THE MISSING MARGINAL: MARGINAL CLOSURE FOR CLASSIFIER-FREE GUIDANCE IN DIFFUSION POST-TRAINING

Anonymous authors
Paper under double-blind review

ABSTRACT

Classifier-Free Guidance (CFG) combines conditional and unconditional predictions to improve diffusion sampling, yet we observe that it can degrade performance after online reinforcement learning (RL). Examining existing post-training objectives, we find that optimizing either the conditional branch alone or a single fixed-scale CFG output leaves the unconditional branch underdetermined. We term this identification gap the missing-marginal problem. Under shared-target squared-error regression, the unconditional optimum equals the posterior average of the conditional optimum given the noised state, and is determined by the regression target and sampling distribution. We formalize this relation between learned predictions as marginal closure and analyze how closure errors affect CFG. Based on this relation, we introduce Marginal Closure Regularization (MCReg) to train both predictions on a shared target. We derive reward-dependent targets for DiffusionNFT. In on-policy diffusion distillation (OPD), we identify the student's unconditional counterpart and train both branches on conditional teacher targets. Experiments across online RL and multitask OPD show that MCReg improves most evaluated generation metrics under CFG.

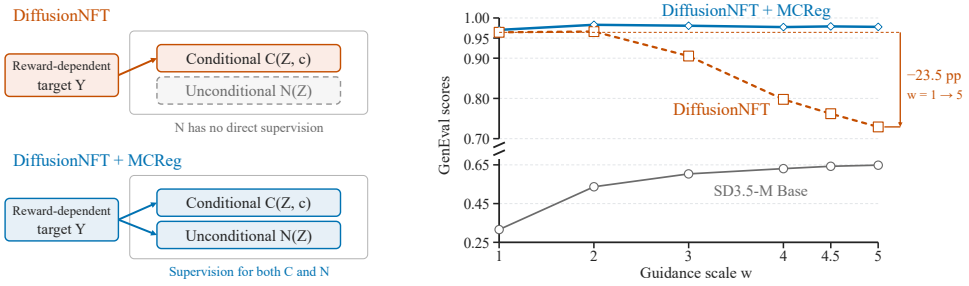

Figure 1: **Shared-target supervision preserves CFG performance.** *Left:* MCReg extends DiffusionNFT supervision to the unconditional prediction using the same target. *Right:* With MCReg, GenEval scores remain high as the guidance scale increases.

1 INTRODUCTION

Online reinforcement learning (RL) improves diffusion and flow models through task-specific reward optimization during post-training (Black et al., 2024; Fan et al., 2023; Liu et al., 2025a; Zheng et al., 2026). Existing formulations include policy gradient objectives over denoising trajectories (Black et al., 2024; Liu et al., 2025a) and score or flow matching objectives on the forward process (Zheng et al., 2026; Xue et al., 2026). However, the relationship between conditional and unconditional predictions under Classifier-Free Guidance (CFG) (Ho & Salimans, 2022) after post-training remains insufficiently characterized.

In particular, several online diffusion RL methods have adopted CFG-free training, with reward-driven updates implicitly integrating reinforcement guidance into the learned policy (Zheng et al.,

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.

2026; Fu et al., 2026). While CFG-free training improves post-training efficiency, we observe substantial performance degradation for DiffusionNFT under CFG-guided sampling, a standard way to improve fidelity (Ho & Salimans, 2022; Nichol et al., 2022). Figure 1 demonstrates this failure mode on GenEval (Ghosh et al., 2023) with SD3.5-M (Stability AI, 2024). At guidance scale 5, its GenEval score is **23.5** percentage points lower than at scale 1. This discrepancy motivates an analysis of whether online RL objectives identify the unconditional prediction corresponding to the updated conditional prediction.

During pretraining, conditional and unconditional predictions are learned jointly through conditioning dropout, which randomly drops the conditioning input while retaining the same regression target (Ho & Salimans, 2022). The same procedure can be used in supervised fine-tuning, providing direct supervision for the unconditional prediction (Zhong et al., 2025). By contrast, online RL updates are based on samples drawn from the current policy and their associated rewards (Liu et al., 2025a; Zheng et al., 2026). For a CFG-free objective such as DiffusionNFT, parameter sharing can alter the unconditional prediction, but the objective directly supervises only the conditional prediction (Zheng et al., 2026). Optimizing only the guided prediction at a fixed guidance scale does not identify the two component predictions separately either (Li et al., 2026a). We refer to this identification gap as *the missing-marginal problem*.

In supervised fine-tuning, the ideal unconditional prediction corresponds to the marginal distribution of the target domain (Zhong et al., 2025). Under population squared-error regression with a shared target, the unconditional optimum equals the posterior average of the conditional optimum given the noised state. With this pair of optima, CFG scales the prompt-dependent component while preserving the shared marginal component. In diffusion post-training, the regression target and sampling distribution together determine this unconditional counterpart. We formalize the corresponding relation between learned predictions as ***marginal closure*** and introduce Marginal Closure Regularization (MCReg) to promote it.

For DiffusionNFT, we rewrite its conditional objective in squared-error form to recover a target for training both predictions at the same state. The unconditional predictor implicitly learns the posterior average by regressing on this target. In on-policy diffusion distillation (OPD) (Li et al., 2026b), separately matching the teacher's conditional and unconditional predictions need not establish marginal closure. Under a fixed student sampling distribution, the same shared-target formulation identifies the student's unconditional optimum as the posterior average of conditional teacher targets. We therefore train both student predictions with a shared conditional teacher target at student-visited states, promoting marginal closure without unconditional teacher queries.

Our contributions are as follows:

- We identify the missing-marginal problem in online RL and characterize the unconditional optimum for a shared regression target. We formalize marginal closure between learned predictions, quantify how closure errors affect CFG, and derive a risk decomposition explaining how shared-target regression promotes marginal closure.
- We introduce MCReg with shared-target supervision for DiffusionNFT and OPD, deriving reward-dependent targets from the former and reusing conditional teacher targets in the latter without unconditional teacher queries. For Flow-GRPO, we add an advantage-weighted unconditional transition fit alongside its native clipped objective.
- MCReg improves guided GenEval and OCR with DiffusionNFT and Flow-GRPO-Fast, and multi-reward scores with DiffusionNFT on SD3.5-Medium and FLUX.2 Klein. It also improves guided compositional generation and preference metrics with both DiffusionOPD (Li et al., 2026b) and DanceOPD (Zhou et al., 2026b).

2 Background

2.1 Post-Training in Diffusion and Flow Models

We use a velocity parameterization for diffusion and flow models (Ho et al., 2020; Lipman et al., 2023). For pretraining, independently draw $(x_0, c) \sim p_{\text{data}}$, $\epsilon \sim \mathcal{N}(0, I)$, and $t \sim \mathcal{U}[0, 1]$. A noise schedule (α_t, σ_t) defines the noised sample and target velocity:

$$x_t = \alpha_t x_0 + \sigma_t \epsilon, \qquad u_t = \dot{\alpha}_t x_0 + \dot{\sigma}_t \epsilon, \tag{1}$$

where overdots denote time derivatives. The velocity predictor v_θ minimizes the flow-matching loss

$$\mathcal{L}_{\mathrm{FM}}(\theta) = \mathbb{E}_{t,(x_0,c),\epsilon}\left[\|v_\theta(x_t,t,c) - u_t\|_2^2\right]. \tag{2}$$

For rectified flow, $\alpha_t = 1 - t$ and $\sigma_t = t$, giving $u_t = \epsilon - x_0$ (Liu et al., 2023; Esser et al., 2024). ODE sampling integrates $\mathrm{d}x_t/\mathrm{d}t = v_\theta(x_t, t, c)$ from pure noise at $t = 1$ to clean data at $t = 0$.

Online RL uses rewards evaluated on samples from the current rollout policy for prompts $c \sim q$, where q is the prompt distribution (Liu et al., 2025a). In DiffusionNFT, rollout samples are re-noised via Eq. (1), and rewards determine the flow-matching objective (Zheng et al., 2026). In OPD, the student matches teacher predictions along its own denoising trajectories (Li et al., 2026b).

2.2 Classifier-Free Guidance

Classifier-Free Guidance (CFG) combines conditional and unconditional predictions from the same model (Ho & Salimans, 2022). With $Z = (x_t, t)$, we denote these predictions by $C(Z, c) = v_\theta(x_t, t, c)$ and $N(Z) = v_\theta(x_t, t, \emptyset)$, respectively, where $\emptyset$ denotes the empty condition. At guidance scale w, CFG forms

$$G_w(Z, c) = N(Z) + w\big(C(Z, c) - N(Z)\big). \tag{3}$$

We use the convention $w = 1$ for the conditional prediction, $G_1 = C$, and $w > 1$ for guidance.

During pretraining, conditioning dropout jointly trains the two predictions by replacing c with $\emptyset$ with probability $\delta \in (0, 1)$, independently of the training sample, while retaining the same target u_t.

3 Marginal Closure Regularization

3.1 Marginal Closure

Conditional and unconditional optima. We first characterize the unconditional prediction associated with a post-training regression target. Let Y denote a square-integrable target in the model output space. For DiffusionNFT, we recover Y from the conditional policy loss; for OPD, it is a conditional teacher prediction. At a given update, the prompt distribution, state sampling, and target construction define the joint law $(Z, c, Y) \sim P$, which we hold fixed for the population analysis. We compare unweighted squared-error regression to the same Y, changing only whether the prompt is provided. The unrestricted population optima are

$$C^\star(Z, c) = \mathbb{E}_P[Y \mid Z, c], \qquad N^\star(Z) = \mathbb{E}_P[Y \mid Z] = \mathbb{E}_P[C^\star(Z, c) \mid Z]. \tag{4}$$

The last equality follows from the law of iterated expectations. We write $(\Pi f)(Z) = \mathbb{E}_P[f(Z, c) \mid Z]$ for posterior averaging over prompts at fixed Z. This operator uses $P(c \mid Z)$, whereas q specifies the prompt sampling distribution.

Marginal closure. Learned predictions need not satisfy the population relation in Eq. (4). For a square-integrable pair (C, N), define the closure residual and absolute closure risk by

$$r(Z) = N(Z) - (\Pi C)(Z), \qquad R_{\mathrm{abs}}(C, N; P) = \mathbb{E}_P[\|r(Z)\|_2^2]. \tag{5}$$

Definition 1. A square-integrable pair of learned predictions (C, N) satisfies *marginal closure* under P if the unconditional prediction equals the posterior average of the conditional prediction, $N = \Pi C$ almost surely. Equivalently, $R_{\mathrm{abs}}(C, N; P) = 0$. Closure concerns the relation between the learned predictions; their accuracy is measured relative to $(C^\star, N^\star)$.

Effect on CFG. The post-training target and sampling distribution determine the unconditional optimum $N^\star$. To distinguish marginal shift from prompt dependence, we use the frozen pretrained unconditional prediction $s_0(Z) = N_0(Z)$ as a reference, assumed square-integrable under P. Define the marginal shift relative to this reference as $d(Z) = N^\star(Z) - s_0(Z)$ and the prompt-dependent component as $l(Z, c) = C^\star(Z, c) - N^\star(Z)$. The target pair decomposes as

$$C^\star = s_0 + d + l, \qquad N^\star = s_0 + d. \tag{6}$$

Under Eq. (3), the target pair defines the oracle CFG prediction $G_w^\star$, and the learned pair has error

$$G_w^\star = s_0 + d + wl, \qquad G_w - G_w^\star = w(C - C^\star) + (1 - w)(N - N^\star). \tag{7}$$

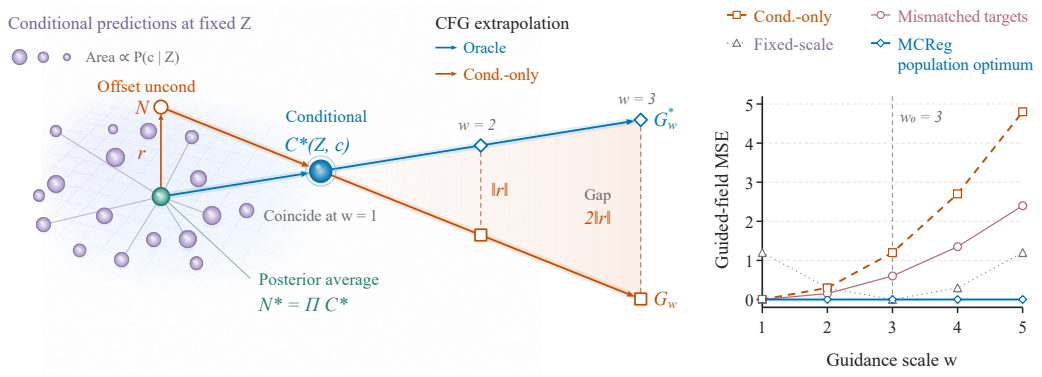

Figure 2: **Marginal closure and CFG error.** *Left:* With $C = C^\star$, an unconditional offset $r = N - \Pi C^\star$ gives CFG error $(1-w)r$; marker areas show posterior prompt probabilities. *Right:* In an analytic Gaussian mixture, the MCReg population optimum matches the oracle $G_w^\star$ across scales; the constructed fixed-scale pair is exact only at $w_0 = 3$.

Since $\Pi l = 0$, oracle guidance scales the prompt-dependent component while preserving the posterior mean $\Pi G_w^\star = N^\star$.

For an exact conditional predictor $C = C^\star$, the closure residual is $r = N - N^\star$. Substitution into CFG gives $G_w = s_0 + d + wl + (1-w)r$. The left panel of Figure 2 shows CFG extrapolating from $N^\star$ or N through the same conditional prediction $C^\star$. The learned and oracle CFG predictions coincide at $w = 1$ but separate by $(w-1)\|r(Z)\|_2$ at each state when $w > 1$. Using the frozen pretrained unconditional guide (Zhong et al., 2025) sets $r = -d$ and $G_w = s_0 + w(d + l)$, scaling the marginal shift together with the prompt-dependent component. Appendix B.3 derives the corresponding decomposition for general learned predictions.

Conditional-only regression to Y provides no direct supervision for N to match $N^\star$. Matching the oracle CFG prediction at a fixed scale $w_0 > 1$ also does not identify the branches separately (Li et al., 2026a): it requires only $w_0(C - C^\star) + (1 - w_0)(N - N^\star) = 0$. Nonzero branch errors can therefore cancel at w_0 despite violating marginal closure, while changing the guide at other scales, as shown in the right panel of Figure 2.

If the branches instead regress to different targets Y_C and Y_N, their optima satisfy $N^\star_{Y_N} - \Pi C^\star_{Y_C} = \mathbb{E}_P[Y_N - Y_C \mid Z]$. The right panel of Figure 2 also illustrates this target mismatch. We therefore supervise both predictions with a shared target.

3.2 MCREG

At update k in the shared-target setting, MCReg evaluates c and $\emptyset$ at the same state and regresses both predictions toward the same realization of Y_k, constructed with the original prompt. The target is detached for both regressions. For the fixed joint law P_k of (Z, c, Y_k), with posterior averaging operator Π_k, the population risk is

$$R_{\rho,k}(C, N) = \frac{1}{1+\rho}\,\mathbb{E}_{P_k}\left[\|C(Z,c) - Y_k\|_2^2 + \rho\|N(Z) - Y_k\|_2^2\right], \tag{8}$$

where $\rho > 0$ weights the unconditional regression. Since N receives only Z, its optimal prediction averages Y_k over prompts compatible with that state. Training requires no explicit evaluation of $P_k(c \mid Z)$ or Π_k.

Theorem 1 (Joint target regression)**.** *Fix P_k with square-integrable Y_k and $\rho > 0$. The unique unrestricted minimizer of $R_{\rho,k}$ is $(C_k^\star, N_k^\star)$ from Eq. (4), up to P_k-almost-sure equality, and satisfies marginal closure. For any square-integrable prediction pair, let $e_C = C - C_k^\star$ and $e_N = N - N_k^\star$. Its excess risk is*

$$\Delta R_{\rho,k} := R_{\rho,k}(C, N) - R_{\rho,k}(C_k^\star, N_k^\star) = \frac{\mathbb{E}_{P_k}\|e_C\|_2^2 + \rho\mathbb{E}_{P_k}\|e_N\|_2^2}{1+\rho}. \tag{9}$$

This risk decomposes into prompt-dependent, marginal, and closure errors:

$$\Delta R_{\rho,k} = \frac{\mathbb{E}_{P_k}\|e_C - \Pi_k e_C\|_2^2}{1+\rho} + \mathbb{E}_{P_k}\left\|\frac{\Pi_k e_C + \rho e_N}{1+\rho}\right\|_2^2 + \frac{\rho}{(1+\rho)^2}\,R_{\mathrm{abs}}(C, N; P_k). \tag{10}$$

Consequently, the sharp closure bound is

$$R_{\mathrm{abs}}(C, N; P_k) \le \frac{(1+\rho)^2}{\rho}\,\Delta R_{\rho,k}. \tag{11}$$

The first term measures prompt-dependent prediction error. The second measures the weighted marginal prediction $(\Pi_k C+\rho N)/(1+\rho)$ relative to $N_k^\star$. The third measures marginal inconsistency. Joint regression penalizes all three, and every $\rho > 0$ has the same unrestricted optimum. In a restricted shared model, ρ changes their relative weights. Attaining the target pair requires joint realizability and population-risk minimization.

We next specify P_k and Y_k for DiffusionNFT and OPD.

3.3 Application to Online RL

For DiffusionNFT, we recover a shared target from its reward-dependent conditional loss on renoised rollouts. With the rollout model and adaptive normalizers fixed, this loss is quadratic in the current conditional prediction.

Target construction. DiffusionNFT (Zheng et al., 2026) uses the implicit positive and negative predictions $C^\pm = C_{\mathrm{old}} \pm \gamma(C - C_{\mathrm{old}})$, where C_{old} is the fixed conditional rollout-model prediction and $\gamma > 0$ is the implicit-policy coefficient. Rewards determine an optimality probability $p_{\mathrm{opt}} \in [0, 1]$ by rescaling advantages clipped to $[-A_{\max}, A_{\max}]$, with $A_{\max} > 0$. At a fixed model snapshot, we compute and detach adaptive loss normalizers $d_\pm > 0$, then set $w_+ = p_{\mathrm{opt}}/d_+$ and $w_- = (1 - p_{\mathrm{opt}})/d_-$. The per-sample policy loss in velocity space is

$$\ell_{\mathrm{policy}}(C) = \frac{A_{\max} t^2}{\gamma}\left[w_+\|C^+ - u_t\|_2^2 + w_-\|C^- - u_t\|_2^2\right]. \tag{12}$$

Here, $u_t = \epsilon - x_0$ is the rectified-flow target in Eq. (1). The factor t^2 follows from $x_t - tC^\pm - x_0 = t(u_t - C^\pm)$. Completing the square gives $\ell_{\mathrm{policy}}(C) = h\|C - T_{\mathrm{policy}}\|_2^2$ up to a C-independent term, where $h = A_{\max} t^2 \gamma(w_+ + w_-)$ and

$$T_{\mathrm{policy}} = C_{\mathrm{old}} + \frac{w_+ - w_-}{\gamma(w_+ + w_-)}(u_t - C_{\mathrm{old}}). \tag{13}$$

Reward weights set the signed displacement from C_{old} along $u_t - C_{\mathrm{old}}$. Adding the reference penalty $\beta_{\mathrm{ref}}\|C - C_0\|_2^2$, with frozen pretrained conditional prediction C_0 and $\beta_{\mathrm{ref}} \ge 0$, gives the shared target

$$Y_k = \frac{hT_{\mathrm{policy}} + \beta_{\mathrm{ref}} C_0}{W}, \qquad W = h + \beta_{\mathrm{ref}} > 0. \tag{14}$$

With target-side quantities detached, $W\|C - Y_k\|_2^2$ preserves the original per-sample conditional gradient. For a local minibatch of B samples, MCReg reuses Y_k for both predictions and replaces W_i with the detached scale $\overline{W}_B = B^{-1}\sum_i W_i$. The joint loss for n-dimensional predictions is

$$\widehat{L}_{\rho,k}^{\mathrm{NFT}} = \frac{\overline{W}_B}{Bn(1+\rho)}\sum_{i=1}^{B}\left[\|C(Z_i, c_i) - Y_i\|_2^2 + \rho\|N(Z_i) - Y_i\|_2^2\right]. \tag{15}$$

This equalizes within-batch coefficients and changes the conditional gradient. The common scale induces the same weighted regression law for both branches, so Theorem 1 applies under that law. Appendix B.2 gives its construction and moment assumptions.

Application to Flow-GRPO. We also extend unconditional supervision to Flow-GRPO (Liu et al., 2025a). We retain its native clipped conditional loss L_{cond} and add an unconditional transition fit L_N, optimizing $(L_{\mathrm{cond}} + \rho L_N)/(1+\rho)$. The unconditional transition mean is affine in model velocity and fitted to next states from conditional rollouts using detached exponentials of clipped advantages normalized within prompt groups. Expanding these weights around zero weighting strength gives a first-order correction with an advantage-weighted gradient, locally analogous to an unclipped policy-gradient update at the frozen rollout policy (Xue et al., 2026). The clipped objective lies outside Theorem 1's shared-target setting. Appendix C details the replay rules and derivation.

3.4 Application to On-Policy Distillation

In OPD, frozen teachers supply targets at student-visited states (Li et al., 2026b).

Student regression optima. At update k, sample a task M and prompt c, then collect states $Z = (x_t, t)$ along the student's trajectory. M selects teacher T_M but is not a student input. Set $Y_k = \mathrm{sg}[C_{T_M}(Z, c)]$, with stop-gradient sg, and let P_k be the joint law of (M, Z, c, Y_k). For square-integrable Y_k, the conditional optimum averages teacher identities given (Z, c); the unconditional optimum also averages prompts under $P_k(M, c \mid Z)$. Eq. (4) gives

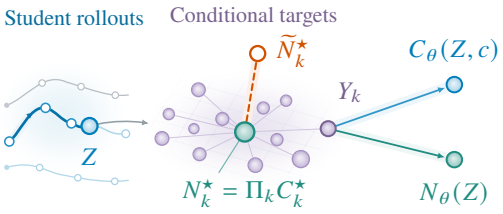

Figure 3: **Shared-target supervision for OPD.** Both predictions fit the conditional teacher target Y_k; teacher-unconditional matching yields $\widetilde{N}_k^\star$. Purple areas indicate posterior task and prompt probabilities.

$$C_k^\star = \mathbb{E}_{P_k}[C_{T_M}(Z, c) \mid Z, c], \qquad N_k^\star = \mathbb{E}_{P_k}[C_{T_M}(Z, c) \mid Z] = \Pi_k C_k^\star. \tag{16}$$

Choice of unconditional target. Regression to square-integrable teacher unconditional targets gives $\widetilde{N}_k^\star = \mathbb{E}_{P_k}[N_{T_M}(Z) \mid Z]$, with mismatch

$$\widetilde{N}_k^\star - N_k^\star = \mathbb{E}_{P_k}[N_{T_M}(Z) - C_{T_M}(Z, c) \mid Z]. \tag{17}$$

Figure 3 shows why Eq. (17) can remain nonzero despite closure under each teacher's own law. Such a teacher's unconditional prediction averages conditional predictions over its prompt posterior; $N_k^\star$ instead averages over the student-rollout posterior $P_k(M, c \mid Z)$, including the task mixture.

We use the conditional teacher target Y_k for both student predictions, with a positive state weight $a(Z)$ shared across the two regression terms:

$$L_{\rho,k}^{\mathrm{OPD}} = \frac{1}{1+\rho}\mathbb{E}_{P_k}\left[a(Z)\left(\|C(Z, c) - Y_k\|_2^2 + \rho\|N(Z) - Y_k\|_2^2\right)\right]. \tag{18}$$

Direct velocity matching in DanceOPD (Zhou et al., 2026b) uses $a(Z) = 1$. For DiffusionOPD (Li et al., 2026b), matching an Euler transition of step $\Delta\sigma_t$ gives $a(Z) = (\Delta\sigma_t)^2/2$.

On supervised timesteps, $a(Z) > 0$ depends only on Z, so reweighting states leaves $P_k(M, c \mid Z)$ unchanged and preserves the optima and closure relation in Eq. (16). After normalization, Eq. (18) has the form of Eq. (8) under the reweighted law, and Theorem 1 applies under the moment conditions in Appendix B.5. Both branches reuse the conditional teacher query without querying the teacher's unconditional branch; states, targets, and weights stay fixed during differentiation.

4 Experiments

4.1 Experimental Setup

Tasks and baselines. We evaluate MCReg in online RL and on-policy diffusion distillation (OPD). For single-reward RL, we apply MCReg to DiffusionNFT (Zheng et al., 2026) and Flow-GRPO-Fast (Liu et al., 2025a;b). For each RL method, we fine-tune separate SD3.5-Medium (Stability AI, 2024) models for GenEval (Ghosh et al., 2023) and OCR, using the corresponding training and test splits from DiffusionNFT. For multi-reward RL, we apply MCReg to DiffusionNFT on SD3.5-Medium and FLUX.2 [klein] 9B Base (Black Forest Labs, 2026). We jointly optimize PickScore, CLIPScore, and HPSv2.1 (Kirstain et al., 2023; Hessel et al., 2021; Wu et al., 2023) on Pick-a-Pic training prompts and report these metrics on the Pick-a-Pic test split and DrawBench (Saharia et al., 2022). On DrawBench, we additionally report ImageReward (Xu et al., 2023), UnifiedReward (Wang et al., 2025), and DeQA-Score (You et al., 2025). For multitask OPD, we apply MCReg within DiffusionOPD (Li et al., 2026b) and DanceOPD (Zhou et al., 2026b), comparing it with the baseline and Positive-Direction Matching (PDM) (Li et al., 2026a) in each framework. All students are evaluated on GenEval, OCR, and the six DrawBench metrics.

Table 1: **Single-reward optimization on SD3.5-Medium.**† Flow-GRPO scores from Choi et al. (2026); AWM and DGPO from Xue et al. (2026); Luo et al. (2026). Bold marks blockwise best values.

Method	GenEval								OCR
	Single Obj.	Two Obj.	Counting	Colors	Position	Color Attr.	**Overall**	$R_{\text{abs}} \downarrow$	
SD3.5-M	0.763	0.283	0.186	0.527	0.048	0.090	0.316	—	0.116
SD3.5-M *with CFG*	0.959	0.720	0.481	0.782	0.195	0.485	0.604	—	0.487
Flow-GRPO†	1.000	0.990	0.950	0.920	0.990	0.860	0.950	—	0.920
AWM†	1.000	1.000	0.990	0.970	0.990	0.940	0.985	—	0.960
DGPO†	1.000	0.990	0.970	0.950	0.990	0.910	0.970	—	0.960
DiffusionNFT	**1.000**	**0.997**	0.969	0.934	0.983	0.903	0.964	0.095	0.969
DiffusionNFT *with CFG*	0.994	0.942	0.972	0.912	0.932	0.679	0.905	0.095	0.967
+ MCReg (Ours) *with CFG*	**1.000**	**0.997**	**0.991**	**0.968**	**0.985**	**0.943**	**0.981**	**0.008**	**0.976**
Flow-GRPO-Fast	0.991	0.990	0.891	0.883	0.930	0.826	0.918	0.112	0.874
Flow-GRPO-Fast *with CFG*	**1.000**	**1.000**	0.882	**0.920**	0.948	**0.881**	0.938	0.112	0.893
+ MCReg (Ours) *with CFG*	**1.000**	0.995	**0.963**	0.915	**0.960**	0.836	**0.945**	**0.041**	**0.912**

Training and evaluation. We fine-tune using LoRA (Hu et al., 2022) with $r = 32$ and $\alpha = 64$, and sample training rollouts at $w = 1$. Within each setting, we compare our trained variants at the same update count. For our evaluations in the main tables, we generate images with 40 sampling steps at 512×512, using $w = 3$ for rows marked *with CFG* and $w = 1$ otherwise. We compute R_{abs} at the pure-noise endpoint for each reported checkpoint, independently of inference CFG. Appendix D.1 provides further training and evaluation details.

4.2 Online Reinforcement Learning

Single-reward optimization. Table 1 shows that MCReg improves guided GenEval and OCR with both RL methods. For DiffusionNFT, CFG lowers GenEval from 0.964 to 0.905, whereas MCReg achieves 0.981 with CFG, exceeding the baseline with or without CFG. For Flow-GRPO-Fast, CFG improves both GenEval and OCR under single-reward training, and MCReg further raises guided GenEval from 0.938 to 0.945. After model-based multi-reward training, however, the Flow-GRPO-Fast baseline also shows performance degradation on several metrics under CFG, indicating that this behavior is not specific to DiffusionNFT. Appendix E.2 reports the per-metric results.

Multi-reward optimization. Table 2 shows that guided MCReg outperforms DiffusionNFT with and without CFG across most metrics on SD3.5-Medium. Relative to the guided baseline, MCReg raises Pick-a-Pic HPSv2.1 from 34.608 to 36.658 and improves all six DrawBench generation metrics. On FLUX.2 Klein, MCReg also achieves higher PickScore on both benchmarks. MCReg reduces endpoint R_{abs} on both backbones.

Qualitative comparison. Figure 4 (left) shows that applying CFG to the DiffusionNFT baseline preserves the overall scene content but can degrade local detail and text accuracy. On SD3.5-Medium, CFG makes the sandwich fillings less distinguishable, whereas MCReg produces clearer ingredient separation than either baseline. On FLUX.2 Klein, the guided baseline writes "TOT" instead of "TO", while MCReg restores the correct "Text to Image" rendering. These observations are consistent with our analysis of the effect of marginal inconsistency on CFG predictions.

4.3 Multitask On-Policy Distillation

Distillation setup. Each variant distills the three frozen SD3.5-Medium DiffusionNFT baseline checkpoints for GenEval, OCR, and multi-reward generation into one student, sampling tasks equally. The OPD baseline and MCReg use the teacher's conditional prediction as the target, with no teacher-side CFG. MCReg also fits the student's unconditional prediction to this target without querying the teacher's unconditional branch. PDM matches the student's conditional prediction and CFG direction $(C - N)$ to their teacher counterparts, using both teacher branches.

Multitask performance. Table 3 shows that MCReg outperforms the baseline and PDM across most task metrics under CFG in both distillation frameworks. On DrawBench, MCReg raises HPSv2.1

Table 2: **Multi-reward optimization across backbones.** DiffusionNFT jointly optimizes PickScore, CLIPScore, and HPSv2.1; the remaining metrics are used only for evaluation.

Method	Pick-a-Pic			DrawBench						
	PickScore	CLIP	HPSv2.1	PickScore	CLIP	HPSv2.1	ImgRwd	UniReward	DeQA	$R_{\text{abs}}\downarrow$
SD3.5-M	19.940	24.289	20.625	20.674	23.851	20.165	-0.554	0.416	3.280	—
SD3.5-M *with CFG*	21.685	28.496	28.421	22.267	28.103	27.093	0.732	0.591	4.033	—
DiffusionNFT	**23.380**	29.350	35.510	23.750	28.567	33.467	1.456	0.667	4.339	0.182
DiffusionNFT *with CFG*	22.894	30.815	34.608	23.403	29.888	33.357	1.502	0.671	4.256	0.182
+ MCReg (Ours) *with CFG*	23.339	**31.877**	**36.658**	**23.761**	**30.207**	**34.875**	**1.574**	**0.681**	**4.344**	**0.018**
FLUX.2 Klein	19.990	25.201	22.450	20.643	24.754	20.714	-0.391	0.451	3.230	—
FLUX.2 Klein *with CFG*	21.712	28.779	29.298	22.470	29.338	27.261	0.918	0.669	3.900	—
DiffusionNFT	23.379	28.263	**36.182**	23.897	28.031	**34.153**	1.342	0.717	4.386	0.426
DiffusionNFT *with CFG*	23.322	29.611	35.841	23.868	29.116	34.011	1.418	0.727	4.389	0.426
+ MCReg (Ours) *with CFG*	**23.459**	**29.633**	35.910	**23.979**	**29.124**	33.974	**1.428**	**0.729**	**4.396**	**0.049**



Figure 4: **Qualitative comparison under CFG.** DrawBench samples from DiffusionNFT multi-reward models (left) and multitask OPD models (right).

from 32.736 to 33.996 in DiffusionOPD relative to the guided baseline. In DanceOPD, it achieves 34.272, compared with 32.935 for the guided baseline and 32.698 for PDM.

Qualitative comparison. The right panel of Figure 4 shows finer mouse fur and foliage with MCReg than with the guided DiffusionOPD baseline. In DanceOPD, MCReg clarifies the train–surfboard connection and avoids the extra person introduced by CFG in the baseline.

4.4 Guidance-Scale Sensitivity and Marginal Closure

Guidance-scale sensitivity. Figure 5 shows that the Original baselines in both OPD frameworks reach their highest DrawBench HPSv2.1 at $w = 2$ and then decline, whereas MCReg continues to improve before leveling off. MCReg exceeds Original and PDM at every tested $w \geq 2$, with a widening advantage over Original as guidance increases. In DiffusionNFT, MCReg also maintains a higher DrawBench PickScore throughout the sweep; its smaller decline widens the gap to the baseline at larger w.

Marginal closure. The right panel of Figure 5 shows that MCReg reduces endpoint R_{abs} relative to Original in DiffusionNFT and both OPD frameworks. In DanceOPD, PDM attains the lowest R_{abs} but still trails MCReg in guided HPSv2.1. PDM matches each teacher's CFG direction rather than using a shared conditional target for both student branches, so the branches can agree while their conditional mean remains displaced from the common evaluation target. Appendix E.1 quantifies this offset and examines the branch and guided-prediction errors.

Table 3: **Multitask on-policy distillation on SD3.5-Medium.** All methods use the same teachers.

Method	GenEval	OCR	DrawBench						
			PickScore	CLIP	HPSv2.1	ImgRwd	UniReward	DeQA	$R_{\text{abs}} \downarrow$
DiffusionOPD	0.958	0.962	23.723	28.390	32.871	1.436	0.665	4.308	0.146
DiffusionOPD *with CFG*	0.973	0.971	23.422	**29.974**	32.736	1.489	0.681	4.258	0.146
+ PDM *with CFG*	0.961	0.954	23.381	29.939	32.693	1.480	0.680	4.221	0.095
+ MCReg (Ours) *with CFG*	**0.977**	**0.973**	**23.843**	29.367	**33.996**	**1.544**	**0.684**	**4.312**	**0.046**
DanceOPD	0.953	0.964	23.706	28.373	33.149	1.415	0.664	**4.330**	0.143
DanceOPD *with CFG*	0.968	**0.973**	23.356	29.831	32.935	1.489	0.678	4.243	0.143
+ PDM *with CFG*	0.958	0.930	23.338	**29.844**	32.698	1.486	0.680	4.244	**0.041**
+ MCReg (Ours) *with CFG*	**0.970**	0.965	**23.711**	29.452	**34.272**	**1.552**	**0.683**	**4.330**	0.055

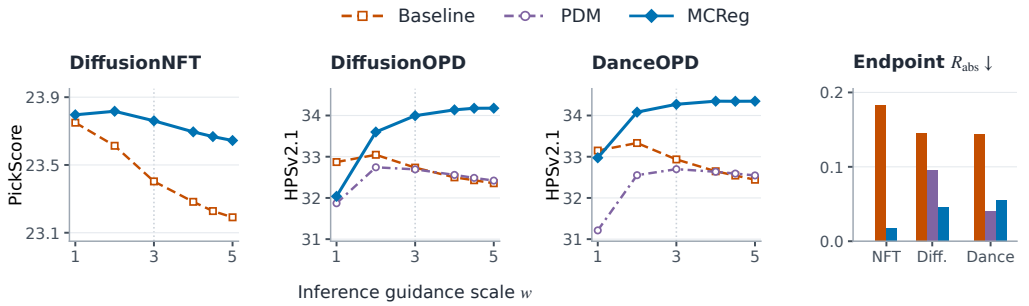

Figure 5: **Guidance-scale sensitivity and marginal closure.** *Left three panels:* DrawBench PickScore for DiffusionNFT and HPSv2.1 for both OPD frameworks as inference guidance scale w varies. *Right:* Marginal closure risk R_{abs} at the pure-noise endpoint; lower is better.

5 Related Work

Guidance in diffusion models. Sampling-time CFG methods refine guidance updates (Chung et al., 2025; Sadat et al., 2025), while analyses characterize guidance dynamics (Bradley & Nakkiran, 2025; Chidambaram et al., 2024; Pavasovic et al., 2026). For fine-tuned models, pretrained unconditional predictions can guide sampling (Phunyaphibarn et al., 2026; Zhong et al., 2025), while DogFit injects a source-model guidance offset into the fine-tuning loss (Bahram et al., 2026). Learn to Guide learns guidance weights with a frozen denoiser (Galashov et al., 2026). MCReg instead directly supervises the unconditional prediction with post-training targets.

Online RL. Online RL uses policy gradients (Black et al., 2024; Liu et al., 2025a; He et al., 2026), reward-driven matching (Zheng et al., 2026; Xue et al., 2026; Fu et al., 2026; Fan et al., 2025), and control regression (Domingo i Enrich et al., 2025). DiffusionOPSD builds reward-gradient targets (Zhou et al., 2026a). MCReg fits both branches to DiffusionNFT's recovered conditional target and adds an unconditional transition fit to Flow-GRPO's clipped objective.

Diffusion distillation. Distillation compresses trajectories or guided predictions (Salimans & Ho, 2022; Meng et al., 2023), matches distributions (Yin et al., 2024), or imitates student trajectories (Chen et al., 2026). DiffusionOPD and DanceOPD match transitions and generative fields (Li et al., 2026b; Zhou et al., 2026b). PDM fits conditional predictions and CFG directions to resolve fixed-scale ambiguity (Li et al., 2026a). MCReg fits both branches to conditional teacher targets, whose posterior average under the student sampling law defines the unconditional optimum.

6 Conclusion

Conditional-only post-training leaves CFG's unconditional prediction unidentified. Marginal closure identifies its shared-target squared-error optimum as the conditional optimum's posterior average at each noised state; joint excess risk bounds closure risk. MCReg fits both predictions to shared targets in DiffusionNFT and OPD and adds an advantage-weighted unconditional transition fit to Flow-GRPO-Fast. Across online RL and multitask OPD, it improves most guided-generation metrics and lowers measured endpoint closure risk relative to Original baselines.

AI USE STATEMENT

We used generative AI tools to assist with manuscript drafting and language polishing, and to support literature retrieval and the identification of related work. We manually reviewed and verified the AI-assisted text and suggested references for accuracy and relevance. We take full responsibility for the final content of the paper.

ETHICS STATEMENT

This work builds on existing text-to-image models and public benchmarks. Potential ethical concerns include biases inherited from pretrained models and the misuse of generated images to produce misleading or harmful content.

REPRODUCIBILITY STATEMENT

The proposed method is described in Section 3, with theoretical analysis and proofs provided in Appendices A–C. Details of the datasets, experimental settings, hyperparameters, and evaluation procedures are presented in Section 4.1 and Appendix D.

REFERENCES

Yara Bahram, Mohammadhadi Shateri, and Eric Granger. DogFit: Domain-guided fine-tuning for efficient transfer learning of diffusion models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 2345–2353, 2026. URL https://arxiv.org/abs/2508.05685.

Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/14f75513f0f1ca01de1e826b52e6b840-Abstract-Conference.html.

Black Forest Labs. FLUX.2 [klein] 9B Base: Model card. https://huggingface.co/black-forest-labs/FLUX.2-klein-base-9B, 2026.

Arwen Bradley and Preetum Nakkiran. Classifier-free guidance is a predictor-corrector. *Transactions on Machine Learning Research*, 2025. URL https://openreview.net/forum?id=zrWNtzSZsf.

Hansheng Chen, Kai Zhang, Hao Tan, Leonidas Guibas, Gordon Wetzstein, and Sai Bi. pi-Flow: Policy-based few-step generation via imitation distillation. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/f593c9c251d4d7cf14d4ab9861dfb7eb-Abstract-Conference.html.

Muthu Chidambaram, Khashayar Gatmiry, Sitan Chen, Holden Lee, and Jianfeng Lu. What does guidance do? a fine-grained analysis in a simple setting. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/9a3942c235daa9c5f62a8598ae81a946-Abstract-Conference.html.

Jaemoo Choi, Yuchen Zhu, Wei Guo, Petr Molodyk, Bo Yuan, Jinbin Bai, Yi Xin, Molei Tao, and Yongxin Chen. Rethinking the design space of reinforcement learning for diffusion models: On the importance of likelihood estimation beyond loss design, 2026. URL https://arxiv.org/abs/2602.04663v2.

Hyungjin Chung, Jeongsol Kim, Geon Yeong Park, Hyelin Nam, and Jong Chul Ye. CFG++: Manifold-constrained classifier free guidance for diffusion models. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/4c9477b9e2c7ec0ad3f4f15077aaf85a-Abstract-Conference.html.

Carles Domingo i Enrich, Michal Drozdzal, Brian Karrer, and Ricky T. Q. Chen. Adjoint matching: Fine-tuning flow and diffusion generative models with memoryless stochastic optimal control. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/852f50969a9e523ec41d26f2f68bd456-Abstract-Conference.html.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 12606–12633, 2024. URL https://proceedings.mlr.press/v235/esser24a.html.

Jiajun Fan, Shuaike Shen, Chaoran Cheng, Yuxin Chen, Chumeng Liang, and Ge Liu. Online reward-weighted fine-tuning of flow matching with wasserstein regularization. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/9690d4746230cfea3d067fca695ba648-Abstract-Conference.html.

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, 2023. URL https://arxiv.org/abs/2305.16381.

Zheming Fu, Ruizhe He, Wei Shang, Xiaoxiao Ma, Lei Wang, Chang Liu, and Siming Fu. FlowAWR: Online adaptive flow reinforcement via advantage-weighted rectification, 2026. URL https://arxiv.org/abs/2606.30376.

Alexandre Galashov, Ashwini Pokle, Arnaud Doucet, Arthur Gretton, Mauricio Delbracio, and Valentin De Bortoli. Learn to guide your diffusion model. In *International Conference on Learning Representations*, 2026.

Dhruba Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. GenEval: An object-focused framework for evaluating text-to-image alignment. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://arxiv.org/abs/2310.11513.

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. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/d75f561eaaf2cb754bc8d7e36d8af362-Abstract-Conference.html.

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. Association for Computational Linguistics, 2021. doi: 10.18653/v1/2021.emnlp-main.595. URL https://aclanthology.org/2021.emnlp-main.595/.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance, 2022. URL https://arxiv.org/abs/2207.12598.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pp. 6840–6851, 2020.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiana, Joe Penna, and Omer Levy. Pick-a-pic: An open dataset of user preferences for text-to-image generation. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/73aacd8b3b05b4b503d58310b523553c-Abstract-Conference.html.

Bingnan Li, Haozhe Wang, Haozhong Xiong, Fangtai Wu, Jinpeng Yu, Yang Shi, Jiaming Liu, and Ruihua Huang. Rethinking classifier-free guidance in on-policy diffusion distillation, 2026a. URL https://arxiv.org/abs/2607.24731.

Quanhao Li, Junqiu Yu, Kaixun Jiang, Yujie Wei, Zhen Xing, Pandeng Li, Ruihang Chu, Shiwei Zhang, Yu Liu, and Zuxuan Wu. DiffusionOPD: A unified perspective of on-policy distillation in diffusion models, 2026b. URL https://arxiv.org/abs/2605.15055.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2210.02747.

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. In *Advances in Neural Information Processing Systems*, volume 38, 2025a. doi: 10.52202/085713-1362. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/3a10c46572628d58cb44fb705f25cbbf-Abstract-Conference.html.

Jie Liu, Gongye Liu, Jiajun Liang, Yangguang Li, Jiaheng Liu, Xintao Wang, Pengfei Wan, Di Zhang, and Wanli Ouyang. Flow-GRPO: Official implementation. https://github.com/yifan123/flow_grpo/tree/879042cf5707f8b90daa98d147d7deac2317c5da, 2025b.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=XVjTT1nw5z.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://arxiv.org/abs/1711.05101.

Yihong Luo, Tianyang Hu, and Jing Tang. Reinforcing diffusion models by direct group preference optimization. In *International Conference on Learning Representations*, 2026. URL https://arxiv.org/abs/2510.08425.

Chenlin Meng, Robin Rombach, Ruiqi Gao, Diederik Kingma, Stefano Ermon, Jonathan Ho, and Tim Salimans. On distillation of guided diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023. URL https://openaccess.thecvf.com/content/CVPR2023/html/Meng_On_Distillation_of_Guided_Diffusion_Models_CVPR_2023_paper.html.

Alexander Quinn Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob Mcgrew, Ilya Sutskever, and Mark Chen. GLIDE: Towards photorealistic image generation and editing with text-guided diffusion models. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 16784–16804, 2022. URL https://proceedings.mlr.press/v162/nichol22a.html.

Krunoslav Lehman Pavasovic, Jakob Verbeek, Giulio Biroli, and Marc Mezard. Overshoot and shrinkage in classifier-free guidance: From theory to practice. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/d7168ad82b4780797ffa8202c531ef6a-Abstract-Conference.html.

Prin Phunyaphibarn, Phillip Y. Lee, Jaihoon Kim, and Minhyuk Sung. Unconditional priors matter! improving conditional generation of fine-tuned diffusion models. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 3982–3992, 2026. doi: 10.48550/arXiv.2503.20240. URL https://arxiv.org/abs/2503.20240.

Seyedmorteza Sadat, Otmar Hilliges, and Romann Weber. Eliminating oversaturation and artifacts of high guidance scales in diffusion models. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/d1450d6c10c6b6cf1b80964357f5fa08-Abstract-Conference.html.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L. Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, Jonathan Ho, David J. Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. In *Advances in Neural Information Processing Systems*, volume 35, 2022. doi: 10.52202/068431-2643. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/ec795aeadae0b7d230fa35cbaf04c041-Abstract.html.

Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=TIdIXIpzhoI.

Stability AI. Stable Diffusion 3.5 Medium: Model card. https://huggingface.co/stabilityai/stable-diffusion-3.5-medium, 2024.

Feng Wang and Zihao Yu. Coefficients-preserving sampling for reinforcement learning with flow matching, 2025. URL https://arxiv.org/abs/2509.05952.

Yibin Wang, Yuhang Zang, Hao Li, Cheng Jin, and Jiaqi Wang. Unified reward model for multimodal understanding and generation, 2025. URL https://arxiv.org/abs/2503.05236.

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

Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. ImageReward: Learning and evaluating human preferences for text-to-image generation. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/33646ef0ed554145eab65f6250fab0c9-Abstract-Conference.html.

Shuchen Xue, Chongjian Ge, Shilong Zhang, Yichen Li, and Zhi-Ming Ma. Advantage weighted matching: Aligning RL with pretraining in diffusion models. In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306, 2026. doi: 10.48550/arXiv.2509.25050. URL https://arxiv.org/abs/2509.25050.

Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Frédo Durand, William T. Freeman, and Taesung Park. One-step diffusion with distribution matching distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6613–6623, 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Yin_One-step_Diffusion_with_Distribution_Matching_Distillation_CVPR_2024_paper.html.

Zhiyuan You, Xin Cai, Jinjin Gu, Tianfan Xue, and Chao Dong. Teaching large language models to regress accurate image quality scores using score distribution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14483–14494, 2025.

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. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/d8e68ddfe22520b45b8fb8d5cbde5a21-Abstract-Conference.html. Official implementation: https://github.com/NVlabs/DiffusionNFT.

Jincheng Zhong, Xiangcheng Zhang, Jianmin Wang, and Mingsheng Long. Domain guidance: A simple transfer approach for a pre-trained diffusion model. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/1e2e0ad8818c91ee5e49da9cfb18724d-Abstract-Conference.html.

Wei Zhou, Xiongwei Zhu, Lingdong Kong, Bo Chen, Lei Zhang, Yongyuan Liang, Xiaoxia Hou, Ye Tian, Xian Sun, Yingshuo Wang, Linfeng Li, Shengqiong Wu, Leigang Qu, Feng Li, Wei Liu, Julian McAuley, and Tat-Seng Chua. On-policy self-distillation in diffusion models, 2026a. URL https://arxiv.org/abs/2608.24646.

Wei Zhou, Xiongwei Zhu, Zelin Xu, Bo Dong, Lixue Gong, Yongyuan Liang, Meng Chu, Leigang Qu, Lingdong Kong, Wei Liu, and Tat-Seng Chua. DanceOPD: On-policy generative field distillation, 2026b. URL https://arxiv.org/abs/2606.27377.

A Theoretical Results and Proofs

We use the notation of Section 3: P_k is the joint law at a fixed update, and Π_k averages conditional predictions over prompts given Z. We first formalize the fixed-scale identification gap and target mismatch, then prove Theorem 1 and derive the DiffusionNFT regression target. Appendix B extends the analysis to weighted regression and the application-specific sampling laws.

A.1 Branch identification under fixed-scale guidance

Proposition 1. *Fix P_k and w_0. Suppose an objective depends on (C, N) only through $G_{w_0} = w_0 C + (1 - w_0)N$. For any square-integrable Z-measurable field H, define*

$$C_H = C + (w_0 - 1)H, \qquad N_H = N + w_0 H.$$

Then $G_{w_0}(C_H, N_H) = G_{w_0}(C, N)$, while $N_H - \Pi_k C_H = N - \Pi_k C + H$.

Proof. The change in the guided output is $w_0(w_0 - 1)H + (1 - w_0)w_0 H = 0$. Since $\Pi_k H = H$, the closure residual changes by $w_0 H - (w_0 - 1)H = H$. □

At $w_0 = 1$, the transformation leaves C fixed and varies N freely. Whether shared parameters realize this output-space symmetry depends on the model class.

A.2 Target mismatch between conditional and unconditional predictions

Lemma 1. *Let Y_C, Y_N be square-integrable targets under a common sampling law P_k, and let $C = \mathbb{E}_{P_k}[Y_C \mid Z, c]$ and $N = \mathbb{E}_{P_k}[Y_N \mid Z]$ be the unrestricted minimizers of their squared-error risks. Then*

$$N - \Pi_k C = \mathbb{E}_{P_k}[Y_N - Y_C \mid Z]. \tag{19}$$

The pair satisfies marginal closure if and only if the right-hand side vanishes P_k-almost surely.

Proof. By the tower property, $\Pi_k C = \mathbb{E}_{P_k}[\mathbb{E}_{P_k}[Y_C \mid Z, c] \mid Z] = \mathbb{E}_{P_k}[Y_C \mid Z]$. Subtracting this identity from $N = \mathbb{E}_{P_k}[Y_N \mid Z]$ gives the result. □

A.3 Proof of Theorem 1

We prove the branchwise excess-risk identity in Eq. (9) and its three-term form in Eq. (10).

Proof. Because $C_k^\star = \mathbb{E}_{P_k}[Y_k \mid Z, c]$, the conditional-expectation residual is orthogonal in $L_2(P_k)$ to every square-integrable function of (Z, c). Therefore,

$$\mathbb{E}_{P_k}\|C - Y_k\|_2^2 = \mathbb{E}_{P_k}\|C - C_k^\star\|_2^2 + \mathbb{E}_{P_k}\|C_k^\star - Y_k\|_2^2. \tag{20}$$

Applying the same identity to $N_k^\star = \mathbb{E}_{P_k}[Y_k \mid Z]$ gives

$$\mathbb{E}_{P_k}\|N - Y_k\|_2^2 = \mathbb{E}_{P_k}\|N - N_k^\star\|_2^2 + \mathbb{E}_{P_k}\|N_k^\star - Y_k\|_2^2. \tag{21}$$

Substituting these identities into Eq. (8) gives Eq. (9). Since both coefficients are positive, the excess risk is zero if and only if $C = C_k^\star$ and $N = N_k^\star$ up to P_k-almost-sure equality. The law of iterated expectations gives $N_k^\star = \Pi_k C_k^\star$.

To obtain the three-term decomposition, write $e_C = C - C_k^\star$ and $e_N = N - N_k^\star$, and define

$$m = \Pi_k e_C, \qquad e_C^\perp = e_C - m, \qquad b_\rho = \frac{m + \rho e_N}{1 + \rho}. \tag{22}$$

The closure residual is $r = N - \Pi_k C = e_N - m$. Since $\Pi_k e_C^\perp = 0$, the conditional error decomposes orthogonally as

$$\mathbb{E}_{P_k}\|e_C\|_2^2 = \mathbb{E}_{P_k}\|e_C^\perp\|_2^2 + \mathbb{E}_{P_k}\|m\|_2^2. \tag{23}$$

Completing the square in the two Z-measurable errors gives

$$\frac{\|m\|_2^2 + \rho\|e_N\|_2^2}{1+\rho} = \|b_\rho\|_2^2 + \frac{\rho}{(1+\rho)^2}\|e_N - m\|_2^2. \tag{24}$$

Taking expectations and substituting into Eq. (9) proves Eq. (10). Dropping its first two nonnegative terms yields Eq. (11). Equality holds precisely when $e_C^\perp = 0$ and $b_\rho = 0$ almost surely. For any nonzero square-integrable Z-measurable field H, the pair $e_C = -\rho H$, $e_N = H$ satisfies these conditions and gives

$$R_{\mathrm{abs}} = (1+\rho)^2\mathbb{E}_{P_k}\|H\|_2^2, \qquad \Delta R_{\rho,k} = \rho\mathbb{E}_{P_k}\|H\|_2^2. \tag{25}$$

Thus the coefficient in the closure bound cannot be reduced. □

Projection interpretation. The closure term in Eq. (10) is the squared distance to the set of marginally closed prediction pairs in the metric induced by the joint risk. Write $\|f\|_k^2 := \mathbb{E}_{P_k}\|f\|_2^2$ and equip pairs of square-integrable functions of (Z, c) and Z with the norm

$$\|(A,B)\|_\rho^2 := \frac{\|A\|_k^2 + \rho\|B\|_k^2}{1+\rho}. \tag{26}$$

Since conditional expectation is a bounded linear operator, $\mathcal{M}_k := \{(C,N) : N = \Pi_k C\}$ is a closed linear subspace.

Proposition 2 (Projection onto marginally closed pairs)**.** *Under the assumptions of Theorem 1, let $r = N - \Pi_k C$. The orthogonal projection of (C,N) onto $\mathcal{M}_k$ in the above metric is*

$$\bar{C} = C + \frac{\rho}{1+\rho}r, \qquad \bar{N} = N - \frac{1}{1+\rho}r. \tag{27}$$

The joint risk satisfies the exact identity

$$R_{\rho,k}(C,N) = R_{\rho,k}(\bar{C},\bar{N}) + \frac{\rho}{(1+\rho)^2}R_{\mathrm{abs}}(C,N;P_k). \tag{28}$$

Proof. Because r is Z-measurable, Eq. (27) gives $\bar{N} = \Pi_k\bar{C}$. For any $(h, \Pi_k h) \in \mathcal{M}_k$, the inner product induced by $\|\cdot\|_\rho$ satisfies

$$\begin{aligned}&\big\langle (C-\bar{C}, N-\bar{N}), (h, \Pi_k h)\big\rangle_\rho \\ &\quad = \frac{\rho}{(1+\rho)^2}\mathbb{E}_{P_k}\big[\langle r, \Pi_k h\rangle - \langle r, h\rangle\big] = 0.\end{aligned} \tag{29}$$

The final equality follows by conditioning on Z. Hence $(\bar{C},\bar{N})$ is the orthogonal projection, with squared distance

$$\|(C-\bar{C}, N-\bar{N})\|_\rho^2 = \frac{\rho}{(1+\rho)^2}\|r\|_k^2. \tag{30}$$

The optimal pair $(C_k^\star, N_k^\star)$ belongs to $\mathcal{M}_k$. Applying the Pythagorean identity to Eq. (9) proves Eq. (28). □

Thus the shared-target risk decomposes into the risk of its projected pair and a penalty for departing from marginal closure. The projected errors are $\bar{C} - C_k^\star = e_C^\perp + b_\rho$ and $\bar{N} - N_k^\star = b_\rho$, recovering the first two terms of Eq. (10).

Consequences for CFG. The same error components describe how joint regression controls guided predictions at any fixed scale under P_k.

Corollary 1 (Guided-prediction error)**.** *Under the assumptions of Theorem 1, use the components in Eq.* (22) *and let $r = N - \Pi_k C$. For any fixed $w \in \mathbb{R}$, define the oracle $G^\star_w = wC^\star_k + (1-w)N^\star_k$. Then*

$$\begin{aligned} \|G_w - G^\star_w\|_k^2 &= w^2\|e_C^\perp\|_k^2 + \|m + (1-w)r\|_k^2 \\ &= w^2\|e_C^\perp\|_k^2 + \left\| b_\rho + \left(\frac{1}{1+\rho} - w\right) r \right\|_k^2, \end{aligned} \tag{31}$$

and

$$\|G_w - G^\star_w\|_k^2 \le (1+\rho)\left(w^2 + \frac{(1-w)^2}{\rho}\right)\Delta R_{\rho,k}. \tag{32}$$

The coefficient in this bound is sharp.

Proof. Since $e_C = e_C^\perp + m$ and $e_N = m + r$, the guided error is

$$\begin{aligned} G_w - G^\star_w &= we_C + (1-w)e_N \\ &= we_C^\perp + b_\rho + \left(\frac{1}{1+\rho} - w\right) r. \end{aligned} \tag{33}$$

The last two terms depend only on Z and are orthogonal to $e_C^\perp$, proving Eq. (31). Weighted Cauchy–Schwarz gives

$$\|we_C + (1-w)e_N\|_k^2 \le \left(w^2 + \frac{(1-w)^2}{\rho}\right)\left(\|e_C\|_k^2 + \rho\|e_N\|_k^2\right). \tag{34}$$

Eq. (9) yields the stated bound. For any nonzero square-integrable Z-measurable H, equality is attained by $e_C = wH$ and $e_N = (1-w)H/\rho$. □

At marginal closure, Eq. (31) reduces to $w^2\|e_C^\perp\|_k^2 + \|b_\rho\|_k^2$. Closure removes the residual term, while conditional and common marginal errors determine the remaining guided-prediction error.

A.4 Target recovery from the DiffusionNFT objective

Fix a parameter snapshot $\bar\theta_k$ and the rollout and reference models. Target construction uses these snapshots, including $\bar\theta_k$ for the adaptive normalizers; the regression variable is θ. The identities below omit the common output-coordinate average. Dividing the losses and their gradients by the output dimension gives the implementation, with all target-side values frozen and the gradient evaluated at $\theta = \bar\theta_k$.

For the adaptive normalizers, let $\bar C^\pm = C_{\mathrm{old}} \pm \gamma(C_{\bar\theta_k} - C_{\mathrm{old}})$ denote the implicit predictions at the fixed snapshot. The official implementation (Zheng et al., 2026) computes

$$d_\pm = \mathrm{sg}\left[\max\left\{\frac{1}{n}\|x_t - t\bar C^\pm - x_0\|_1,\ 10^{-5}\right\}\right],$$

where n is the number of output coordinates per sample and sg denotes stop-gradient. Thus each denominator is a sample-wise mean absolute reconstruction error with a numerical floor. For normalized advantage A, the optimality probability is $p_{\mathrm{opt}} = \frac{1}{2} + \mathrm{clip}(A, -A_{\max}, A_{\max})/(2A_{\max})$. The factor $A_{\max}/\gamma$ in the policy loss below follows the same implementation's loss normalization.

Lemma 2. *Fix one sample and hold the adaptive normalizers and all target-side quantities constant with respect to the current model output. Let C_{old} be the rollout-model prediction, $u_t = \epsilon - x_0$ the rectified-flow target, $\gamma > 0$ the DiffusionNFT implicit-policy coefficient, $p_{\mathrm{opt}} \in [0,1]$ the optimality probability, and $d_+, d_- > 0$ the adaptive loss normalizers. Define*

$$\begin{aligned} w_+ &:= \frac{p_{\mathrm{opt}}}{d_+}, & C^+ &:= (1-\gamma)C_{\mathrm{old}} + \gamma C, \\ w_- &:= \frac{1-p_{\mathrm{opt}}}{d_-}, & C^- &:= (1+\gamma)C_{\mathrm{old}} - \gamma C. \end{aligned} \tag{35}$$

For advantage-clipping scale $A_{\max} > 0$, the sample-level policy loss

$$\ell_{\text{policy}}(C) := \frac{A_{\max} t^2}{\gamma} \left(w_+ \|C^+ - u_t\|_2^2 + w_- \|C^- - u_t\|_2^2 \right) \tag{36}$$

equals $h\|C - T_{\text{policy}}\|_2^2$ up to a term independent of C, where $h = A_{\max} t^2 \gamma (w_+ + w_-)$ and

$$T_{\text{policy}} = C_{\text{old}} + \frac{w_+ - w_-}{\gamma(w_+ + w_-)}(u_t - C_{\text{old}}). \tag{37}$$

After adding the reference penalty $\beta_{\text{ref}}\|C - C_0\|_2^2$, let $W = h + \beta_{\text{ref}} > 0$. Completing the square gives the regression target

$$Y_k = \frac{h T_{\text{policy}} + \beta_{\text{ref}} C_0}{W}, \tag{38}$$

and its C-gradient equals that of $W\|C - Y_k\|_2^2$.

Proof. For the rectified path, $x_t - tC^{\pm} - x_0 = t(u_t - C^{\pm})$, giving the factor t^2 when the original reconstruction errors are expressed in velocity space. Let $\Delta = C - C_{\text{old}}$ and $D = u_t - C_{\text{old}}$. Then $C^+ - u_t = \gamma\Delta - D$ and $C^- - u_t = -(\gamma\Delta + D)$, so

$$\begin{aligned} &w_+ \|C^+ - u_t\|_2^2 + w_- \|C^- - u_t\|_2^2 \\ &\quad = \gamma^2 (w_+ + w_-) \|\Delta\|_2^2 + 2\gamma(w_- - w_+)\langle \Delta, D \rangle + (w_+ + w_-)\|D\|_2^2. \end{aligned} \tag{39}$$

Completing the square in Δ gives Eq. (37). Completing the square once more after adding the reference penalty yields

$$h\|C - T_{\text{policy}}\|_2^2 + \beta_{\text{ref}}\|C - C_0\|_2^2 = W\|C - Y_k\|_2^2 + \frac{h\beta_{\text{ref}}}{W}\|T_{\text{policy}} - C_0\|_2^2, \tag{40}$$

which proves the gradient identity because the final term is independent of C. □

B Extended Theory

B.1 Shared weighting changes the regression measure

Proposition 3. *Let $W > 0$ be integrable under P_k, and define the W-weighted law Q_k by*

$$\frac{\mathrm{d}Q_k}{\mathrm{d}P_k} = \frac{W}{\mathbb{E}_{P_k}[W]}. \tag{41}$$

Suppose Y_k is square-integrable under Q_k. If the two regressions use the same target and the same weight,

$$R_{\rho,k}^W(C, N) := \frac{1}{1+\rho}\mathbb{E}_{P_k}\left[W \left(\|C - Y_k\|_2^2 + \rho\|N - Y_k\|_2^2 \right) \right], \tag{42}$$

then their unrestricted minimizers are

$$C_W^\star = \mathbb{E}_{Q_k}[Y_k \mid Z, c], \qquad N_W^\star = \mathbb{E}_{Q_k}[Y_k \mid Z]. \tag{43}$$

Consequently, $N_W^\star = \mathbb{E}_{Q_k}[C_W^\star \mid Z]$: the pair satisfies marginal closure under Q_k.

Proof. Dividing $R_{\rho,k}^W$ by the positive constant $\mathbb{E}_{P_k}[W]$ gives Eq. (8) with P_k replaced by Q_k. The population minimizers and closure relation then follow from Theorem 1 and the law of iterated expectations under Q_k. □

B.2 Minibatch scaling

For a local minibatch of B samples, with n output coordinates per sample, the implemented loss is

$$\widehat{L}_{\rho,k}^{\text{NFT}}(\theta) = \frac{\overline{W}_B}{Bn(1+\rho)} \sum_{i=1}^{B} \left[\|C_\theta(Z_i, c_i) - Y_i\|_2^2 + \rho\|N_\theta(Z_i) - Y_i\|_2^2 \right], \tag{44}$$

where $\overline{W}_B = B^{-1}\sum_i W_i$ is computed separately for each local minibatch at each training timestep. The targets Y_i and weights W_i are detached.

We first derive the population risk for independent samples, then extend the result to the dependent minibatches used in training. Let $(W_i, Z_i, c_i, Y_i)_{i=1}^B$ be an i.i.d. minibatch from the fixed target construction, with $W_i > 0$ and the displayed moments finite. We omit the common factor $1/n$ below and define the per-sample joint regression loss

$$\ell_i(\theta) := \frac{1}{1+\rho}\left(\|C_\theta(Z_i, c_i) - Y_i\|_2^2 + \rho\|N_\theta(Z_i) - Y_i\|_2^2\right). \tag{45}$$

For $\overline{W}_B = B^{-1}\sum_i W_i$ and $\overline{\ell}_B = B^{-1}\sum_i \ell_i$, direct expansion gives

$$\begin{aligned}\mathbb{E}[\overline{W}_B\,\overline{\ell}_B] &= \frac{B-1}{B}\mathbb{E}[W]\mathbb{E}[\ell] + \frac{1}{B}\mathbb{E}[W\ell] \\ &= \mathbb{E}[W]\mathbb{E}[\ell] + \frac{1}{B}\operatorname{Cov}(W, \ell).\end{aligned} \tag{46}$$

Let $\mu_W = \mathbb{E}_{P_k}[W]$ and let Q_k be the weighted law from Proposition 3. Define

$$Q_{k,B} = \left(1 - \frac{1}{B}\right)P_k + \frac{1}{B}Q_k. \tag{47}$$

Then Eq. (46) gives

$$\frac{\mathbb{E}[\overline{W}_B\,\overline{\ell}_B]}{\mu_W} = \mathbb{E}_{Q_{k,B}}[\ell]. \tag{48}$$

Thus the unrestricted minimizers are $C_B^\star = \mathbb{E}_{Q_{k,B}}[Y_k \mid Z, c]$ and $N_B^\star = \mathbb{E}_{Q_{k,B}}[Y_k \mid Z]$, which satisfy closure under $Q_{k,B}$.

A common scalar preserves equal sample weights within each realized minibatch while changing the expected regression measure. Our implementation uses repeated-prompt sampling, group-based advantages, and minibatches drawn without replacement. To describe the resulting dependent batch law, fix the model snapshots and let I be uniform on $\{1, \ldots, B\}$, independently of the minibatch. Write $X_i = (Z_i, c_i, Y_i)$, $P_{k,I} = \mathcal{L}(X_I)$, $S = \overline{W}_B$, and $\ell_i(\theta) = \ell(X_i; \theta)$. For $W_i > 0$ almost surely and $0 < \mathbb{E}[S] < \infty$, define

$$\frac{\mathrm{d}\widetilde{Q}_{k,B}}{\mathrm{d}P_{k,I}}(x) = \frac{\mathbb{E}[S \mid X_I = x]}{\mathbb{E}[S]}. \tag{49}$$

Conditioning first on the batch and then on X_I gives

$$\frac{\mathbb{E}[S\overline{\ell}_B]}{\mathbb{E}[S]} = \frac{\mathbb{E}[S\ell(X_I;\theta)]}{\mathbb{E}[S]} = \mathbb{E}_{\widetilde{Q}_{k,B}}[\ell(X;\theta)]. \tag{50}$$

With Y square-integrable under $\widetilde{Q}_{k,B}$, Theorem 1 therefore gives the unrestricted minimizers $\widetilde{C}_B^\star = \mathbb{E}_{\widetilde{Q}_{k,B}}[Y \mid Z, c]$ and $\widetilde{N}_B^\star = \mathbb{E}_{\widetilde{Q}_{k,B}}[Y \mid Z]$. They satisfy $\widetilde{N}_B^\star = \mathbb{E}_{\widetilde{Q}_{k,B}}[\widetilde{C}_B^\star \mid Z]$, so closure holds under this induced law. For i.i.d. samples, this reduces to $Q_{k,B}$ on (Z, c, Y). A fixed positive scale, or a scale independent of the minibatch, gives $\widetilde{Q}_{k,B} = P_{k,I}$.

B.3 Reference decomposition of learned predictions

For any square-integrable learned conditional prediction C and fixed square-integrable Z-measurable reference s_0, define $d_{C,k} := \Pi_k C - s_0$ and $l_{C,k} := C - \Pi_k C$. Then $\Pi_k l_{C,k} = 0$. Writing $r_k = N - \Pi_k C$ gives

$$\begin{aligned}C &= s_0 + d_{C,k} + l_{C,k}, \qquad N = s_0 + d_{C,k} + r_k, \\ G_w &= s_0 + d_{C,k} + w l_{C,k} + (1-w) r_k.\end{aligned} \tag{51}$$

These identities follow by adding and subtracting $\Pi_k C$; the centering condition makes the decomposition unique for a fixed reference. At $C = C_k^\star$, the components reduce to the marginal shift d and centered prompt component l in Eq. (6), with $P = P_k$. Under closure, the guided field consists of the reference, marginal change, and scaled prompt component.

For $C = C_k^\star$, Eq. (7) reduces to guided error $(1-w)r_k$, as analyzed in Section 3.1.

If the unconditional prediction is kept at $N_0 = s_0$, its residual is $r_k = -d_{C,k}$ and CFG becomes $s_0 + wd_{C,k} + wl_{C,k}$. It therefore scales the change in the marginal together with the centered prompt component. Replacing N_0 by $\Pi_k C$ restores the coefficient of $d_{C,k}$ to one, with the same conditional prediction throughout this comparison.

B.4 Gaussian mixture example

Figure 2 (right) uses four analytically constructed prediction pairs. In this example, $\tau \in [0,1]$ runs from noise to data. Let c be uniform on $\{-1,1\}$, draw $x \mid c \sim \mathcal{N}(1.5c, 0.25)$ and independently $\epsilon \sim \mathcal{N}(0,1)$, and set

$$z_\tau = (1-\tau)\epsilon + \tau x, \qquad Y = x - \epsilon, \qquad Z = (z_\tau, \tau),$$

with independent $\tau \sim \mathcal{U}[0,1]$. Let P denote the resulting joint law, and let $(C^\star, N^\star)$ be the unrestricted squared-error optima for Y. Define

$$h(\tau) = 4\tau(1-\tau), \qquad m(Z) = \mathbb{E}_P[c \mid Z] = \tanh\left(\frac{1.5\tau z_\tau}{(1-\tau)^2 + 0.25\tau^2}\right).$$

With $s_\tau^2 = (1-\tau)^2 + 0.25\tau^2$, Gaussian conditioning gives

$$a_\tau = \frac{0.25\tau - (1-\tau)}{s_\tau^2}, \qquad b_\tau = \frac{1.5(1-\tau)}{s_\tau^2},$$
$$C^\star(Z,c) = a_\tau z_\tau + b_\tau c, \qquad N^\star(Z) = a_\tau z_\tau + b_\tau m(Z).$$

The four pairs are

$$\begin{aligned}(C_{\text{cond}}, N_{\text{cond}}) &= (C^\star, N^\star + 0.75h),\\ (C_{\text{fix}}, N_{\text{fix}}) &= (C^\star + 1.5h, N^\star + 2.25h),\\ (C_{\text{mis}}, N_{\text{mis}}) &= (C^\star, N^\star + \gamma hm),\\ (C_{\text{MCReg}}, N_{\text{MCReg}}) &= (C^\star, N^\star).\end{aligned}$$

Here $\gamma = \sqrt{0.15/\mathbb{E}_P[h^2m^2]} \approx 0.65343$. The conditional-only pair retains the conditional optimum and introduces an unconditional offset. The fixed-scale pair uses compensating offsets to preserve the oracle guided prediction at $w_0 = 3$. The mismatched pair consists of the population optima for targets $Y_C = Y$ and $Y_N = Y + \gamma h(\tau)c$, respectively, since $\mathbb{E}_P[Y_N - Y_C \mid Z] = \gamma hm$. The MCReg population pair uses the common target Y.

For each pair, we evaluate $\text{MSE}(w) = \mathbb{E}_P[|G_w(Z,c) - G_w^\star(Z,c)|^2]$ under the same joint law, where $G_w^\star = N^\star + w(C^\star - N^\star)$. Since $\mathbb{E}_P[h^2] = 8/15$, the four MSE curves are $0.3(w-1)^2$, $0.3(w-3)^2$, $0.15(w-1)^2$, and 0, respectively. Figure 2 (right) reports these values at $w \in \{1,2,3,4,5\}$.

B.5 Conditional teacher targets for on-policy distillation

At a fixed student update k, sample a task and teacher index M and a prompt c from the training mixture, then collect states $Z = (x_t, t)$ along the current student's conditional trajectory. The resulting joint law is P_k. The teacher index selects the supervision source and is not a separate input to the student. With student rollouts and teacher queries at CFG scale 1, define the detached velocity target $Y_k = \text{sg}[C_{T_M}(Z,c)]$.

Both predictions regress to this target through Eq. (18). DanceOPD uses $a(Z) = 1$; the transition-matching weight for DiffusionOPD is derived below.

Source of the target mismatch. Consider a single teacher at a state Z where its reference prompt posterior $\pi_T(c \mid Z)$ and the student's prompt posterior $\pi_S(c \mid Z)$ are defined. Suppose $C_T(Z,\cdot)$ has finite second moments under both. Write $N_S^\star(Z) = \mathbb{E}_{\pi_S}[C_T(Z,c)]$ and $r_T(Z) = N_T(Z) - \mathbb{E}_{\pi_T}[C_T(Z,c)]$, with all expectations taken at this fixed state. Adding and subtracting the teacher's posterior mean gives

$$N_T(Z) - N_S^\star(Z) = r_T(Z) + \big(\mathbb{E}_{\pi_T} - \mathbb{E}_{\pi_S}\big)[C_T(Z,c)]. \tag{52}$$

The first term is the teacher's closure residual, and the second is the change in the posterior average of its conditional prediction. At a state where the teacher satisfies closure, the mismatch equals this change in posterior average. For multiple teachers, apply Eq. (52) to each T_m using its reference posterior $\pi_{T_m}(c \mid Z)$ and the student posterior $P_k(c \mid Z, M = m)$ at the states considered. Averaging over $P_k(M = m \mid Z)$ gives the mismatch in Eq. (17).

Consequences for closure and CFG. Suppose $C_{T_M}(Z, c)$ and $N_{T_M}(Z)$ are square-integrable under P_k, and let $\delta = \widetilde{N}_k^\star - N_k^\star$ denote the target mismatch in Eq. (17). Separately regressing the two student predictions on the corresponding teacher predictions yields the unrestricted population pair $(C_k^\star, \widetilde{N}_k^\star) = (C_k^\star, N_k^\star + \delta)$. For any fixed scale w, compare the guided predictions $G_w^\star = wC_k^\star + (1-w)N_k^\star$ and $\widetilde{G}_w^\star = wC_k^\star + (1-w)\widetilde{N}_k^\star$. Then

$$\begin{aligned} R_{\mathrm{abs}}(C_k^\star, \widetilde{N}_k^\star; P_k) &= \mathbb{E}_{P_k}\|\delta\|_2^2, \\ \widetilde{G}_w^\star - G_w^\star &= (1-w)\delta. \end{aligned} \tag{53}$$

Indeed, $N_k^\star = \Pi_k C_k^\star$ makes the closure residual of the separately matched pair exactly δ. Subtracting the two guided predictions gives the second identity.

Transition matching. Our DiffusionOPD instantiation uses deterministic Euler transitions (Li et al., 2026b). For signed step $\Delta\sigma_t \neq 0$, the conditional student, unconditional student, and teacher transition means at the same state are

$$\begin{aligned} &\mu_C = x_t + \Delta\sigma_t C_\theta(Z, c), \qquad \mu_N = x_t + \Delta\sigma_t N_\theta(Z), \\ &\mu_T = x_t + \Delta\sigma_t Y_k. \end{aligned} \tag{54}$$

Both student means are matched to the same cached teacher mean:

$$\begin{aligned} L_{\rho,k}^{\mathrm{OPD}} &= \frac{1}{2(1+\rho)}\mathbb{E}_{P_k}\left[\|\mu_C - \mu_T\|_2^2 + \rho\|\mu_N - \mu_T\|_2^2\right] \\ &= \frac{1}{1+\rho}\mathbb{E}_{P_k}\left[a(Z)\left(\|C_\theta - Y_k\|_2^2 + \rho\|N_\theta - Y_k\|_2^2\right)\right], \end{aligned} \tag{55}$$

where $a(Z) = (\Delta\sigma_t)^2/2 > 0$ on the supervised timesteps. The implementation additionally averages over output coordinates, introducing the same constant factor for both regressions. States, teacher targets, and transition coefficients are held fixed during each update.

Assume $0 < \mathbb{E}_{P_k}[a(Z)] < \infty$ and $\mathbb{E}_{P_k}[a(Z)\|Y_k\|_2^2] < \infty$, and define Q_k by $\mathrm{d}Q_k/\mathrm{d}P_k = a(Z)/\mathbb{E}_{P_k}[a(Z)]$. Since this weight depends only on Z, it leaves the conditional laws of (Y_k, c) given Z unchanged. Consequently,

$$\begin{aligned} C_k^\star &= \mathbb{E}_{Q_k}[Y_k \mid Z, c] = \mathbb{E}_{P_k}[Y_k \mid Z, c], \\ N_k^\star &= \mathbb{E}_{Q_k}[Y_k \mid Z] = \mathbb{E}_{P_k}[Y_k \mid Z] = \Pi_k C_k^\star. \end{aligned} \tag{56}$$

Thus retaining the OPD timestep weighting preserves the population marginal-closure relation under the student visitation law. The closure bound in Eq. (11) applies to the normalized risk $L_{\rho,k}^{\mathrm{OPD}}/\mathbb{E}_{P_k}[a(Z)]$ and closure risk under Q_k. The state-dependent weight also preserves $P_k(M, c \mid Z)$, so $\widetilde{N}_k^\star$ and δ are unchanged. With finite weighted second moments for both teacher targets, Eq. (53) holds with closure risk and expectation taken under Q_k.

In our multi-teacher setting, equal-sized prompt batches from multi-reward, OCR, and GenEval teachers define an equal task mixture. The unconditional population optimum is $\mathbb{E}_{P_k}[Y_k \mid Z]$ under this pooled law. At intermediate states, the averaging weights are the posterior task and prompt probabilities given Z. Under independent pure-noise probes, they reduce to the prescribed task and prompt prior.

C Flow-GRPO-Fast Implementation

We use the Flow-GRPO-Fast implementation of Flow-GRPO. Its conditional branch uses a clipped surrogate objective and therefore does not specify a unique parameter-independent regression target. Its MCReg instantiation uses conditional rollout transitions and reward-dependent weights to fit the

unconditional transition mean. Sampling uses Coefficients-Preserving Sampling (CPS) (Wang & Yu, 2025) from the official implementation (Liu et al., 2025b). For each prompt group and selected timestep, let

$$\overline{A}_{i,t} := \mathrm{clip}(A_{i,t}, -A_{\max}, A_{\max}), \qquad \omega_{i,t} := \frac{\exp(\eta \overline{A}_{i,t})}{|\mathcal{G}_i|^{-1} \sum_{j \in \mathcal{G}_i} \exp(\eta \overline{A}_{j,t})}, \tag{57}$$

where $\mathcal{G}_i$ contains all rollout samples sharing prompt c_i. We use $\eta = 1/A_{\max} = 0.2$. The detached weights are positive and have empirical mean one within each group. With S_θ denoting the negative mean squared transition residual, the unconditional branch minimizes

$$L_{N,t}(\theta) = -\mathbb{E}_i \left[\omega_{i,t} S_\theta(x_{t-1}^{(i)} \mid x_t^{(i)}, t, \emptyset) \right]. \tag{58}$$

For the CPS transition used in our experiments, the implementation averages the squared residual over the D_x non-batch output dimensions:

$$S_\theta(x_{t-1} \mid x_t, t, \emptyset) = -\frac{1}{D_x} \|x_{t-1} - \mu_\theta(x_t, t, \emptyset)\|_2^2, \tag{59}$$

and μ_θ is affine in the model velocity. Eq. (58) is therefore a squared fit of the unconditional transition mean to the exponentially reweighted rollout distribution.

The complete training loss is

$$L_{\rho,t}^{\mathrm{Flow}}(\theta) = \frac{L_{\mathrm{cond},t}(\theta) + \rho L_{N,t}(\theta)}{1 + \rho}, \tag{60}$$

where $L_{\mathrm{cond},t}$ is the native conditional loss, including its reference penalty when enabled by the configuration. Its replay rule filters trajectories with zero advantages at every selected timestep and restores samples as needed to complete batches. The unconditional term uses the full conditional rollout bank, including those trajectories.

For the population-normalized weight $\omega_\eta \propto \exp(\eta \overline{A})$, the expansion at $\eta = 0$ is

$$\omega_\eta = 1 + \eta \left(\overline{A} - \mathbb{E}[\overline{A} \mid c, t] \right) + O(\eta^2). \tag{61}$$

Substituting this expansion into Eq. (58) gives an unweighted regression loss and the first-order correction $-\eta \mathbb{E}[(\overline{A} - \mathbb{E}[\overline{A} \mid c, t]) S_\theta]$. Differentiating this correction with rollout samples and weights held fixed gives an advantage-weighted gradient of S_θ, analogous to an unclipped policy-gradient update (Xue et al., 2026). The implemented objective uses finite prompt groups, clipped advantages, replay, and the discretized CPS transition. Its conditional branch remains the native clipped Flow-GRPO-Fast objective, while Eq. (58) trains the unconditional branch without a separate unconditional rollout or ratio. Theorem 1 characterizes two-branch shared-target regression; the Flow-GRPO-Fast experiment tests the related weighted unconditional fit within native clipped policy optimization.

D Experimental Details

D.1 Training and evaluation protocol

Models and common settings. We fine-tune both backbones with LoRA (Hu et al., 2022), using rank 32 and $\alpha = 64$. The SD3.5-M adapters act on the transformer attention projections; the FLUX.2 Klein Base 9B adapters also cover feed-forward projections. All training runs use NVIDIA H100 GPUs: eight for single-reward SD3.5-M RL and OPD, and 32 for multi-reward RL on either backbone. Local task evaluations use 40 sampling steps at 512×512 resolution.

RL settings. The recorded SD3.5-M runs use fp16 with AdamW (Loshchilov & Hutter, 2019), learning rate 3×10^{-4}, $(\beta_1, \beta_2) = (0.9, 0.999)$, $\epsilon = 10^{-8}$, weight decay 10^{-4}, and gradient clipping at 1. The controlled configurations use training seed 42; the seed of the older archived DiffusionNFT GenEval Original in Figure 1 and Table 1 is not recorded. DiffusionNFT clips advantages to $[-A_{\max}, A_{\max}]$ with $A_{\max} = 5$, uses $\beta_{\mathrm{ref}} = 10^{-4}$, and sets the implicit-policy coefficient γ to 1 for GenEval and 0.1 for OCR and multi-reward training. Its $w = 1$ rollouts use deterministic DPM2 sampling with ten steps for GenEval and OCR and 25 for multi-reward training; the sampled timestep fraction is 0.99. Each rollout round has 48 prompt groups of 24 images. GenEval and multi-reward training take one optimizer update per round (1,152 images per update), while OCR takes two (576 images per update).

Compositional generation and text rendering. The GenEval and OCR runs use the training and test prompt splits provided with the corresponding task data. Both DiffusionNFT and Flow-GRPO-Fast use matched 1,000-update runs for GenEval and separate 150-update runs for OCR. In Table 1, MCReg uses $\rho = 0.01$ and 0.03 for GenEval with DiffusionNFT and Flow-GRPO-Fast, respectively, and $\rho = 0.03$ and 0.003 for OCR. Figure 1 shows the pretrained reference and 1,000-update GenEval checkpoints.

The single-reward Flow-GRPO-Fast runs use $w = 1$ and ten-step CPS rollouts with noise level 0.8, 24 images per prompt, and 48 prompt groups per round. Each round yields two optimizer updates (576 images per update) with three replay timesteps. The PPO ratio clip is 10^{-5}, the reference-KL coefficient is zero, and the EMA decay is 0.9. Both tasks use the AdamW settings above.

GenEval has 2,212 test samples and OCR has 1,018. The DiffusionNFT and Flow-GRPO-Fast evaluations retain distributed-sampler padding, producing 2,216 and 1,024 aggregated rows, respectively. The OPD evaluations use exact distributed sampling without padding and aggregate the 2,212 and 1,018 unique rows.

The Flow-GRPO reference rows follow Choi et al. (2026, Tables 2 and 5, version 2); the AWM rows follow Xue et al. (2026, Tables 1 and 2), using its extended-training OCR result. DGPO GenEval and OCR scores follow Luo et al. (2026, Tables 1 and 2). These reference rows retain their source evaluation settings. We evaluate the SD3.5-M pretrained rows at CFG $= 1$ and $= 3$.

Preference and multi-reward optimization. Multi-reward training jointly optimizes PickScore, CLIPScore, and HPSv2.1. The SD3.5-Medium block in Table 2 uses matched 500-update checkpoints from 1,000-update runs, with MCReg $\rho = 0.03$. The FLUX.2 Klein block uses matched 200-update checkpoints from 250-update runs, with MCReg $\rho = 0.002$. FLUX training uses the Pick-a-Pic split with 25,432 training prompts and 2,048 test prompts. Each checkpoint is evaluated on Pick-a-Pic and DrawBench at both guidance scales, with the constituent reward scores reported separately.

The training reward is the unit-weight sum of the three constituent scores on their native scales. For SD3.5-M, the advantage tracker subtracts the prompt's accumulated mean reward and divides by the current rollout's global reward standard deviation plus 10^{-4}. FLUX subtracts each current prompt-group mean and likewise divides by the global reward standard deviation plus 10^{-4}. FLUX uses bf16, seed 42, 25-step deterministic DPM2 training rollouts at $w = 1$ with noise level 0.7, 24 images for each of 48 prompts per round, and one optimizer update per round (1,152 images). It samples 24 discrete training timesteps with shift 3.0. Its implicit-policy coefficient is $\gamma = 0.1$, with $A_{\max} = 5$ and $\beta_{\text{ref}} = 10^{-4}$; it uses the AdamW settings above. FLUX evaluation uses the native 40-step flow ODE sampler, and exported checkpoints use EMA decay 0.9.

We also evaluate the SD3.5-M and FLUX.2 Klein pretrained rows at $w = 1$ and $w = 3$, using 2,048 Pick-a-Pic prompts and 200 DrawBench prompts. DrawBench uses five generated samples per prompt.

Multi-task on-policy distillation. Table 3 uses SD3.5-Medium students trained for 2,000 steps with DiffusionOPD (Li et al., 2026b) and DanceOPD (Zhou et al., 2026b). Both frameworks use the same three DiffusionNFT teachers: the GenEval checkpoint at 1,000 updates, the OCR checkpoint at 150 updates, and the multi-reward checkpoint at 500 updates. Students start from the pretrained backbone with fresh LoRA adapters. Each optimizer attempt accumulates three task-specific microbatches, one for each teacher, with three rollout images per GPU in each microbatch (72 rollout images in total). The task index selects the teacher and is not an input to the student. Student rollouts and conditional teacher queries use $w = 1$ and ten sampling steps. The baseline and MCReg use conditional teacher targets; PDM additionally queries the teacher's unconditional prediction to match the conditional-minus-unconditional CFG direction. Within each framework, the variants use the same task mixture and optimization budget, with one inner epoch per rollout batch.

The OPD students use AdamW (Loshchilov & Hutter, 2019) with learning rate 3×10^{-4}, $(\beta_1, \beta_2) = (0.9, 0.999)$, $\epsilon = 10^{-8}$, weight decay 10^{-4}, gradient clipping at 1, fp16 training, and EMA checkpoints. DanceOPD uses grouped task routing and one supervised pre-Euler state per trajectory, selected with its low-timestep $\text{Beta}(5, 2)$ rule. MCReg uses $\rho = 0.1$ in both frameworks. PDM (Li

et al., 2026a) uses $\lambda = 2$ to weight CFG-direction matching relative to conditional-prediction matching.

OPD target ablation. Table 5 uses the same 2,000-step protocol, with $\rho = 0.1$ for MCReg and Teacher-null and $\lambda = 2$ for PDM. Teacher-null replaces MCReg's unconditional regression target with the selected teacher's unconditional prediction, retaining the conditional objective and branch weights. At each supervised state, Original OPD and MCReg use a conditional teacher query, which MCReg reuses for both student losses. Teacher-null and PDM additionally obtain the teacher's unconditional prediction. DanceOPD retains group size three, one inner update, and its low-timestep sampling rule. Its C2000 checkpoints correspond to 2,000 optimizer attempts; all four Dance variants completed 2,000 successful updates with no AMP-skipped step.

Training cost. GPU-hours are elapsed time multiplied by the number of allocated GPUs. For the matched eight-GPU, 150-update DiffusionNFT OCR runs in Table 1, Original takes 23.714 GPU-hours and MCReg ($\rho = 0.03$) takes 28.484 GPU-hours, a 20.1% increase. This measurement starts after model and optimizer initialization, includes rollout generation, reward evaluation, updates, synchronization, and checkpointing, and ends before external evaluation.

For OPD, timing instead runs from the job entering its running state to checkpoint weight availability, including initialization and checkpoint export. At C2000, DiffusionOPD takes 50.7, 50.4, and 55.2 GPU-hours for Original, MCReg, and PDM, respectively; DanceOPD takes 19.3, 20.0, and 22.8 GPU-hours in the same order.

Metrics and table conventions. GenEval Overall is the mean of its six category accuracies. OCR measures normalized edit-distance similarity after lowercasing and removing spaces, with full credit when the recognized text contains the requested string. PickScore uses its original scale; CLIPScore and HPSv2.1 are reported at $100\times$ their stored values.

CFG denotes inference guidance, and native sampling uses CFG $= 1$. In the main tables, bold marks the best locally evaluated trained-model value at CFG $= 1$ or $= 3$ within each comparison block, including gray-shaded Original rows. Source-reported literature rows and pretrained-model references are excluded from ranking. Ties are determined at displayed precision, and columns with missing comparison values are not ranked. Dashes denote unavailable results. CLIP, ImgRwd, and UniReward abbreviate CLIPScore, ImageReward, and UnifiedReward, respectively. Metric arrows indicate whether higher or lower is better.

D.2 OPD ENDPOINT PREDICTION DIAGNOSTICS

For the 2,000-step OPD students in Table 3, the evaluation law P_{eval} uses 768 shared pure-noise states at $t = 1$ and $\sigma = 1$, independent of prompts. At each state, we enumerate all 200 distinct DrawBench prompts with equal weight; this is equivalent to averaging the 1,000 source rows, in which each prompt occurs five times. Every student is compared with the same Original DiffusionNFT multi-reward teacher T_{mr} at 500 updates. Its conditional prediction and DrawBench marginal define the evaluation target pair

$$C^\star(Z,c) = C_{T_{\mathrm{mr}}}(Z,c), \qquad N^\star(Z) = \mathbb{E}_{c\sim q_{\mathrm{DB}}}[C_{T_{\mathrm{mr}}}(Z,c)], \tag{62}$$

where q_{DB} is uniform over the 200 prompts. The students retain their three-task training with three teachers; JER evaluates them against this separate fixed target.

We average squared errors over the output dimension n and define $R_C = n^{-1}\mathbb{E}_{P_{\mathrm{eval}}}\|C - C^\star\|_2^2$ and $R_N = n^{-1}\mathbb{E}_{P_{\mathrm{eval}}}\|N - N^\star\|_2^2$. The joint excess risk (JER) is

$$\mathrm{JER}(P_{\mathrm{eval}}) = \frac{R_C + R_N}{2}. \tag{63}$$

By Eq. (9), this is the per-coordinate, equal-weight excess risk for the fixed evaluation target $Y = C_{T_{\mathrm{mr}}}(Z,c)$. At equal branch weight, the closure term contributes one quarter of the coordinate-averaged R_{abs} to JER in Eq. (10); the remaining terms measure teacher-relative prediction error. These evaluation weights do not depend on training ρ or inference w. Full prompt enumeration eliminates prompt-sampling correction for R_N and R_{abs}.

Evaluations use model timestep 1000, 512×512 resolution, and fp16 model predictions. Confidence intervals use 1,000 paired-state bootstrap draws, preserving the original three strata of 256 noise states. The same resampled states are used for every method and paired difference. These intervals describe state-sampling uncertainty for the fixed checkpoints and prompt population. Table 5 reports the risks, and Table 6 reports paired changes.

D.3 Endpoint closure statistics

Table 4 reports closure diagnostics for GenEval, OCR, and multi-reward training, including normalization denominators. At the pure-noise endpoint, we average the conditional prediction over $K = 64$ prompts at each evaluation noise state. For a bank drawn without replacement from a test file of M rows, the finite-K correction subtracts $(1 - K/M)\widehat{V}/K$ from the coordinate-averaged squared closure residual, where $\widehat{V}$ is the unbiased sample variance of the conditional predictions across sampled prompts. This diagnostic compares a checkpoint's two predictions at pure noise; it is separate from the rollout-state regression loss used for training.

Table 4: **Marginal closure across tasks and RL backends.** Absolute closure risk and normalized closure error are measured at the pure-noise endpoint.

Method	$R_{\text{abs}} \downarrow$	D	E_{norm} (%) $\downarrow$
DiffusionNFT / GenEval			C1000
Original	0.07059	0.75112	9.397
+ MCReg	**0.00813**	1.57567	**0.516**
DiffusionNFT / OCR			C150
Original	0.05540	0.33002	16.787
+ MCReg	**0.00289**	0.26729	**1.082**
Flow-GRPO-Fast / GenEval			C1000
Original	0.11182	0.12090	92.487
+ MCReg	**0.04098**	0.27959	**14.658**
DiffusionNFT / Multi-reward; evaluated on Pick-a-Pic			C1000
Original	0.14824	0.18075	82.013
+ MCReg	**0.02001**	0.15577	**12.846**
DiffusionNFT / Multi-reward; evaluated on DrawBench			C1000
Original	0.18880	0.16130	117.053
+ MCReg	**0.02789**	0.13584	**20.529**

R_{abs}: absolute closure risk; $E_{\text{norm}} = 100 R_{\text{abs}}/D$. Blue: lower risk or error than the corresponding baseline. C denotes the training update. The DiffusionNFT/GenEval Original here differs from the Original in Table 1.

For the single-reward GenEval and OCR rows of Table 4, evaluation uses one endpoint noise state per test row and one rollout repeat. The test files contain 2,212 GenEval or 1,018 OCR rows. At each state, $K = 64$ bank rows are sampled without replacement. Matched methods use identical noise states and prompt-bank indices within each task. The multi-reward measurements use 1,000-update checkpoints.

Normalized closure error is $E_{\text{norm}} = 100 R_{\text{abs}}/D$, where D is the empirical mean squared distance between each checkpoint's conditional prompt average and the pretrained unconditional prediction. All squared distances are averaged over output coordinates. The denominator itself is not finite-K corrected. Each checkpoint uses its own conditional prompt average and denominator. For DiffusionNFT/GenEval, Table 4 and Table 1 use different Original checkpoints.

The R_{abs} values in Table 2 instead use the same exported EMA checkpoints as its performance measurements: 500 updates for SD3.5-Medium and 200 updates for FLUX.2 Klein. Each checkpoint is evaluated on 1,000 shared pure-noise states at 512×512 resolution. At each state, we sample $K = 64$ rows without replacement from the 1,000-row DrawBench test file (200 distinct prompts repeated five times), using the same prompt banks across checkpoints. The correction above uses

$M = 1{,}000$ for this file. SD3.5-Medium uses fp16 predictions at model timestep 1000; FLUX uses bf16 predictions on its native packed noise latents at model timestep 1. Both branches use the same checkpoint, with an empty prompt for the unconditional prediction. Risk is independent of inference CFG, and magnitudes are compared within each backbone.

E Additional Results

The fixed-teacher diagnostics relate branch accuracy to endpoint closure. Additional RL comparisons show native and guided performance, followed by guidance and training-budget trajectories and selected paired examples.

E.1 OPD target ablation and fixed-teacher diagnostics

Table 5 compares MCReg with Teacher-null, Original OPD, and PDM under the fixed teacher. MCReg reduces JER relative to all three comparators in both frameworks, with all six paired 95% confidence intervals below zero. Relative to Original OPD, its JER is 30.4% lower in DiffusionOPD and 17.9% lower in DanceOPD under the fixed evaluation law.

The branch errors in Figure 6 locate the JER gain. MCReg's reduction in R_N drives its advantage over Original OPD in both frameworks. Relative to Teacher-null, MCReg reduces R_N in DiffusionOPD and both branch errors in DanceOPD. In DanceOPD, MCReg's lower R_C explains its JER advantage over PDM; their R_N values are close (paired difference -0.00086, 95% CI $[-0.00375, 0.00215]$). Table 6 gives the paired differences.

Table 5: **Fixed-teacher OPD prediction risks.** The 2,000-step students are compared under the common evaluation target. Bold marks the lowest risk within each framework.

Method	$R_C \downarrow$	$R_N \downarrow$	JER $\downarrow$	$R_{\mathrm{abs}} \downarrow$
DiffusionOPD				
Original OPD	**0.08920**	0.14277	0.11599	0.14574
PDM	0.20155	0.20651	0.20403	0.09488
Teacher-null	0.10255	0.11530	0.10893	0.09980
MCReg	0.10363	**0.05787**	**0.08075**	**0.04569**
DanceOPD				
Original OPD	**0.08995**	0.13636	0.11316	0.14309
PDM	0.13506	0.08669	0.11087	**0.04055**
Teacher-null	0.11557	0.15550	0.13553	0.09543
MCReg	0.10007	**0.08583**	**0.09295**	0.05523

The multi-reward teacher has $R_{\mathrm{abs}} = 0.18179$ with 95% CI $[0.18099, 0.18257]$. Evaluation protocol: Section D.2.

Table 6: **Paired changes in prediction risk.** Entries are MCReg minus the indicated student comparator, with 95% confidence intervals. Negative values indicate lower risk for MCReg.

Risk	Original OPD	PDM	Teacher-null
DiffusionOPD			
R_C	$0.0144\ [0.0143, 0.0146]$	$-0.0979\ [-0.0982, -0.0976]$	$0.0011\ [0.0010, 0.0011]$
R_N	$-0.0849\ [-0.0856, -0.0841]$	$-0.1486\ [-0.1491, -0.1481]$	$-0.0574\ [-0.0595, -0.0554]$
JER	$-0.0352\ [-0.0356, -0.0349]$	$-0.1233\ [-0.1236, -0.1230]$	$-0.0282\ [-0.0293, -0.0271]$
R_{abs}	$-0.1001\ [-0.1007, -0.0993]$	$-0.0492\ [-0.0497, -0.0487]$	$-0.0541\ [-0.0561, -0.0522]$
DanceOPD			
R_C	$0.0101\ [0.0099, 0.0103]$	$-0.0350\ [-0.0357, -0.0343]$	$-0.0155\ [-0.0161, -0.0149]$
R_N	$-0.0505\ [-0.0544, -0.0461]$	$-0.0009\ [-0.0038, 0.0021]$	$-0.0697\ [-0.0721, -0.0674]$
JER	$-0.0202\ [-0.0222, -0.0180]$	$-0.0179\ [-0.0192, -0.0165]$	$-0.0426\ [-0.0440, -0.0412]$
R_{abs}	$-0.0879\ [-0.0905, -0.0850]$	$0.0147\ [0.0119, 0.0176]$	$-0.0402\ [-0.0417, -0.0387]$

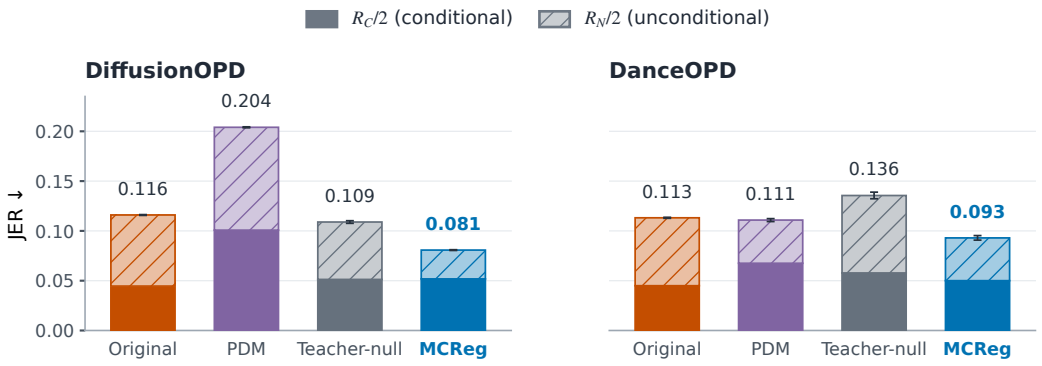

Figure 6: **JER branch contributions.** For the same students and evaluation as Table 5, each stack is $R_C/2 + R_N/2 = \mathrm{JER}$. Whiskers show paired-state 95% confidence intervals; lower is better.

Unconditional-target ablation. At the main-table guidance scale $w = 3$, Table 7 compares Teacher-null, which fits the teacher's unconditional prediction, with MCReg, which reuses the conditional teacher target. With the same 2,000-update budget, MCReg scores higher on six of eight metrics in DiffusionOPD and seven of eight in DanceOPD, including GenEval, OCR, PickScore, HPSv2.1, ImageReward, and UnifiedReward in both frameworks. These generation results complement the fixed-teacher JER analysis above.

Table 7: **Unconditional-target ablation on generation tasks.** Teacher-null and MCReg use the same 2,000-step OPD training and CFG $= 3$ evaluation protocol within each framework. Bold marks the higher value within each framework.

Method	GenEval ↑	OCR ↑	PickScore ↑	CLIP ↑	HPSv2.1 ↑	ImgRwd ↑	UniReward ↑	DeQA ↑
DiffusionOPD								
Teacher-null	0.967	0.965	23.615	**29.738**	33.352	1.518	0.680	**4.314**
MCReg	**0.977**	**0.973**	**23.843**	29.367	**33.996**	**1.544**	**0.684**	4.312
DanceOPD								
Teacher-null	0.961	0.956	23.407	**29.708**	33.271	1.505	0.676	4.254
MCReg	**0.970**	**0.965**	**23.711**	29.452	**34.272**	**1.552**	**0.683**	**4.330**

MCReg values are repeated from Table 3. GenEval is the mean of six category accuracies. PickScore is reported on its original scale; CLIPScore and HPSv2.1 are $100\times$ the stored scores.

Closure and teacher-relative error. DanceOPD illustrates the difference between self-closure and shared-target accuracy. MCReg achieves lower JER than PDM (0.09295 versus 0.11087) and higher DrawBench HPSv2.1 at $w = 3$ (34.272 versus 32.698), even though PDM has lower R_{abs} (0.04055 versus 0.05523). PDM's $R_{\mathrm{abs}}/4$ contribution is smaller by 0.00367, while its other teacher-relative errors are larger by 0.02159.

In DanceOPD, MCReg reduces the conditional-mean error $n^{-1}\mathbb{E}_{P_{\mathrm{eval}}}\|\mathbb{E}_{c\sim q_{\mathrm{DB}}}[C_\theta - C^\star]\|_2^2$ from PDM's 0.04244 to 0.00868, locating its shared-target advantage in the prompt-averaged conditional prediction. Eq. (17) shows why PDM's teacher-unconditional direction target can differ from the marginal of its conditional teacher target under student rollouts. This difference separates internal branch agreement from accuracy against the common teacher.

At the main inference scale $w = 3$, the same fixed-teacher evaluation compares $3C_\theta - 2N_\theta$ with $3C^\star - 2N^\star$. The per-coordinate squared prediction error is 1.04595 for MCReg, versus 1.35375 for Original OPD and 1.41856 for PDM in DiffusionOPD. In DanceOPD, the corresponding values are 1.00828, 1.34444, and 1.03086. This endpoint diagnostic measures prediction error against the common teacher; the guidance sweeps below measure generated-image quality.

E.2 ADDITIONAL RL COMPARISONS

Single-reward PickScore training. We also train SD3.5-Medium with DiffusionNFT using PickScore alone. The matched Original and MCReg ($\rho = 0.001$) checkpoints receive 2,000 updates and use the evaluation in Section D.1. At $w = 3$ and $w = 5$, MCReg has higher PickScore and HPSv2.1 on both prompt sets and higher DrawBench ImageReward (Table 8). Its PickScore advantage over Original grows from 0.336 to 0.571 on Pick-a-Pic and from 0.299 to 0.477 on DrawBench between these two guidance scales.

Table 8: **Single-reward PickScore training under guidance.** Matched SD3.5-Medium DiffusionNFT checkpoints after 2,000 updates. Both methods use the same inference guidance within each comparison. Higher is better for every metric.

CFG	Method	Pick-a-Pic ↑		DrawBench ↑		
		PickScore	HPSv2.1	PickScore	HPSv2.1	ImgRwd
$w = 3$	DiffusionNFT	23.252	33.083	23.582	32.260	1.379
	+ MCReg	**23.588**	**33.966**	**23.881**	**32.839**	**1.432**
$w = 5$	DiffusionNFT	22.753	32.418	23.178	31.672	1.387
	+ MCReg	**23.324**	**33.825**	**23.655**	**32.803**	**1.432**

Training optimizes PickScore alone; HPSv2.1 and ImageReward are evaluation metrics. MCReg uses $\rho = 0.001$. PickScore is on its original scale, and HPSv2.1 is $100\times$ its stored score.

Native SD3.5-Medium multi-reward performance. At $w = 1$, the 500-update DiffusionNFT MCReg checkpoint in Table 2 exceeds Original on all nine Pick-a-Pic and DrawBench metrics (Table 9). This comparison isolates the trained checkpoints before inference guidance is applied.

Table 9: **Native SD3.5-Medium multi-reward performance.** The 500-update DiffusionNFT checkpoints from Table 2 are evaluated at $w = 1$. Higher is better throughout.

Method	Pick-a-Pic ↑			DrawBench ↑					
	PickScore	CLIP	HPSv2.1	PickScore	CLIP	HPSv2.1	ImgRwd	UniReward	DeQA
DiffusionNFT	23.380	29.350	35.510	23.750	28.567	33.467	1.456	0.667	4.339
MCReg ($\rho = 0.03$)	**23.440**	**30.837**	**36.039**	**23.796**	**29.825**	**33.957**	**1.500**	**0.674**	**4.342**

Flow-GRPO-Fast multi-reward performance. We additionally train SD3.5-Medium with Flow-GRPO-Fast on Pick-a-Pic using the unit-weight sum of PickScore, CLIPScore, and HPSv2.1. Original and MCReg ($\rho = 0.003$) each receive 4,000 optimizer updates on 32 GPUs. Training uses $w = 1$, 10-step CPS rollouts, three replay timesteps, groups of 24 images per prompt, 576 rollout images per update, learning rate 3×10^{-4}, and no reference KL penalty. We evaluate the final EMA checkpoints with 40 sampling steps at 512×512 on 2,048 Pick-a-Pic test prompts and 200 DrawBench prompts with five samples each.

Table 10 shows that Original's PickScore and HPSv2.1 decrease under $w = 3$ on both prompt sets. At the same guidance scale, MCReg scores higher on seven of the nine reported metrics, including PickScore and HPSv2.1 on both sets and ImageReward, UnifiedReward, and DeQA on DrawBench.

Table 10: **Flow-GRPO-Fast multi-reward generation.** Final SD3.5-Medium checkpoints after 4,000 updates. The MCReg row uses $\rho = 0.003$; bold marks the best score among the displayed settings.

Method	Pick-a-Pic ↑			DrawBench ↑					
	PickScore	CLIP	HPSv2.1	PickScore	CLIP	HPSv2.1	ImgRwd	UniReward	DeQA
Original ($w = 1$)	**23.434**	30.369	36.652	**23.700**	29.088	34.540	1.458	0.636	4.019
Original ($w = 3$)	23.116	**31.746**	35.702	23.470	**30.239**	34.227	1.514	0.651	3.963
MCReg ($w = 3$)	23.335	31.529	**37.034**	23.665	30.064	**35.067**	**1.549**	**0.657**	**4.057**

PickScore is on its original scale (stored score $\times 26$); CLIPScore and HPSv2.1 are $100\times$ stored scores. ImageReward, UnifiedReward, and DeQA retain their stored scales.

E.3 Guidance sweeps

Main-figure protocol. Figure 5 uses the SD3.5-Medium DiffusionNFT checkpoints in Table 2 (500 updates, MCReg $\rho = 0.03$) and the DiffusionOPD and DanceOPD students in Table 3 (2,000 updates, MCReg $\rho = 0.1$). The OPD panels include PDM at the same training budget. All three performance curves use DrawBench. The closure panel uses the DiffusionNFT protocol in Section D.3 and the OPD evaluation law in Section D.2; risks are compared within each framework. Figure 1 retains its separately specified GenEval configurations.

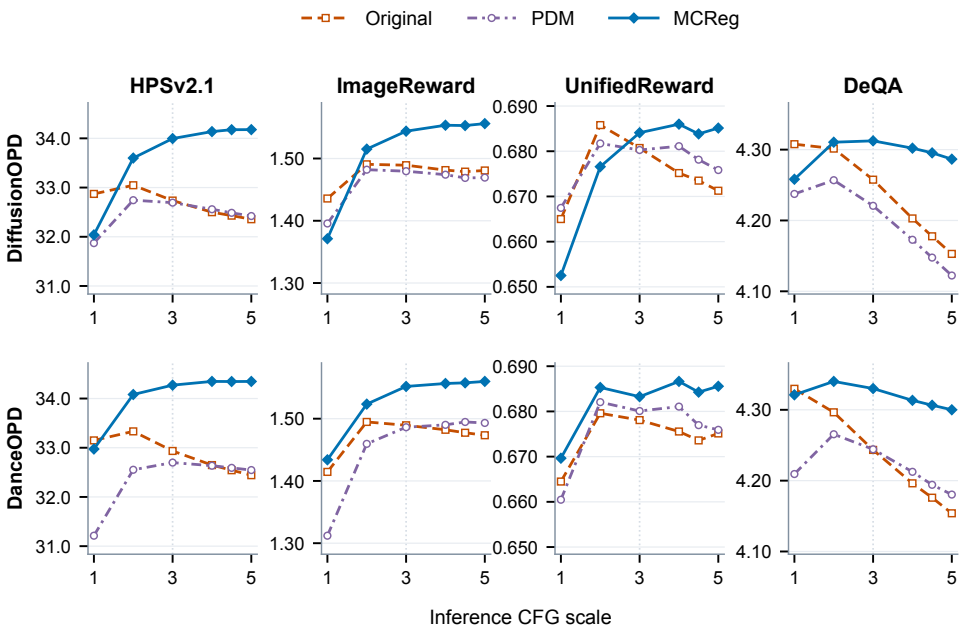

Figure 7: **DrawBench quality metrics under CFG.** HPSv2.1, ImageReward, UnifiedReward, and DeQA for the 2,000-update DiffusionOPD and DanceOPD checkpoints in Table 3. Each panel shows all six evaluated guidance scales for Original OPD, PDM, and MCReg ($\rho = 0.1$). Higher is better for every metric; vertical scales differ by metric.

DrawBench quality trajectories. Figure 7 traces four DrawBench quality metrics for the 2,000-update OPD checkpoints. MCReg exceeds Original at every measured $w \geq 2$ on HPSv2.1, ImageReward, and DeQA in both frameworks. Its HPSv2.1 margin widens from 0.553 to 1.821 in DiffusionOPD and from 0.749 to 1.905 in DanceOPD between $w = 2$ and $w = 5$. At $w = 5$, MCReg leads Original and PDM on all four plotted DrawBench metrics in both frameworks. Figure 8 shows three of these metrics for the 500-update DiffusionNFT checkpoints and OCR for its

separate 150-update single-reward checkpoints. Both figures cover all six measured CFG scales, with each panel using its matched evaluation task.

ImageReward, UnifiedReward, and DeQA are outside the PickScore/CLIPScore/HPSv2.1 reward used to train the multi-reward teacher. At $w = 5$, MCReg also leads Original and PDM on all three in both OPD frameworks, extending the quality trend beyond the teacher's reward components. At the same guidance scale, it exceeds Original on every metric in Figure 8, covering DrawBench quality and the separate OCR task.

Task scores at high guidance. Table 11 gives all eight task metrics at the largest evaluated guidance scale. MCReg has the highest value on seven metrics in DiffusionOPD and six in DanceOPD, extending the quality trajectories in Figure 7 to GenEval, OCR, PickScore, and CLIPScore.

Table 11: **OPD performance at CFG $w = 5$.** The 2,000-update checkpoints and evaluation sets match Table 3. Bold marks the highest value within each framework.

Method	Tasks ↑		DrawBench ↑					
	GenEval	OCR	PickScore	CLIP	HPSv2.1	ImgRwd	UniReward	DeQA
DiffusionOPD								
Original	0.964	0.966	23.180	29.985	32.356	1.481	0.671	4.153
PDM	0.954	0.931	23.134	**29.991**	32.421	1.469	0.676	4.122
MCReg	**0.972**	**0.972**	**23.721**	29.457	**34.177**	**1.556**	**0.685**	**4.287**
DanceOPD								
Original	0.963	**0.970**	23.134	**29.911**	32.442	1.473	0.675	4.154
PDM	0.940	0.897	23.135	29.871	32.546	1.493	0.676	4.180
MCReg	**0.968**	0.962	**23.597**	29.612	**34.347**	**1.560**	**0.686**	**4.300**

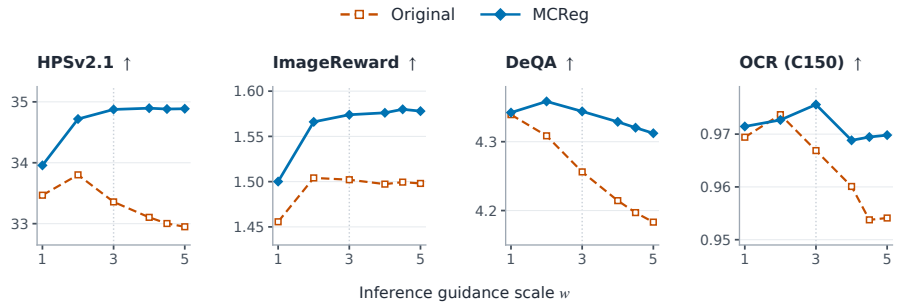

Figure 8: **DiffusionNFT guidance response on DrawBench and OCR.** The first three panels use the Original and MCReg ($\rho = 0.03$) SD3.5-Medium multi-reward checkpoints at 500 updates from Table 2; HPSv2.1 uses the paper's $100\times$ scale. The OCR panel uses the separate 150-update single-reward checkpoints in Table 1. All panels show six CFG scales. Higher is better.

E.4 Performance across OPD training budgets

Figure 9 follows the C1000, C2000, C3000, and C4000 checkpoints at $w = 3$. MCReg has higher DrawBench HPSv2.1 than Original and PDM at every recorded budget in both OPD frameworks. The C2000 points match the checkpoints in Table 3.

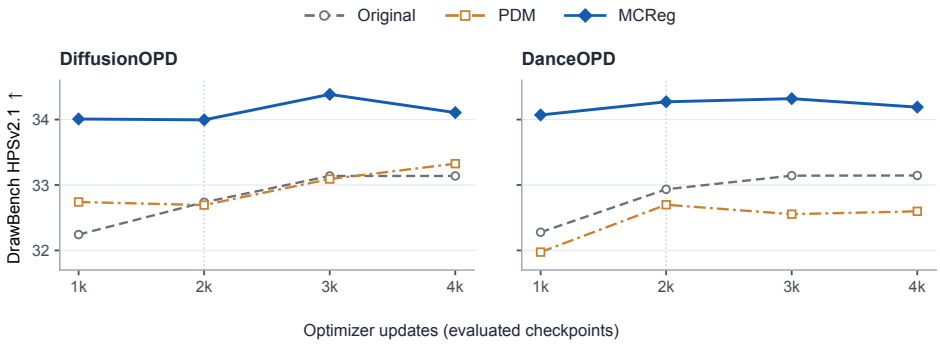

Figure 9: **DrawBench HPSv2.1 across OPD training budgets.** Original OPD, PDM, and MCReg ($\rho = 0.1$) are evaluated at four saved checkpoints with CFG $w = 3$ in each framework. The vertical guide marks the 2,000-update main-table checkpoint; higher is better.

E.5 Effect of regularization strength

Figure 10 shows the PickScore response of matched FLUX.2 Klein multi-reward checkpoints at 200 updates. Across the three evaluated low-strength settings, MCReg scores higher than Original on both Pick-a-Pic and DrawBench at every measured guided scale $w \geq 2$. The response at native inference ($w = 1$) is mixed, so this ablation supports a guided PickScore gain over the tested strength range.

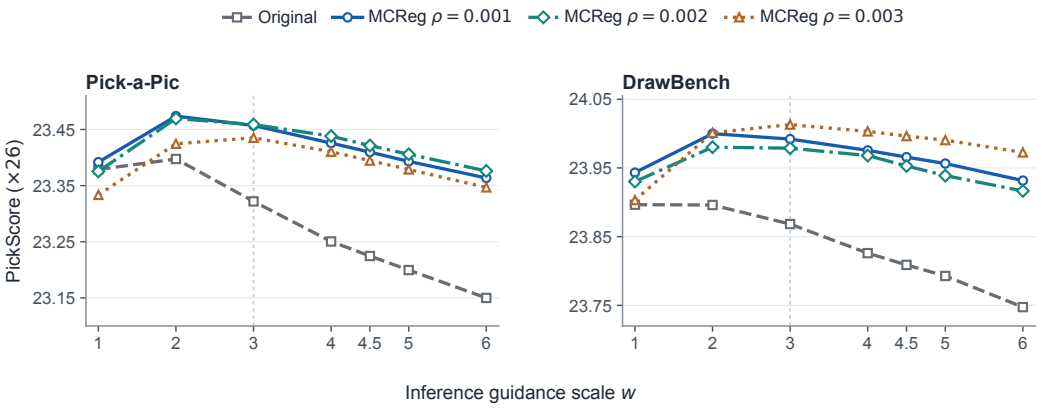

Figure 10: **PickScore response to MCReg strength in FLUX.2 Klein 9B multi-reward DiffusionNFT.** Matched seed-42 Original and MCReg ($\rho \in \{0.001, 0.002, 0.003\}$) checkpoints after 200 updates, evaluated on 2,048 Pick-a-Pic test prompts and 200 DrawBench prompts with five samples each. Curves join the seven measured guidance scales, $w \in \{1, 2, 3, 4, 4.5, 5, 6\}$, using 40 sampling steps at 512×512 resolution; the dotted guide marks $w = 3$. PickScore is the raw scorer output multiplied by 26; higher is better.

Table 12 gives the exact native and main-guidance scores for these checkpoints. At $w = 3$, all three strengths exceed the matched Original on both prompt sets; the $\rho = 0.002$ row is the setting used in Table 2.

Table 12: **FLUX.2 Klein multi-reward strength sweep at 200 updates.** PickScore (raw scorer output $\times 26$) on the 2,048-prompt Pick-a-Pic test split and 200 DrawBench prompts with five samples each, at native ($w = 1$) and main ($w = 3$) guidance. Higher is better; bold marks the highest score in each column.

Method	ρ	Pick-a-Pic ↑		DrawBench ↑	
		$w = 1$	$w = 3$	$w = 1$	$w = 3$
Original	–	23.3787	23.3219	23.8965	23.8683
MCReg	0.001	**23.3918**	23.4571	**23.9432**	23.9920
MCReg	0.002	23.3754	**23.4588**	23.9302	23.9787
MCReg	0.003	23.3335	23.4352	23.9032	**24.0130**

DrawBench strength ablation. Figure 11 reports the complete nine-strength DanceOPD sweep at 1,000 updates and $w = 3$. The choice of ρ has a sizable effect on the DrawBench HPSv2.1 gain: MCReg scores range from 33.468 to 34.101, versus 32.278 for Original. The main setting $\rho = 0.1$ reaches 34.071 in this sweep.

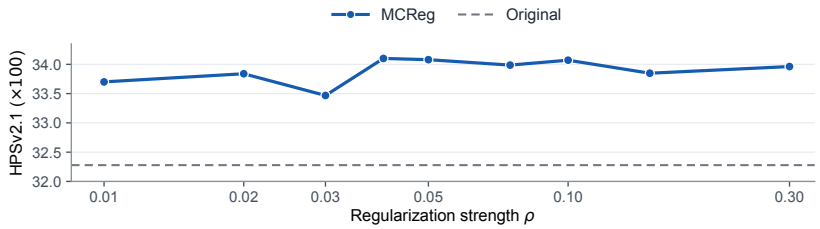

Figure 11: **Effect of ρ on DrawBench HPSv2.1.** DanceOPD after 1,000 updates at $w = 3$. Dots mark all nine evaluated MCReg strengths; the dashed line is Original. The ρ axis is logarithmic.

E.6 Guided single-reward PickScore response

Figure 12 extends Table 8 across the measured guided range. For both Pick-a-Pic and DrawBench, PickScore declines more slowly for MCReg than for Original as w increases from 2 to 5, and MCReg scores higher at every shown scale.

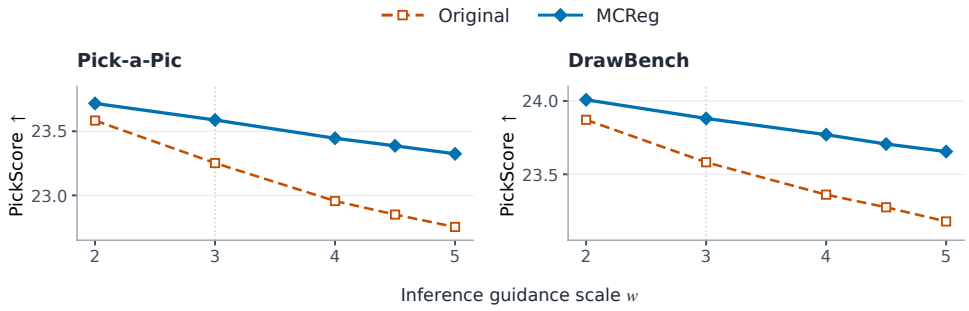

Figure 12: **Guided PickScore after single-reward training.** The matched 2,000-update SD3.5-Medium DiffusionNFT checkpoints from Table 8 are evaluated at $w \in \{2, 3, 4, 4.5, 5\}$. The plot connects measured scores; the dotted guide marks the main inference setting $w = 3$. Higher is better.

E.7 ADDITIONAL QUALITATIVE COMPARISONS

Figures 13 and 14 extend Figure 4 with selected DrawBench outputs from the 2,000-update OPD checkpoints. Each row matches prompt, sample ID, and initial noise across Original ($w = 1, 3$), PDM ($w = 3$), and MCReg ($w = 3$). MCReg has the highest per-image HPSv2.1, PickScore, and ImageReward among the four displayed settings for these rows; Table 3 reports aggregate test-set results. The DiffusionOPD examples cover color attributes, a multi-person scene, rendered text, and object design.



Figure 13: **Selected DiffusionOPD DrawBench examples.** All columns use the 2,000-update checkpoints from Table 3, with 40 sampling steps at 512×512 resolution. The rows show a rainbow-coloured penguin (a), an Egyptian painting of an argument over taking out trash (b), fireworks spelling "Google Research Pizza Cafe" above a skyline (c), and a pig-shaped airship over a wheat field (d).

Figure 14 gives the same four-way comparison for DanceOPD. The rows test object attributes, storefront text, poster typography, and an object used as an umbrella. In row (c), MCReg renders "Climate Change" correctly, while the Original and PDM images omit or alter letters.

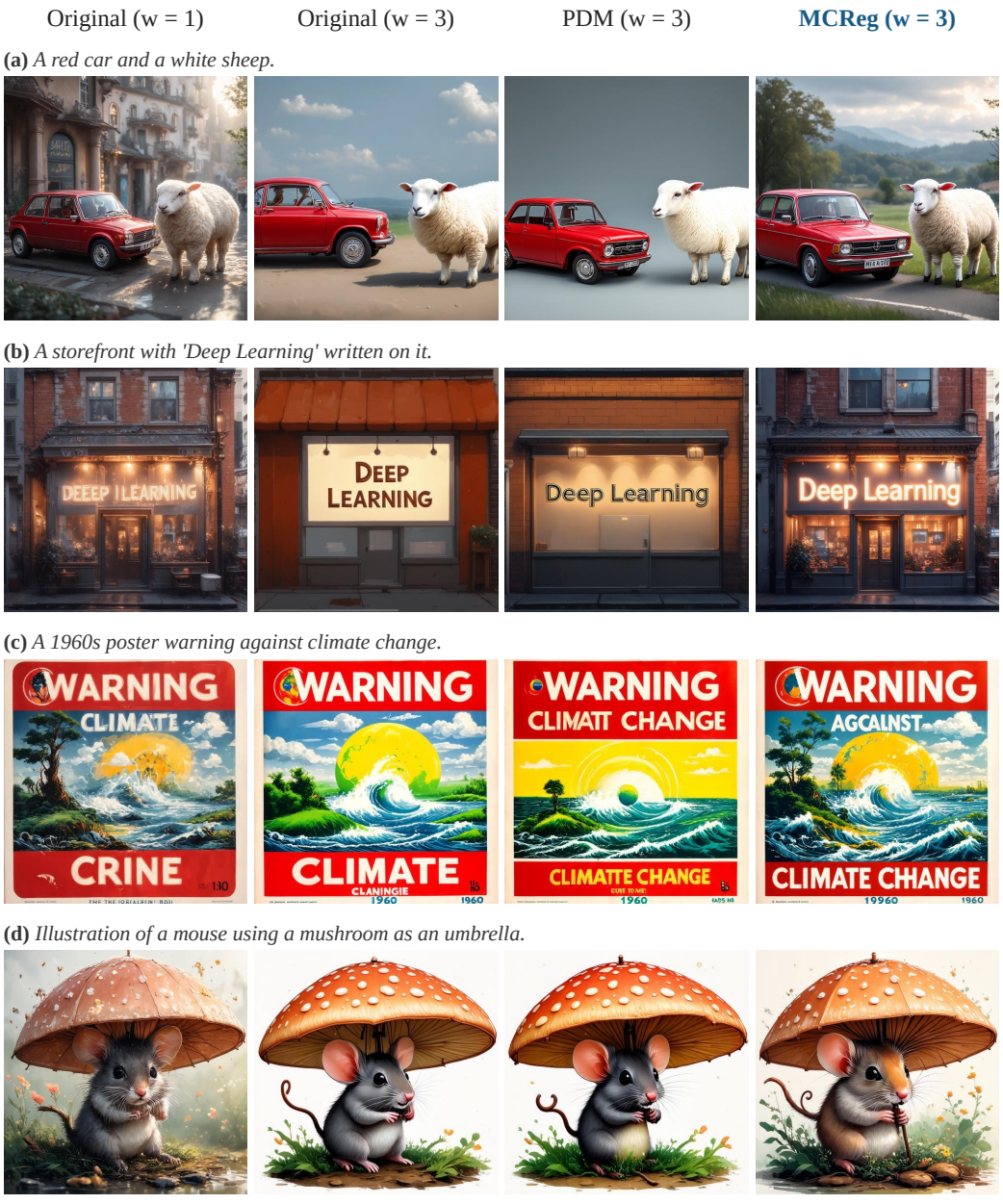

Figure 14: **Selected DanceOPD DrawBench examples.** The columns match Figure 13 and use the DanceOPD 2,000-update checkpoints from Table 3. Each row shares the full prompt and initial noise across Original, PDM, and MCReg. Row (c) depicts a 1960s poster warning against climate change; row (d) depicts a mouse using a mushroom as an umbrella.

Figure 15 extends the DiffusionNFT side of Figure 4. The first two rows use the SD3.5-Medium multi-reward checkpoints, while the third uses FLUX.2 Klein 9B. Within each row, the three columns use the same prompt and sample index. The examples compare object form and text rendering, including the spelling of "Toronto" in the fireworks in row (c).

Original (w = 1) Original (w = 3) **MCReg (w = 3)**

(a) *SD3.5-Medium: Lego Arnold Schwarzenegger.*



(b) *SD3.5-Medium: An old photograph of a 1920s airship shaped like a pig, floating over a wheat field.*



(c) *FLUX.2 Klein 9B: New York Skyline with 'Google Brain Toronto' written with fireworks on the sky.*



Figure 15: **Selected DiffusionNFT DrawBench examples.** Rows (a,b) use SD3.5-Medium at 500 updates (MCReg $\rho = 0.03$); row (c) uses FLUX.2 Klein 9B at 200 updates (MCReg $\rho = 0.002$). The columns show Original at $w = 1$, Original at $w = 3$, and MCReg at $w = 3$. The prompts specify a Lego portrait (a), a pig-shaped airship (b), and fireworks spelling "Google Brain Toronto" above a skyline (c).

Figure 16 follows Original images from $w = 1$ through $w = 5$ for four selected DrawBench prompts, alongside MCReg at $w = 3$. The storefront and fireworks examples highlight text rendering as guidance increases, while the statue example shows changes in object detail. Across these selected rows, MCReg has higher HPSv2.1, PickScore, and ImageReward than all three Original settings. Test-set guidance curves for the DanceOPD and SD3.5-Medium checkpoints appear in Figures 7 and 8.



Figure 16: **Selected DrawBench examples across guidance scales.** Columns show Original at $w = 1, 3, 5$ and MCReg at $w = 3$. Row (a) uses DanceOPD at 2,000 updates ($\rho = 0.1$); row (b) uses SD3.5-Medium DiffusionNFT at 500 updates ($\rho = 0.03$); rows (c,d) use FLUX.2 Klein 9B DiffusionNFT at 200 updates ($\rho = 0.002$). All images use 40 sampling steps and 512×512 resolution. Prompts and sample IDs match within each row.

Figure 17 shows selected samples from the separate, single-task DiffusionNFT GenEval experiment at 1,000 updates. The three prompts test color and object-attribute binding. The settings share each row's prompt and sample ID. MCReg receives strict GenEval accuracy 1 on each displayed sample, versus 0 for both displayed Original settings; Table 1 reports aggregate category accuracies over the full test set.



Figure 17: **Selected DiffusionNFT GenEval examples.** SD3.5-Medium Original and MCReg use 1,000-update single-task checkpoints; MCReg uses $\rho = 0.01$. Columns show Original at $w = 1$, Original at $w = 3$, and MCReg at $w = 3$. The selected rows cover an orange handbag with a green carrot, a white banana with a black elephant, and a purple carrot. Each row shares its prompt and sample ID.

Figure 18 turns to the OCR task within the multi-task DiffusionOPD evaluation. Row labels quote the target strings; the full prompts place them on a city monument, an arcade screen, and an airport display, respectively. The Original images contain missing or substituted characters, while MCReg receives an OCR score of 1.0 on each selected case.



Figure 18: **Selected DiffusionOPD OCR examples.** The columns use the same 2,000-update Original and MCReg checkpoints as Table 3. Each row shares its full prompt, sample ID, and initial noise; $w = 1$ and $w = 3$ are the inference guidance scales. The quoted target text is abbreviated from each full scene prompt for readability.