon \sim \mathcal{N}(0, I)\,, \tag{1}$$

so that $\mathbf{x}_0$ is the clean sample at $t = 0$ and $\mathbf{x}_1 = \epsilon$ is the noise sample at $t = 1$. The corresponding conditional target velocity is

$$u_t(\mathbf{x}_t \mid \mathbf{x}_0) = \frac{\mathrm{d}\mathbf{x}_t}{\mathrm{d}t} = \epsilon - \mathbf{x}_0\,. \tag{2}$$

Flow matching trains a velocity field $v_\theta(\mathbf{x}_t, c, t)$ to regress to this target, typically by minimizing

$$\mathcal{L}_{\text{FM}}(\theta) = \mathbb{E}_{c, \mathbf{x}_0, t, \epsilon}\left[\|v_\theta(\mathbf{x}_t, c, t) - u_t(\mathbf{x}_t \mid \mathbf{x}_0)\|_2^2\right]\,. \tag{3}$$

Although diffusion and flow matching use different parameterizations, they are closely connected: under suitable probability paths, the denoising model, score function, and velocity field describe the same underlying transport from noise to data (Lai et al., 2025; Holderrieth & Erives, 2025; Sun et al., 2025b; Lipman et al., 2024).

3.2 PREFERENCE OPTIMIZATION

RLHF (Ouyang et al., 2022) maximizes expected reward $r(\mathbf{x}_0, c)$ while regularizing a policy π_θ toward a frozen reference π_{ref}. DPO (Rafailov et al., 2023) instead learns from fixed preference pairs $\mathcal{D} = \{(c, \mathbf{x}_0^w, \mathbf{x}_0^l)\}$ using reference-relative log-likelihood ratios, where w and l denote preferred and dispreferred samples. For continuous-time generators, exact likelihood evaluation requires costly ODE integration. DiffusionDPO (Wallace et al., 2024) substitutes reference-relative denoising errors, and FlowDPO (Liu et al., 2025b) extends this surrogate to velocity-field regression. The full DPO and DiffusionDPO objectives are given in App. F.

3.3 Forward and Reverse KL Divergences

Let $\pi(\mathbf{x}_0 \mid c)$ denote a target distribution and $\pi_\theta(\mathbf{x}_0 \mid c)$ denote the trainable model distribution. Later in the paper, π will be instantiated by the preferred and dispreferred target distributions π^+ and π^-, while the model side of the objective will be represented by branch-specific distributions q_θ^+ and q_θ^- rather than by a single π_θ. The two KL directions are

$$\begin{aligned} \text{Reverse-KL: } \mathcal{D}_{KL}(\pi_\theta \| \pi) &= \mathbb{E}_{\mathbf{x}_0 \sim \pi_\theta(\cdot|c)} \left[\log \frac{\pi_\theta(\mathbf{x}_0 \mid c)}{\pi(\mathbf{x}_0 \mid c)} \right] \\ \text{Forward-KL: } \mathcal{D}_{KL}(\pi \| \pi_\theta) &= \mathbb{E}_{\mathbf{x}_0 \sim \pi(\cdot|c)} \left[\log \frac{\pi(\mathbf{x}_0 \mid c)}{\pi_\theta(\mathbf{x}_0 \mid c)} \right] \end{aligned} \tag{4}$$

The key difference is the sampling distribution inside the expectation. In the reverse-KL term $\mathcal{D}_{KL}(\pi_\theta \| \pi)$, errors are weighted by the current model π_θ, so target regions that are rarely visited by π_θ contribute little. In contrast, the forward-KL term $\mathcal{D}_{KL}(\pi \| \pi_\theta)$ is weighted by the target distribution π. If $\pi_\theta(\mathbf{x}_0 \mid c)$ is too small in regions where $\pi(\mathbf{x}_0 \mid c)$ is large, the penalty becomes severe, and finiteness requires $\mathrm{supp}(\pi) \subseteq \mathrm{supp}(\pi_\theta)$.

4 Methodology

This section develops FlowCPO from a unified divergence-based formulation of preference alignment. We first show that RLHF admits a contrastive reverse-KL interpretation, then derive its offline forward-KL counterpart and reduce the resulting objective to a practical flow matching loss.

4.1 Probabilistic Reward Formulation

Let c denote the conditioning context and $\mathbf{x}_0$ the generated content. Following control-as-inference (Levine, 2018), we introduce a binary *optimality variable* $o \in \{0, 1\}$, where $o = 1$ means $\mathbf{x}_0$ is preferred and $o = 0$ means $\mathbf{x}_0$ is dispreferred. This gives a normalized probabilistic view of any unbounded reward $r(\mathbf{x}_0, c)$:

$$\begin{aligned} p(o = 1 \mid \mathbf{x}_0, c) &= \frac{\exp(\omega \cdot r(\mathbf{x}_0, c))}{\exp(\omega \cdot r(\mathbf{x}_0, c)) + \exp(-\omega \cdot r(\mathbf{x}_0, c))} = \sigma(2\omega \cdot r(\mathbf{x}_0, c)) \\ p(o = 0 \mid \mathbf{x}_0, c) &= \frac{\exp(-\omega \cdot r(\mathbf{x}_0, c))}{\exp(\omega \cdot r(\mathbf{x}_0, c)) + \exp(-\omega \cdot r(\mathbf{x}_0, c))} = \sigma(-2\omega \cdot r(\mathbf{x}_0, c)) \end{aligned} \tag{5}$$

Here $\omega > 0$ is an inverse temperature and $\sigma(x) = \frac{1}{1+\exp(-x)}$ is the sigmoid function. Conditioning the reference policy $\pi_{\text{ref}}(\mathbf{x}_0 \mid c)$ on the two optimality events, $o = 1$ and $o = 0$, yields preferred and dispreferred target distributions:

$$\begin{aligned} \pi^+(\mathbf{x}_0 \mid c) &:= p(\mathbf{x}_0 \mid o = 1, c) = \frac{\pi_{\text{ref}}(\mathbf{x}_0 \mid c)\, p(o = 1 \mid \mathbf{x}_0, c)}{p_{\pi_{\text{ref}}}(o = 1 \mid c)} \\ \pi^-(\mathbf{x}_0 \mid c) &:= p(\mathbf{x}_0 \mid o = 0, c) = \frac{\pi_{\text{ref}}(\mathbf{x}_0 \mid c)\, p(o = 0 \mid \mathbf{x}_0, c)}{p_{\pi_{\text{ref}}}(o = 0 \mid c)} \end{aligned} \tag{6}$$

We use π^+ and π^- as the positive and negative target distributions throughout the paper. In practice, offline preference construction may only produce empirical approximations to these posteriors. We defer this distinction to App. B.1.

By inverting equation 6 and substituting into equation 5, the reward $r(\mathbf{x}_0, c)$ becomes

$$r(\mathbf{x}_0, c) = \frac{1}{2\omega} \log \frac{p(o = 1 \mid \mathbf{x}_0, c)}{p(o = 0 \mid \mathbf{x}_0, c)} = \frac{1}{2\omega} \left[\log \frac{\pi^+(\mathbf{x}_0 \mid c)}{\pi^-(\mathbf{x}_0 \mid c)} + \log \frac{p_{\pi_{\text{ref}}}(o = 1 \mid c)}{p_{\pi_{\text{ref}}}(o = 0 \mid c)} \right]. \tag{7}$$

Thus, up to a context-only constant, reward is a contrastive log-density ratio: high reward corresponds to regions where π^+ dominates π^-.

4.2 Re-formalizing Reinforcement Learning from Human Feedback

We begin with the unregularized expected-reward term $\mathbb{E}_{c, \mathbf{x}_0 \sim \pi_\theta(\cdot|c)}[r(\mathbf{x}_0, c)]$ that appears inside RLHF objectives. Using equation 7, we can rewrite this term directly in divergence form in Thm. 4.1 (see App. B.2 for the proof):

Theorem 4.1 (Expected reward as contrastive reverse KL)**.** *Up to the positive scale factor* $\frac{1}{2\omega}$ *and an additive term depending only on* c*, maximizing expected reward is equivalent to minimizing*

$$\mathbb{E}_c \left[\mathcal{D}_{KL}(\pi_\theta \| \pi^+) - \mathcal{D}_{KL}(\pi_\theta \| \pi^-) \right] . \tag{8}$$

Equation 8 shows that expected reward induces a contrastive reverse-KL term: the policy is attracted to π^+ and repelled from π^- under its own sampling distribution. Reference-policy KL penalties used in practical RLHF methods remain additional terms and are not absorbed by this identity. Due to the high computational cost of online sampling, we focus on the offline regime in this work. To derive an offline counterpart, we reverse the direction of the KL divergence in equation 8 and assume fixed sampling distributions:

$$\min_\theta \mathcal{L}_{\text{Offline}} = \mathbb{E}_c \left[\mathcal{D}_{KL}(\pi^+ \| \pi_\theta^+) + \lambda \cdot \mathcal{D}_{KL}(\pi^- \| \pi_\theta^-) \right] , \qquad \lambda \geq 0 . \tag{9}$$

Here π_θ^+ and π_θ^- are branch-specific model distributions. For FLOWCPO itself, we focus on the attractive regime $\lambda \geq 0$. In Sec. 5.3, we additionally instantiate $\lambda < 0$ under the generalized framework to analyze repulsive baselines.

To avoid evaluating flow-model likelihoods directly, we parameterize the two branches through symmetric mixed velocity fields:

$$\mu_\theta = (1-\beta)\, v_{\text{old}} + \beta\, v_\theta , \qquad \nu_\theta = (1+\beta)\, v_{\text{old}} - \beta\, v_\theta .$$

Here v_{old} is an EMA copy of the trainable field, and $\beta > 0$ controls the symmetric displacement of the two branches from that reference. The implicit distributions induced from the common Gaussian endpoint by μ_θ and ν_θ are denoted by π_θ^+ and π_θ^-. The idealized score-field interpretation of this interpolation is given in App. B.3.

The remaining bridge is from cross-entropy to regression. Each forward-KL term equals a target entropy plus an expected model negative log-likelihood; under the uniform regularity conditions in App. A, the latter is bounded by conditional flow matching objective plus a constant independent of θ. Applying this argument separately to the two branches yields Thm. 4.2.

Theorem 4.2 (A flow-matching upper bound for offline forward KL)**.** *Under the regularity conditions in App. A, we have:*

$$\begin{aligned} \mathcal{L}_{\textit{Offline}}(\theta) \leq C &+ \mathbb{E}_{c,\, \mathbf{x}_0^+ \sim \pi^+(\cdot|c) \atop t,\, \epsilon} \left[\left\| \mu_\theta(\mathbf{x}_t^+, t, c) - u_t(\mathbf{x}_t^+ \mid \mathbf{x}_0^+) \right\|_2^2 \right] \\ &+ \lambda \cdot \mathbb{E}_{c,\, \mathbf{x}_0^- \sim \pi^-(\cdot|c) \atop t,\, \epsilon} \left[\left\| \nu_\theta(\mathbf{x}_t^-, t, c) - u_t(\mathbf{x}_t^- \mid \mathbf{x}_0^-) \right\|_2^2 \right] , \end{aligned} \tag{10}$$

where C *is independent of* θ*.*

The proof of Thm. 4.2 is deferred to App. B.4. This result justifies a tractable upper-bound surrogate. In practice, we replace the idealized source distributions π^+, π^- with empirical preferred and dispreferred datasets, $\mathcal{D}^+$ and $\mathcal{D}^-$, and minimize the following contrastive flow matching loss:

$$\begin{aligned} \min_\theta \mathcal{L}_{\text{FLOWCPO}}(\theta) = \quad & \mathbb{E}_{(c, \mathbf{x}_0^w) \sim \mathcal{D}^+ \atop t, \epsilon} \left[\left\| \mu_\theta(\mathbf{x}_t^w, t, c) - u_t(\mathbf{x}_t^w \mid \mathbf{x}_0^w) \right\|_2^2 \right] \\ & + \lambda \cdot \mathbb{E}_{(c, \mathbf{x}_0^l) \sim \mathcal{D}^- \atop t, \epsilon} \left[\left\| \nu_\theta(\mathbf{x}_t^l, t, c) - u_t(\mathbf{x}_t^l \mid \mathbf{x}_0^l) \right\|_2^2 \right] . \end{aligned} \tag{11}$$

Pseudo-code is given in Alg. 1. For $\lambda \geq 0$, equation 11 is bounded below by zero, whereas simplified FlowDPO's signed objective can be unbounded below. App. B.5 provides the comparison.

4.3 UNIFIED VIEW AND CONNECTIONS TO PRIOR WORK

Tab. 2 summarizes connections between equation 12 and prior works, and App. C elaborates the assumptions behind each mapping. FLOWCPO is derived from the forward-KL objective in equation 9, which is a special case of equation 12. The generalized objective organizes flow-based

Table 2: **Comparison of flow-based alignment methods and related special cases under equation 12.** Rows are organized by optimization regime and divergence choice. BT indicates Bradley–Terry preference model, and Ref. KL indicates whether the method uses reference-model KL regularization.

Method	Regime	Divergence	Optimized Process	α	γ	q_θ^+	q_θ^-	BT	Ref. KL
FlowGRPO	Online	Reverse KL	Reverse Process	1	-1	π_θ	π_θ	×	✓
DiffusionNFT	Online	Forward KL	Forward Process	$p(o=1\|c)$	$p(o=0\|c)$	π_θ^+	π_θ^-	×	×
AWM	Online	Reverse KL	Forward Process	1	-1	π_θ	π_θ	×	×
RAM	Online	Reverse KL	Forward Process	1	-1	π_θ	π_θ	×	✓
SFT	Offline	Forward KL	-	$p(o=1\|c)$	$p(o=0\|c)$	π_θ	π_θ	×	×
RFT	Offline	Forward KL	-	1	0	π_θ	–	×	×
FlowDPO	Offline	Implicit reverse KL	-	1	-1	π_θ	π_θ	✓	✓
FLOWCPO	**Offline**	**Forward KL**	-	**1**	$\boldsymbol{\lambda}$	$\boldsymbol{\pi_\theta^+}$	$\boldsymbol{\pi_\theta^-}$	×	×

preference-alignment methods (FlowDPO, FlowGRPO, DiffusionNFT) and regression baselines (SFT, RFT):

$$\min_\theta \mathcal{L}_{\text{General}} = \mathbb{E}_c \left[\alpha \cdot \underbrace{\mathcal{D}_f(\pi^+ \| q_\theta^+)}_{\text{positive alignment}} + \gamma \cdot \underbrace{\mathcal{D}_f(\pi^- \| q_\theta^-)}_{\text{negative regularization}} \right] \tag{12}$$

Here $\alpha > 0$, and $\gamma \in \mathbb{R}$. In the forward-KL branch we use $\mathcal{D}_f(\pi^\pm \| q) = \mathcal{D}_{KL}(\pi^\pm \| q)$, and in the reverse-KL branch we use $\mathcal{D}_f(\pi^\pm \| q) = \mathcal{D}_{KL}(q \| \pi^\pm)$. When $\gamma \geq 0$, the negative regularization matches a negative distribution, and it acts as repulsive regularization when $\gamma < 0$.

5 EXPERIMENTS

We study FLOWCPO along three axes: offline alignment with in-domain and out-of-domain preference data, the effect of negative regularization, and sensitivity to the core hyperparameters (β, λ, η). The main text focuses on the central quantitative and qualitative results, while the appendix collects the training pseudo-code, per-domain tables, and extended ablations.

5.1 EXPERIMENTAL SETUP

Data Construction and Implementation. We evaluate FLOWCPO on `Stable Diffusion 3.5 Medium (SD3.5-M)` (Esser et al., 2024) in a strictly offline setting. Our offline preference data come from two sources. In the *in-domain* regime, we build winner/loser pairs from a frozen copy of the reference model using prompts drawn from the target benchmark sources. In the *out-of-domain* (OOD) regime, we use the public Open Image Preferences v1 dataset, whose images are generated by other open models. All models are trained under the same offline protocol, and we defer the exact data-generation pipeline, dataset sizes, reward filtering, and training hyperparameters to App. D.1. In the in-domain setting, each target capability uses its own specialist model, so the Tab. 3 summarizes task-specific fine-tuning results rather than a single universal model.

Baselines. We compare against the two published approaches that are most directly comparable under the same fixed-data protocol:

- **RFT (Rejection Sampling Fine-Tuning)** (Xiong et al., 2025; Chen et al., 2025): A positive-only offline baseline that can be viewed as a forward-KL special case, learning exclusively from preferred data and ignoring dispreferred data.
- **FlowDPO** (Liu et al., 2025b): A flow-based DPO baseline that uses preferred/dispreferred pairs through a reference-relative likelihood-ratio surrogate.

Evaluation Metrics. We evaluate two kinds of behavior. For targeted capabilities, we use task-specific benchmarks: `GenEval` (Ghosh et al., 2023) for multi-concept compositional generation and an `OCR`-based metric for visual text rendering. For general preference alignment, we run inference on

Table 3: **Quantitative comparison under *in-domain* and *out-of-domain* (OOD) offline training regimes** on `SD3.5-M`. We report results with and without Classifier-Free Guidance (CFG). When CFG is enabled, we report separate rows for guidance scales $\{1.0, 3.0, 4.5\}$. Under the *in-domain* setting, fine-tuned methods are trained separately for each target domain (Concept, Typography, and General Preference). Under the *OOD* setting, a single model is trained on the OOD offline dataset and evaluated across all domains and metrics. The two blocks therefore correspond to different evaluation protocols and should be compared primarily within, rather than across, regimes. The main table reports means only and rounds all entries to 2-3 decimal places; the corresponding mean $\pm$ standard deviation statistics over 5 independent runs for fine-tuned `SD3.5-M` variants are reported in the App. D.2, whereas pretrained baselines are single evaluations. Within each training regime, the best result is highlighted in **bold** and the second-best result is underlined.

		Concept	Typography	General Preference					
Model	**CFG**	**GenEval ↑**	**OCR ↑**	**PickScore ↑**	**CLIPSc. ↑**	**HPSv2.1 ↑**	**Aes. ↑**	**ImgRwd ↑**	**UniRwd ↑**
Pretrained model baselines									
SD-XL	–	0.55	0.14	22.42	0.287	0.280	5.60	0.76	2.93
SD3.5-L	–	0.71	0.68	22.91	0.289	0.288	5.50	0.96	3.25
FLUX.1-Dev	–	0.66	0.59	22.84	0.295	0.274	5.71	0.96	3.27
Reference model: SD3.5-M									
Base Model	1.0	0.24	0.12	20.51	0.237	0.204	5.13	−0.58	2.02
	3.0	0.59	0.47	22.28	0.287	0.284	5.38	0.71	2.96
	4.5	0.63	0.59	22.34	0.285	0.279	5.36	0.85	3.03
In-domain offline fine-tuning on SD3.5-M									
+ RFT	1.0	0.59	0.35	21.91	0.279	0.276	5.35	0.57	2.66
	3.0	0.74	0.70	22.60	0.296	0.304	5.41	1.06	3.14
	4.5	0.75	0.72	22.57	0.297	0.305	5.42	1.11	3.15
+ FlowDPO	1.0	0.59	0.51	20.72	0.240	0.221	5.22	−0.46	2.11
	3.0	0.81	0.74	22.76	0.297	0.301	5.56	1.04	3.10
	4.5	0.81	0.75	22.89	**0.301**	**0.311**	5.56	1.15	3.18
+ FlowCPO ($\beta = 0.5$, Ours)	1.0	0.76	0.83	22.48	0.280	0.290	**5.64**	0.90	2.93
	3.0	**0.84**	**0.87**	**22.94**	0.299	**0.311**	5.54	**1.25**	**3.30**
	4.5	0.82	0.86	22.67	0.300	0.303	5.46	1.22	3.28
Out-of-domain (OOD) offline fine-tuning on SD3.5-M									
+ RFT	1.0	0.44	0.17	21.59	0.268	0.261	5.49	0.31	2.56
	3.0	0.67	0.53	22.64	0.297	0.303	5.48	1.05	3.18
	4.5	0.68	**0.60**	**22.69**	**0.299**	**0.307**	5.47	**1.12**	**3.23**
+ FlowDPO	1.0	0.23	0.12	20.83	0.243	0.226	**5.72**	−0.39	2.27
	3.0	0.61	0.48	22.54	0.292	0.292	5.52	0.91	3.10
	4.5	0.65	0.54	22.62	0.295	0.299	5.49	0.99	3.17
+ FlowCPO ($\beta = 0.5$, Ours)	1.0	0.57	0.24	22.02	0.281	0.275	5.35	0.63	2.85
	3.0	0.69	0.49	22.32	0.296	0.288	5.37	1.00	3.15
	4.5	0.68	0.49	22.12	0.294	0.284	5.31	0.95	3.11
+ FlowCPO ($\beta = 1$, Ours)	1.0	0.47	0.24	21.81	0.276	0.270	5.35	0.52	2.77
	3.0	**0.70**	0.56	22.46	0.297	0.295	5.41	1.02	**3.23**
	4.5	**0.70**	0.59	22.36	0.294	0.294	5.38	1.02	3.22

the `DrawBench` prompt set and report `PickScore` (Kirstain et al., 2023), `CLIP Score` (Hessel et al., 2021), `HPS v2.1` (Wu et al., 2023), `Aesthetics` (Schuhmann, 2022), `ImageReward` (Xu et al., 2023), and `UnifiedReward` (Wang et al., 2025). Because the in-domain general-preference subset is filtered with `PickScore`, `CLIP Score`, and `HPS v2.1`, we treat these as optimization-aligned metrics and use `Aesthetics`, `ImageReward`, and `UnifiedReward` as a cleaner check of transfer beyond the filtering pipeline.

5.2 RQ1: Main Results with In-Domain and Out-of-Domain Data

To evaluate method-level performance, we study three capability groups: semantic alignment, typographic generation, and general preference.

We summarize the main quantitative results in Tab. 3. The two regimes answer different questions, so we read them separately: the *in-domain* block evaluates in-domain offline fine-tuning, whereas the *out-of-domain* (OOD) block evaluates a single model trained once on an OOD offline dataset.

- **In the *in-domain* setting, FlowCPO has higher mean GenEval and OCR scores than RFT and FlowDPO.** Tab. 3 aggregates separate task-specific fine-tunes for the three capability groups; FlowCPO attains the best GenEval mean (**0.84** at CFG 3.0) and the best OCR mean (**0.87** at CFG 3.0), exceeding FlowDPO by 0.03 and 0.12, respectively. On general preference, the picture is narrower but still favorable: FlowCPO is best or tied-best on `PickScore`, `HPS`

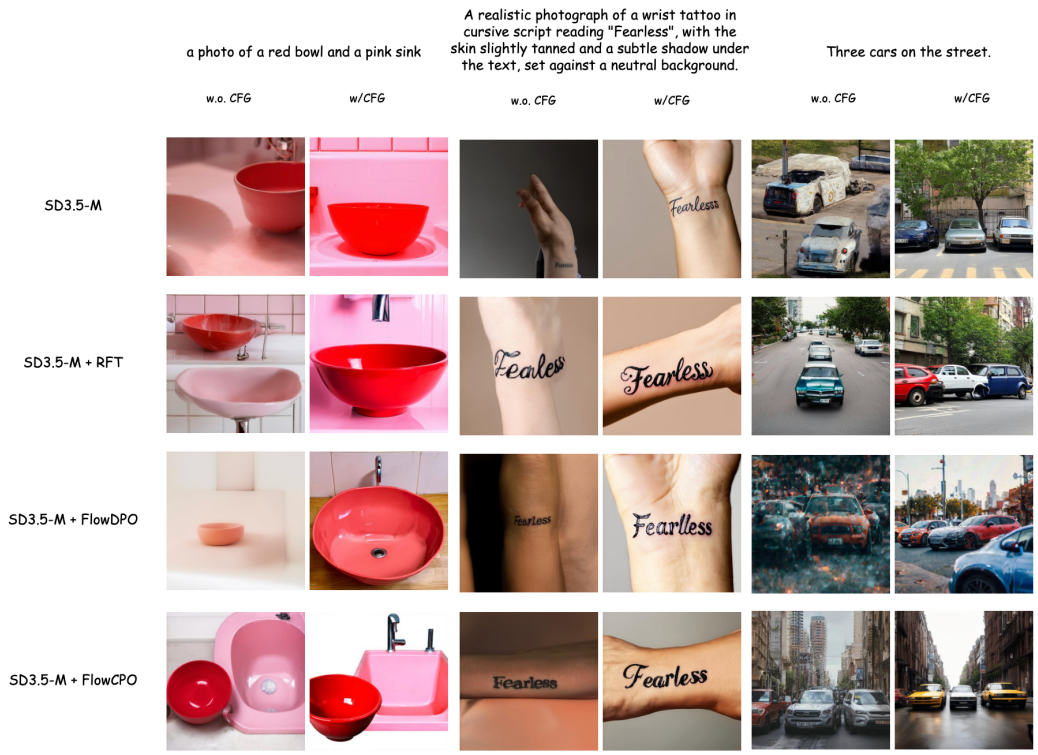


Figure 2: **Representative qualitative comparison of FLOWCPO and the baselines** across three preference optimization tasks.

`v2.1`, `ImageReward`, and `UnifiedReward`, while `CLIP Score` is effectively tied with FlowDPO. Since `PickScore`, `CLIP Score`, and `HPS v2.1` also appear in the in-domain filtering pipeline, we put more weight on `Aesthetics`, `ImageReward`, and `UnifiedReward`, together with the qualitative comparison in Fig. 2 and the extended examples in App. D.2, when judging transfer beyond the optimization-aligned signals.

- **In the *out-of-domain* setting, the advantage is metric-dependent.** FLOWCPO gives the best GenEval score and remains competitive on OCR and `UnifiedReward`, whereas RFT performs better on several reward-model metrics. Thus, retaining both sides of each preference pair does not guarantee improvement when the fixed data are generated by models other than the reference model. This result motivates the regularization analysis in RQ2 and identifies out-of-domain data as an empirical boundary of the method.

5.3 RQ2: EFFECTIVENESS OF DIFFERENT NEGATIVE REGULARIZATION

We compare positive-only training with repulsive ($\lambda < 0$) and attractive ($\lambda > 0$) negative regularization under the unified framework.

Tab. 4 **favors moderate attractive matching**: $\beta = 0.5, \lambda = 1$ achieves the best GenEval score (0.84), while $\beta = 1, \lambda = 0.1$ gives the best OCR, `ImageReward`, and `UnifiedReward`. Repulsion with $\lambda = -1$ sharply degrades all metrics, and both signs diverge at $|\lambda| = 10$. Since training optimizes only GenEval, the other gains measure cross-metric transfer.

5.4 RQ3: SENSITIVITY OF FLOWCPO TO ITS CORE HYPERPARAMETERS

Fig. 9 favors moderate interpolation ($\beta \in [0.5, 1.0]$), balanced negative weight ($\lambda = 1$), and slow EMA ($\eta = 0.99$). Extreme weights destabilize training, while small weights weaken the negative contribution. We therefore use $\beta = 0.5$, $\lambda = 1$, and $\eta = 0.99$ as the default settings. Complete sweeps appear in App. D.3.2.

Table 4: **Analysis of different regularization mechanisms** on `SD3.5-M` under the unified divergence-based framework of equation 12, with all models trained only on *GenEval*. We consider three representative paradigms: positive-only regularization, which uses only preferred samples; repulsive negative regularization, which explicitly pushes the model away from dispreferred regions; and attractive negative regularization, which incorporates negative samples through distributional matching. For attractive negative regularization, we report two instantiations with different mixing coefficients β. Results are reported with and without Classifier-Free Guidance (CFG). Metrics are grouped into *Concept*, *Typography*, and *General Preference* for consistency with the main results table and to assess cross-metric generalization beyond the training signal. Best results are highlighted in **bold** and second-best results are underlined.

λ	CFG	Concept	Typography	General Preference					
		GenEval ↑	OCR ↑	PickScore ↑	CLIPSc. ↑	HPSv2.1 ↑	Aes. ↑	ImgRwd ↑	UniRwd ↑
Positive-only regularization (RFT-like): $\mu_\theta = v_\theta,\ \nu_\theta = 0$									
0	1.0	0.59	0.15	21.67	0.27	0.268	5.32	0.43	2.63
0	3.0	0.72	0.51	**22.5**	0.295	0.301	**5.41**	1.03	3.13
0	4.5	0.75	0.56	**22.5**	0.296	**0.304**	**5.41**	1.08	3.16
Repulsive negative regularization (DPO-like): $\mu_\theta = \nu_\theta = v_\theta$									
−0.1	1.0	0.64	0.15	21.76	0.277	0.270	5.30	0.53	2.66
−0.1	3.0	0.77	0.50	22.48	0.297	0.300	5.37	1.07	3.17
−0.1	4.5	0.76	0.58	22.47	0.297	**0.304**	5.40	1.10	3.17
−1	1.0	0.25	0.08	19.46	0.201	0.154	4.22	-1.66	1.40
−1	3.0	0.70	0.48	21.39	0.270	0.229	4.70	-0.28	2.51
−1	4.5	0.71	0.55	21.50	0.276	0.240	4.86	-0.01	2.65
−10	–	Diverged	Diverged	Diverged	Diverged	Diverged	Diverged	Diverged	Diverged
Attractive negative regularization (CPO-like): $\mu_\theta = v_\theta,\ \nu_\theta = 2v_{\text{old}} - v_\theta$									
0.1	1.0	0.64	0.17	21.72	0.278	0.266	5.24	0.50	2.71
0.1	3.0	0.77	0.54	22.47	0.297	0.301	5.35	1.06	3.19
0.1	4.5	0.77	**0.59**	22.48	0.298	**0.304**	5.38	**1.14**	**3.23**
1	1.0	0.78	0.25	21.74	0.280	0.258	5.23	0.57	2.74
1	3.0	0.78	0.51	21.69	0.291	0.266	5.09	0.76	2.96
1	4.5	0.67	0.55	21.20	0.285	0.245	4.89	0.43	2.75
10	–	Diverged	Diverged	Diverged	Diverged	Diverged	Diverged	Diverged	Diverged
Attractive negative regularization (CPO-like): $\mu_\theta = 0.5v_{\text{old}} + 0.5v_\theta,\ \nu_\theta = 1.5v_{\text{old}} - 0.5v_\theta$									
0.1	1.0	0.63	0.17	21.48	0.275	0.259	5.22	0.33	2.55
0.1	3.0	0.78	0.51	22.41	0.298	0.297	5.33	1.04	3.14
0.1	4.5	0.77	0.58	22.43	**0.299**	0.299	5.35	1.10	3.19
1	1.0	0.76	0.26	21.84	0.280	0.268	5.26	0.64	2.79
1	3.0	**0.84**	0.53	22.20	0.296	0.288	5.27	1.01	3.10
1	4.5	0.83	0.55	21.94	0.294	0.282	5.16	0.95	3.04
10	–	Diverged	Diverged	Diverged	Diverged	Diverged	Diverged	Diverged	Diverged

6 Conclusion and Limitations

We presented a divergence-based framework that separates sampling regime from divergence direction and derived FlowCPO as its offline forward-KL instance. Coupled preferred and dispreferred branches enable regression on fixed preference pairs without online rollouts. Our loss decomposition also connects FlowCPO to simplified FlowDPO and shows how the difference between the current model and its EMA enters the objective. The nonnegative loss avoids simplified FlowDPO's potentially unbounded negative regression term.

On SD3.5-M, FlowCPO achieves higher mean in-domain GenEval and OCR scores than RFT and FlowDPO, reaching 0.84 and 0.87 at CFG 3.0. Out-of-domain gains are mixed, so the present results do not establish consistent transfer across generators. The ablations favor moderate attractive matching of dispreferred samples, while excessive weights can destabilize training. The diversity analysis further shows that stronger guidance improves target quality at the cost of per-prompt diversity.

The forward-KL bound assumes linear interpolation and uniform regularity, and empirical preference pairs need not match its idealized target distributions. Loss nonnegativity alone does not guarantee stable training. Our fine-tuning evaluation is limited to SD3.5-M; performance on other backbones remains untested. Extending the bound to other paths, improving robustness to data mismatch, and better balancing quality and diversity are concrete directions for future work.

AI USE STATEMENT

Generative AI tools were used to edit the manuscript text for clarity and readability. We also used generative AI to draft portions of the theoretical proofs, which the authors subsequently revised manually to improve readability. During preparation of this submission, an OpenAI Codex assistant was also used to adapt the existing manuscript to the ICLR 2027 template, edit submission-specific text, prepare the accompanying statements, and check compilation and layout. The authors are responsible for the final content of this work, including any AI-assisted text or artifacts.

ETHICS STATEMENT

Preference alignment can improve the usefulness and controllability of text-to-image models, but it can also facilitate deceptive synthetic content and inherit biases from preference datasets and automated reward models. Our evaluation documents mixed out-of-domain results and does not establish safety or fairness guarantees. We discuss these risks and release considerations in App. E. The experiments use existing models and datasets under their original access conditions; any future release should preserve the upstream licenses and usage restrictions.

REPRODUCIBILITY STATEMENT

Sec. 4 specifies the training objective, and Alg. 1 gives the training procedure. App. A and App. B.4 state the assumptions and proofs supporting the flow-matching bound; App. C details connections to prior methods. App. D.1 documents data construction, optimization, and checkpoint selection, while App. D.2 reports per-domain results and run-level uncertainty. Further ablations appear in App. D.3.2. No new code package, dataset, or model checkpoints accompany this submission.

REFERENCES

Michael S Albergo and Eric Vanden-Eijnden. Building normalizing flows with stochastic interpolants. *arXiv preprint arXiv:2209.15571*, 2022.

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

Andreas Bergmeister, Stefanie Jegelka, Nikolas Nüsken, Carles Domingo-Enrich, and Jakiw Pidstrigach. Reinforce adjoint matching: Scaling rl post-training of diffusion and flow-matching models. *arXiv preprint arXiv:2605.10759*, 2026.

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*, 2024.

Huayu Chen, Kaiwen Zheng, Qinsheng Zhang, Ganqu Cui, Yin Cui, Haotian Ye, Tsung-Yi Lin, Ming-Yu Liu, Jun Zhu, and Haoxiang Wang. Bridging supervised learning and reinforcement learning in math reasoning. *arXiv preprint arXiv:2505.18116v1*, 2025.

Kevin Clark, Paul Vicol, Kevin Swersky, and David J. Fleet. Directly fine-tuning diffusion models on differentiable rewards. In *The Twelfth International Conference on Learning Representations*, 2024.

Data Is Better Together. Open image preferences v1 results. https://huggingface.co/datasets/data-is-better-together/open-image-preferences-v1-results, 2024. Hugging Face dataset, accessed March 26, 2026.

Zheng Ding and Weirui Ye. Treegrpo: Tree-advantage grpo for online rl post-training of diffusion models. *arXiv preprint arXiv:2512.08153*, 2025.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024.

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. In *Advances in Neural Information Processing Systems*, volume 36, pp. 79858–79885, 2023.

Dan Friedman and Adji Bousso Dieng. The vendi score: A diversity evaluation metric for machine learning. *Transactions on Machine Learning Research*, 2023. URL https://openreview.net/forum?id=g97OHbQyk1.

Dhruba Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. Geneval: An object-focused framework for evaluating text-to-image alignment. *Advances in Neural Information Processing Systems*, 36: 52132–52152, 2023.

Dongyoung Go, Tomasz Korbak, Germán Kruszewski, Jos Rozen, Nahyeon Ryu, and Marc Dymetman. Aligning language models with preferences through f-divergence minimization. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 11546–11583. PMLR, 2023. URL https://proceedings.mlr.press/v202/go23a.html.

Jiaqi Han, Mingjian Jiang, Yuxuan Song, Stefano Ermon, and Minkai Xu. f-po: Generalizing preference optimization with f-divergence minimization. In *Proceedings of the 28th International Conference on Artificial Intelligence and Statistics*, volume 258 of *Proceedings of Machine Learning Research*, pp. 1144–1152. PMLR, 2025. URL https://proceedings.mlr.press/v258/han25a.html.

Yansen Han, Hongxin Sun, and Tao Lin. When can conditional flow matching replace pointwise negative log-likelihood? *arXiv preprint arXiv:2608.28010*, 2026.

Xiaoxuan He, Siming Fu, Yuke Zhao, Wanli Li, Jian Yang, Dacheng Yin, Fengyun Rao, and Bo Zhang. Tempflow-grpo: When timing matters for grpo in flow models. *arXiv preprint arXiv:2508.04324*, 2025.

Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. Clipscore: A reference-free evaluation metric for image captioning. In *Proceedings of the 2021 conference on empirical methods in natural language processing*, pp. 7514–7528, 2021.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Peter Holderrieth and Ezra Erives. An introduction to flow matching and diffusion models. *arXiv preprint arXiv:2506.02070*, 2025.

Jiwoo Hong, Sayak Paul, Noah Lee, Kashif Rasul, James Thorne, and Jongheon Jeong. Margin-aware preference optimization for aligning diffusion models without reference. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 4744–4752, 2026. doi: 10.1609/aaai.v40i6.42476.

Haozhe Ji, Cheng Lu, Yilin Niu, Pei Ke, Hongning Wang, Jun Zhu, Jie Tang, and Minlie Huang. Towards efficient exact optimization of language model alignment. *arXiv preprint arXiv:2402.00856*, 2024.

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. *Advances in neural information processing systems*, 36:36652–36663, 2023.

Chieh-Hsin Lai, Yang Song, Dongjun Kim, Yuki Mitsufuji, and Stefano Ermon. The principles of diffusion models. *arXiv preprint arXiv:2510.21890*, 2025.

Sergey Levine. Reinforcement learning and control as probabilistic inference: Tutorial and review. *arXiv preprint arXiv:1805.00909*, 2018.

Binxu Li, Minkai Xu, Jiaqi Han, Meihua Dang, and Stefano Ermon. Divergence minimization preference optimization for diffusion model alignment. *arXiv preprint arXiv:2507.07510*, 2025a.

Junzhe Li, Yutao Cui, Tao Huang, Yinping Ma, Chun Fan, Miles Yang, and Zhao Zhong. Mixgrpo: Unlocking flow-based grpo efficiency with mixed ode-sde. *arXiv preprint arXiv:2507.21802v1*, 2025b.

Yuming Li, Yikai Wang, Yuying Zhu, Zhongyu Zhao, Ming Lu, Qi She, and Shanghang Zhang. Branchgrpo: Stable and efficient grpo with structured branching in diffusion models. *arXiv preprint arXiv:2509.06040*, 2025c.

Zhanhao Liang, Yuhui Yuan, Shuyang Gu, Bohan Chen, Tiankai Hang, Mingxi Cheng, Ji Li, and Liang Zheng. Aesthetic post-training diffusion models from generic preferences with step-by-step preference optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13199–13208, 2025.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Yaron Lipman, Marton Havasi, Peter Holderrieth, Neta Shaul, Matt Le, Brian Karrer, Ricky TQ Chen, David Lopez-Paz, Heli Ben-Hamu, and Itai Gat. Flow matching guide and code. *arXiv preprint arXiv:2412.06264*, 2024.

Jie Liu, Gongye Liu, Jiajun Liang, Yangguang Li, Jiaheng Liu, Xintao Wang, Pengfei Wan, Di Zhang, and Wanli Ouyang. Flow-grpo: Training flow matching models via online rl. *arXiv preprint arXiv:2505.05470*, 2025a.

Jie Liu, Gongye Liu, Jiajun Liang, Ziyang Yuan, Xiaokun Liu, Mingwu Zheng, Xiele Wu, Qiulin Wang, Menghan Xia, Xintao Wang, et al. Improving video generation with human feedback. *arXiv preprint arXiv:2501.13918*, 2025b.

Cheng Lu, Kaiwen Zheng, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Maximum likelihood training for score-based diffusion ODEs by high-order denoising score matching. *arXiv preprint arXiv:2206.08265*, 2022.

Calvin Luo. Understanding diffusion models: A unified perspective. *arXiv preprint arXiv:2208.11970*, 2022.

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

Mihir Prabhudesai, Anirudh Goyal, Deepak Pathak, and Katerina Fragkiadaki. Aligning text-to-image diffusion models with reward backpropagation. *arXiv preprint arXiv:2310.03739v2*, 2023. Withdrawn in 2024; subsumed by arXiv:2407.08737.

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

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

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pp. 2256–2265. pmlr, 2015.

Yang Song, Conor Durkan, Iain Murray, and Stefano Ermon. Maximum likelihood training of score-based diffusion models. *Advances in neural information processing systems*, 34:1415–1428, 2021.

Huashan Sun, Shengyi Liao, Yansen Han, Yu Bai, Yang Gao, Cheng Fu, Weizhou Shen, Fanqi Wan, Ming Yan, Ji Zhang, et al. Solopo: Unlocking long-context capabilities in llms via short-to-long preference optimization. *arXiv preprint arXiv:2505.11166*, 2025a.

Peng Sun, Yi Jiang, and Tao Lin. Unified continuous generative models. *arXiv preprint arXiv:2505.07447*, 2025b.

Yunhao Tang, Zhaohan Daniel Guo, Zeyu Zheng, Daniele Calandriello, Rémi Munos, Mark Rowland, Pierre Harvey Richemond, Michal Valko, Bernardo Ávila Pires, and Bilal Piot. Generalized preference optimization: A unified approach to offline alignment. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 47725–47742. PMLR, 2024. URL https://proceedings.mlr.press/v235/tang24b.html.

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

Chaoqi Wang, Yibo Jiang, Chenghao Yang, Han Liu, and Yuxin Chen. Beyond reverse KL: Generalizing direct preference optimization with diverse divergence constraints. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=2cRzmWXK9N.

Yibin Wang, Yuhang Zang, Hao Li, Cheng Jin, and Jiaqi Wang. Unified reward model for multimodal understanding and generation. *arXiv preprint arXiv:2503.05236*, 2025.

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

Wei Xiong, Jiarui Yao, Yuhui Xu, Bo Pang, Lei Wang, Doyen Sahoo, Junnan Li, Nan Jiang, Tong Zhang, Caiming Xiong, et al. A minimalist approach to llm reasoning: from rejection sampling to reinforce. *arXiv preprint arXiv:2504.11343*, 2025.

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

Shuchen Xue, Chongjian Ge, Shilong Zhang, Yichen Li, and Zhi-Ming Ma. Advantage weighted matching: Aligning rl with pretraining in diffusion models. *arXiv preprint arXiv:2509.25050*, 2025.

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

Shentao Yang, Tianqi Chen, and Mingyuan Zhou. A dense reward view on aligning text-to-image diffusion with preference. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 55998–56032, 2024b.

Yao Zhao, Rishabh Joshi, Tianqi Liu, Misha Khalman, Mohammad Saleh, and Peter J Liu. Slic-hf: Sequence likelihood calibration with human feedback. *arXiv preprint arXiv:2305.10425*, 2023.

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

CONTENTS

1 Introduction **1**

2 Related Work **2**

3 Preliminaries **3**
3.1 Diffusion Models and Flow Matching . . . 3
3.2 Preference Optimization . . . 3
3.3 Forward and Reverse KL Divergences . . . 4

4 Methodology **4**
4.1 Probabilistic Reward Formulation . . . 4
4.2 Re-formalizing Reinforcement Learning from Human Feedback . . . 4
4.3 Unified View and Connections to Prior Work . . . 5

5 Experiments **6**
5.1 Experimental Setup . . . 6
5.2 RQ1: Main Results with In-Domain and Out-of-Domain Data . . . 7
5.3 RQ2: Effectiveness of Different Negative Regularization . . . 8
5.4 RQ3: Sensitivity of FLOWCPO to Its Core Hyperparameters . . . 8

6 Conclusion and Limitations **9**

A Theoretical Relation between Log-Likelihood and Flow Matching Loss **15**
A.1 Pointwise Relation between Log-Likelihood and Flow Matching Loss . . . 15
A.2 Expectation-level Relation between Log-Likelihood and Flow Matching Loss . . . 15

B Theoretical Proofs and Results **20**
B.1 Idealized Posteriors and Empirical Source Distributions . . . 20
B.2 Proof of Theorem 4.1 . . . 20
B.3 An Idealized Interpretation of Interpolated Vector Fields . . . 20
B.4 Proof of Theorem 4.2 . . . 21
B.5 Relation between FlowCPO and Simplified FlowDPO . . . 21

C Connections to Existing Methods Under the Unified Divergence Framework **24**
C.1 Connection to FlowGRPO . . . 24
C.2 Connection to DiffusionNFT . . . 24
C.3 Connection to AWM . . . 24
C.4 Connection to RAM . . . 25
C.5 Connection to SFT . . . 25
C.6 Connection to RFT . . . 26
C.7 Connection to FlowDPO . . . 26

D Experimental Details **28**
D.1 Experimental Details of Sec. 5.2 . . . 28
D.2 Experimental Results on Optimization Targets . . . 29
D.3 Additional Analyses Beyond Main Results . . . 37
D.3.1 Understanding the Role of Negative Regularization . . . 37
D.3.2 Hyperparameter Sensitivity and Ablation . . . 37
D.4 Per-Prompt Diversity Evaluation . . . 37

E Broader Impacts and Release Considerations **38**

F Additional Background on Preference Optimization **39**
F.1 RLHF and Direct Preference Optimization (DPO) . . . 39
F.2 Preference Optimization in Continuous-Time Models . . . 39

A THEORETICAL RELATION BETWEEN LOG-LIKELIHOOD AND FLOW MATCHING LOSS

In this section, we discuss the relationship between log-likelihood and flow matching loss. The theoretical results mainly follow from (Song et al., 2021; Han et al., 2026; Lu et al., 2022).

A.1 POINTWISE RELATION BETWEEN LOG-LIKELIHOOD AND FLOW MATCHING LOSS

The theoretical results in this subsection are from (Han et al., 2026). Fix a sample $x_0 \in \mathbb{R}^d$,

$$\mathbf{x}_t = (1-t)x_0 + t\mathbf{x}_1 \sim q_t^{x_0} = \mathcal{N}((1-t)x_0, t^2 I_d), \qquad \mathbf{x}_1 \sim \mathcal{N}(0, I_d).$$

Let $u_t^{x_0}(\mathbf{x}) = \frac{\mathbf{x}-x_0}{t}$ be the velocity field corresponding to $q_t^{x_0}$. We compare this conditional path $(q_t^{x_0}, u_t^{x_0})$ to the model path (p_t^θ, v_θ) through the velocity and score gaps:

$$\Delta v_t^\theta(\mathbf{x}; x_0) := v_\theta(\mathbf{x}, t) - u_t^{x_0}(\mathbf{x}), \qquad \Delta s_t^\theta(\mathbf{x}; x_0) := \nabla \log q_t^{x_0}(\mathbf{x}) - \nabla \log p_t^\theta(\mathbf{x})$$

we also define the following terms:

$$\ell_\varepsilon(\theta; x_0) := \mathbb{E}_{\mathbf{x}_\varepsilon \sim q_\varepsilon^{x_0}}\left[-\log p_\varepsilon^\theta(\mathbf{x}_\varepsilon)\right], \quad \mathcal{J}_w^{[\varepsilon,1]}(\theta; x_0) := \int_\varepsilon^1 w(t)\, \mathbb{E}_{\mathbf{x}_t \sim q_t^{x_0}}\left[\|\Delta v_t^\theta(\mathbf{x}_t; x_0)\|^2\right] dt$$

Assumption A.1 (Pathwise regularity). *For the fixed x_0, the densities $q_t^{x_0}$ and p_t^θ are strictly positive and C^1 in x for $t \in (0,1)$. Their continuity equations hold in strong form, $t \mapsto \mathrm{KL}(q_t^{x_0} \| p_t^\theta)$ is differentiable, and the integrations by parts used below have no boundary terms.*

Assumption A.2 (Endpoint regularity). *The map $f_\theta(x,t) := -\log p_t^\theta(x)$ is jointly continuous at $(x_0, 0)$. Moreover, for some $\varepsilon_0 > 0$, $C > 0$, and $m \geq 1$,*

$$|f_\theta(x,t)| \leq C(1 + \|x\|^m), \qquad x \in \mathbb{R}^d, \quad t \in [0, \varepsilon_0].$$

Theorem A.3 (Pointwise NLL is a CFM term plus residuals (Han et al., 2026)). *Suppose Assump. A.1 and Assump. A.2 hold, and suppose that $p_1^\theta = q_1^{x_0} = \mathcal{N}(0, I_d)$. Then, for every positive weight w and every $\varepsilon \in (0,1)$,*

$$\ell_\varepsilon(\theta; x_0) = H(q_\varepsilon^{x_0}) + \int_\varepsilon^1 \mathbb{E}_{q_t^{x_0}}\left[\langle \Delta v_t^\theta(\mathbf{x}_t; x_0), \Delta s_t^\theta(\mathbf{x}_t; x_0)\rangle\right] dt, \tag{13}$$

$$-\log p_0^\theta(x_0) = H(q_\varepsilon^{x_0}) + \mathcal{J}_w^{[\varepsilon,1]}(\theta; x_0) + \mathcal{G}_{\varepsilon,w}(\theta; x_0) + \mathcal{B}_\varepsilon(\theta; x_0), \tag{14}$$

where

$$\mathcal{G}_{\varepsilon,w}(\theta; x_0) := \int_\varepsilon^1 \mathbb{E}_{q_t^{x_0}}\left[\langle \Delta v_t^\theta(\mathbf{x}_t; x_0), \Delta s_t^\theta(\mathbf{x}_t; x_0) - w(t)\Delta v_t^\theta(\mathbf{x}_t; x_0)\rangle\right] dt,$$

$$H(q_\varepsilon^{x_0}) = \frac{d}{2}\log(2\pi e\, \varepsilon^2), \quad \mathcal{B}_\varepsilon(\theta; x_0) := -\log p_0^\theta(x_0) - \ell_\varepsilon(\theta; x_0), \quad \lim_{\varepsilon \downarrow 0} \mathcal{B}_\varepsilon(\theta; x_0) = 0.$$

A.2 EXPECTATION-LEVEL RELATION BETWEEN LOG-LIKELIHOOD AND FLOW MATCHING LOSS

Throughout this subsection, let

$$t \sim \mathrm{Unif}(0,1), \qquad \mathbf{x}_0 \sim p_{\mathrm{data}}, \qquad \mathbf{x}_1 \sim \mathcal{N}(0, I_d),$$

where $\mathbf{x}_0$ and $\mathbf{x}_1$ are independent, and let

$$\mathbf{x}_t = (1-t)\mathbf{x}_0 + t\mathbf{x}_1 \sim q_t.$$

The marginal velocity field of $(q_t)_{t\in[0,1]}$ is

$$u_t(x) := \mathbb{E}[\mathbf{x}_1 - \mathbf{x}_0 \mid \mathbf{x}_t = x].$$

Lemma A.4 (Optimal solution of Flow Matching Loss)**.** *Let us consider the following optimization problem:*

$$\min_{\theta} \mathbb{E}_{t,\mathbf{x}_0,\mathbf{x}_1}\left[\|v_\theta((1-t)\mathbf{x}_0 + t\mathbf{x}_1, t) - (\mathbf{x}_1 - \mathbf{x}_0)\|_2^2\right] \tag{15}$$

Then, for every $t \in (0,1)$, the optimal solution is given by

$$v^\star(x,t) = \mathbb{E}_{\mathbf{x}_0,\mathbf{x}_1}\left[\mathbf{x}_1 - \mathbf{x}_0 \mid \mathbf{x}_t = x\right] \tag{16}$$

where $\mathbf{x}_t = (1-t)\mathbf{x}_0 + t\mathbf{x}_1$.

Proof.

$$\begin{aligned}
&\mathbb{E}_{t,\mathbf{x}_0,\mathbf{x}_1}\left[\|v_\theta((1-t)\mathbf{x}_0 + t\mathbf{x}_1, t) - (\mathbf{x}_1 - \mathbf{x}_0)\|_2^2\right] \\
&= \mathbb{E}_{t,\mathbf{x}_0,\mathbf{x}_1}\left[\|v_\theta(\mathbf{x}_t, t) - v^\star(\mathbf{x}_t, t) + v^\star(\mathbf{x}_t, t) - (\mathbf{x}_1 - \mathbf{x}_0)\|_2^2\right] \\
&= \mathbb{E}_{t,\mathbf{x}_t}[\mathbb{E}_{\mathbf{x}_0,\mathbf{x}_1}\left[\|v_\theta(\mathbf{x}_t, t) - v^\star(\mathbf{x}_t, t) + v^\star(\mathbf{x}_t, t) - (\mathbf{x}_1 - \mathbf{x}_0)\|_2^2 \mid \mathbf{x}_t\right]] \\
&\quad \text{(by tower property of expectation)} \\
&= \mathbb{E}_{t,\mathbf{x}_t}[\mathbb{E}_{\mathbf{x}_0,\mathbf{x}_1}[\|v_\theta(\mathbf{x}_t, t) - v^\star(\mathbf{x}_t, t)\|_2^2 + \|v^\star(\mathbf{x}_t, t) - (\mathbf{x}_1 - \mathbf{x}_0)\|_2^2 \\
&\qquad + 2\langle v_\theta(\mathbf{x}_t, t) - v^\star(\mathbf{x}_t, t), v^\star(\mathbf{x}_t, t) - (\mathbf{x}_1 - \mathbf{x}_0)\rangle \mid \mathbf{x}_t]]
\end{aligned}$$

Now, we prove that $\mathbb{E}_{t,\mathbf{x}_t}[\mathbb{E}_{\mathbf{x}_0,\mathbf{x}_1}\left[\langle v_\theta(\mathbf{x}_t, t) - v^\star(\mathbf{x}_t, t), v^\star(\mathbf{x}_t, t) - (\mathbf{x}_1 - \mathbf{x}_0)\rangle \mid \mathbf{x}_t\right]] = 0$.

$$\begin{aligned}
&\mathbb{E}_{t,\mathbf{x}_t}[\mathbb{E}_{\mathbf{x}_0,\mathbf{x}_1}\left[\langle v_\theta(\mathbf{x}_t, t) - v^\star(\mathbf{x}_t, t), v^\star(\mathbf{x}_t, t) - (\mathbf{x}_1 - \mathbf{x}_0)\rangle \mid \mathbf{x}_t\right]] \\
&= \mathbb{E}_{t,\mathbf{x}_t}[\langle v_\theta(\mathbf{x}_t, t) - v^\star(\mathbf{x}_t, t), v^\star(\mathbf{x}_t, t) - \mathbb{E}_{\mathbf{x}_0,\mathbf{x}_1}\left[(\mathbf{x}_1 - \mathbf{x}_0) \mid \mathbf{x}_t\right]\rangle] \\
&\quad \text{(by the linearity of expectation)} \\
&= \mathbb{E}_{t,\mathbf{x}_t}[\langle v_\theta(\mathbf{x}_t, t) - v^\star(\mathbf{x}_t, t), v^\star(\mathbf{x}_t, t) - v^\star(\mathbf{x}_t, t)\rangle] \quad (\text{by } v^\star(\mathbf{x}_t, t) = \mathbb{E}_{\mathbf{x}_0,\mathbf{x}_1}\left[(\mathbf{x}_1 - \mathbf{x}_0) \mid \mathbf{x}_t\right]) \\
&= 0
\end{aligned}$$

Therefore,

$$\begin{aligned}
&\mathbb{E}_{t,\mathbf{x}_0,\mathbf{x}_1}\left[\|v_\theta((1-t)\mathbf{x}_0 + t\mathbf{x}_1, t) - (\mathbf{x}_1 - \mathbf{x}_0)\|_2^2\right] \\
&= \mathbb{E}_{t,\mathbf{x}_0,\mathbf{x}_1}[\|v_\theta(\mathbf{x}_t, t) - v^\star(\mathbf{x}_t, t)\|_2^2 + \|v^\star(\mathbf{x}_t, t) - (\mathbf{x}_1 - \mathbf{x}_0)\|_2^2]
\end{aligned}$$

Clearly, the optimal solution is attained when $v_\theta(x,t) = v^\star(x,t)$. □

Lemma A.5. *Under the optimal solution $v^\star(x,t)$, we have:*

$$v^\star(x,t) = -\frac{1}{1-t}x - \frac{t}{1-t}\nabla \log q_t(x) \tag{17}$$

where $\mathbf{x}_t = (1-t)\mathbf{x}_0 + t\mathbf{x}_1 \sim q_t(x)$.

Proof. Let $q_t^{x_0}(x) = q_t(x \mid x_0)$. Because $\mathbf{x}_t = (1-t)\mathbf{x}_0 + t\mathbf{x}_1$ and $\mathbf{x}_1 \sim \mathcal{N}(0, I)$, then

$$q_t^{x_0}(\mathbf{x}_t) = \mathcal{N}((1-t)x_0, t^2 I) \tag{18}$$

and we have:

$$\nabla \log q_t^{x_0}(\mathbf{x}_t) = -\frac{1}{t}\mathbf{x}_1 \tag{19}$$

Therefore,

$$\nabla \log q_t(\mathbf{x}_t) = \mathbb{E}[\nabla \log q_t^{x_0}(\mathbf{x}_t) \mid \mathbf{x}_t] = -\frac{1}{t}\mathbb{E}[\mathbf{x}_1 \mid \mathbf{x}_t] \tag{20}$$

Then, for the optimal solution $v^\star(x,t)$, we have:

$$\begin{aligned}
v^\star(x,t) &= \mathbb{E}[\mathbf{x}_1 - \mathbf{x}_0 \mid \mathbf{x}_t = x] && (21)\\
&= \mathbb{E}[\mathbf{x}_1 - \frac{\mathbf{x}_t - t\mathbf{x}_1}{1-t} \mid \mathbf{x}_t = x] && (22)\\
&= -\frac{1}{1-t}x + \frac{1}{1-t}\mathbb{E}[\mathbf{x}_1 \mid \mathbf{x}_t = x] && (23)\\
&= -\frac{1}{1-t}x - \frac{t}{1-t}\nabla \log q_t(x) && (24)
\end{aligned}$$

□

Theorem A.6. *Suppose Assump. A.1 and Assump. A.2 hold and let $p_{\text{data}}(\mathbf{x}_0)$ be the data distribution. Then,*

$$\mathbb{E}_{\mathbf{x}_0\sim p_{\text{data}}(\mathbf{x}_0)}[-\log p_0^\theta(\mathbf{x}_0)] \leq H(p_{\text{data}}) + \sqrt{\mathbb{E}\big[\|v_\theta(\mathbf{x}_t,t)-(\mathbf{x}_1-\mathbf{x}_0)\|_2^2\big]\cdot\mathbb{E}\big[\|\nabla\log q_t(\mathbf{x}_t)-\nabla\log p_t^\theta(\mathbf{x}_t)\|_2^2\big]}$$

In the unrestricted, well-specified population setting, the marginal target field $v^\star(\mathbf{x}_t,t) = \mathbb{E}[\mathbf{x}_1-\mathbf{x}_0 \mid \mathbf{x}_t]$ minimizes the flow matching loss and induces the data endpoint distribution, which also minimizes cross-entropy.

Proof. By Cauchy-Schwarz inequality, we have:

$$(\mathbb{E}_{t,\mathbf{x}_0,\mathbf{x}_t}[\langle v_\theta(\mathbf{x}_t,t)-(\mathbf{x}_1-\mathbf{x}_0),\nabla\log q_t(\mathbf{x}_t)-\nabla\log p_t^\theta(\mathbf{x}_t)\rangle])^2 \tag{25}$$
$$\leq \mathbb{E}_{t,\mathbf{x}_0,\mathbf{x}_t}[\|v_\theta(\mathbf{x}_t,t)-(\mathbf{x}_1-\mathbf{x}_0)\|_2^2]\cdot\mathbb{E}_{t,\mathbf{x}_0,\mathbf{x}_t}[\|\nabla\log q_t(\mathbf{x}_t)-\nabla\log p_t^\theta(\mathbf{x}_t)\|_2^2] \tag{26}$$

Then, by Thm. A.3, we have:

$$\mathbb{E}_{\mathbf{x}_0\sim p_{\text{data}}(\mathbf{x}_0)}[-\log p_0^\theta(\mathbf{x}_0)] \tag{27}$$
$$= H(p_{\text{data}}) + \mathbb{E}\big[\langle v_\theta(\mathbf{x}_t,t)-(\mathbf{x}_1-\mathbf{x}_0),\nabla\log q_t(\mathbf{x}_t)-\nabla\log p_t^\theta(\mathbf{x}_t)\rangle\big] \tag{28}$$
$$\leq H(p_{\text{data}}) + (\mathbb{E}\big[\|v_\theta(\mathbf{x}_t,t)-(\mathbf{x}_1-\mathbf{x}_0)\|_2^2\big]\mathbb{E}\big[\|\nabla\log q_t(\mathbf{x}_t)-\nabla\log p_t^\theta(\mathbf{x}_t)\|_2^2\big])^{1/2} \tag{29}$$

By Lem. A.4, the unrestricted population minimizer is $v^\star(\mathbf{x}_t,t) = \mathbb{E}[\mathbf{x}_1-\mathbf{x}_0 \mid \mathbf{x}_t]$. This field generates the prescribed marginal path q_t, so $p_t^{\theta^\star} = q_t$ in the well-specified case and the score-gap term vanishes. Consequently the induced endpoint distribution equals p_{data}, which minimizes cross-entropy. This population statement does not imply equality of the two objectives, nor identical minimizers in a restricted parameter class.

□

Theorem A.7 (Flow matching upper bound under uniform regularity)**.** *Suppose Assump. A.1 and Assump. A.2 hold, and let $p_1^\theta = \mathcal{N}(0, I_d)$. Assume $M_2 := \mathbb{E}_{p_{\text{data}}}[\|\mathbf{x}_0\|^2] < \infty$ and that there exist constants $C_0, C_1, C_2 < \infty$, independent of θ, such that, for every $x \in \mathbb{R}^d$ and $t \in [0,1]$,*

$$\|v_\theta(0,t)\| \leq C_0, \qquad \|\nabla v_\theta(x,t)\| \leq C_1, \qquad \|\nabla^\top\nabla\cdot v_\theta(x,t)\| \leq C_2. \tag{30}$$

Assume additionally that the data-path score is uniformly square-integrable,

$$C_q := \sup_{t\in(0,1]}\mathbb{E}_{q_t}[\|\nabla\log q_t(\mathbf{x}_t)\|^2] < \infty. \tag{31}$$

Then, there exists a constant $C_s < \infty$, independent of θ, such that:

$$\mathbb{E}_{\mathbf{x}_0\sim p_{\text{data}}}[-\log p_0^\theta(\mathbf{x}_0)] \leq \mathbb{E}\big[\|v_\theta(\mathbf{x}_t,t)-(\mathbf{x}_1-\mathbf{x}_0)\|^2\big] + H(p_{\text{data}}) + \frac{C_s}{4}. \tag{32}$$

In the unrestricted, well-specified population setting, the marginal target field minimizes both sides.

Proof. Let $s_t^\theta(x) = \nabla\log p_t^\theta(x)$, then we consider the continuity equation:

$$\partial_t p_t^\theta(x) + \nabla\cdot(p_t^\theta(x)v_\theta(x,t)) = 0 \tag{33}$$
$$\partial_t p_t^\theta(x) + \nabla p_t^\theta(x)\cdot v_\theta(x,t) + p_t^\theta(x)\nabla\cdot v_\theta(x,t) = 0 \tag{34}$$
$$\partial_t\log p_t^\theta(x) + v_\theta(x,t)\cdot\nabla\log p_t^\theta(x) + \nabla\cdot v_\theta(x,t) = 0 \tag{35}$$

Taking ∇ on both sides, we get:

$$\partial_t\nabla\log p_t^\theta(x) + \nabla v_\theta(x,t)\cdot\nabla\log p_t^\theta(x) + v_\theta(x,t)\nabla^\top\nabla\log p_t^\theta(x) + \nabla^\top\nabla\cdot v_\theta(x,t) = 0 \tag{36}$$
$$\partial_t s_t^\theta(x) + \nabla v_\theta(x,t)\cdot s_t^\theta(x) + v_\theta(x,t)\nabla^\top s_t^\theta(x) + \nabla^\top\nabla\cdot v_\theta(x,t) = 0 \tag{37}$$

Let X_t denote the trajectory, then we have:

$$\frac{d}{dt}s_t^\theta(X_t) + \nabla v_\theta(X_t,t)\cdot s_t^\theta(X_t) + \nabla^\top\nabla\cdot v_\theta(X_t,t) = 0 \tag{38}$$

$$\frac{d}{dt}s_t^\theta(X_t) = -\nabla v_\theta(X_t,t)\cdot s_t^\theta(X_t) - \nabla^\top\nabla\cdot v_\theta(X_t,t) \tag{39}$$

Therefore, we need to control $\|\nabla v_\theta\|$ and $\|\nabla^\top\nabla\cdot v_\theta\|$. We assume v_θ is smooth and constrained in a compact set, then we can have:

$$\|\nabla v_\theta\| \le C_1 \tag{40}$$

$$\|\nabla^\top\nabla\cdot v_\theta\| \le C_2 \tag{41}$$

We further assume that $\|v_\theta(0,t)\| \le C_0$ uniformly over the parameter set. Then, we define $g(\lambda) = v_\theta(\lambda x, t)$, $\lambda\in[0,1]$, and we have:

$$g(1) - g(0) = \int_0^1 \frac{d}{d\lambda}g(\lambda)d\lambda \tag{42}$$

This is actually:

$$v_\theta(x,t) - v_\theta(0,t) = \int_0^1 \nabla v_\theta(\lambda x,t)x d\lambda \tag{43}$$

Therefore, we have:

$$\|v_\theta(x,t)\| = \|v_\theta(0,t) + \int_0^1 \nabla v_\theta(\lambda x,t)x d\lambda\| \tag{44}$$

$$\le \|v_\theta(0,t)\| + \|\int_0^1 \nabla v_\theta(\lambda x,t)x d\lambda\| \tag{45}$$

$$\le C_0 + \int_0^1 \|\nabla v_\theta(\lambda x,t)\|\|x\| d\lambda \tag{46}$$

$$\le C_0 + C_1\|x\| \tag{47}$$

Let X_r be the characteristic starting from $X_t = x$, namely,

$$X_r = x + \int_t^r v_\theta(X_s,s)\,ds, \qquad r\in[t,1]. \tag{48}$$

Using the linear-growth bound above, we obtain

$$\|X_r\| \le \|x\| + \int_t^r \|v_\theta(X_s,s)\|\,ds \tag{49}$$

$$\le \|x\| + \int_t^r (C_0 + C_1\|X_s\|)\,ds. \tag{50}$$

Therefore, Grönwall's inequality gives

$$\|X_r\| \le e^{C_1(r-t)}\|x\| + \frac{C_0}{C_1}\big(e^{C_1(r-t)} - 1\big), \qquad r\in[t,1], \tag{51}$$

Since $p_1^\theta = \mathcal{N}(0,I)$, we have $s_1^\theta(x) = -x$. Then, by equation 39, we have:

$$\frac{d}{dt}s_t^\theta(X_t) = -\nabla v_\theta(X_t,t)\cdot s_t^\theta(X_t) - \nabla^\top\nabla\cdot v_\theta(X_t,t) \tag{52}$$

$$\int_t^1 ds_t^\theta(X_t) = \int_t^1 (-\nabla v_\theta(X_t,t)\cdot s_t^\theta(X_t) - \nabla^\top\nabla\cdot v_\theta(X_t,t))dt \tag{53}$$

$$s_1^\theta(X_1) - s_t^\theta(X_t) = \int_t^1 (-\nabla v_\theta(X_t,t)\cdot s_t^\theta(X_t) - \nabla^\top\nabla\cdot v_\theta(X_t,t))dt \tag{54}$$

$$\|s_t^\theta(X_t)\| \le \|s_1^\theta(X_1)\| + \int_t^1 \big(C_1\|s_r^\theta(X_r)\| + C_2\big)\,dr \tag{55}$$

$$= \|X_1\| + \int_t^1 \big(C_1\|s_r^\theta(X_r)\| + C_2\big)\,dr. \tag{56}$$

Applying Grönwall's inequality once more and using equation 51 gives

$$\|s_t^\theta(x)\| \le e^{C_1(1-t)}\|X_1\| + \frac{C_2}{C_1}\big(e^{C_1(1-t)} - 1\big) \tag{57}$$

$$\le K_0 + K_1\|x\|, \tag{58}$$

where $K_0, K_1 < \infty$ depend only on C_0, C_1, C_2 and are independent of θ. If $M_2 := \mathbb{E}_{p_{\text{data}}}[\|\mathbf{x}_0\|^2] < \infty$, then

$$\mathbb{E}[\|\mathbf{x}_t\|^2] = \mathbb{E}[\|(1-t)\mathbf{x}_0 + t\mathbf{x}_1\|^2] \tag{59}$$

$$= \mathbb{E}[(1-t)^2\|\mathbf{x}_0\|^2 + t^2\|\mathbf{x}_1\|^2 + 2t(1-t)\langle\mathbf{x}_0, \mathbf{x}_1\rangle] \tag{60}$$

$$= (1-t)^2\mathbb{E}[\|\mathbf{x}_0\|^2] + t^2\mathbb{E}[\|\mathbf{x}_1\|^2] + 2t(1-t)\mathbb{E}[\langle\mathbf{x}_0, \mathbf{x}_1\rangle] \tag{61}$$

$$= (1-t)^2\mathbb{E}[\|\mathbf{x}_0\|^2] + t^2\mathbb{E}[\|\mathbf{x}_1\|^2] + 2t(1-t)\langle\mathbb{E}[\mathbf{x}_0], \mathbb{E}[\mathbf{x}_1]\rangle] \quad (\text{by } \mathbf{x}_0 \perp \mathbf{x}_1) \tag{62}$$

$$= (1-t)^2\mathbb{E}\|\mathbf{x}_0\|^2 + t^2\mathbb{E}\|\mathbf{x}_1\|^2 \quad (\text{by } \mathbb{E}[\mathbf{x}_1] = 0) \tag{63}$$

$$= (1-t)^2 M_2 + t^2 d \quad (\text{by } \mathbf{x}_1 \sim \mathcal{N}(0, I_d)) \tag{64}$$

$$\le M_2 + d \quad (\text{by } t \in [0,1]) \tag{65}$$

and consequently

$$\sup_{\theta, t\in[0,1]} \mathbb{E}_{q_t}[\|\nabla \log p_t^\theta(\mathbf{x}_t)\|^2] \le 2K_0^2 + 2K_1^2(M_2 + d) =: C_p < \infty. \tag{66}$$

It remains to control the score of the data path. By the identity in Lem. A.5,

$$\nabla \log q_t(\mathbf{x}_t) = \mathbb{E}\left[-\frac{\mathbf{x}_1}{t}\,\middle|\,\mathbf{x}_t\right]. \tag{67}$$

Hence, on every truncated interval $t \in [\varepsilon, 1]$,

$$\mathbb{E}_{q_t}[\|\nabla \log q_t(\mathbf{x}_t)\|^2] = \mathbb{E}_{q_t}[\|-\frac{1}{t}\mathbb{E}[\mathbf{x}_1 \mid \mathbf{x}_t]\|^2] \tag{68}$$

$$= \frac{1}{t^2}\mathbb{E}_{q_t}[\|\mathbb{E}[\mathbf{x}_1 \mid \mathbf{x}_t]\|^2] \tag{69}$$

$$\le \frac{1}{t^2}\mathbb{E}_{q_t}[\mathbb{E}[\|\mathbf{x}_1\|^2 \mid \mathbf{x}_t]] \tag{70}$$

$$= \frac{1}{t^2}\mathbb{E}[\|\mathbf{x}_1\|^2] \quad (\text{by tower property}) \tag{71}$$

$$\le \frac{d}{\varepsilon^2}. \tag{72}$$

For the full interval, we use the endpoint integrability condition in the theorem,

$$C_q := \sup_{t\in(0,1]} \mathbb{E}_{q_t}[\|\nabla \log q_t(\mathbf{x}_t)\|^2] < \infty. \tag{73}$$

Combining the last two bounds yields

$$\mathbb{E}\big[\|\nabla \log q_t(\mathbf{x}_t) - \nabla \log p_t^\theta(\mathbf{x}_t)\|^2\big] \le 2(C_q + C_p) =: C_s < \infty, \tag{74}$$

uniformly over θ. Substituting this estimate into Thm. A.6, and then using Young's inequality, gives, for every $\kappa > 0$,

$$\mathbb{E}_{\mathbf{x}_0\sim p_{\text{data}}}[-\log p_0^\theta(\mathbf{x}_0)] \le H(p_{\text{data}}) + \sqrt{C_s\,\mathbb{E}[\|v_\theta(\mathbf{x}_t, t) - (\mathbf{x}_1 - \mathbf{x}_0)\|^2]} \tag{75}$$

$$\le \kappa\,\mathbb{E}[\|v_\theta(\mathbf{x}_t, t) - (\mathbf{x}_1 - \mathbf{x}_0)\|^2] + H(p_{\text{data}}) + \frac{C_s}{4\kappa}. \tag{76}$$

Taking $\kappa = 1$ proves the bound. The population-minimizer statement follows from Thm. A.6 under the stated well-specified interpretation. □

B Theoretical Proofs and Results

B.1 Idealized Posteriors and Empirical Source Distributions

The main text defines π^+ and π^- as posterior target distributions induced by the binary optimality variable. In offline preference optimization, however, the data available for training are usually fixed empirical source distributions $\hat{\pi}^+$ and $\hat{\pi}^-$ constructed by a separate pipeline, such as best-of-N winner/loser selection from a frozen generator. The formal identities in the main text only require that the source distributions be fixed during optimization. The practical modeling question is therefore whether the offline construction is a faithful enough surrogate for the idealized targets. In our experiments, the data construction pipeline is frozen, so the resulting preferred and dispreferred distributions remain stationary throughout training.

B.2 Proof of Theorem 4.1

Proof. For each context c, equation 7 gives

$$r(\mathbf{x}_0, c) = \frac{1}{2\omega}\left[\log \pi^+(\mathbf{x}_0 \mid c) - \log \pi^-(\mathbf{x}_0 \mid c) + \log \frac{p_{\pi_{\text{ref}}}(o=1 \mid c)}{p_{\pi_{\text{ref}}}(o=0 \mid c)}\right].$$

Taking expectation under $\pi_\theta(\cdot \mid c)$ yields

$$\begin{aligned}\mathbb{E}_{\mathbf{x}_0\sim\pi_\theta(\cdot|c)}[r(\mathbf{x}_0,c)] =& \frac{1}{2\omega}\mathbb{E}_{\mathbf{x}_0\sim\pi_\theta(\cdot|c)}\big[\log \pi^+(\mathbf{x}_0 \mid c) - \log \pi^-(\mathbf{x}_0 \mid c)\big] \\ &+ \frac{1}{2\omega}\log \frac{p_{\pi_{\text{ref}}}(o=1 \mid c)}{p_{\pi_{\text{ref}}}(o=0 \mid c)}.\end{aligned}$$

The second term depends only on c, hence is irrelevant for optimization over θ. For the first term,

$$\begin{aligned}\mathbb{E}_{\pi_\theta}[\log \pi^+] &= -\mathcal{D}_{KL}(\pi_\theta \| \pi^+) - \mathcal{H}(\pi_\theta), \\ \mathbb{E}_{\pi_\theta}[\log \pi^-] &= -\mathcal{D}_{KL}(\pi_\theta \| \pi^-) - \mathcal{H}(\pi_\theta),\end{aligned}$$

where $\mathcal{H}(\pi_\theta)$ is the conditional differential entropy of $\pi_\theta(\cdot \mid c)$. Subtracting the two identities cancels the entropy term:

$$\mathbb{E}_{\pi_\theta}[\log \pi^+ - \log \pi^-] = -\mathcal{D}_{KL}(\pi_\theta \| \pi^+) + \mathcal{D}_{KL}(\pi_\theta \| \pi^-).$$

Taking expectation over c and dropping the positive constant $\frac{1}{2\omega}$ proves the claim. □

B.3 An Idealized Interpretation of Interpolated Vector Fields

Proposition 1 of (Holderrieth & Erives, 2025) gives the following relation when a velocity field and a score describe the same Gaussian marginal path (see also Lem. A.5):

$$u_t(\mathbf{x}) = -\frac{t}{1-t}\nabla \log p_t(\mathbf{x}) - \frac{1}{1-t}\mathbf{x}. \tag{77}$$

For an idealized interpretation, suppose that both v_θ and v_{old} satisfy this compatibility condition:

$$v_\theta(\mathbf{x}_t, t) = -\frac{t}{1-t}\nabla \log p_t^\theta(\mathbf{x}_t) - \frac{1}{1-t}\mathbf{x}_t, \tag{78}$$

$$v_{\text{old}}(\mathbf{x}_t, t) = -\frac{t}{1-t}\nabla \log p_t^{\text{old}}(\mathbf{x}_t) - \frac{1}{1-t}\mathbf{x}_t. \tag{79}$$

Then the positive interpolation satisfies

$$v^+(\mathbf{x}_t, t) = v_{\text{old}}(\mathbf{x}_t, t) + \beta(v_\theta(\mathbf{x}_t, t) - v_{\text{old}}(\mathbf{x}_t, t)) \tag{80}$$

$$= -\frac{t}{1-t}\nabla \log \left[p_t^{\text{old}}(\mathbf{x}_t)\left(\frac{p_t^\theta(\mathbf{x}_t)}{p_t^{\text{old}}(\mathbf{x}_t)}\right)^\beta\right] - \frac{1}{1-t}\mathbf{x}_t. \tag{81}$$

Similarly,

$$v^-(\mathbf{x}_t, t) = v_{\text{old}}(\mathbf{x}_t, t) - \beta(v_\theta(\mathbf{x}_t, t) - v_{\text{old}}(\mathbf{x}_t, t)) \tag{82}$$

$$= -\frac{t}{1-t}\nabla \log \left[p_t^{\text{old}}(\mathbf{x}_t)\left(\frac{p_t^{\text{old}}(\mathbf{x}_t)}{p_t^\theta(\mathbf{x}_t)}\right)^\beta\right] - \frac{1}{1-t}\mathbf{x}_t. \tag{83}$$

Thus, at each fixed t, v^+ and v^- admit a score-field interpretation analogous to the CFG (Luo, 2022; Ho & Salimans, 2022) interpolation $p_t(\mathbf{x}_t, \emptyset)\left(\frac{p_t(\mathbf{x}_t,c)}{p_t(\mathbf{x}_t,\emptyset)}\right)^\beta$ over condition space.

B.4 Proof of Theorem 4.2

Proof of Thm. 4.2. Expanding each KL term in equation 9 into cross-entropy plus entropy gives

$$\mathcal{L}_{\text{Offline}}(\theta) = \mathbb{E}_c\Big[\mathbb{E}_{\mathbf{x}_0^+\sim\pi^+(\cdot|c)}[-\log \pi_\theta^+(\mathbf{x}_0^+ \mid c)] + \lambda \mathbb{E}_{\mathbf{x}_0^-\sim\pi^-(\cdot|c)}[-\log \pi_\theta^-(\mathbf{x}_0^- \mid c)]\Big] + C_{\text{ent}},$$

where $C_{\text{ent}} = -\mathbb{E}_c[\mathcal{H}(\pi^+(\cdot \mid c)) + \lambda \mathcal{H}(\pi^-(\cdot \mid c))]$ is independent of θ.

From Thm. A.7, we have:

$$\mathbb{E}_{\mathbf{x}_0\sim p_{\text{data}}}[-\log p_0^\theta(\mathbf{x}_0)] \le \mathbb{E}\big[\|v_\theta(\mathbf{x}_t, t) - (\mathbf{x}_1 - \mathbf{x}_0)\|^2\big] + C. \tag{84}$$

Therefore, we have:

$$\mathbb{E}_{\mathbf{x}_0^+\sim\pi^+(\cdot|c)}[-\log \pi_\theta^+(\mathbf{x}_0^+ \mid c)] \le C_+(c) + \mathbb{E}_{\substack{\mathbf{x}_0^+\sim\pi^+(\cdot|c)\\ t,\,\epsilon}}[\|\mu_\theta(\mathbf{x}_t^+, t, c) - u_t(\mathbf{x}_t^+ \mid \mathbf{x}_0^+)\|_2^2],$$

$$\mathbb{E}_{\mathbf{x}_0^-\sim\pi^-(\cdot|c)}[-\log \pi_\theta^-(\mathbf{x}_0^- \mid c)] \le C_-(c) + \mathbb{E}_{\substack{\mathbf{x}_0^-\sim\pi^-(\cdot|c)\\ t,\,\epsilon}}[\|\nu_\theta(\mathbf{x}_t^-, t, c) - u_t(\mathbf{x}_t^- \mid \mathbf{x}_0^-)\|_2^2],$$

Then,

$$\begin{aligned}\mathcal{L}_{\text{Offline}}(\theta) \le\ & \mathbb{E}_{c,\mathbf{x}_0^+\sim\pi^+(\cdot|c)\atop t,\,\epsilon}[\|\mu_\theta(\mathbf{x}_t^+, t, c) - u_t(\mathbf{x}_t^+ \mid \mathbf{x}_0^+)\|_2^2] \\ & + \lambda \cdot \mathbb{E}_{c,\mathbf{x}_0^-\sim\pi^-(\cdot|c)\atop t,\,\epsilon}[\|\nu_\theta(\mathbf{x}_t^-, t, c) - u_t(\mathbf{x}_t^- \mid \mathbf{x}_0^-)\|_2^2] + Constant.\end{aligned}$$

This establishes an upper-bound surrogate for equation 9. If the two branch fields can simultaneously realize their population targets, then both branch endpoint distributions match their corresponding sources. □

B.5 Relation between FlowCPO and Simplified FlowDPO

We isolate the contrastive core of FlowDPO by omitting its sigmoid weighting and frozen-reference terms. Using the preferred/dispreferred notation from equation 11, the simplified objective is

$$\min_\theta \mathcal{L}_{\text{FlowDPO}}^{\text{simple}}(\theta) = \mathbb{E}\Big[\|v_\theta(\mathbf{x}_t^w, t, c) - (\epsilon - \mathbf{x}_0^w)\|_2^2 - \|v_\theta(\mathbf{x}_t^l, t, c) - (\epsilon - \mathbf{x}_0^l)\|_2^2\Big]. \tag{85}$$

It captures the per-pair contrastive gradient direction, without retaining the full FlowDPO objective. Below, $\mathbb{E}$ averages over the same fixed offline pairs and sampled t, ϵ as in equation 11. We set $\lambda = 1$ to match the unweighted preference comparison above.

Theorem B.1 (Relation between FlowCPO and Simplified FlowDPO). *Let $\beta > 0$, $\lambda = 1$, and hold v_{old} fixed during each gradient step. Assuming the displayed expectations are finite, the FlowCPO loss satisfies*

$$\begin{aligned}\mathcal{L}_{\text{FlowCPO}}(\theta) = \beta \mathcal{L}_{\text{FlowDPO}}^{\text{simple}}(\theta) & \\ + \beta(\beta-1)\mathbb{E}\big[\|v_\theta(\mathbf{x}_t^w, t, c) - v_{\text{old}}(\mathbf{x}_t^w, t, c)\|_2^2\big] & \\ + \beta(\beta+1)\mathbb{E}\big[\|v_\theta(\mathbf{x}_t^l, t, c) - v_{\text{old}}(\mathbf{x}_t^l, t, c)\|_2^2\big] + C, &\end{aligned} \tag{86}$$

where C is independent of θ for the current detached v_{old}.

Proof. The preferred branch uses $\mathbf{x}_t^w$ and target $u_t(\mathbf{x}_t^w \mid \mathbf{x}_0^w) = \epsilon - \mathbf{x}_0^w$, while the dispreferred branch uses $\mathbf{x}_t^l$ and target $u_t(\mathbf{x}_t^l \mid \mathbf{x}_0^l) = \epsilon - \mathbf{x}_0^l$. We expand each branch separately.

Preferred branch. Substituting $\mu_\theta = (1-\beta)v_{\text{old}} + \beta v_\theta$ and collecting terms gives

$$\begin{aligned}&\mu_\theta(\mathbf{x}_t^w, t, c) - u_t(\mathbf{x}_t^w \mid \mathbf{x}_0^w) \\ &\quad = v_{\text{old}}(\mathbf{x}_t^w, t, c) - u_t(\mathbf{x}_t^w \mid \mathbf{x}_0^w) + \beta\big[v_\theta(\mathbf{x}_t^w, t, c) - v_{\text{old}}(\mathbf{x}_t^w, t, c)\big].\end{aligned}$$

Expanding the squared norm yields

$$\begin{aligned}&\|\mu_\theta(\mathbf{x}_t^w, t, c) - u_t(\mathbf{x}_t^w \mid \mathbf{x}_0^w)\|_2^2 \\ &\quad = \|v_{\text{old}}(\mathbf{x}_t^w, t, c) - u_t(\mathbf{x}_t^w \mid \mathbf{x}_0^w)\|_2^2 \\ &\qquad + \beta^2\|v_\theta(\mathbf{x}_t^w, t, c) - v_{\text{old}}(\mathbf{x}_t^w, t, c)\|_2^2 \\ &\qquad + 2\beta\Big\langle v_{\text{old}}(\mathbf{x}_t^w, t, c) - u_t(\mathbf{x}_t^w \mid \mathbf{x}_0^w), v_\theta(\mathbf{x}_t^w, t, c) - v_{\text{old}}(\mathbf{x}_t^w, t, c)\Big\rangle.\end{aligned}$$

The residual of the current model is the sum of the EMA residual and $v_\theta(\mathbf{x}_t^w, t, c) - v_{\text{old}}(\mathbf{x}_t^w, t, c)$. Expanding the squared norm of this sum and solving for the cross term gives

$$\begin{aligned}&2\Big\langle v_{\text{old}}(\mathbf{x}_t^w, t, c) - u_t(\mathbf{x}_t^w \mid \mathbf{x}_0^w), v_\theta(\mathbf{x}_t^w, t, c) - v_{\text{old}}(\mathbf{x}_t^w, t, c)\Big\rangle \\ &\quad = \|v_\theta(\mathbf{x}_t^w, t, c) - u_t(\mathbf{x}_t^w \mid \mathbf{x}_0^w)\|_2^2 \\ &\qquad - \|v_{\text{old}}(\mathbf{x}_t^w, t, c) - u_t(\mathbf{x}_t^w \mid \mathbf{x}_0^w)\|_2^2 - \|v_\theta(\mathbf{x}_t^w, t, c) - v_{\text{old}}(\mathbf{x}_t^w, t, c)\|_2^2.\end{aligned}$$

Substituting this identity into the expansion gives coefficients β, $\beta^2 - \beta = \beta(\beta - 1)$, and $1 - \beta$ for the three squared errors, respectively:

$$\begin{aligned}\|\mu_\theta(\mathbf{x}_t^w, t, c) - u_t(\mathbf{x}_t^w \mid \mathbf{x}_0^w)\|_2^2 = \beta\|v_\theta(\mathbf{x}_t^w, t, c) - u_t(\mathbf{x}_t^w \mid \mathbf{x}_0^w)\|_2^2 \\ + \beta(\beta - 1)\|v_\theta(\mathbf{x}_t^w, t, c) - v_{\text{old}}(\mathbf{x}_t^w, t, c)\|_2^2 \\ + (1 - \beta)\|v_{\text{old}}(\mathbf{x}_t^w, t, c) - u_t(\mathbf{x}_t^w \mid \mathbf{x}_0^w)\|_2^2.\end{aligned}$$

Dispreferred branch. Substituting $\nu_\theta = (1 + \beta)v_{\text{old}} - \beta v_\theta$ now gives

$$\begin{aligned}&\nu_\theta(\mathbf{x}_t^l, t, c) - u_t(\mathbf{x}_t^l \mid \mathbf{x}_0^l) \\ &\quad = v_{\text{old}}(\mathbf{x}_t^l, t, c) - u_t(\mathbf{x}_t^l \mid \mathbf{x}_0^l) - \beta\big[v_\theta(\mathbf{x}_t^l, t, c) - v_{\text{old}}(\mathbf{x}_t^l, t, c)\big].\end{aligned}$$

The minus sign changes the sign of the cross term, while the coefficient of the squared distance remains β^2:

$$\begin{aligned}&\|\nu_\theta(\mathbf{x}_t^l, t, c) - u_t(\mathbf{x}_t^l \mid \mathbf{x}_0^l)\|_2^2 \\ &\quad = \|v_{\text{old}}(\mathbf{x}_t^l, t, c) - u_t(\mathbf{x}_t^l \mid \mathbf{x}_0^l)\|_2^2 \\ &\qquad + \beta^2\|v_\theta(\mathbf{x}_t^l, t, c) - v_{\text{old}}(\mathbf{x}_t^l, t, c)\|_2^2 \\ &\qquad - 2\beta\Big\langle v_{\text{old}}(\mathbf{x}_t^l, t, c) - u_t(\mathbf{x}_t^l \mid \mathbf{x}_0^l), v_\theta(\mathbf{x}_t^l, t, c) - v_{\text{old}}(\mathbf{x}_t^l, t, c)\Big\rangle.\end{aligned}$$

Evaluating the same norm identity on the dispreferred input gives

$$\begin{aligned}&2\Big\langle v_{\text{old}}(\mathbf{x}_t^l, t, c) - u_t(\mathbf{x}_t^l \mid \mathbf{x}_0^l), v_\theta(\mathbf{x}_t^l, t, c) - v_{\text{old}}(\mathbf{x}_t^l, t, c)\Big\rangle \\ &\quad = \|v_\theta(\mathbf{x}_t^l, t, c) - u_t(\mathbf{x}_t^l \mid \mathbf{x}_0^l)\|_2^2 \\ &\qquad - \|v_{\text{old}}(\mathbf{x}_t^l, t, c) - u_t(\mathbf{x}_t^l \mid \mathbf{x}_0^l)\|_2^2 - \|v_\theta(\mathbf{x}_t^l, t, c) - v_{\text{old}}(\mathbf{x}_t^l, t, c)\|_2^2.\end{aligned}$$

Substituting this identity gives coefficients $-\beta$, $\beta^2 + \beta = \beta(\beta + 1)$, and $1 + \beta$:

$$\begin{aligned}\|\nu_\theta(\mathbf{x}_t^l, t, c) - u_t(\mathbf{x}_t^l \mid \mathbf{x}_0^l)\|_2^2 = -\beta\|v_\theta(\mathbf{x}_t^l, t, c) - u_t(\mathbf{x}_t^l \mid \mathbf{x}_0^l)\|_2^2 \\ + \beta(\beta + 1)\|v_\theta(\mathbf{x}_t^l, t, c) - v_{\text{old}}(\mathbf{x}_t^l, t, c)\|_2^2 \\ + (1 + \beta)\|v_{\text{old}}(\mathbf{x}_t^l, t, c) - u_t(\mathbf{x}_t^l \mid \mathbf{x}_0^l)\|_2^2.\end{aligned}$$

Take expectations over the preferred and dispreferred samples, respectively, and add. The first terms give $\beta\mathcal{L}_{\text{FlowDPO}}^{\text{simple}}$, the second terms give the two corrections in equation 86, and the last terms form C. The latter use their respective sample targets and are independent of θ because the data and v_{old} are fixed during differentiation. □

Interpretation. The theorem expresses FlowCPO as β times simplified FlowDPO, plus two corrections determined by the distance between v_θ and v_{old} and a term C that is constant during each gradient step. At $\beta = 1$, only the quadratic constraint on dispreferred inputs remains, with coefficient 2. For $0 < \beta < 1$, including our default $\beta = 0.5$, the preferred correction has a negative coefficient, so the two corrections should not both be described as positive regularizers. The ablation analysis of β is in Fig. 9.

Why use EMA for interpolation? We use EMA so that the interpolation reference follows the model being trained. Thm. B.1 shows that the two correction terms depend on $v_\theta - v_{\text{old}}$ at the corresponding preferred or dispreferred input. These corrections depend on the squared prediction differences, so a reference that falls far behind can substantially change the objective. If we use the frozen v_{ref} for interpolation, the reference stays at initialization while v_θ changes during fine-tuning. EMA updates the parameters of v_{old} from recent training iterates, with the aim of keeping this gap small. Thus, v_{ref} provides the fixed in-domain offline data, while v_{old} follows training and is used to form μ_θ and ν_θ.

Advantages of FlowCPO over FlowDPO. For $\lambda = 1$, the two losses can be compared directly:

$$\begin{aligned}\mathcal{L}^{\text{simple}}_{\text{FlowDPO}}(\theta) &= \mathbb{E}\Big[\|v_\theta(\mathbf{x}_t^w, t, c) - (\epsilon - \mathbf{x}_0^w)\|_2^2 - \|v_\theta(\mathbf{x}_t^l, t, c) - (\epsilon - \mathbf{x}_0^l)\|_2^2\Big], \\ \mathcal{L}_{\text{FLOWCPO}}(\theta) &= \mathbb{E}\Big[\|\mu_\theta(\mathbf{x}_t^w, t, c) - (\epsilon - \mathbf{x}_0^w)\|_2^2 + \|\nu_\theta(\mathbf{x}_t^l, t, c) - (\epsilon - \mathbf{x}_0^l)\|_2^2\Big].\end{aligned} \tag{87}$$

The minus sign allows simplified FlowDPO to decrease without bound if the rejected error grows while the preferred error stays bounded. FLOWCPO adds two nonnegative errors, so its loss is bounded below by zero. This removes one potential source of instability, although a lower bound alone does not guarantee stable training.

C Connections to Existing Methods Under the Unified Divergence Framework

Sec. 4.3 and Tab. 2 summarize the connections under the unified divergence-based framework.

C.1 Connection to FlowGRPO

FlowGRPO (Liu et al., 2025a) is an online RL algorithm for flow matching models. Ignoring the KL term added in practice for stabilization, its core objective is

$$\max_{\theta} \mathbb{E}_{c,\,\mathbf{x}_0\sim\pi_\theta(\cdot|c)}[r(\mathbf{x}_0,c)]$$

By Thm. 4.1, this is exactly equivalent to

$$\min_{\theta} \mathbb{E}_c\left[\mathcal{D}_{KL}(\pi_\theta\|\pi^+) - \mathcal{D}_{KL}(\pi_\theta\|\pi^-)\right]$$

Hence FlowGRPO belongs to the reverse-KL branch of equation 12 with

$$q_\theta^+ = q_\theta^- = \pi_\theta, \qquad \alpha = 1, \qquad \gamma = -1,$$

plus the usual stabilization term $\mathcal{D}_{KL}(\pi_\theta\|\pi_{\text{ref}})$ used in practice.

C.2 Connection to DiffusionNFT

DiffusionNFT (Zheng et al., 2025) optimizes a supervised loss of the form

$$\mathcal{L}_{\text{NFT}}(\theta) = \mathbb{E}_{\substack{c,\,\mathbf{x}_0\sim\pi_{\text{old}}(\cdot|c)\\ t,\,\epsilon}}\Big[p(o=1\mid\mathbf{x}_0,c)\,\|\mu_\theta(\mathbf{x}_t,t,c) - u_t(\mathbf{x}_t\mid\mathbf{x}_0)\|_2^2 + p(o=0\mid\mathbf{x}_0,c)\,\|\nu_\theta(\mathbf{x}_t,t,c) - u_t(\mathbf{x}_t\mid\mathbf{x}_0)\|_2^2\Big]. \tag{88}$$

Similar to equation 6, we can have:

$$\pi_{\text{old}}(\mathbf{x}_0\mid c)\,p(o=1\mid\mathbf{x}_0,c) = \pi^+(\mathbf{x}_0\mid c)\,p_{\pi_{\text{old}}}(o=1\mid c)$$
$$\pi_{\text{old}}(\mathbf{x}_0\mid c)\,p(o=0\mid\mathbf{x}_0,c) = \pi^-(\mathbf{x}_0\mid c)\,p_{\pi_{\text{old}}}(o=0\mid c)$$

Therefore, with formula equation 32 in Thm. A.7, equation 88 can be rewritten as

$$\begin{aligned}
\mathcal{L}_{\text{NFT}}(\theta) =& \mathbb{E}_c\Big[p_{\pi_{\text{old}}}(o=1\mid c)\,\mathbb{E}_{\substack{\mathbf{x}_0\sim\pi^+(\cdot|c)\\ t,\,\epsilon}}\|\mu_\theta(\mathbf{x}_t,t,c) - u_t\|_2^2\Big] \\
&+ \mathbb{E}_c\Big[p_{\pi_{\text{old}}}(o=0\mid c)\,\mathbb{E}_{\substack{\mathbf{x}_0\sim\pi^-(\cdot|c)\\ t,\,\epsilon}}\|\nu_\theta(\mathbf{x}_t,t,c) - u_t\|_2^2\Big] \\
\geq\ & \mathbb{E}_c\Big[p_{\pi_{\text{old}}}(o=1\mid c)\,\mathbb{E}_{\pi^+}\left[-\log\pi_\theta^+(\mathbf{x}_0\mid c)\right] \\
&\quad + p_{\pi_{\text{old}}}(o=0\mid c)\,\mathbb{E}_{\pi^-}\left[-\log\pi_\theta^-(\mathbf{x}_0\mid c)\right]\Big] + C \\
=\ & \mathbb{E}_c[p_{\pi_{\text{old}}}(o=1\mid c)D_{KL}(\pi^+\|\pi_\theta^+) + p_{\pi_{\text{old}}}(o=0\mid c)D_{KL}(\pi^-\|\pi_\theta^-)] + C
\end{aligned}$$

where C is a constant independent of θ. The both sides achieve the same optimal solution when $\mu_{\theta^\star}(\mathbf{x}_t^+,t,c) = \mathbb{E}[\mathbf{x}_1 - \mathbf{x}_0^+ \mid \mathbf{x}_t^+,t,c]$ and $\nu_{\theta^\star}(\mathbf{x}_t^-,t,c) = \mathbb{E}[\mathbf{x}_1 - \mathbf{x}_0^- \mid \mathbf{x}_t^-,t,c]$.

Therefore, DiffusionNFT corresponds to the forward-KL branch of equation 12 with

$$q_\theta^+ = \pi_\theta^+, \qquad q_\theta^- = \pi_\theta^-, \qquad \alpha = p_{\pi_{\text{old}}}(o=1\mid c), \qquad \gamma = p_{\pi_{\text{old}}}(o=0\mid c).$$

C.3 Connection to AWM

AWM (Xue et al., 2025) aims to optimize the following objective:

$$\max_{\theta} \mathbb{E}_{c,\,\mathbf{x}_0\sim\pi_\theta(\cdot|c)}[r(\mathbf{x}_0,c)] \tag{89}$$

From equation 14, we have:

$$-\log \pi_\theta(x_0 \mid c) = H(q_\varepsilon^{x_0}) + \mathbb{E}_{t,\,\epsilon}[w(t)\|v_\theta(\mathbf{x}_t, t, c) - (\mathbf{x}_1 - x_0)\|^2] + \mathcal{G}_{\varepsilon,w}(\theta; x_0) + \mathcal{B}_\varepsilon(\theta; x_0) \tag{90}$$

We set $\bar{\theta}$ as the stop gradient of θ, we have:

$$\begin{aligned}\log \frac{\pi_\theta(\mathbf{x}_0 \mid c)}{\pi_{\bar{\theta}}(\mathbf{x}_0 \mid c)} &= -\mathbb{E}_{t,\epsilon}[w(t)\|v_\theta(\mathbf{x}_t, t, c) - (\mathbf{x}_1 - \mathbf{x}_0)\|^2] \\ &\quad + \mathbb{E}_{t,\epsilon}[w(t)\|v_{\bar{\theta}}(\mathbf{x}_t, t, c) - (\mathbf{x}_1 - \mathbf{x}_0)\|^2] \\ &:= \Delta\mathcal{L}_{\mathrm{FM}}\end{aligned}$$

Therefore, the objective equation 89 can be rewritten as

$$\begin{aligned}&\max_\theta \mathbb{E}_{c,\,\mathbf{x}_0\sim\pi_\theta(\cdot|c)}[r(\mathbf{x}_0, c)] \\ &= \mathbb{E}_{c,\,\mathbf{x}_0\sim\pi_{\bar{\theta}}(\cdot|c)}\left[\frac{\pi_\theta(\mathbf{x}_0 \mid c)}{\pi_{\bar{\theta}}(\mathbf{x}_0 \mid c)} r(\mathbf{x}_0, c)\right] \\ &= \mathbb{E}_{c,\,\mathbf{x}_0\sim\pi_{\bar{\theta}}(\cdot|c)}\left[\exp(\Delta\mathcal{L}_{\mathrm{FM}}) r(\mathbf{x}_0, c)\right]\end{aligned}$$

Therefore, AWM is optimized over forward process, while its derivation of $q_\theta^+, q_\theta^-, \alpha, \gamma$ is the same as FlowGRPO. Thus,

$$q_\theta^+ = q_\theta^- = \pi_\theta, \qquad \alpha = 1, \qquad \gamma = -1,$$

Because $\Delta\mathcal{L}_{\mathrm{FM}} = 0$ due to $\bar{\theta}$ is the stop gradient of θ, we know $\exp(x) = 1 + x$ when x is small. Therefore, we can simplify the objective as

$$\begin{aligned}&\max_\theta \mathbb{E}_{c,\,\mathbf{x}_0\sim\pi_\theta(\cdot|c)}[r(\mathbf{x}_0, c)] \\ &= \mathbb{E}_{c,\,\mathbf{x}_0\sim\pi_{\bar{\theta}}(\cdot|c)}\left[\exp(\Delta\mathcal{L}_{\mathrm{FM}}) r(\mathbf{x}_0, c)\right] \\ &= \mathbb{E}_{c,\,\mathbf{x}_0\sim\pi_{\bar{\theta}}(\cdot|c)}\left[(1 + \Delta\mathcal{L}_{\mathrm{FM}}) r(\mathbf{x}_0, c)\right] \\ &= \mathbb{E}_{c,\,\mathbf{x}_0\sim\pi_{\bar{\theta}}(\cdot|c)}\left[r(\mathbf{x}_0, c) + \Delta\mathcal{L}_{\mathrm{FM}} r(\mathbf{x}_0, c)\right] \\ &= -\mathbb{E}_{c,\,\mathbf{x}_0\sim\pi_{\bar{\theta}}(\cdot|c)}\left[r(\mathbf{x}_0, c)\mathbb{E}_{t,\epsilon}[w(t)\|v_\theta(\mathbf{x}_t, t, c) - (\mathbf{x}_1 - \mathbf{x}_0)\|^2]\right] + C \\ &= -\mathbb{E}_{c,\mathbf{x}_0\sim\pi_{\bar{\theta}}(\cdot|c),t,\epsilon}\left[w(t) r(\mathbf{x}_0, c)\|v_\theta(\mathbf{x}_t, t, c) - (\mathbf{x}_1 - \mathbf{x}_0)\|^2\right] + C\end{aligned}$$

C.4 Connection to RAM

First, let's remind the objective of RAM (Bergmeister et al., 2026):

$$\mathcal{L}_{\mathrm{RAM}}(\theta) = \mathbb{E}_{c,t}[\|v_\theta(\mathbf{x}_t, t, c) - sg(v_{\mathrm{ref}}(\mathbf{x}_t, t, c) + r(\mathbf{x}_0, c)((\mathbf{x}_1 - \mathbf{x}_0) - v_\theta(\mathbf{x}_t, t, c)))\|^2]$$

where sg is the stop gradient operator.

Let us define a new objective:

$$\mathcal{L}_2(\theta) = \mathbb{E}_{c,t}[r(\mathbf{x}_0, c)\|v_\theta(\mathbf{x}_t, t, c) - (\mathbf{x}_1 - \mathbf{x}_0)\|^2] + \mathbb{E}_{c,t}[\|v_\theta(\mathbf{x}_t, t, c) - v_{\mathrm{ref}}(\mathbf{x}_t, t, c)\|^2]$$

It is easy to check that $\nabla_\theta\mathcal{L}_2(\theta) = \nabla_\theta\mathcal{L}_{\mathrm{RAM}}(\theta)$.

Therefore, RAM can be considered as AWM objective plus a reference term.

Thus, it is also optimized over forward process, while its derivation of $q_\theta^+, q_\theta^-, \alpha, \gamma$ is the same as FlowGRPO. Thus,

$$q_\theta^+ = q_\theta^- = \pi_\theta, \qquad \alpha = 1, \qquad \gamma = -1,$$

C.5 Connection to SFT

The reference distribution decomposes as a mixture of the positive and negative posteriors.

Lemma C.1. *For every context c,*

$$\pi_{\text{ref}}(\mathbf{x}_0 \mid c) = p_{\pi_{\text{ref}}}(o=1 \mid c)\,\pi^+(\mathbf{x}_0 \mid c) + p_{\pi_{\text{ref}}}(o=0 \mid c)\,\pi^-(\mathbf{x}_0 \mid c).$$

Proof. Multiplying the definitions in equation 6 by the corresponding normalizing constants and summing the two identities gives

$$\begin{aligned} &p_{\pi_{\text{ref}}}(o=1 \mid c)\,\pi^+(\mathbf{x}_0 \mid c) + p_{\pi_{\text{ref}}}(o=0 \mid c)\,\pi^-(\mathbf{x}_0 \mid c) \\ &= \pi_{\text{ref}}(\mathbf{x}_0 \mid c)\,[p(o=1 \mid \mathbf{x}_0, c) + p(o=0 \mid \mathbf{x}_0, c)] \\ &= \pi_{\text{ref}}(\mathbf{x}_0 \mid c). \end{aligned}$$

□

Standard SFT maximizes the log-likelihood of samples from π_{ref}. By Thm. A.7, we know that when $\mathbb{E}_{\mathbf{x}_0\sim\pi_{\text{ref}}(\cdot|c)}\left[\|v_\theta(\mathbf{x}_t,t) - (\mathbf{x}_1 - \mathbf{x}_0)\|^2\right]$ achieves the minimum, $\mathbb{E}_{\mathbf{x}_0\sim\pi_{\text{ref}}(\cdot|c)}[-\log \pi_\theta(\mathbf{x}_0 \mid c)]$ also achieves the minimum. Therefore, we have: By Lem. C.1,

$$\begin{aligned} &= \arg\min_\theta \mathbb{E}_{\mathbf{x}_0\sim\pi_{\text{ref}}(\cdot|c)}[-\log \pi_\theta(\mathbf{x}_0 \mid c)] \\ &= \int \pi_{\text{ref}}(\mathbf{x}_0 \mid c)[-\log \pi_\theta(\mathbf{x}_0 \mid c)]d\mathbf{x}_0 \\ &= \int (p_{\pi_{\text{ref}}}(o=1 \mid c)\,\pi^+(\mathbf{x}_0 \mid c) + p_{\pi_{\text{ref}}}(o=0 \mid c)\,\pi^-(\mathbf{x}_0 \mid c))[-\log \pi_\theta(\mathbf{x}_0 \mid c)]d\mathbf{x}_0 \\ &= p_{\pi_{\text{ref}}}(o=1 \mid c)\,\mathcal{D}_{KL}(\pi^+\|\pi_\theta) + p_{\pi_{\text{ref}}}(o=0 \mid c)\,\mathcal{D}_{KL}(\pi^-\|\pi_\theta) \end{aligned}$$

Thus SFT corresponds to the forward-KL branch of equation 12 with

$$q_\theta^+ = q_\theta^- = \pi_\theta, \qquad \alpha = p_{\pi_{\text{ref}}}(o=1 \mid c), \qquad \gamma = p_{\pi_{\text{ref}}}(o=0 \mid c).$$

C.6 Connection to RFT

RFT keeps only preferred samples. By Thm. A.7, we know that when $\mathbb{E}_{\mathbf{x}_0\sim\pi_{\text{ref}}(\cdot|c)}\left[\|v_\theta(\mathbf{x}_t,t) - (\mathbf{x}_1 - \mathbf{x}_0)\|^2\right]$ achieves the minimum, $\mathbb{E}_{\mathbf{x}_0\sim\pi_{\text{ref}}(\cdot|c)}[-\log \pi_\theta(\mathbf{x}_0 \mid c)]$ also achieves the minimum. Therefore, we have:

$$\begin{aligned} &= \arg\min_\theta \mathbb{E}_{\mathbf{x}_0\sim\pi^+(\cdot|c)}[-\log \pi_\theta(\mathbf{x}_0 \mid c)] \\ &= \arg\min_\theta \mathcal{D}_{KL}(\pi^+\|\pi_\theta) \end{aligned}$$

Therefore RFT corresponds to the forward-KL branch of equation 12 with

$$q_\theta^+ = \pi_\theta, \qquad \alpha = 1, \qquad \gamma = 0.$$

C.7 Connection to FlowDPO

The connection to FlowDPO (Liu et al., 2025b) is heuristic rather than formal. Starting from the KL-regularized RLHF objective, DPO uses the reward parameterization

$$r(\mathbf{x}_0, c) = \eta \log \frac{\pi_\theta(\mathbf{x}_0 \mid c)}{\pi_{\text{ref}}(\mathbf{x}_0 \mid c)} + \eta \log Z(c), \tag{91}$$

which, when substituted into the Bradley–Terry preference likelihood, yields the usual DPO objective

$$\mathcal{L}_{\text{DPO}}(\theta) = \mathbb{E}_{c,\mathbf{x}_0^w,\mathbf{x}_0^l}\left[-\log\sigma\left(\eta \log \frac{\pi_\theta(\mathbf{x}_0^w \mid c)}{\pi_{\text{ref}}(\mathbf{x}_0^w \mid c)} - \eta \log \frac{\pi_\theta(\mathbf{x}_0^l \mid c)}{\pi_{\text{ref}}(\mathbf{x}_0^l \mid c)}\right)\right]$$

From equation 14, we have:

$$-\log \pi_\theta(x_0 \mid c) = H(q_\varepsilon^{x_0}) + \mathbb{E}_{t,\,\epsilon}[w(t)\|v_\theta(\mathbf{x}_t,t,c) - (\mathbf{x}_1 - x_0)\|^2] + \mathcal{G}_{\varepsilon,w}(\theta; x_0) + \mathcal{B}_\varepsilon(\theta; x_0)$$

Therefore, we have:

$$\log \frac{\pi_\theta(\mathbf{x}_0 \mid c)}{\pi_{\text{ref}}(\mathbf{x}_0 \mid c)} = \Delta\mathcal{L}_{\text{FM}}(\mathbf{x}_0) + \Delta\mathcal{G}_{\varepsilon,w}(\mathbf{x}_0) + \Delta\mathcal{B}_\varepsilon(\mathbf{x}_0) \tag{92}$$

where

$$\Delta\mathcal{L}_{\mathrm{FM}}(\mathbf{x}_0) := \mathbb{E}_{t,\,\epsilon}[-w(t)\|v_\theta(\mathbf{x}_t,t,c)-(\mathbf{x}_1-\mathbf{x}_0)\|^2 + w(t)\|v_{\mathrm{ref}}(\mathbf{x}_t,t,c)-(\mathbf{x}_1-\mathbf{x}_0)\|^2]$$
$$\Delta\mathcal{G}_{\varepsilon,w}(\mathbf{x}_0) := -\mathcal{G}_{\varepsilon,w}(\theta;\mathbf{x}_0) + \mathcal{G}_{\varepsilon,w}(\mathrm{ref};\mathbf{x}_0)$$
$$\Delta\mathcal{B}_{\varepsilon}(\mathbf{x}_0) := -\mathcal{B}_{\varepsilon}(\theta;\mathbf{x}_0) + \mathcal{B}_{\varepsilon}(\mathrm{ref};\mathbf{x}_0)$$

Therefore, we can simplify the DPO objective in the following:

$$\begin{aligned}
\mathcal{L}_{\mathrm{DPO}}(\theta) &= \mathbb{E}_{c,\mathbf{x}_0^w,\mathbf{x}_0^l}\big[-\log\sigma\big(\eta\{\Delta\mathcal{L}_{\mathrm{FM}}(\mathbf{x}_0^w)+\Delta\mathcal{G}_{\varepsilon,w}(\mathbf{x}_0^w)+\Delta\mathcal{B}_{\varepsilon}(\mathbf{x}_0^w)\\
&\qquad\qquad -\Delta\mathcal{L}_{\mathrm{FM}}(\mathbf{x}_0^l)-\Delta\mathcal{G}_{\varepsilon,w}(\mathbf{x}_0^l)-\Delta\mathcal{B}_{\varepsilon}(\mathbf{x}_0^l)\}\big)\\
&= \mathbb{E}_{c,\mathbf{x}_0^w,\mathbf{x}_0^l}\big[-\log\sigma\big(\eta(\Delta\mathcal{L}_{\mathrm{FM}}(\mathbf{x}_0^w)-\Delta\mathcal{L}_{\mathrm{FM}}(\mathbf{x}_0^l))\\
&\qquad\qquad +\eta(\Delta\mathcal{G}_{\varepsilon,w}(\mathbf{x}_0^w)+\Delta\mathcal{B}_{\varepsilon}(\mathbf{x}_0^w)-\Delta\mathcal{G}_{\varepsilon,w}(\mathbf{x}_0^l)-\Delta\mathcal{B}_{\varepsilon}(\mathbf{x}_0^l))\}\big)\big]\\
&\quad\text{because } -\log\sigma \text{ is convex and then by Jensen's inequality}\\
&\le \mathbb{E}_{c,\mathbf{x}_0^w,\mathbf{x}_0^l}\big[-\frac{1}{2}\log\sigma\big(2\eta(\Delta\mathcal{L}_{\mathrm{FM}}(\mathbf{x}_0^w)-\Delta\mathcal{L}_{\mathrm{FM}}(\mathbf{x}_0^l))\big)\\
&\qquad\qquad -\frac{1}{2}\log\sigma\big(2\eta(\Delta\mathcal{G}_{\varepsilon,w}(\mathbf{x}_0^w)+\Delta\mathcal{B}_{\varepsilon}(\mathbf{x}_0^w)-\Delta\mathcal{G}_{\varepsilon,w}(\mathbf{x}_0^l)-\Delta\mathcal{B}_{\varepsilon}(\mathbf{x}_0^l))\big)\big]\\
&\le \frac{1}{2}\mathcal{L}_{\mathrm{FlowDPO}}(\theta)\\
&\qquad +\mathbb{E}_{c,\mathbf{x}_0^w,\mathbf{x}_0^l}\big[-\frac{1}{2}\log\sigma\big(2\eta(\Delta\mathcal{G}_{\varepsilon,w}(\mathbf{x}_0^w)+\Delta\mathcal{B}_{\varepsilon}(\mathbf{x}_0^w)-\Delta\mathcal{G}_{\varepsilon,w}(\mathbf{x}_0^l)-\Delta\mathcal{B}_{\varepsilon}(\mathbf{x}_0^l))\big)\big]
\end{aligned}$$

where

$$\delta_\theta(\mathbf{x}_0) := \|v_\theta(\mathbf{x}_t,t,c)-(\mathbf{x}_1-\mathbf{x}_0)\|^2 - \|v_{\mathrm{ref}}(\mathbf{x}_t,t,c)-(\mathbf{x}_1-\mathbf{x}_0)\|^2.$$

$$\mathcal{L}_{\mathrm{FlowDPO}}(\theta) = \mathbb{E}_{c,\mathbf{x}_0^w,\mathbf{x}_0^l,t,\epsilon}\big[-\log\sigma\big(2\eta w(t)[\delta_\theta(\mathbf{x}_0^l)-\delta_\theta(\mathbf{x}_0^w)]\big)\big].$$

Therefore, the FlowDPO objective plus a residual term is an upper bound of the DPO objective.

Algorithm 1 Flow Contrastive Preference Optimization (FlowCPO)

Require: Pretrained v_{ref}, dataset $\mathcal{D} = \{(c, \mathbf{x}_0^w, \mathbf{x}_0^l)\}$, flow interpolation coefficient β, negative regularization weight λ, EMA coefficient η, learning rate ι.

Init: $v_\theta \leftarrow v_{\text{ref}}$, $v_{\text{old}} \leftarrow v_{\text{ref}}$.
for each training iteration **do**
  Sample batch $(c, \mathbf{x}_0^w, \mathbf{x}_0^l) \sim \mathcal{D}$, $t \sim \mathcal{U}(0,1)$, $\epsilon \sim \mathcal{N}(0, I)$.
  Forward: $\mathbf{x}_t^{\{w,l\}} = (1-t) \cdot \mathbf{x}_0^{\{w,l\}} + t \cdot \epsilon$; Target $u_t^{\{w,l\}} = \epsilon - \mathbf{x}_0^{\{w,l\}}$.
  Flow Mixing:
  $\mu_\theta = (1-\beta)v_{\text{old}} + \beta v_\theta$, $\nu_\theta = (1+\beta)v_{\text{old}} - \beta v_\theta$.
  Loss Update:
  $\mathcal{L} = \|\mu_\theta(\mathbf{x}_t^w, c, t) - u_t(\mathbf{x}_t^w|\mathbf{x}_0^w)\|_2^2 + \lambda\|\nu_\theta(\mathbf{x}_t^l, c, t) - u_t(\mathbf{x}_t^l|\mathbf{x}_0^l)\|_2^2$
  $\theta \leftarrow \theta - \iota \cdot \nabla_\theta \mathcal{L}$.
  $v_{\text{old}} \leftarrow EMA_\eta(v_\theta, v_{\text{old}})$.
end for
return v_θ

D Experimental Details

D.1 Experimental Details of Sec. 5.2

In-Domain Offline Preference Data. For the *in-domain* offline training setting, we build a static preference dataset from a frozen copy of the pretrained reference model `Stable Diffusion 3.5 Medium (SD3.5-M)`. We sample prompts from the three target sources: the GenEval prompt set, the OCR prompt set, and the general-preference prompt set. The resulting in-domain prompt pools contain 50,000 training prompts for GenEval, 19,653 training prompts for OCR, and 25,432 training prompts for general preference. Their held-out evaluation splits contain 2212, 1018, and 2048 prompts, respectively. For each prompt, we generate 16 candidate images with the frozen reference model and convert them into a preferred/dispreferred pair using a fixed offline pipeline: the highest-scoring sample becomes the preferred and the lowest-scoring sample becomes the dispreferred. The GenEval and OCR training data use single-metric filtering aligned with their target benchmark. The general-preference training data uses an equal-weight multi-reward score with `PickScore`:`HPS v2.1`:`CLIP Score` = 1:1:1 during preferred/dispreferred construction. Because the generator parameters stay fixed during data construction, the resulting training distribution remains stationary throughout optimization. These preferred/dispreferred pairs should be read as empirical source distributions induced by the fixed best-of-16 pipeline, rather than as exact samples from the idealized posteriors π^+ and π^- introduced in the theory.

All images used for data construction and evaluation are generated at a resolution of 512×512 with 40 inference steps using the ODE sampler. Unless otherwise noted, we use a guidance scale of 4.5 during dataset construction. We leave negative prompts blank. The VAE and text encoder are frozen, and training prompts do not overlap with validation prompts or evaluation benchmarks. The complete reported experiment suite used approximately 288 aggregate A100 GPU-hours.

Out-of-Domain Offline Preference Data. For the *out-of-domain* offline training setting, we use Open Image Preferences v1 Results from the Data Is Better Together collection on Hugging Face (Data Is Better Together, 2024). This dataset contains roughly 10K text-to-image preference pairs with community annotations, and the images are generated by open models including FLUX.1-Dev and SD3.5-Large. We use it to evaluate the out-of-domain regime because the training samples are not generated by the reference model used in our fine-tuning experiments.

Training Configuration. We fine-tune `SD3.5-M` with LoRA on 8×A100 GPUs in fp16 precision. Unless otherwise specified, the LoRA hyperparameters are $r = 32$ and $\alpha = 64$. We optimize with AdamW using a learning rate of 3e-4, cosine decay, and weight decay of 1e-4. The global batch size is 16. The EMA decay η is 0.99, the default value of λ is 1, and reported checkpoints are selected with the corresponding validation score for each training target.

Checkpoint Selection. All fine-tuning runs are monitored for 2K training steps. For the in-domain specialists, we report the checkpoint with the best task-aligned validation score within that 2K-step

budget: the GenEval model is selected by GenEval validation score, the OCR model by OCR validation score, and the general-preference model by the equal-weight composite validation score over `PickScore`, `HPS v2.1`, and `CLIP Score`. For the OOD setting, where one model is shared across all downstream evaluations, we select the checkpoint with the highest validation `PickScore` within the same 2K-step budget. We apply the same protocol to FlowCPO and the offline baselines in the corresponding setting.

D.2 Experimental Results on Optimization Targets

We report results for three capabilities: semantic alignment, typographic generation, and general preference alignment. Tab. 5, Tab. 6, and Tab. 7 correspond to *in-domain* fine-tuning with domain-specific reward filtering, whereas Tab. 8 reports the *out-of-domain* setting. The first three tables therefore summarize separate in-domain specialist models, while the OOD table evaluates single models trained once on out-of-domain data. Unless otherwise noted, each fine-tuned `SD3.5-M` entry is the mean over five independent training and evaluation runs with different random seeds, and the uncertainty is the sample standard deviation of the five run-level metric values. Pretrained baselines are shown as single evaluations for reference.

Tab. 5 reports quantitative results on `GenEval`, with representative qualitative comparisons in Fig. 3. Tab. 6 summarizes typographic generation performance measured by OCR accuracy, with examples in Fig. 4. Tab. 7 presents the in-domain results for general preference alignment; because training data in that setting are filtered with `PickScore`, `CLIP Score`, and `HPS v2.1`, the remaining metrics are the more informative check of transfer beyond the filtering pipeline. Tab. 8 reports the out-of-domain setting, with qualitative examples in Fig. 6, Fig. 7, and Fig. 8. Unless otherwise noted, we use FlowDPO temperature 100 and FlowCPO $\beta = 0.5$ as the default settings.

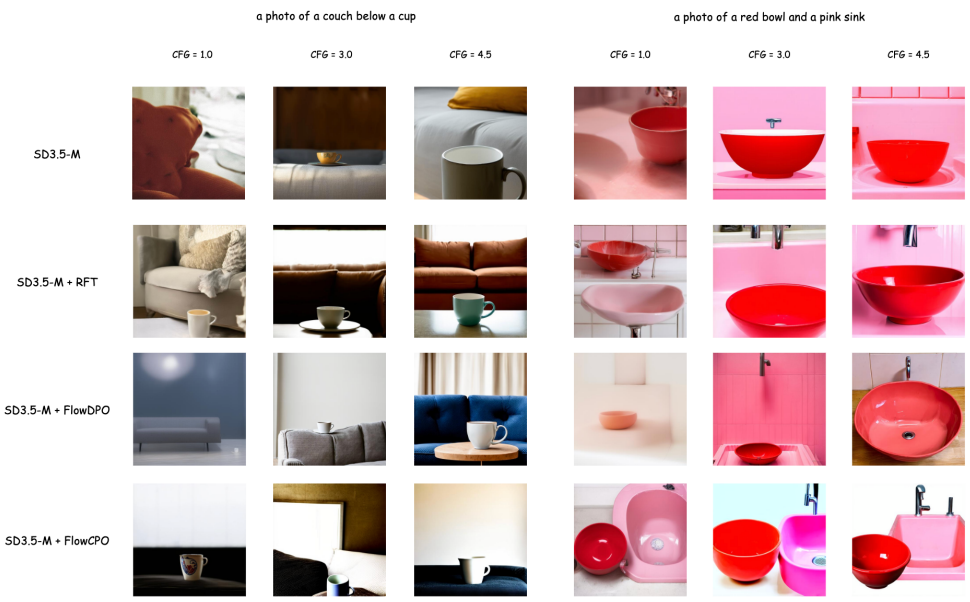


Figure 3: Qualitative comparison on the GenEval benchmark. All samples are generated by models trained with *in-domain* preference data.

Table 5: Quantitative comparison of fine-tuning SD3.5-M using different methods under the *in-domain* offline training setting. The fine-tuning data are filtered solely by the **GenEval** reward function. We evaluate each model with several Classifier-Free Guidance (CFG) scales. The broad set of auxiliary metrics is included to monitor whether optimizing only for `GenEval` degrades other capabilities. Fine-tuned SD3.5-M variants are reported as mean ± standard deviation over 5 runs; pretrained baselines are listed as single evaluations. Within the SD3.5-M group, the best results are highlighted in **bold**, and the second best are underlined.

Model	CFG	GenEval	OCR	PickScore	ClipScore
Pretrained Model Baselines					
SD-XL	–	0.55	0.14	22.42	0.287
SD3.5-L	–	0.71	0.68	22.91	0.289
FLUX.1-Dev	–	0.66	0.59	22.84	0.295
SD3.5-M Fine-Tuning (filtered by GenEval)					
Base Model	1.0	0.24	0.12	20.51	0.237
	3.0	0.59	0.47	22.28	0.287
	4.5	0.63	**0.59**	22.34	0.285
+ RFT	1.0	0.5883 ± 0.0100	0.1421 ± 0.0033	21.6937 ± 0.0125	0.2735 ± 0.0009
	3.0	0.7423 ± 0.0097	0.5003 ± 0.0070	$\underline{22.4661 \pm 0.0212}$	0.2941 ± 0.0006
	4.5	0.7489 ± 0.0045	0.5562 ± 0.0098	$\mathbf{22.4864 \pm 0.0099}$	0.2966 ± 0.0005
+ FlowDPO	1.0	0.5892 ± 0.0046	0.2327 ± 0.0053	21.4067 ± 0.0324	0.2696 ± 0.0010
	3.0	0.8118 ± 0.0044	0.4534 ± 0.0075	22.3095 ± 0.0182	$\underline{0.2970 \pm 0.0006}$
	4.5	0.8107 ± 0.0046	0.5067 ± 0.0074	22.3091 ± 0.0138	$\mathbf{0.2979 \pm 0.0005}$
+ FlowCPO ($\beta = 0.5$, Ours)	1.0	0.7596 ± 0.0106	0.2535 ± 0.0098	21.8045 ± 0.0157	0.2804 ± 0.0001
	3.0	$\mathbf{0.8415 \pm 0.0036}$	0.5203 ± 0.0055	22.1952 ± 0.0184	0.2957 ± 0.0005
	4.5	$\underline{0.8161 \pm 0.0052}$	$\underline{0.5702 \pm 0.0082}$	21.9438 ± 0.0086	0.2933 ± 0.0005

Model	CFG	HPSv2.1	Aesthetic	ImgRwd	UniRwd
Pretrained Model Baselines					
SD-XL	–	0.280	5.60	0.76	2.93
SD3.5-L	–	0.288	5.50	0.96	3.25
FLUX.1-Dev	–	0.274	5.71	0.96	3.27
SD3.5-M Fine-Tuning (filtered by GenEval)					
Base Model	1.0	0.204	5.13	−0.58	2.02
	3.0	0.284	5.38	0.71	2.96
	4.5	0.279	5.36	0.85	3.03
+ RFT	1.0	0.2686 ± 0.0005	5.3183 ± 0.0155	0.4381 ± 0.0179	2.6175 ± 0.0135
	3.0	$\underline{0.3010 \pm 0.0006}$	$\underline{5.3998 \pm 0.0092}$	1.0200 ± 0.0172	3.1360 ± 0.0145
	4.5	$\mathbf{0.3031 \pm 0.0002}$	$\mathbf{5.4070 \pm 0.0050}$	1.0808 ± 0.0074	$\underline{3.1615 \pm 0.0080}$
+ FlowDPO	1.0	0.2562 ± 0.0007	5.2817 ± 0.0058	0.4560 ± 0.0225	2.5870 ± 0.0205
	3.0	0.2938 ± 0.0009	5.3570 ± 0.0112	$\underline{1.0811 \pm 0.0094}$	$\underline{3.1615 \pm 0.0200}$
	4.5	0.2963 ± 0.0003	5.3479 ± 0.0052	$\mathbf{1.1256 \pm 0.0078}$	$\mathbf{3.1680 \pm 0.0060}$
+ FlowCPO ($\beta = 0.5$, Ours)	1.0	0.2673 ± 0.0008	5.2487 ± 0.0139	0.6231 ± 0.0160	2.8040 ± 0.0125
	3.0	0.2885 ± 0.0007	5.2662 ± 0.0085	0.9999 ± 0.0076	3.1170 ± 0.0085
	4.5	0.2807 ± 0.0004	5.1610 ± 0.0062	0.9184 ± 0.0094	3.0245 ± 0.0150

Table 6: Quantitative comparison of fine-tuning SD3.5-M using different methods (RFT, FlowDPO, FlowCPO) under the *in-domain* offline training setting. The fine-tuning data are filtered solely by the **OCR** reward metric. We evaluate the models with several Classifier-Free Guidance (CFG) scales. The broad set of auxiliary metrics is included to monitor whether optimizing only for OCR degrades other generation capabilities. Fine-tuned SD3.5-M variants are reported as mean ± standard deviation over 5 runs; pretrained baselines are listed as single evaluations. Within the SD3.5-M fine-tuning group, the best results are highlighted in **bold**, and the second best are underlined.

Model	CFG	GenEval	OCR	PickScore	ClipScore
Pretrained Model Baselines					
SD-XL	–	0.55	0.14	22.42	0.287
SD3.5-L	–	0.71	0.68	22.91	0.289
FLUX.1-Dev	–	0.66	0.59	22.84	0.295
SD3.5-M Fine-Tuning (filtered by OCR)					
Base Model	1.0	0.24	0.12	20.51	0.237
	3.0	0.59	0.47	22.28	0.287
	4.5	0.63	0.59	22.34	0.285
+ RFT	1.0	0.5127 ± 0.0108	0.3530 ± 0.0057	21.8831 ± 0.0192	0.2784 ± 0.0007
	3.0	$\underline{0.6577 \pm 0.0106}$	0.6981 ± 0.0105	$\mathbf{22.5177 \pm 0.0194}$	$\underline{0.2964 \pm 0.0008}$
	4.5	$\mathbf{0.6698 \pm 0.0101}$	0.7202 ± 0.0083	$\underline{22.5098 \pm 0.0157}$	$\mathbf{0.2979 \pm 0.0009}$
+ FlowDPO	1.0	0.3140 ± 0.0051	0.5136 ± 0.0050	20.9171 ± 0.0090	0.2565 ± 0.0008
	3.0	0.5848 ± 0.0098	0.7389 ± 0.0096	22.3210 ± 0.0104	0.2927 ± 0.0004
	4.5	0.6149 ± 0.0153	0.7476 ± 0.0041	22.4249 ± 0.0144	0.2960 ± 0.0006
+ FlowCPO ($\beta = 0.5$, Ours)	1.0	0.4023 ± 0.0062	0.8349 ± 0.0033	21.2178 ± 0.0185	0.2689 ± 0.0006
	3.0	0.5962 ± 0.0113	$\mathbf{0.8737 \pm 0.0034}$	22.3139 ± 0.0090	0.2930 ± 0.0002
	4.5	0.6124 ± 0.0102	$\underline{0.8588 \pm 0.0078}$	22.3656 ± 0.0154	0.2953 ± 0.0008

Model	CFG	HPSv2.1	Aesthetic	ImgRwd	UniRwd
Pretrained Model Baselines					
SD-XL	–	0.280	5.60	0.76	2.93
SD3.5-L	–	0.288	5.50	0.96	3.25
FLUX.1-Dev	–	0.274	5.71	0.96	3.27
SD3.5-M Fine-Tuning (filtered by OCR)					
Base Model	1.0	0.204	5.13	−0.58	2.02
	3.0	0.284	5.38	0.71	2.96
	4.5	0.279	5.36	0.85	3.03
+ RFT	1.0	0.2741 ± 0.0005	5.3589 ± 0.0096	0.5248 ± 0.0191	2.7170 ± 0.0160
	3.0	$\underline{0.3008 \pm 0.0003}$	$\mathbf{5.3954 \pm 0.0043}$	$\underline{1.0491 \pm 0.0072}$	$\mathbf{3.1630 \pm 0.0200}$
	4.5	$\mathbf{0.3023 \pm 0.0005}$	$\underline{5.3927 \pm 0.0068}$	$\mathbf{1.0816 \pm 0.0100}$	$\underline{3.1605 \pm 0.0175}$
+ FlowDPO	1.0	0.2219 ± 0.0006	5.1246 ± 0.0127	-0.3015 ± 0.0273	2.2580 ± 0.0175
	3.0	0.2873 ± 0.0005	5.3620 ± 0.0025	0.8559 ± 0.0143	3.0315 ± 0.0220
	4.5	0.2943 ± 0.0005	5.3795 ± 0.0064	0.9493 ± 0.0182	3.0955 ± 0.0195
+ FlowCPO ($\beta = 0.5$, Ours)	1.0	0.2430 ± 0.0011	5.1770 ± 0.0102	0.0952 ± 0.0182	2.4955 ± 0.0185
	3.0	0.2943 ± 0.0005	5.3506 ± 0.0079	0.9560 ± 0.0115	3.0760 ± 0.0085
	4.5	0.2982 ± 0.0005	5.3591 ± 0.0048	1.0438 ± 0.0048	3.1160 ± 0.0035

Table 7: Quantitative comparison of fine-tuning SD3.5-M using different methods (RFT, FlowDPO, and FlowCPO) under the *in-domain* offline training setting. The fine-tuning data are filtered by a combination of multiple metrics (**PickScore**, **CLIP Score**, and **HPSv2.1**), which are highlighted in gray. We evaluate the models with several Classifier-Free Guidance (CFG) scales. Because the shaded metrics are directly coupled to the filtering pipeline, the remaining metrics are especially useful for assessing transfer beyond the optimization-aligned rewards. Fine-tuned SD3.5-M variants are reported as mean ± standard deviation over 5 runs; pretrained baselines are listed as single evaluations. Within the SD3.5-M fine-tuning group, the best results are highlighted in **bold**, and the second best are underlined.

Model	CFG	GenEval	OCR	PickScore	ClipScore
Pretrained Model Baselines					
SD-XL	–	0.55	0.14	22.42	0.287
SD3.5-L	–	0.71	0.68	22.91	0.289
FLUX.1-Dev	–	0.66	0.59	22.84	0.295
SD3.5-M Fine-Tuning					
Base Model	1.0	0.24	0.12	20.51	0.237
	3.0	0.59	0.47	22.28	0.287
	4.5	0.63	0.59	22.34	0.285
+ RFT	1.0	0.5312 ± 0.0111	0.2351 ± 0.0045	21.9056 ± 0.0124	0.2793 ± 0.0002
	3.0	$\underline{0.6894 \pm 0.0086}$	0.5722 ± 0.0045	22.5975 ± 0.0069	0.2963 ± 0.0007
	4.5	$\mathbf{0.6974 \pm 0.0056}$	$\mathbf{0.6180 \pm 0.0097}$	22.5664 ± 0.0104	0.2972 ± 0.0009
+ FlowDPO	1.0	0.0363 ± 0.0117	0.2132 ± 0.0061	20.7158 ± 0.0179	0.2399 ± 0.0006
	3.0	0.3903 ± 0.0100	0.5549 ± 0.0072	22.7637 ± 0.0209	0.2974 ± 0.0008
	4.5	0.4610 ± 0.0137	0.5791 ± 0.0071	$\underline{22.8912 \pm 0.0192}$	$\mathbf{0.3010 \pm 0.0008}$
+ FlowCPO ($\beta = 0.5$, Ours)	1.0	0.5325 ± 0.0035	0.3426 ± 0.0053	22.4805 ± 0.0155	0.2796 ± 0.0007
	3.0	0.6487 ± 0.0088	$\underline{0.5888 \pm 0.0096}$	$\mathbf{22.9432 \pm 0.0164}$	0.2994 ± 0.0005
	4.5	0.6526 ± 0.0079	0.5762 ± 0.0136	22.6725 ± 0.0088	$\underline{0.2997 \pm 0.0008}$

Model	CFG	HPSv2.1	Aesthetic	ImgRwd	UniRwd
Pretrained Model Baselines					
SD-XL	–	0.280	5.60	0.76	2.93
SD3.5-L	–	0.288	5.50	0.96	3.25
FLUX.1-Dev	–	0.274	5.71	0.96	3.27
SD3.5-M Fine-Tuning					
Base Model	1.0	0.204	5.13	−0.58	2.02
	3.0	0.284	5.38	0.71	2.96
	4.5	0.279	5.36	0.85	3.03
+ RFT	1.0	0.2759 ± 0.0008	5.3507 ± 0.0111	0.5694 ± 0.0213	2.6590 ± 0.0065
	3.0	0.3041 ± 0.0005	5.4139 ± 0.0093	1.0616 ± 0.0096	3.1370 ± 0.0100
	4.5	0.3054 ± 0.0005	5.4207 ± 0.0061	1.1068 ± 0.0063	3.1520 ± 0.0020
+ FlowDPO	1.0	0.2205 ± 0.0010	5.2202 ± 0.0113	-0.4610 ± 0.0222	2.1075 ± 0.0120
	3.0	0.3014 ± 0.0007	$\underline{5.5590 \pm 0.0089}$	1.0403 ± 0.0089	3.0915 ± 0.0195
	4.5	$\underline{0.3107 \pm 0.0006}$	5.5562 ± 0.0107	1.1543 ± 0.0072	3.1840 ± 0.0190
+ FlowCPO ($\beta = 0.5$, Ours)	1.0	0.2900 ± 0.0003	$\mathbf{5.6392 \pm 0.0100}$	0.9014 ± 0.0139	2.9300 ± 0.0200
	3.0	$\mathbf{0.3114 \pm 0.0004}$	5.5395 ± 0.0065	$\mathbf{1.2480 \pm 0.0071}$	$\mathbf{3.3020 \pm 0.0110}$
	4.5	0.3030 ± 0.0004	5.4569 ± 0.0045	$\underline{1.2174 \pm 0.0116}$	$\underline{3.2775 \pm 0.0095}$

Table 8: Quantitative comparison of fine-tuning SD3.5-M using different methods (RFT, FlowDPO, and FlowCPO) under the *out-of-domain* offline training setting. The fine-tuning data come from the open-source Open Image Preferences v1 Results dataset and were generated by models other than the SD3.5-M reference policy. We evaluate the models with several Classifier-Free Guidance (CFG) scales. Fine-tuned SD3.5-M variants are reported as mean ± standard deviation over 5 runs; pretrained baselines are listed as single evaluations. Within the SD3.5-M fine-tuning group, the best results are highlighted in **bold**, and the second best are underlined.

Model	CFG	GenEval	OCR	PickScore	ClipScore
Pretrained Model Baselines					
SD-XL	–	0.55	0.14	22.42	0.287
SD3.5-L	–	0.71	0.68	22.91	0.289
FLUX.1-Dev	–	0.66	0.59	22.84	0.295
SD3.5-M Fine-Tuning					
Base Model	1.0	0.24	0.12	20.51	0.237
	3.0	0.59	0.47	22.28	0.287
	4.5	0.63	0.59	22.34	0.285
+ RFT	1.0	0.4398 ± 0.0086	0.1723 ± 0.0061	21.5913 ± 0.0194	0.2678 ± 0.0016
	3.0	0.6650 ± 0.0076	0.5310 ± 0.0078	$\underline{22.6427 \pm 0.0108}$	$\underline{0.2974 \pm 0.0008}$
	4.5	0.6845 ± 0.0078	$\mathbf{0.5968 \pm 0.0059}$	$\mathbf{22.6854 \pm 0.0063}$	$\mathbf{0.2991 \pm 0.0007}$
+ FlowDPO	1.0	0.2290 ± 0.0084	0.1158 ± 0.0054	20.8273 ± 0.0155	0.2425 ± 0.0011
	3.0	0.6065 ± 0.0141	0.4789 ± 0.0053	22.5399 ± 0.0212	0.2919 ± 0.0007
	4.5	0.6513 ± 0.0132	0.5406 ± 0.0071	22.6201 ± 0.0165	0.2946 ± 0.0003
+ FlowCPO ($\beta = 1$, Ours)	1.0	0.4670 ± 0.0110	0.2351 ± 0.0027	21.8088 ± 0.0198	0.2762 ± 0.0011
	3.0	$\underline{0.6958 \pm 0.0050}$	0.5561 ± 0.0085	22.4609 ± 0.0188	0.2966 ± 0.0003
	4.5	$\mathbf{0.6995 \pm 0.0060}$	$\underline{0.5912 \pm 0.0069}$	22.3570 ± 0.0111	0.2942 ± 0.0007
+ FlowCPO ($\beta = 0.5$, Ours)	1.0	0.5706 ± 0.0124	0.2360 ± 0.0047	22.0231 ± 0.0146	0.2813 ± 0.0006
	3.0	0.6859 ± 0.0099	0.4919 ± 0.0071	22.3229 ± 0.0064	0.2959 ± 0.0010
	4.5	0.6815 ± 0.0040	0.4912 ± 0.0054	22.1181 ± 0.0128	0.2939 ± 0.0009

Model	CFG	HPSv2.1	Aesthetic	ImgRwd	UniRwd
Pretrained Model Baselines					
SD-XL	–	0.280	5.60	0.76	2.93
SD3.5-L	–	0.288	5.50	0.96	3.25
FLUX.1-Dev	–	0.274	5.71	0.96	3.27
SD3.5-M Fine-Tuning					
Base Model	1.0	0.204	5.13	−0.58	2.02
	3.0	0.284	5.38	0.71	2.96
	4.5	0.279	5.36	0.85	3.03
+ RFT	1.0	0.2611 ± 0.0010	5.4923 ± 0.0129	0.3110 ± 0.0216	2.5551 ± 0.0138
	3.0	$\underline{0.3032 \pm 0.0002}$	5.4780 ± 0.0082	$\underline{1.0514 \pm 0.0134}$	3.1766 ± 0.0202
	4.5	$\mathbf{0.3074 \pm 0.0003}$	5.4733 ± 0.0074	$\mathbf{1.1228 \pm 0.0133}$	$\mathbf{3.2299 \pm 0.0084}$
+ FlowDPO	1.0	0.2264 ± 0.0006	$\mathbf{5.7158 \pm 0.0147}$	-0.3851 ± 0.0285	2.2693 ± 0.0198
	3.0	0.2917 ± 0.0008	$\underline{5.5203 \pm 0.0032}$	0.9063 ± 0.0109	3.0972 ± 0.0170
	4.5	0.2993 ± 0.0003	5.4927 ± 0.0108	0.9934 ± 0.0164	3.1694 ± 0.0117
+ FlowCPO ($\beta = 1$, Ours)	1.0	0.2704 ± 0.0003	5.3546 ± 0.0107	0.5230 ± 0.0140	2.7723 ± 0.0152
	3.0	0.2952 ± 0.0004	5.4140 ± 0.0029	1.0166 ± 0.0075	$\underline{3.2263 \pm 0.0100}$
	4.5	0.2942 ± 0.0002	5.3810 ± 0.0056	1.0159 ± 0.0089	3.2173 ± 0.0184
+ FlowCPO ($\beta = 0.5$, Ours)	1.0	0.2752 ± 0.0005	5.3451 ± 0.0106	0.6294 ± 0.0064	2.8523 ± 0.0200
	3.0	0.2882 ± 0.0003	5.3718 ± 0.0074	0.9967 ± 0.0062	3.1464 ± 0.0089
	4.5	0.2842 ± 0.0005	5.3073 ± 0.0069	0.9452 ± 0.0108	3.1068 ± 0.0117



Figure 4: Qualitative comparison on the OCR benchmark. All samples are generated by models trained with *in-domain* preference data.

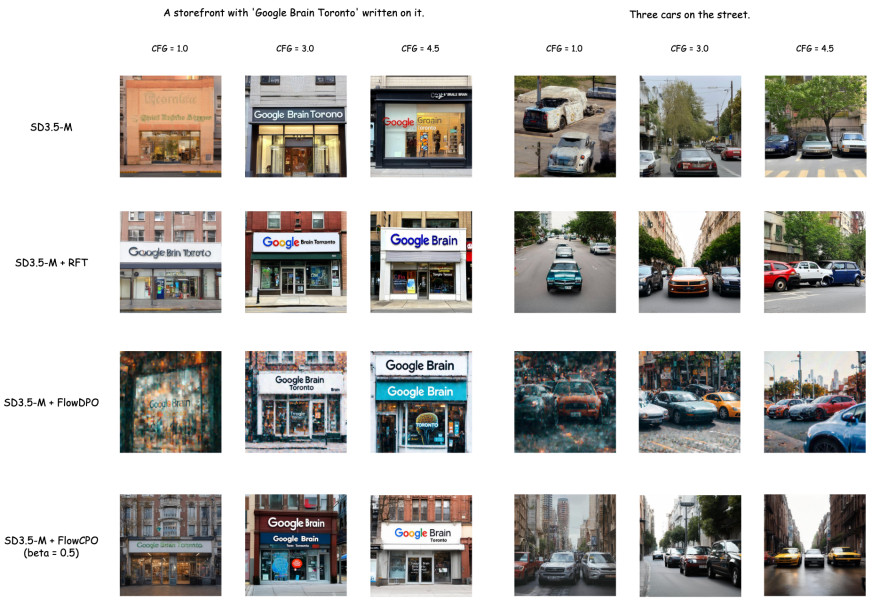


Figure 5: Qualitative comparison on the `DrawBench` benchmark. All samples are generated by models trained with *in-domain* preference data.

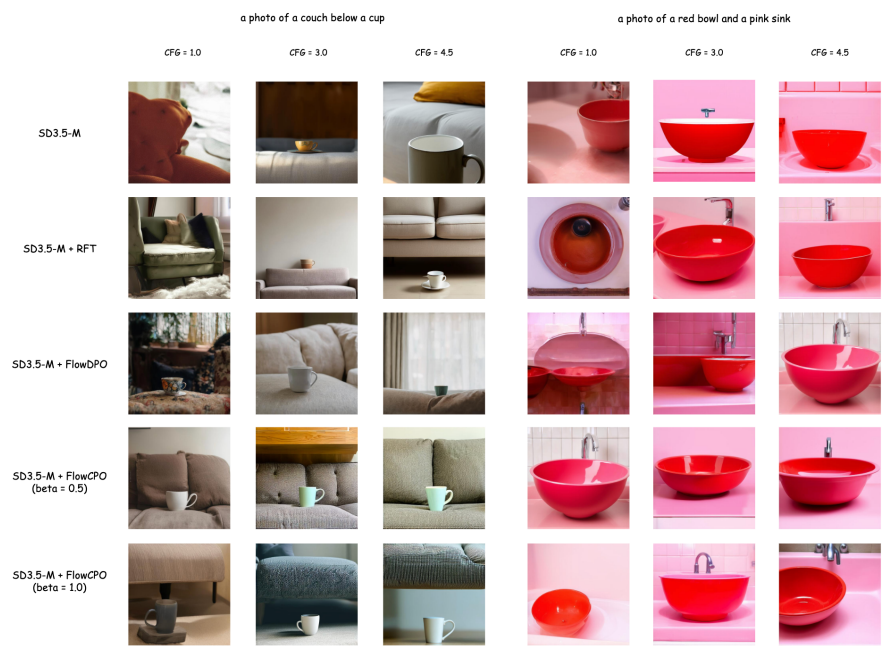


Figure 6: Qualitative comparison on the GenEval benchmark. All samples are generated by models trained with *out-of-domain* preference data.



Figure 7: Qualitative comparison on the OCR benchmark. All samples are generated by models trained with *out-of-domain* preference data.

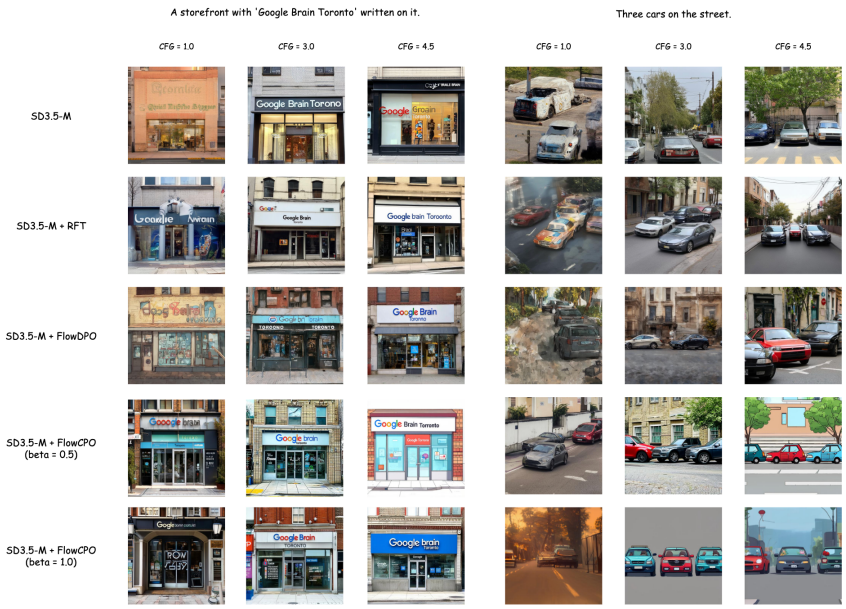


Figure 8: Qualitative comparison on the `DrawBench` benchmark. All samples are generated by models trained with *out-of-domain* preference data.

D.3 Additional Analyses Beyond Main Results

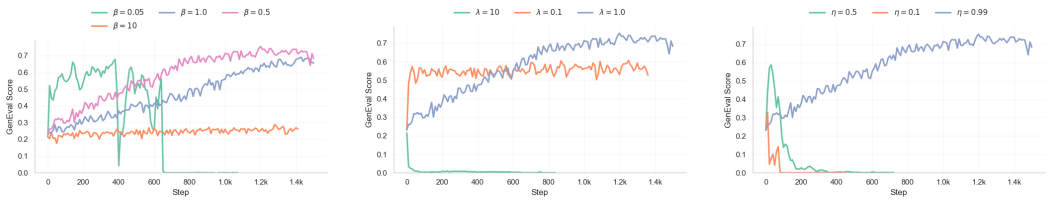
D.3.1 Understanding the Role of Negative Regularization

Tab. 4 shows a clear pattern within the *GenEval-only preference training* setup: how negative samples enter the objective matters at least as much as whether they enter at all. Positive-only regularization provides a stable baseline and remains competitive across all three metric groups, especially under stronger CFG. Repulsive negative regularization is less reliable. Mild repulsion ($\lambda = -0.1$) stays close to positive-only training, but stronger repulsion ($\lambda = -1$) hurts most metrics, and overly aggressive repulsion ($\lambda = -10$) leads to numerical instability and collapsed training. This is consistent with the view that explicitly pushing the model away from dispreferred samples can over-amplify contrastive signals and destabilize optimization.

At moderate weights, *attractive* negative regularization gives the best trade-off in this ablation between alignment strength and training stability. The configuration with $\beta = 0.5$ and $\lambda = 1$ achieves the highest GenEval score (0.84), whereas repulsion with $\lambda = -1$ degrades all reported metrics. The configuration with $\beta = 1$ and $\lambda = 0.1$ also attains the best OCR (0.59), ImgReward (1.14), and UniReward (3.23). Because all models in this table are trained with GenEval-only supervision, these gains on typography and general-preference metrics should be interpreted as cross-metric transfer rather than direct optimization. Both signs diverge at magnitude 10, so the evidence supports moderate attractive matching rather than a blanket stability claim.

D.3.2 Hyperparameter Sensitivity and Ablation

To study the hyperparameter sensitivity of the proposed objective function (equation 11), we conduct ablations on the GenEval benchmark and track performance across training steps.



(a) Effect of β (b) Effect of λ (c) Effect of η

Figure 9: **Hyperparameter sensitivity within the GenEval-only setup.** We monitor the GenEval score during training to analyze the impact of (a) the flow interpolation coefficient β, (b) the negative loss weight λ, and (c) the EMA parameter η. The figure is intended to show relative stability trends for the tested values.

Flow Interpolation Coefficient (β). β controls how far the target flows move away from the reference prior. As shown in Fig. 9a, moderate values ($\beta \in [0.5, 1.0]$) perform best in this sweep on GenEval. In particular, $\beta = 0.5$ achieves the highest peak score.

Negative Regularization Weight (λ). Balancing positive alignment and negative regularization is important. Fig. 9b shows that a balanced choice ($\lambda = 1.0$) yields the most robust long-term performance among the tested values on GenEval.

Reference Prior Stability (η). FlowCPO relies on a stable reference prior inside the flow-matching objective. Fig. 9c shows that updating the reference model v_{old} with a high-EMA decay rate ($\eta = 0.99$) yields the most stable behavior in this sweep.

D.4 Per-Prompt Diversity Evaluation

The primary empirical claim in RQ2 is that FlowCPO provides stable offline preference optimization with a forward-KL objective in the evaluated regime, not that it universally produces more diverse samples than FlowDPO. To evaluate sample diversity directly, we compute the Vendi Score (Friedman & Dieng, 2023) separately over eight samples generated for each prompt

and report the mean per-prompt score. We evaluate two classifier-free guidance (CFG) scales, 1.0 and 4.5. For each target column in Tab. 9, the quality score and Vendi Score are obtained from the corresponding specialist: GenEval-only fine-tuning for `GenEval`, OCR-only fine-tuning for `OCR`, and multi-reward fine-tuning for `PickScore, CLIPScore, HPSv2.1`. The main entry in each cell is the quality score, while the parenthesized entry is the Vendi Score. Higher is better for both.

Table 9: Target quality and per-prompt diversity on `SD3.5-M`. Vendi Score is computed from eight samples per prompt and reported in parentheses. Each metric column evaluates the specialist fine-tuned for that target. Bold independently marks the highest quality score and the highest Vendi Score in each column.

Method	CFG	GenEval ↑ (Vendi ↑)	OCR ↑ (Vendi ↑)	PickScore ↑ (Vendi ↑)
SD3.5-M	1.0	0.24 (**2.1573**)	0.12 (2.3073)	20.51 (**2.2824**)
	4.5	0.63 (1.6540)	0.59 (1.7778)	22.34 (1.7742)
RFT	1.0	0.59 (1.8210)	0.35 (2.2417)	21.91 (1.9740)
	4.5	0.75 (1.4547)	0.72 (1.4984)	22.57 (1.5544)
FlowDPO	1.0	0.59 (1.8926)	0.51 (**2.5117**)	20.72 (1.8671)
	4.5	0.81 (1.5594)	0.75 (1.6310)	**22.89** (1.5966)
FLOWCPO ($\beta = 0.5$)	1.0	0.76 (1.7385)	0.83 (2.0578)	22.48 (1.8936)
	4.5	**0.82** (1.4773)	**0.86** (1.4955)	22.69 (1.5251)

Results. The results show a consistent quality–diversity trade-off: stronger CFG improves target quality while reducing per-prompt diversity. Across all comparisons in Tab. 9, increasing CFG from 1.0 to 4.5 raises the target quality score and lowers the corresponding Vendi Score. At CFG 1.0, FLOWCPO achieves the highest quality among the compared methods for all three targets (0.76 GenEval, 0.83 OCR, and 22.48 PickScore), while retaining higher diversity than its own CFG-4.5 outputs. In particular, lowering CFG from 4.5 to 1.0 increases the Vendi Score of FLOWCPO by 0.2612, 0.5623, and 0.3685 on the three targets, respectively, with corresponding quality decreases of 0.06, 0.03, and 0.21. However, FLOWCPO does not maximize raw Vendi Score: the pretrained model is most diverse in the GenEval and PickScore columns.

E BROADER IMPACTS AND RELEASE CONSIDERATIONS

Potential positive impacts. By turning preference alignment into a strictly offline optimization problem, our method can reduce the need for repeated online rollouts during alignment and make experimentation on continuous generative models more compute-efficient. Better alignment on structure-sensitive tasks such as compositional generation and text rendering can also improve the controllability and practical usefulness of text-to-image systems in benign applications such as design ideation, educational content creation, and accessibility-related graphics.

Potential negative impacts. The same improvement in preference alignment can increase the capability of image generators to produce more convincing synthetic content, including misleading text-in-image content or other deceptive media. In addition, the method can inherit biases from offline preference data and from the automatic reward signals used to construct winner/loser pairs, and the out-of-domain results in Sec. 5 together with the limitations discussed in Sec. 6 indicate that behavior may degrade when the offline data distribution differs from the reference model.

Release considerations. We do not release new model checkpoints, a new dataset, or a code package with this submission; the paper describes the training objective and experimental protocol only. The experiments rely on existing pretrained models, public benchmarks, and a public OOD preference dataset under their original access conditions, and any future release of code or checkpoints should preserve the upstream licenses, terms of use, and usage restrictions of those assets.

F Additional Background on Preference Optimization

F.1 RLHF and Direct Preference Optimization (DPO)

Reinforcement Learning from Human Feedback (RLHF) (Ouyang et al., 2022) typically aligns a generative policy $\pi_\theta(\mathbf{x}_0|c)$ by maximizing expected reward while regularizing deviation from a reference policy, where $\mathbf{x}_0$ is the generated content and c is the condition. We use $r(\mathbf{x}_0, c)$ to denote the reward function, $\mathbf{x}_0^w$ and $\mathbf{x}_0^l$ to denote the preferred and dispreferred contents, respectively. DPO (Rafailov et al., 2023; Sun et al., 2025a; Tang et al., 2024) bypasses the explicit reward modeling step by directly optimizing the policy using a closed-form solution to the KL-constrained reward maximization problem. Given a dataset $\mathcal{D} = \{(c, \mathbf{x}_0^w, \mathbf{x}_0^l)\}$ of preferred ($\mathbf{x}_0^w$) and dispreferred ($\mathbf{x}_0^l$) contents, DPO minimizes the following negative log-likelihood loss:

$$\mathcal{L}_{\text{DPO}}(\theta) = -\mathbb{E}_{(c,\mathbf{x}_0^w,\mathbf{x}_0^l)\sim\mathcal{D}}\left[\log\sigma\left(\beta\log\frac{\pi_\theta(\mathbf{x}_0^w|c)}{\pi_{\text{ref}}(\mathbf{x}_0^w|c)} - \beta\log\frac{\pi_\theta(\mathbf{x}_0^l|c)}{\pi_{\text{ref}}(\mathbf{x}_0^l|c)}\right)\right], \tag{93}$$

where π_{ref} is the frozen reference policy and β is a temperature parameter controlling the strength of the KL constraint.

F.2 Preference Optimization in Continuous-Time Models

Applying equation 93 directly to the diffusion and flow-matching models introduced above is non-trivial. Calculating the exact log-likelihood $\log\pi_\theta(\mathbf{x}_0|c)$ requires solving the probability flow ODE, which is computationally prohibitive during training (Zheng et al., 2025). DiffusionDPO (Wallace et al., 2024) addresses this intractability by optimizing the Evidence Lower Bound (ELBO) as a proxy for the likelihood. The loss function is reformulated using the denoising error at timestep t:

$$\mathcal{L}_{\text{Diff-DPO}}(\theta) = -\mathbb{E}_{c,\mathbf{x}_0^w,\mathbf{x}_0^l,t,\epsilon}\Big[\log\sigma\Big(\beta\rho_t\Big(\underbrace{\left(\|\epsilon_\theta(\mathbf{x}_t^l) - \epsilon\|^2 - \|\epsilon_{\text{ref}}(\mathbf{x}_t^l) - \epsilon\|^2\right)}_{\text{Advantage of Loser}} - \underbrace{\left(\|\epsilon_\theta(\mathbf{x}_t^w) - \epsilon\|^2 - \|\epsilon_{\text{ref}}(\mathbf{x}_t^w) - \epsilon\|^2\right)}_{\text{Advantage of Winner}}\Big)\Big)\Big], \tag{94}$$

where $\mathbf{x}_t$ is the linear interpolation of $\mathbf{x}_0$ and noise ϵ, and ρ_t is a weighting function. FlowDPO (Liu et al., 2025b) extends this paradigm to flow matching. By leveraging the connection between the score function and the vector field, FlowDPO replaces the noise prediction error with the flow matching loss, enabling efficient gradient-based preference optimization directly on the velocity field without ODE integration.