DIFFUSIONRVGE: REINFORCEMENT LEARNING VIA REGRESSION-BASED VALUE GRADIENT ESTIMATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Reinforcement learning improves diffusion models using rewards on generated images, but repeated denoising and gradient estimation make training expensive. We introduce DiffusionRVGE, an efficient method that constructs useful update directions by regressing centered rewards on centered final denoised images. It requires neither reward derivatives nor adjoint backpropagation through the sampling trajectory. A unified first-variation formulation identifies the value-gradient target with or without a trajectory-KL penalty. A population analysis under Gaussian noising explains the regression geometry, while a subspace analysis motivates clean-image coordinates. The implementation adapts rollout bandwidth and query weights to reward continuity: local rollouts suit smooth rewards, whereas broader groups provide informative contrasts for discrete rewards. Trained jointly on five rewards, one SD3.5-Medium model matches or exceeds DiffusionNFT's joint model on seven of eight metrics, reaching GenEval 0.94, OCR 0.92, and PickScore 23.82 at about one fifth of its estimated training compute. In single-reward training, DiffusionRVGE reaches matched reward thresholds 2.1–$6.7\times$ faster than reproduced DiffusionNFT on identical hardware. Ablations support the clean-image estimator, direct direction transfer, and reward-dependent rollout choices. Our code is provided in the supplementary material.

1 INTRODUCTION

Diffusion and flow matching models generate images by transforming noise into data (Ho et al., 2020; Song et al., 2021; Lipman et al., 2023; Esser et al., 2024). Reinforcement learning (RL) adapts these generators to scalar rewards measuring preferences, text accuracy, or compositional correctness. Methods including DDPO, Flow-GRPO, and DiffusionNFT improve generation through reward-based fine-tuning (Black et al., 2023; Liu et al., 2025; Zheng et al., 2026). Training remains expensive: each update requires image generation and an estimate of how model changes affect the final reward. Our aim is to obtain effective update directions at substantially lower training cost.

We introduce DiffusionRVGE, which constructs reward-informed update directions by regression on final denoised images. To motivate the estimator, consider the Gaussian noising path

$$\mathbf{X}_t = \alpha_t \mathbf{Z} + \sigma_t \boldsymbol{\epsilon}, \qquad \mathbf{Z} \sim \rho, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0, I), \tag{1}$$

where $\mathbf{Z}$ and $\boldsymbol{\epsilon}$ are independent, $\mathbf{Z}$ is a clean image or latent, and generation reverses the noising process. Let $P_{\boldsymbol{\theta}}$ denote the law of a stochastic denoising trajectory. A common RL objective, $\mathbb{E}_{P_{\boldsymbol{\theta}}} R(\mathbf{Z}) - \tau \,\mathrm{KL}(P_{\boldsymbol{\theta}} \| P_{\mathrm{ref}})$, balances reward against deviation from a reference model. For $\tau > 0$, under an independent-noise initialization, its optimal-control formulation identifies a reward-tilted terminal target $\pi^* \propto \rho e^{R/\tau}$. The policy first variation gives a common cross-term plus quadratic-penalty loss for every $\tau \geq 0$, including unregularized reward optimization. This supplies a target for efficient approximation, rather than requiring differentiation through every denoising step.

Our estimator fits centered reward differences to clean-image contrasts using normalized ridge regression. Under exact conditioning in the Gaussian noising model, the population normal equations connect clean-image regression to a covariance-preconditioned value gradient. A subspace analysis characterizes how denoised coordinates improve directional estimation, and our alignment experiments demonstrate this advantage in the evaluated models. We transfer the fitted directions directly to model updates without an adjoint backward pass.

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.

Regression uses reward values and benefits from informative variation across the rollout bank. Smooth scores permit local perturbations; discrete scores can remain constant across these perturbations and benefit from independent-seed groups. We therefore adapt rollout bandwidth and query weights to reward continuity while retaining the same centered squared-regression estimator. This gives one practical method for both continuous preferences and discrete task scores.

Our contributions are:

- **Efficient regression-based updates.** Clean-image regression constructs update directions without reward derivatives or adjoint backpropagation, guided by gradient matching and supported by directional analysis and empirical validation.
- **Reward-adaptive implementation.** Local trajectory banks and broader rollout groups share a centered regression formulation, with reward-dependent query weights validated by ablations.
- **Strong results at lower compute.** Our joint model matches or exceeds DiffusionNFT on seven of eight metrics at about one fifth of its estimated training compute. Single-reward runs reach matched thresholds 2.1–$6.7\times$ faster on identical hardware.

2 Related work

RL for diffusion and flow models. DDPO and Flow-GRPO optimize stochastic denoising policies (Black et al., 2023; Liu et al., 2025); DiffusionNFT uses positive and negative generations in forward-process matching (Zheng et al., 2026). DPOK, Diffusion-DPO, DanceGRPO, and Centered Reward Distillation provide other reward- or preference-based objectives (Fan et al., 2023; Wallace et al., 2023; Xue et al., 2025; Zhu et al., 2026). A path-space perspective identifies value-gradient estimation as a shared component (Xu et al., 2026). DRaFT differentiates rewards through sampling, whereas FDFO transfers paired reward differences into velocity updates (Clark et al., 2023; McAllister et al., 2026). Our joint regression accounts for correlations among clean-image differences. Temporal and stepwise methods change branching or credit assignment (He et al., 2025; Savani et al., 2026; Zhou et al., 2025; Chen et al., 2025); our bandwidth choices address reward continuity and discreteness.

Regression and controlled dynamics. Adjoint Matching learns stochastic-control targets through adjoint regression (Domingo-Enrich et al., 2024). Reward Score Matching and path-space analyses relate diffusion RL to reward-conditioned score or value gradients (Lee et al., 2026; Xu et al., 2026). Our analysis distinguishes ordinary reward regression from the soft-value oracle of trajectory-KL control. Dimension-dependent derivative-free estimation and subspace-guided search (Nesterov & Spokoiny, 2017; Maheswaranathan et al., 2019) motivate the clean-coordinate comparison.

3 Gradient matching with clean-image regression

We first identify the RL update target, then study the geometry of clean-image regression. Population and subspace analyses explain the role of preconditioning and the choice of coordinates; the corresponding proofs and technical conditions are in Appendices A and B.

3.1 The gradient-matching objective

Use generation time $s = T - t$ and a reference diffusion with drift $\mathbf{b}_{\mathrm{ref}}$ and noise coefficient $\gamma_s > 0$. A learned correction $\mathbf{u}_{\boldsymbol{\theta}}$ defines

$$\mathrm{d}\mathbf{Y}_s = (\mathbf{b}_{\mathrm{ref}} + \mathbf{u}_{\boldsymbol{\theta}})(\mathbf{Y}_s, s)\mathrm{d}s + \gamma_s \mathrm{d}\mathbf{B}_s, \qquad J_\tau = \mathbb{E}_{P_{\boldsymbol{\theta}}} R(\mathbf{Y}_T) - \tau\,\mathrm{KL}(P_{\boldsymbol{\theta}} \| P_{\mathrm{ref}}). \tag{2}$$

For $\tau > 0$, tilting reference paths by $e^{R/\tau}$ gives a law Q^* with terminal distribution $\pi^* \propto e^{R/\tau}\rho$ and

$$J_\tau = \tau \log Z_\tau - \tau\,\mathrm{KL}(P_{\boldsymbol{\theta}} \| Q^*), \qquad Z_\tau = \mathbb{E}_\rho e^{R/\tau}. \tag{3}$$

Under the independent-noise initialization, this identifies the desired terminal target. Define the soft value $W_t = \tau \log H_t$, where $H_t(\mathbf{x}) = \mathbb{E}_{\mathrm{ref}}[e^{R(\mathbf{Z})/\tau} \mid \mathbf{X}_t = \mathbf{x}]$.

Theorem 1 (Gradient matching for trajectory-KL RL). *For $\tau > 0$, reference and controlled processes with the same initial law and diffusion coefficient satisfy, under the regularity conditions in Appendix A,*

$$C_\tau - J_\tau = \frac{\tau}{2}\mathbb{E}_{P_{\boldsymbol{\theta}}} \int_0^T \frac{\|\mathbf{u}_{\boldsymbol{\theta}} - \mathbf{u}_s^*\|^2}{\gamma_s^2} \mathrm{d}s, \qquad \mathbf{u}_s^* = \frac{\gamma_s^2}{\tau} \nabla W_{T-s}, \tag{4}$$

where $C_\tau = \mathbb{E}[W_T(\mathbf{Y}_0)]$ is independent of $\boldsymbol{\theta}$.

For a common local update valid for every $\tau \geq 0$, use the continuation value of the frozen collection policy $\boldsymbol{\theta}_{\mathrm{old}}$:

$$F_t^{\boldsymbol{\theta}_{\mathrm{old}},\tau}(\mathbf{x}) = \mathbb{E}_{P_{\boldsymbol{\theta}_{\mathrm{old}}}} \left[R(\mathbf{Y}_T) - \frac{\tau}{2} \int_{T-t}^T \frac{\|\mathbf{u}_{\boldsymbol{\theta}_{\mathrm{old}}}(\mathbf{Y}_s, s)\|^2}{\gamma_s^2} \mathrm{d}s \,\middle|\, \mathbf{Y}_{T-t} = \mathbf{x} \right]. \tag{5}$$

Equivalently, $F_t^{\boldsymbol{\theta}_{\mathrm{old}},\tau} = V_t^{\boldsymbol{\theta}_{\mathrm{old}}} - \tau C_t^{\boldsymbol{\theta}_{\mathrm{old}}}$, where $V_t^{\boldsymbol{\theta}_{\mathrm{old}}}$ is expected terminal reward and $C_t^{\boldsymbol{\theta}_{\mathrm{old}}}$ is expected future control energy. Thus $F_t^{\boldsymbol{\theta}_{\mathrm{old}},0} = V_t^{\boldsymbol{\theta}_{\mathrm{old}}}$; at the optimal policy for $\tau > 0$, $F_t^{*,\tau} = W_t$. The same value definition covers both cases.

For the velocity parameterization $\mathbf{b}_{\boldsymbol{\theta}} = A_t \mathbf{x} - 2\mathbf{v}_{\boldsymbol{\theta}}$ with $A_t = \dot{\alpha}_t/\alpha_t$, write $\delta\mathbf{v}_{\boldsymbol{\theta}} = \mathbf{v}_{\boldsymbol{\theta}} - \mathbf{v}_{\mathrm{ref}}$ and $\nu_t^2 = \gamma_{T-t}^2$. Then $\mathbf{u}_{\boldsymbol{\theta}} = -2\delta\mathbf{v}_{\boldsymbol{\theta}}$. The first variation of J_τ gives the detached loss

$$\mathcal{L}_{\mathrm{GM},\tau} = \mathbb{E}_{\mu_{\boldsymbol{\theta}_{\mathrm{old}}}} \left[w_t \left(\underbrace{2\langle \delta\mathbf{v}_{\boldsymbol{\theta}}, \mathrm{sg}[\nabla F_t^{\boldsymbol{\theta}_{\mathrm{old}},\tau}] \rangle}_{\text{value-gradient update}} + \underbrace{\frac{2\tau}{\nu_t^2} \|\delta\mathbf{v}_{\boldsymbol{\theta}}\|^2}_{\text{trajectory-KL penalty}} \right) \right], \quad \tau \geq 0. \tag{6}$$

With time-sampling compensation w_t, its gradient at $\boldsymbol{\theta} = \boldsymbol{\theta}_{\mathrm{old}}$ equals $-\nabla_{\boldsymbol{\theta}} J_\tau|_{\boldsymbol{\theta}_{\mathrm{old}}}$ (Appendix A.1). Here sg stops gradients. The quadratic term is simply omitted at $\tau = 0$; the remaining cross term follows directly from the first variation. This is a local gradient identity, whereas Theorem 1 characterizes the positive-penalty optimum. Local regression on query-state perturbations estimates $\nabla F_t^{\boldsymbol{\theta}_{\mathrm{old}},\tau}$ from sampled rewards minus future control costs (Appendix A.1); the latter are absent when $\tau = 0$.

3.2 Population interpretation and directional analysis

We study population regression under the exact conditional law induced by the Gaussian noising model in Eq. (1). Let $V_t(\mathbf{x}) = \mathbb{E}[R(\mathbf{Z}) \mid \mathbf{X}_t = \mathbf{x}]$ be its ordinary value. At the reference policy, $\mathbf{u}_{\boldsymbol{\theta}_{\mathrm{old}}} = 0$ and $F_t^{\boldsymbol{\theta}_{\mathrm{old}},\tau} = V_t$ for every $\tau \geq 0$. Write $\mathbf{m}_t = \mathbb{E}[\mathbf{Z} \mid \mathbf{x}]$, $\Sigma_t = \mathrm{Cov}(\mathbf{Z} \mid \mathbf{x})$, $\mathbf{c}_t = \mathrm{Cov}(\mathbf{Z}, R \mid \mathbf{x})$, and $\kappa_t = \alpha_t/\sigma_t^2$. We fit centered rewards to centered clean images:

$$\mathbf{g}_t = \kappa_t \arg\min_{\mathbf{u}} \mathbb{E}\Big[\big(R(\mathbf{Z}) - V_t - \mathbf{u}^\top (\mathbf{Z} - \mathbf{m}_t) \big)^2 \mid \mathbf{X}_t = \mathbf{x} \Big], \tag{7}$$

choosing the minimum-norm coefficient.

Proposition 2 (Population geometry of clean-image regression). *At $\alpha_t, \sigma_t > 0$, for Gaussian corruption with finite conditional moments and justified differentiation under the conditional integral,*

$$\underbrace{\nabla_{\mathbf{x}} V_t = \kappa_t \mathbf{c}_t}_{\textit{value gradient}}, \qquad \underbrace{\mathbf{g}_t = \kappa_t \Sigma_t^\dagger \mathbf{c}_t = \Sigma_t^\dagger \nabla_{\mathbf{x}} V_t}_{\textit{preconditioned direction}}. \tag{8}$$

No derivative of R is needed. The population direction has positive inner product with ∇V_t whenever the latter is nonzero.

For an affine reward with slope $\mathbf{w}$ on the identifiable clean subspace, $\mathbf{c}_t = \Sigma_t \mathbf{w}$ while the population regression coefficient $\mathbf{g}_t/\kappa_t$ recovers $\mathbf{w}$. Thus regression accounts for covariance scaling of affine reward slopes on that subspace. For $\tau > 0$, the optimal soft value has the analogous covariance formula with $e^{R/\tau}$; its ordinary-to-soft-value residual is discussed in Appendix C.

Directional benefit of clean-image coordinates. In an isotropic Gaussian clean-subspace model of dimension $k < d$ with an affine reward, consider the same $m \geq 2$ Gaussian perturbations and mean-centered observations. With $n = m - 1$ independent contrasts, the unregularized minimum-norm fits satisfy

$$\mathbb{E}\cos^2(\widehat{\mathbf{h}}_{\text{noisy}}, \nabla V_t) = \min(n/d, 1), \qquad \mathbb{E}\cos^2(\widehat{\mathbf{g}}_{\text{clean}}, \nabla V_t) = \min(n/k, 1). \tag{9}$$

In this model, clean-image regression identifies the useful direction from fewer observations. The subspace comparison, proved in Appendix B.1, motivates the choice of clean-image coordinates; Section 6.1 evaluates their directional alignment in the trained model. Ridge regularization additionally bounds coefficient sensitivity even for a singular design (Appendix B.2). Directly using the fitted clean-image direction avoids an adjoint backward pass; Section 6.2 compares it with adjoint transport.

A supplementary idealized analysis of Langevin-corrected sampling under additional score-error and mixing assumptions is provided in Appendix C.

4 Reward-adaptive rollout and regression

DiffusionRVGE constructs update directions by regressing reward differences on clean-image contrasts. The rollout bank determines which contrasts enter the fit, and centered ridge regression combines them into a direction for each query. We adapt rollout bandwidth to the reward's variation and use reward-dependent query weights to determine where the fitted directions are applied.

Reward-dependent rollout bandwidth. For smooth rewards, small-noise trajectory banks capture local reward variation by reusing denoised completions along a stochastic rollout. For discrete or nearly piecewise-constant rewards, independent-seed groups provide broader contrasts across reward plateaus. Both constructions produce a bank $\mathcal{B}_i$ of clean images and rewards for query i. Appendix F specifies the observation sets and rollout schedules.

Centered clean-image regression. We form clean-image and reward contrasts using rollout-dependent centers. Trajectory banks use the query's own ODE completion and its reward; independent-seed groups use leave-one-out group means (Section 5.1). Denoting these centers by $\widehat{\mathbf{m}}_i$ and $\widehat{\bar{r}}_i$, we define

$$\mathbf{d}_{ij} = \mathbf{z}_j - \widehat{\mathbf{m}}_i, \qquad b_{ij} = r_j - \widehat{\bar{r}}_i, \qquad j \in \mathcal{B}_i. \tag{10}$$

For each nonzero displacement, divide the feature and label by the same length $\ell_{ij} = \|\mathbf{d}_{ij}\|$. Stack $\mathbf{d}_{ij}^\top/\ell_{ij}$ in Q_i and b_{ij}/ℓ_{ij} in $\mathbf{b}_i$. The coefficient and its scaled regression direction are

$$\widehat{\mathbf{u}}_i = \arg\min_{\mathbf{u}} \left\{ \|Q_i\mathbf{u} - \mathbf{b}_i\|^2 + \lambda_{\text{sum}}\|\mathbf{u}\|^2 \right\}, \qquad \widehat{\mathbf{g}}_{i,t} = \kappa_t \widehat{\mathbf{u}}_i, \tag{11}$$

with $\lambda_{\text{sum}} = 0.3$ in the reported experiments. A small solve in the number of observations gives

$$\widehat{\mathbf{u}}_i = Q_i^\top (Q_i Q_i^\top + \lambda_{\text{sum}} I)^{-1} \mathbf{b}_i. \tag{12}$$

Centers are computed before removing zero-displacement rows; an empty fit returns zero. Constant rewards also give zero direction. Normalization and ridge stabilize the finite fit; we do not normalize the fitted direction again.

Reward-dependent query weights. Weights a_i act on the policy update after fitting. Trajectory banks use $a_i = 1$; the GenEval recipe selects the six lowest-reward queries per prompt. All other clean images, including full-score images, remain regression observations. Section 6.4 compares this rule with continuous saturation weights and other selections. Query weighting does not change the ridge coefficient.

Policy update without trajectory backpropagation. Following Eq. (6), we insert the scaled regression direction $\chi_t \widehat{\mathbf{g}}_{i,t}$ into the value-gradient term, where $\chi_t = \sigma_t^2/\alpha_t^2$. With stopped query weights and a shared normalization $Z_a = \mathbb{E}_{\mu_{\boldsymbol{\theta}_{\text{old}}}}[a_i] > 0$, this yields

$$\mathcal{L}_\tau = \frac{1}{\text{sg}[Z_a]} \mathbb{E}_{\mu_{\boldsymbol{\theta}_{\text{old}}}} \left[a_i w_t \left(2\chi_t \langle \delta \mathbf{v}_{\boldsymbol{\theta}}, \text{sg}[\widehat{\mathbf{g}}_{i,t}] \rangle + \frac{2\tau}{\nu_t^2} \|\delta \mathbf{v}_{\boldsymbol{\theta}}\|^2 \right) \right]. \tag{13}$$

Next, center the cross term at $\mathbf{v}_{\boldsymbol{\theta}_{\text{old}}}$, the frozen policy that collected the current rollout bank. Replacing $\mathbf{v}_{\text{ref}}$ by $\text{sg}[\mathbf{v}_{\boldsymbol{\theta}_{\text{old}}}]$ in this term changes the loss only by a parameter-independent quantity, so the gradient is identical for every frozen batch. Using $\widehat{\mathbf{g}}_{i,t} = \kappa_t \widehat{\mathbf{u}}_i$ and $\omega_t = w_t \chi_t \kappa_t$ yields the equivalent centered loss

$$\mathcal{L}_\tau^{\text{ctr}} = \frac{1}{\text{sg}[Z_a]} \mathbb{E}_{\mu_{\boldsymbol{\theta}_{\text{old}}}} \Big[a_i \Big(2\omega_t \langle \mathbf{v}_{\boldsymbol{\theta}} - \text{sg}[\mathbf{v}_{\boldsymbol{\theta}_{\text{old}}}], \text{sg}[\widehat{\mathbf{u}}_i] \rangle + \frac{2\tau w_t}{\nu_t^2} \|\mathbf{v}_{\boldsymbol{\theta}} - \mathbf{v}_{\text{ref}}\|^2 \Big) \Big]. \tag{14}$$

The centered cross term is zero samplewise at $\boldsymbol{\theta} = \boldsymbol{\theta}_{\text{old}}$, removing loss fluctuations caused by the fixed-reference offset at the collection policy. Its gradient is preserved exactly; the quadratic penalty remains measured from $\mathbf{v}_{\text{ref}}$. Since the gradients agree samplewise, their covariance is unchanged. Only the velocity evaluation receives gradients: rollouts, rewards, and regression are detached, so no reward derivative or adjoint through denoising is required. At $\tau = 0$ the quadratic term is omitted, including for ODE rollouts; positive trajectory KL requires a nonzero diffusion coefficient. In practice we take $\tau = 0$ and set the time weight $\omega_t = t$, which we found to work best, normalized to unit mean over the sampling grid; Appendix F gives the discrete loss.

5 Main Experiments

5.1 Setup and Evaluation

We fine-tune SD3.5-Medium (Esser et al., 2024) at 512×512 with rank-32, scale-64 LoRA (Hu et al., 2021) and ten-step rollouts. PickScore (Kirstain et al., 2023) uses Pick-a-Pic training prompts. DiffusionNFT and FDFO are reproduced from their official code on the same $8\times$A800-40GB hardware. Compute is wall time multiplied by eight. Each rollout collection gives one parameter update through Algorithm 1; Appendix F gives the configurations. The two-seed PickScore comparison is reported in Appendix E.4.

Both the single-reward and five-reward comparisons use evaluation protocols and datasets aligned with DiffusionNFT for direct comparison: GenEval (Ghosh et al., 2023) uses 553 prompts with four images each; OCR uses 1,016 evaluated prompts from the released OCR test split with PaddleOCR scoring; and DrawBench (Saharia et al., 2022) uses 200 prompts with five images each for PickScore, ClipScore, HPSv2.1, Aesthetic, ImageReward, and UnifiedReward. All our models are trained and evaluated with CFG $= 1$. Evaluation uses a first-order ODE sampler with 40 steps, aligned with DiffusionNFT and Flow-GRPO. Published results are reported as released.

Each recipe defines the image and reward centers in Eq. (10) from its rollout construction.

PickScore: samples along one trajectory. We collect 24×24 trajectory-bank rollouts with $\eta = 0.03$ and fit each direction from clean completions along the same stochastic trajectory. The query state's own ODE completion and its reward serve as the image and reward centers. All query trajectories receive actor updates.

GenEval: samples across trajectories. We collect $K = 24$ independent ODE trajectories for each of 48 prompts and select the six lowest-reward trajectories for actor updates. Each selected query is fitted from the other 23 trajectories, whose mean clean image and mean reward serve as the regression centers. The query itself is excluded because its own reward selected it among the lowest six. The fitted direction is reused along its trajectory.

OCR. OCR combines graded credit for partially correct text with discrete recognition changes, placing it between smooth PickScore and plateaued GenEval. Both sampling schemes are effective in the ablation in Section 6.3. The main OCR-only run uses 8×12 trajectory-bank rollouts with $\eta = 0.5$; the group variant uses lowest-six actor selection and also supplies the OCR quality-floor run.

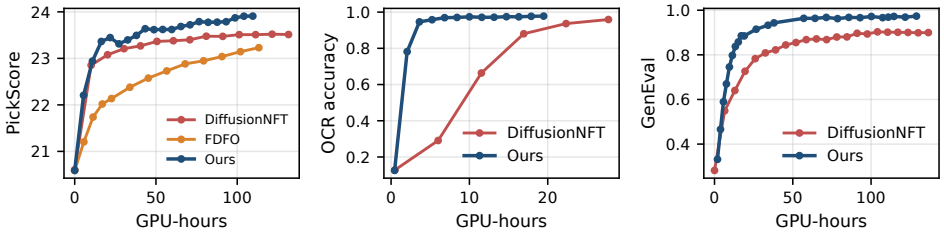


Figure 1: Single-reward learning on 8×A800 hardware: each curve optimizes the named reward. All curves use the aligned DrawBench 200×1, OCR 1,016, and official GenEval 553×4 datasets. FDFO appears only in the PickScore panel because FDFO does not evaluate the non-continuous OCR and GenEval rewards.

5.2 Joint optimization of five rewards

We train one model on the five rewards of DiffusionNFT's joint experiment—GenEval, OCR, PickScore, ClipScore, and HPSv2.1—and also evaluate Aesthetic, ImageReward, and UnifiedReward. Training alternates the two sampling schemes of Algorithm 1: group updates on mixed GenEval, OCR, and Pick-a-Pic prompts, and trajectory-bank updates on Pick-a-Pic prompts with the composite quality reward $q = \mathrm{PickScore}/26 + \mathrm{ClipScore} + \mathrm{HPSv2.1}$. On GenEval and OCR prompts, q enters through the group-relative floor of Eq. (43). Starting from SD3.5-Medium, the model uses 1,000 updates and 404 GPU-hours (Appendix F.1).

Table 1 gives the result. Our model matches DiffusionNFT's joint model on GenEval, exceeds it on OCR, PickScore, ClipScore, HPSv2.1, ImageReward, and UnifiedReward, and trails only on Aesthetic (5.95 vs. 6.01). DiffusionNFT does not report GPU-hours for this model; measured per-iteration runtime of its official code on our hardware implies at least 2,000 GPU-hours for the 1,600 forty-step iterations in its described stages (its table lists 1.7k), roughly five times our budget.

Table 1: Five-reward comparison on the metric panel of DiffusionNFT Table 1. Gray marks the rewards used to train each model. Baseline rows are published results and retain their published inference protocols; † denotes official checkpoints. DiffusionNFT lists 1.7k iterations for its five-stage joint model, whose described stages total 1,600 (Appendix F.1); ours uses 1,000 updates (404 GPU-hours). Bold marks the best value in each column.

Model	Iter.	GenEval	OCR	PickScore	ClipScore	HPSv2.1	Aesthetic	ImgRwd	UniRwd
SD3.5-M	–	0.63	0.59	22.34	0.285	0.279	5.36	0.85	3.03
+ FlowGRPO† (GenEval)	>5k	**0.95**	0.66	22.51	0.293	0.274	5.32	1.06	3.18
+ FlowGRPO† (OCR)	2k	0.66	**0.92**	22.41	0.290	0.280	5.32	0.95	3.15
+ FlowGRPO† (PickScore)	4k	0.54	0.68	23.50	0.280	0.316	5.90	1.29	3.37
+ DiffusionNFT (joint)	1.7k	0.94	0.91	23.80	0.293	0.331	**6.01**	1.49	3.49
+ Ours (joint)	1.0k	0.94	**0.92**	**23.82**	**0.303**	**0.339**	5.95	**1.55**	**3.50**

5.3 Single-reward results and learning efficiency

Table 2 reports all eight metrics for the single-reward checkpoints and a separate OCR quality-floor run. The single-reward models reach GenEval 0.98, OCR 0.98, and PickScore 23.99. The quality-floor run uses a composite training reward and is excluded from the single-reward speed comparison.

Figure 1 and Table 3 compare single-reward training time to matched reward thresholds, with each model optimized for the reward in its row. Clean-image regression is 2.1–2.6× faster than DiffusionNFT on PickScore, 6.2–6.7× on OCR, and 2.6–4.5× on GenEval.

FDFO (McAllister et al., 2026) does not evaluate discrete rewards such as OCR and GenEval, so Figure 1 compares it on PickScore only. Our reward-adaptive rollouts address such discrete rewards directly (Section 4). Even on the continuous PickScore reward, FDFO needs 109.3 GPU-hours to reach 23.20, versus 14.3 for our method, and stays below 23.50 throughout its 113-GPU-hour run.

Table 2: Single-reward results on the panel of DiffusionNFT Table 2, with a separate OCR quality-floor run. Gray marks the primary training reward; blue marks the three floor components (PickScore, ClipScore, HPSv2.1). Published rows retain their original protocols. Metric precision matches Table 1: two decimals for GenEval, OCR, PickScore, Aesthetic, ImageReward, and UnifiedReward; three for ClipScore and HPSv2.1.

Model	Reward	Iter. (GPU-h)	GenEval	OCR	PickScore	ClipScore	HPSv2.1	Aesthetic	ImgRwd	UniRwd
SD3.5-M	–	–	0.63	0.59	22.34	0.285	0.279	5.36	0.85	3.03
FlowGRPO	GenEval	4k	0.97	0.30	21.78	0.277	0.248	5.15	0.74	2.87
FlowGRPO	OCR	1k	0.66	0.96	21.94	0.280	0.257	5.18	0.31	2.86
FlowGRPO	PickScore	4k	0.54	0.60	23.62	0.257	0.295	6.42	1.17	3.17
DiffusionNFT	GenEval	1k	0.98	0.36	21.92	0.271	0.251	5.33	0.68	2.91
DiffusionNFT	OCR	150	0.54	0.97	21.63	0.281	0.246	5.19	0.37	2.81
DiffusionNFT	PickScore	2k	0.53	0.64	24.03	0.270	0.315	6.17	1.29	3.40
DiffusionNFT (repr.)	GenEval	210 (137)	0.90	0.42	22.03	0.276	0.256	5.20	0.62	2.99
DiffusionNFT (repr.)	OCR	50 (28)	0.25	0.96	20.94	0.254	0.206	4.94	-0.53	2.41
DiffusionNFT (repr.)	PickScore	260 (132)	0.63	0.56	23.50	0.266	0.299	6.07	1.19	3.24
Ours	GenEval	400 (129)	0.98	0.48	21.94	0.287	0.252	5.11	0.77	3.07
Ours	OCR	120 (19.5)	0.15	0.98	20.32	0.236	0.198	4.65	-0.93	2.00
Ours	PickScore	200 (110)	0.65	0.63	23.99	0.275	0.328	6.28	1.38	3.45
Ours	OCR + quality floor	150 (41)	0.51	0.97	22.17	0.276	0.292	5.64	0.87	3.02

Table 3: Measured GPU-hours for single-reward training to shared reward thresholds in Figure 1.

Reward	Target	Ours	DiffusionNFT	Speed-up
PickScore	23.20	14.3	29.7	2.1×
	23.50	38.2	98.5	2.6×
OCR	0.80	2.2	14.9	6.7×
	0.95	4.1	25.6	6.2×
GenEval	0.80	11.5	30.3	2.6×
	0.90	22.8	102.0	4.5×

Image quality beyond the trained reward. OCR-only training reaches its scalar reward quickly but often replaces requested scenes with large text on a flat background. After observing this reward hacking, we ran a separate, scheduled, group-relative quality-floor intervention. It reaches OCR 0.97 and DrawBench PickScore 22.17 after 150 updates (41 GPU-hours) while retaining scenes in the inspected samples (Table 2). Its training reward also uses PickScore, ClipScore, and HPSv2.1. Appendix H.1 gives the reward formulation, schedule, failed alternatives, and qualitative comparisons.

At comparable trained-reward scores, our GenEval and PickScore checkpoints are numerically higher than the published DiffusionNFT single-reward rows on $6/7$ auxiliary metrics each (Table 2). For the OCR quality-floor checkpoint, GenEval, Aesthetic, ImageReward, and UnifiedReward did not enter its training reward; these scores and the inspected images complement the floor-component scores.

6 Ablation Study

We test directional alignment, direction transport, reward contrasts, and actor weights. Appendix E gives the detailed results.

6.1 Clean-image versus noisy-state regression: directional alignment

We compare noisy-state and clean-image fits using identical PickScore perturbations and reward differences, with the gradient through the numerical ODE as a reference (Appendix E.3). At a 200-update checkpoint, direct clean-image regression raises the time-weighted mean signed cosine from 0.009 to 0.107: a paired gain of 0.098 (95% prompt-bootstrap CI $[0.092, 0.106]$). The initial model shows the same trend (gain 0.204).

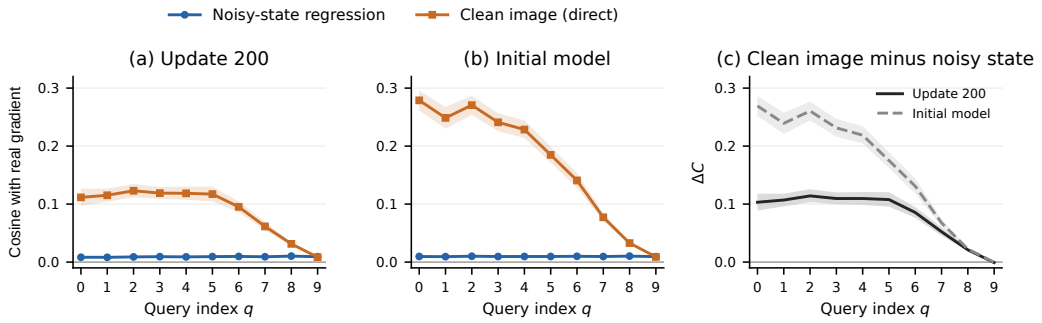


Figure 2: PickScore cosine with the real gradient through the numerical ODE. Left and middle: noisy-state and direct clean-image regression. Right: paired clean-image-minus-noisy-state cosine. Shading denotes 95% confidence intervals from resampling prompts.

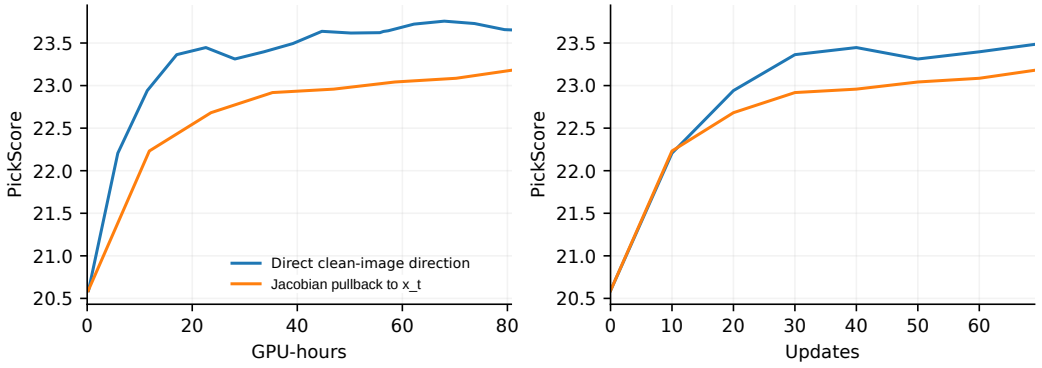


Figure 3: PickScore training with direct clean-image directions or their norm-matched Jacobian pullback to $\mathbf{x}_t$, shown against compute (left) and updates (right). Both arms use the same trajectory bank and actor settings.

6.2 Direct clean-image directions versus adjoint transport

Keeping the PickScore recipe fixed, we replace each direct clean-image direction by its numerical ODE suffix pullback along the sample's own completion (Appendix D). We rescale the pullback to the original direction's norm for every sample and step, so transport changes only the direction and not the per-step update magnitudes. At 81 GPU-hours (69 updates), norm-matched transport reaches 23.18, versus 23.65 for direct transfer at equal compute and 23.48 at equal updates, and each transported update costs $2.1\times$ as much (Figure 3; Appendix E.2).

6.3 Rollout bandwidth depends on reward continuity

PickScore: local contrasts outperform independent-seed rollouts. Holding the 24×24 batch and optimizer fixed, we replace the small-noise bank by ODE-group regression, train all 24 query trajectories, and use batch-standard-deviation reward scaling for the group arm. At 20, 56, and 112 GPU-hours, the bank scores 23.41, 23.63, and 23.78, versus 22.89, 23.21, and 23.18 for the group arm. The latter remains 0.4–0.7 behind at the matched budgets from 20 GPU-hours onward. This is consistent with the local approximation in Eq. (24): independent-seed rollouts give useful diversity but much longer clean-image differences.

GenEval: increasing local noise does not close the gap. We compare the 48×24 group recipe with all-state SDE banks using 48×6 trajectories at $\eta \in \{0.03, 0.3, 0.7\}$. At 28 GPU-hours, group regression reaches 0.92, versus 0.84 and 0.85 for the two larger noise levels; $\eta = 0.03$ collapses to

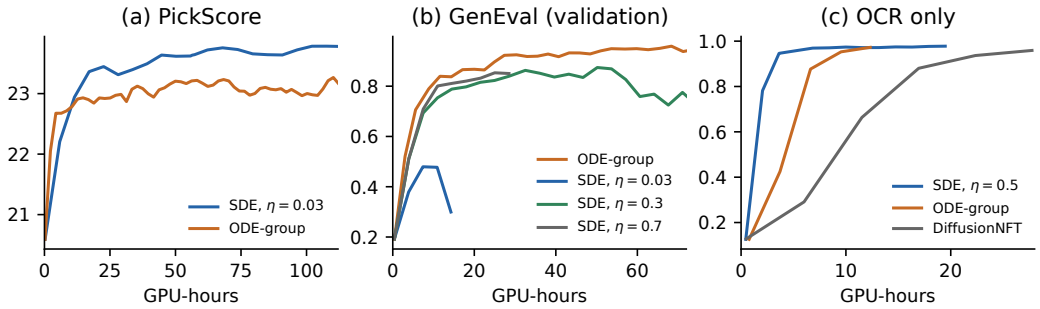


Figure 4: **Rollout comparison across three rewards.** Training curves for trajectory-bank and independent-seed group regression. Batch sizes differ in GenEval and OCR, so these panels compare practical recipes at equal compute.

Table 4: GenEval actor-weight ablation. Scores are interpolated within the recorded validation curves at equal GPU-hours and rounded to two decimals; bold marks the best unrounded value in each column. Only the selected query trajectories receive actor gradients; all sibling clean images remain regression observations.

Stopped weight / selection	10 GPU-h	20 GPU-h	28 GPU-h
$a_i = 1$ (all 24)	0.72	0.84	0.86
$a_i = 1 - r_i$	0.71	0.88	0.89
$a_i = \mathbf{1}\{r_i < 1\}$	0.74	**0.88**	0.88
Random 6	0.81	0.86	0.87
Highest 6	0.62	0.67	0.75
Lowest 6 (ours)	**0.81**	0.87	**0.92**

about 0.30 by update 40. Extending $\eta = 0.3$ to 72 GPU-hours gives 0.76, versus 0.94 for the group recipe. The group recipe leads at both reported budgets.

OCR. At update 40, the trajectory bank and ODE-group both reach 0.97, using 6.8 and 12.3 GPU-hours, respectively. Section 5.3 reports the image-quality outcomes and the quality-floor extension.

6.4 Effect of reward-adaptive weights on training

Keeping the GenEval group regression fixed, we vary the stopped actor weights in Eq. (42). Table 4 compares the resulting training scores at equal compute. Selecting the lowest six rewards gives the best score at 28 GPU-hours; Appendix E.1 provides the full curves.

7 Conclusion

DiffusionRVGE uses centered clean-image regression to construct useful RL directions without adjoint backpropagation. Gradient matching identifies the update target, while population and directional analyses motivate regression in clean-image coordinates. Reward-adaptive rollouts and weights handle continuous and discrete scores within one regression method. A joint model matches or exceeds DiffusionNFT on seven of eight metrics at about one fifth of its estimated training compute; single-reward runs reach matched thresholds 2.1–$6.7\times$ faster on identical hardware.

AI USE STATEMENT

Generative AI tools assisted drafting, translation, literature organization, LaTeX preparation, and plotting saved diagnostics. All reported numbers come from recorded evaluation files.

REPRODUCIBILITY STATEMENT

Sections 3 and 4 define the population theory and evaluated estimator. Appendices A and B give the control and regression proofs; Appendix C contains the supplementary conditional sampling analysis. Appendices E–G provide ablations, implementation details, and evaluation protocols; Appendix H documents reward exploitation. Our code is provided in the supplementary material, and checkpoints will be released.

REFERENCES

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

Kaijie Chen, Zhiyang Xu, Ying Shen, Zihao Lin, Yuguang Yao, and Lifu Huang. SuperFlow: Training flow matching models with RL on the fly. *arXiv preprint arXiv:2512.17951*, 2025. URL https://arxiv.org/abs/2512.17951.

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

Carles Domingo-Enrich, Michal Drozdzal, Brian Karrer, and Ricky T. Q. Chen. Adjoint matching: Fine-tuning flow and diffusion generative models with memoryless stochastic optimal control. *arXiv preprint arXiv:2409.08861*, 2024. URL https://arxiv.org/abs/2409.08861.

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, Kyle Lacey, Alex Goodwin, Yannik Marek, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis. *arXiv preprint arXiv:2403.03206*, 2024. URL https://arxiv.org/abs/2403.03206.

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. *arXiv preprint arXiv:2305.16381*, 2023. URL https://arxiv.org/abs/2305.16381.

Dhruba Ghosh, Hanna Hajishirzi, and Ludwig Schmidt. GenEval: An object-focused framework for evaluating text-to-image alignment. *arXiv preprint arXiv:2310.11513*, 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. *arXiv preprint arXiv:2508.04324*, 2025. URL https://arxiv.org/abs/2508.04324.

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

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. *arXiv preprint arXiv:2106.09685*, 2021. URL https://arxiv.org/abs/2106.09685.

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. *arXiv preprint arXiv:2305.01569*, 2023. URL https://arxiv.org/abs/2305.01569.

Jeongjae Lee, Jinho Chang, Jeongsol Kim, and Jong Chul Ye. Reward score matching: Unifying reward-based fine-tuning for flow and diffusion models. *arXiv preprint arXiv:2604.17415*, 2026. URL https://arxiv.org/abs/2604.17415.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 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. *arXiv preprint arXiv:2505.05470*, 2025. URL https://arxiv.org/abs/2505.05470.

Niru Maheswaranathan, Luke Metz, George Tucker, Dami Choi, and Jascha Sohl-Dickstein. Guided evolutionary strategies: Augmenting random search with surrogate gradients. In *International Conference on Machine Learning*, volume 97, pp. 4264–4273, 2019. URL https://proceedings.mlr.press/v97/maheswaranathan19a.html.

David McAllister, Miika Aittala, Tero Karras, Janne Hellsten, Angjoo Kanazawa, Timo Aila, and Samuli Laine. Finite difference flow optimization for RL post-training of text-to-image models. *arXiv preprint arXiv:2603.12893*, 2026. URL https://arxiv.org/abs/2603.12893.

Yurii Nesterov and Vladimir Spokoiny. Random gradient-free minimization of convex functions. *Foundations of Computational Mathematics*, 17(2):527–566, 2017. doi: 10.1007/s10208-015-9296-2. URL https://doi.org/10.1007/s10208-015-9296-2.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S. Sara Mahdavi, Rapha Gontijo Lopes, Tim Salimans, Jonathan Ho, David J. Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. *arXiv preprint arXiv:2205.11487*, 2022. URL https://arxiv.org/abs/2205.11487.

Yash Savani, Branislav Kveton, Yuchen Liu, Yilin Wang, Jing Shi, Subhojyoti Mukherjee, Nikos Vlassis, and Krishna Kumar Singh. Stepwise credit assignment for GRPO on flow-matching models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026. URL https://arxiv.org/abs/2603.28718.

Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2021. URL https://arxiv.org/abs/2011.13456.

Bram Wallace, Meihua Dang, Rafael Rafailov, Linqi Zhou, Aaron Lou, Senthil Purushwalkam, Stefano Ermon, Caiming Xiong, Shafiq Joty, and Nikhil Naik. Diffusion model alignment using direct preference optimization. *arXiv preprint arXiv:2311.12908*, 2023. URL https://arxiv.org/abs/2311.12908.

Yixian Xu, Yuanrui Zhang, Shengjie Luo, Liwei Wang, and Di He. Designing reinforcement learning for diffusion models: A unified path-space view. *arXiv preprint arXiv:2608.14430*, 2026. URL https://arxiv.org/abs/2608.14430.

Zeyue Xue, Jie Wu, Yu Gao, Fangyuan Kong, Lingting Zhu, Mengzhao Chen, Zhiheng Liu, Wei Liu, Qiushan Guo, Weilin Huang, and Ping Luo. DanceGRPO: Unleashing GRPO on visual generation. *arXiv preprint arXiv:2505.07818*, 2025. URL https://arxiv.org/abs/2505.07818.

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://arxiv.org/abs/2509.16117.

Yujie Zhou, Pengyang Ling, Jiazi Bu, Yibin Wang, Yuhang Zang, Jiaqi Wang, Li Niu, and Guangtao Zhai. G^2RPO: Granular GRPO for precise reward in flow models. *arXiv preprint arXiv:2510.01982*, 2025. URL https://arxiv.org/abs/2510.01982.

Yuanzhi Zhu, Xi Wang, Stéphane Lathuilière, and Vicky Kalogeiton. Diffusion reinforcement learning via centered reward distillation. *arXiv preprint arXiv:2603.14128*, 2026. URL https://arxiv.org/abs/2603.14128.

A PROOF OF THE TRAJECTORY-KL CONTROL IDENTITY

Assumptions and diffusion parameterization. Assume well-posed reference and controlled diffusions with the same initial law and diffusion coefficient, finite expected reward and control energy, and Girsanov integrability. The conditional exponential moment is positive and finite; its soft value is regular enough for Itô's formula, with a zero-mean martingale term. Singular endpoints are treated by limits of truncated intervals with uniform integrability of the displayed terms.

For the Gaussian noising path, set $A_t = \dot{\alpha}_t/\alpha_t$ and $\nu_t^2 = \partial_t \sigma_t^2 - 2A_t\sigma_t^2 > 0$ on the interior. The linear forward SDE $\mathrm{d}\mathbf{X}_t = A_t\mathbf{X}_t\mathrm{d}t + \nu_t\mathrm{d}\mathbf{W}_t$ has the specified conditional Gaussian kernels. Its time reversal, initialized at the forward terminal law, has drift

$$\mathbf{b}_{\mathrm{ref}} = -A_t\mathbf{x} + \nu_t^2\nabla \log p_t = A_t\mathbf{x} - 2\mathbf{v}_{\mathrm{ref}}, \qquad \mathbf{v}_{\mathrm{ref}} = A_t\mathbf{x} - \tfrac{1}{2}\nu_t^2\nabla \log p_t.$$

Its conditional clean law is the Gaussian noising posterior. The parameterization $\mathbf{b}_{\boldsymbol{\theta}} = A_t\mathbf{x} - 2\mathbf{v}_{\boldsymbol{\theta}}$ therefore gives $\mathbf{u}_{\boldsymbol{\theta}} = -2\delta\mathbf{v}_{\boldsymbol{\theta}}$ and $\gamma_s^2 = \nu_{T-s}^2$.

Proof of Theorem 1. Write $\mathcal{W}_s = W_{T-s}$ and $\mathcal{H}_s = H_{T-s}$. The reference backward equation for $\mathcal{H}_s = e^{\mathcal{W}_s/\tau}$ yields

$$(\partial_s + \mathcal{L}_s^{\mathrm{ref}})\mathcal{W}_s + \frac{\gamma_s^2}{2\tau}\|\nabla\mathcal{W}_s\|^2 = 0, \qquad \mathcal{W}_T = R, \qquad \mathcal{L}_s^{\mathrm{ref}} = \mathbf{b}_{\mathrm{ref}}\cdot\nabla + \tfrac{1}{2}\gamma_s^2\Delta. \tag{15}$$

Girsanov's identity and Itô's formula under the controlled process give

$$\begin{aligned} \mathrm{KL}(P_{\boldsymbol{\theta}}\|P_{\mathrm{ref}}) &= \frac{1}{2}\mathbb{E}_{P_{\boldsymbol{\theta}}}\int_0^T \gamma_s^{-2}\|\mathbf{u}_{\boldsymbol{\theta}}\|^2\mathrm{d}s, \\ \mathbb{E}_{P_{\boldsymbol{\theta}}}R(\mathbf{Y}_T) - C_\tau &= \mathbb{E}_{P_{\boldsymbol{\theta}}}\int_0^T \left[\mathbf{u}_{\boldsymbol{\theta}}^\top\nabla\mathcal{W}_s - \frac{\gamma_s^2}{2\tau}\|\nabla\mathcal{W}_s\|^2\right]\mathrm{d}s. \end{aligned} \tag{16}$$

Subtracting τ times the first identity from the second and completing the square proves Eq. (4). The maximizing correction is $\mathbf{u}_s^* = \gamma_s^2\nabla\mathcal{W}_s/\tau$ when admissible. Substituting $\mathbf{u}_{\boldsymbol{\theta}} = -2\delta\mathbf{v}_{\boldsymbol{\theta}}$ proves Eq. (17), including its positive residual sign. Appendix A.1 gives the separate first-variation argument for the frozen-policy training loss.

The equivalent velocity identity is

$$C_\tau - J_\tau = 2\tau\mathbb{E}_{P_{\boldsymbol{\theta}}}\int_0^T \frac{\|\delta\mathbf{v}_{\boldsymbol{\theta}} + \gamma_s^2\nabla W_{T-s}/(2\tau)\|^2}{\gamma_s^2}\mathrm{d}s. \tag{17}$$

Reward tilt and initialization. The path measure $\mathrm{d}Q^*/\mathrm{d}P_{\mathrm{ref}} = e^{R(\mathbf{Y}_T)/\tau}/Z_\tau$ has terminal law $\pi^* \propto e^{R/\tau}\rho$. Substitution into relative entropy proves Eq. (3). Its initial law is $\mu^*(\mathrm{d}\mathbf{x}) = H_T(\mathbf{x})\mu(\mathrm{d}\mathbf{x})/Z_\tau$. The oracle drift started from μ^* realizes Q^*. With an arbitrary fixed initial law μ, the same drift instead realizes the conditional tilt $e^{R(\mathbf{Y}_T)/\tau}/H_T(\mathbf{Y}_0)$. The two coincide when H_T is constant, as at the independent-noise entrance in the main text.

A.1 A COMMON FIRST VARIATION FOR ALL PENALTY STRENGTHS

In addition to the shared initial law and policy-independent diffusion, assume that $F_t^{\boldsymbol{\theta}_{\mathrm{old}},\tau}$ is $C^{1,2}$ on interior intervals, the following Itô terms are integrable, and differentiation of the objective and endpoint limits are justified. For $\tau > 0$ assume Girsanov integrability and finite expected control energy; no exponential moment or soft-value limit is needed for the first variation at $\tau = 0$.

Write $\mathcal{F}_s = F_{T-s}^{\boldsymbol{\theta}_{\mathrm{old}},\tau}$ and $c_{\boldsymbol{\theta},s} = \tau\|\mathbf{u}_{\boldsymbol{\theta}}\|^2/(2\gamma_s^2)$, taking $c_{\boldsymbol{\theta},s} = 0$ when $\tau = 0$. Its backward equation is

$$(\partial_s + \mathcal{L}_s^{\boldsymbol{\theta}_{\mathrm{old}}})\mathcal{F}_s = c_{\boldsymbol{\theta}_{\mathrm{old}},s}, \qquad \mathcal{F}_T = R, \qquad \mathcal{L}_s^{\boldsymbol{\theta}_{\mathrm{old}}} = \mathbf{b}_{\boldsymbol{\theta}_{\mathrm{old}}}\cdot\nabla + \tfrac{1}{2}\gamma_s^2\Delta.$$

Itô's formula under $P_{\boldsymbol{\theta}}$ gives the exact performance difference

$$J_\tau(\boldsymbol{\theta}) - J_\tau(\boldsymbol{\theta}_{\mathrm{old}}) = \mathbb{E}_{P_{\boldsymbol{\theta}}}\int_0^T \left[\langle\mathbf{u}_{\boldsymbol{\theta}} - \mathbf{u}_{\boldsymbol{\theta}_{\mathrm{old}}}, \nabla\mathcal{F}_s\rangle - c_{\boldsymbol{\theta},s} + c_{\boldsymbol{\theta}_{\mathrm{old}},s}\right]\mathrm{d}s. \tag{18}$$

The integrand vanishes at $\boldsymbol{\theta} = \boldsymbol{\theta}_{\mathrm{old}}$, so differentiating there leaves no first-order occupancy term. Consequently,

$$\nabla_{\boldsymbol{\theta}} J_\tau|_{\boldsymbol{\theta}_{\mathrm{old}}} = \mathbb{E}_{P_{\boldsymbol{\theta}_{\mathrm{old}}}} \int_0^T (\partial_{\boldsymbol{\theta}} \mathbf{u}_{\boldsymbol{\theta}}|_{\boldsymbol{\theta}_{\mathrm{old}}})^\top \left(\nabla \mathcal{F}_s - \frac{\tau}{\gamma_s^2} \mathbf{u}_{\boldsymbol{\theta}_{\mathrm{old}}} \right) \mathrm{d}s. \tag{19}$$

Substituting $\mathbf{u}_{\boldsymbol{\theta}} = -2\delta \mathbf{v}_{\boldsymbol{\theta}}$ proves the gradient identity for Eq. (6), with $w_t = 1/\varrho(t)$ when t has sampling density ϱ. At zero penalty the cost terms are omitted before evaluation. For the reward-only ODE with generation drift $-\mathbf{v}_{\boldsymbol{\theta}}$, the same proof uses the transport equation and gives cross-term factor 1 instead of 2, assuming a differentiable continuation value. For $\tau > 0$, $F_t^{\boldsymbol{\theta}_{\mathrm{old}},\tau}$ agrees with the optimal soft value W_t at the oracle policy, linking the local update to Theorem 1.

Regression uses the same return definition for all penalties. Perturb a query by $h\boldsymbol{\epsilon}$ with $\mathbb{E}\boldsymbol{\epsilon} = 0$ and $M = \mathbb{E}[\boldsymbol{\epsilon}\boldsymbol{\epsilon}^\top] \succ 0$. Let $\mathcal{R}_\tau$ be the terminal reward minus the sampled future frozen-policy control cost in Eq. (5). Its conditional mean is $F_t^{\boldsymbol{\theta}_{\mathrm{old}},\tau}(\mathbf{x} + h\boldsymbol{\epsilon})$. With a bounded Hessian on the perturbed states, finite second moments of $\mathcal{R}_\tau$ and $\mathbb{E}\|\boldsymbol{\epsilon}\|^3 < \infty$, the population regression satisfies

$$\begin{aligned} \mathbf{a}_{h,\tau} &= \arg\min_{\mathbf{a}} \mathbb{E}\big[(\mathcal{R}_\tau - \mathbb{E}\mathcal{R}_\tau - \mathbf{a}^\top h\boldsymbol{\epsilon})^2\big] \\ &= \frac{1}{h} M^{-1} \mathbb{E}[\boldsymbol{\epsilon} F_t^{\boldsymbol{\theta}_{\mathrm{old}},\tau}(\mathbf{x} + h\boldsymbol{\epsilon})] = \nabla F_t^{\boldsymbol{\theta}_{\mathrm{old}},\tau}(\mathbf{x}) + O(h). \end{aligned} \tag{20}$$

The normal equations give the middle equality; Taylor expansion gives the last. Thus the same regression principle applies for each fixed $\tau \geq 0$. At zero penalty its labels are ordinary terminal rewards. For a fixed bank, centered feature matrix, row normalization and ridge coefficient, linearity of the fit also gives $\widehat{\mathbf{u}}_\tau = \widehat{\mathbf{u}}_R - \tau \widehat{\mathbf{u}}_C$, where the two coefficients fit terminal rewards and sampled future control energies.

B Covariance preconditioning and regression stability

Covariance identities and proof of Proposition 2. For finite conditional moments and differentiation under the conditional integral, Bayes' rule gives

$$q_t(\mathrm{d}\mathbf{z} \mid \mathbf{x}) \propto e^{-\|\mathbf{x} - \alpha_t \mathbf{z}\|^2/(2\sigma_t^2)} \rho(\mathrm{d}\mathbf{z}), \qquad \nabla_{\mathbf{x}} \log q_t(\mathbf{z} \mid \mathbf{x}) = \kappa_t(\mathbf{z} - \mathbf{m}_t).$$

The derivative is of the density relative to ρ, so singular clean laws are allowed. Consequently $\nabla_{\mathbf{x}} \mathbb{E}[f(\mathbf{Z}) \mid \mathbf{x}] = \kappa_t \operatorname{Cov}(\mathbf{Z}, f(\mathbf{Z}) \mid \mathbf{x})$. Taking $f = R$ proves Eq. (8); taking $f = e^{R/\tau}$ yields

$$\nabla_{\mathbf{x}} W_t = \tau \kappa_t \frac{\operatorname{Cov}(\mathbf{Z}, e^{R/\tau} \mid \mathbf{x})}{H_t}. \tag{21}$$

Neither identity differentiates the reward. The normal equations in Eq. (7) give $\Sigma_t \mathbf{g}_t = \kappa_t \mathbf{c}_t$. If $\mathbf{a} \in \ker \Sigma_t$, then $\mathbf{a}^\top(\mathbf{Z} - \mathbf{m}_t) = 0$ almost surely, so $\mathbf{a}^\top \mathbf{c}_t = 0$. Thus $\mathbf{c}_t \in \operatorname{Range}(\Sigma_t)$ and the minimum-norm solution is $\mathbf{g}_t = \kappa_t \Sigma_t^\dagger \mathbf{c}_t$. Its inner product with ∇V_t is $(\nabla V_t)^\top \Sigma_t^\dagger \nabla V_t > 0$ for nonzero ∇V_t.

For an affine reward with slope $\mathbf{w} \in \operatorname{Range}(\Sigma_t)$, $\mathbf{c}_t = \Sigma_t \mathbf{w}$. The maps $\mathbf{w} \mapsto \nabla V_t$ and $\mathbf{w} \mapsto \mathbf{g}_t$ therefore have condition numbers $\kappa_+(\Sigma_t)$ and 1 on that subspace, where κ_+ uses the nonzero eigenvalues.

B.1 Directional advantage with mean-centered observations

Let $P = UU^\top$ with $U^\top U = I_k$, $1 \leq k < d$, and $\mathbf{Z} \sim \mathcal{N}(0, \sigma_{\mathrm{data}}^2 P)$. For $R(\mathbf{z}) = b + \mathbf{w}^\top \mathbf{z}$, $P\mathbf{w} = \mathbf{w} \neq 0$, Gaussian conditioning gives

$$\mathbf{m}_t(\mathbf{x}) = a_t P \mathbf{x}, \quad \Sigma_t = \lambda_t P, \quad a_t = \frac{\alpha_t \sigma_{\mathrm{data}}^2}{\alpha_t^2 \sigma_{\mathrm{data}}^2 + \sigma_t^2}, \quad \lambda_t = \frac{\sigma_{\mathrm{data}}^2 \sigma_t^2}{\alpha_t^2 \sigma_{\mathrm{data}}^2 + \sigma_t^2}. \tag{22}$$

Since $W_t = V_t + \lambda_t \|\mathbf{w}\|^2/(2\tau)$, both gradients equal $a_t \mathbf{w}$, and $\mathbf{g}_t = \kappa_t \mathbf{w}$. Draw $m \geq 2$ independent perturbations $\boldsymbol{\delta}_j \sim \mathcal{N}(0, \eta^2 I_d)$ and form a matrix D with rows $\boldsymbol{\delta}_j^\top$. Write $C = I_m - \mathbf{1}\mathbf{1}^\top/m$ for the centering matrix. The centered noisy rows, centered denoised means, and centered labels are

$$D_c = CD, \qquad Q_c = a_t D_c P, \qquad \mathbf{b}_c = a_t D_c \mathbf{w} = Q_c \mathbf{w}.$$

The unnormalized minimum-norm fits are

$$\widehat{\mathbf{h}}_{\text{noisy}} = D_c^\dagger \mathbf{b}_c = a_t \Pi_S \mathbf{w}, \qquad \widehat{\mathbf{g}}_{\text{clean}} = \kappa_t Q_c^\dagger \mathbf{b}_c = \kappa_t \Pi_L \mathbf{w},$$

where $S = \text{row}(D_c)$ and $L = \text{row}(D_c P)$. If $B \in \mathbb{R}^{m\times(m-1)}$ is an orthonormal basis of $\mathbf{1}^\perp$, then $C = BB^\top$ and $B^\top D$ has $m-1$ independent isotropic Gaussian rows. Hence S has the same distribution as the row space of an $(m-1)\times d$ Gaussian matrix. Rotational invariance gives

$$\mathbb{E}\Pi_S = \min((m-1)/d, 1) I_d, \qquad \mathbb{E}\Pi_L = \min((m-1)/k, 1) P.$$

For either nonzero projection, $\cos^2(\Pi_S \mathbf{w}, \mathbf{w}) = \|\Pi_S \mathbf{w}\|^2 / \|\mathbf{w}\|^2$. Taking expectations proves Eq. (9), including positive alignment almost surely.

B.2 Ridge controls coefficient sensitivity

For the actual sum-loss convention in Eq. (11), let $Q = U \,\text{diag}(s_j) V^\top$ be a singular value decomposition. Then

$$\widehat{\mathbf{u}} = V \,\text{diag}\left(\frac{s_j}{s_j^2 + \lambda_{\text{sum}}}\right) U^\top \mathbf{b}, \qquad \|\widehat{\mathbf{u}}\| \le \frac{\|\mathbf{b}\|}{2\sqrt{\lambda_{\text{sum}}}}. \tag{23}$$

The bound follows from $s/(s^2+\lambda) \le 1/(2\sqrt{\lambda})$. For fixed Q, perturbing the normalized labels by $\Delta\mathbf{b}$ changes the coefficient by at most $\|\Delta\mathbf{b}\|/(2\sqrt{\lambda_{\text{sum}}})$. No rank assumption is required; this includes the linear dependence introduced by mean centering.

Local approximation and rollout radius. For R with Hessian norm at most L on a convex neighborhood containing the bank, set $\mathbf{d}_j = \mathbf{z}_j - \bar{\mathbf{z}}$ and $\mathbf{w} = \nabla R(\bar{\mathbf{z}})$. Taylor's theorem and $\sum_j \mathbf{d}_j = 0$ give

$$r_j - \bar{r} = \mathbf{w}^\top \mathbf{d}_j + e_j - \bar{e}, \qquad |e_j - \bar{e}| \le \frac{L}{2}\left(\|\mathbf{d}_j\|^2 + \frac{1}{m}\sum_k \|\mathbf{d}_k\|^2\right). \tag{24}$$

This explains why smaller clean-image displacements can improve a local fit for smooth rewards.

C Supplementary analysis of Langevin-corrected sampling

This appendix provides a conditional consistency result for an idealized sampling dynamics at $\tau > 0$. It assumes exact reference scores, ideal realization of the regression field, a vanishing score discrepancy, and a mixing inequality. Under these assumptions, the corrected dynamics tracks the reward-tilted target.

Idealized score approximation. For $\tau > 0$, to examine the structural effect of preconditioning, use $\chi_t = \sigma_t^2/\alpha_t^2$ and replace ∇W_t by $\chi_t \mathbf{g}_t$. In normalized coordinates $\mathbf{y} = \mathbf{x}/\alpha_t$, $\chi = e^{-2\ell}$, define $\mathbf{G}_\ell = \alpha_t \chi_t \mathbf{g}_t = \Sigma_\ell^\dagger \mathbf{c}_\ell$. Let $p_\ell = \rho * \mathcal{N}(0, \chi I)$ and $\pi_\ell^* = \pi^* * \mathcal{N}(0, \chi I)$. The regression score is $\nabla \log p_\ell + \mathbf{G}_\ell/\tau$, with error

$$\mathbf{d}_\ell = \frac{\mathbf{G}_\ell - \nabla_{\mathbf{y}} W_\ell}{\tau} = \frac{(I - \Sigma_\ell/\chi)\mathbf{G}_\ell + \nabla_{\mathbf{y}} V_\ell - \nabla_{\mathbf{y}} W_\ell}{\tau}. \tag{25}$$

The two terms isolate covariance geometry and the ordinary-to-soft-value approximation. The following conditional result states when this preconditioning preserves the terminal target.

Theorem 3 (Conditional consistency of idealized corrected sampling)**.** *Assume an exact reference score, ideal realization of the regression field, and the regularity conditions stated below. For a fixed $a_0 > 0$, let*

$$\mathrm{d}\mathbf{Y}_\ell = (\chi + a_0)\left(\nabla \log p_\ell + \frac{\mathbf{G}_\ell}{\tau}\right)(\mathbf{Y}_\ell)\mathrm{d}\ell + \sqrt{2a_0}\mathrm{d}\mathbf{B}_\ell. \tag{26}$$

If π_ℓ^* satisfies a uniform log-Sobolev inequality, the initial relative entropy is finite, and $\mathbb{E}\|\mathbf{d}_\ell(\mathbf{Y}_\ell)\|^2 \to 0$, then

$$\mathrm{KL}(\mu_\ell \| \pi_\ell^*) \to 0, \qquad \mu_\ell \Rightarrow \pi^* \quad (\ell \to \infty). \tag{27}$$

Ideal control realization. The comparison isolates the regression approximation by assuming exact reference scores and

$$\mathbf{v}_{\boldsymbol{\theta}} - \mathbf{v}_{\mathrm{ref}} = -\frac{\nu_t^2 \chi_t}{2\tau}\mathbf{g}_t, \qquad \mathbf{s}_t^{\boldsymbol{\theta}} = \mathbf{s}_t^{\mathrm{ref}} + \frac{\chi_t \mathbf{g}_t}{\tau}, \qquad \widetilde{\mathbf{u}}_t = \frac{\nu_t^2 \chi_t}{\tau}\mathbf{g}_t. \tag{28}$$

Here $\mathbf{s}_t^{\mathrm{ref}} = \nabla_{\mathbf{x}} \log p_t$ and $\mathbf{u}_t^* = \nu_t^2 \nabla_{\mathbf{x}} W_t/\tau$. In normalized coordinates $\mathbf{y} = \mathbf{x}/\alpha_t$, write P_χ for Gaussian convolution of variance χ. Reindex V, W, Σ and $\mathbf{c}$ at $(\alpha_t \mathbf{y}, t)$, and write $\mathbf{k}_\ell = \nabla_{\mathbf{y}} W_\ell/\tau = \nabla \log \pi_\ell^* - \nabla \log p_\ell$.

Scale and oracle control. Let $\chi_t = \sigma_t^2/\alpha_t^2$, $\mathbf{y} = \mathbf{x}/\alpha_t$, and $\chi = e^{-2\ell}$. Since $\alpha_t \chi_t \kappa_t = 1$, the normalized regression field is $\mathbf{G}_\ell = \alpha_t \chi_t \mathbf{g}_t = \Sigma_\ell^\dagger \mathbf{c}_\ell$. The control targets satisfy the exact identity

$$\widetilde{\mathbf{u}}_t = \chi_t \Sigma_t^\dagger \mathbf{u}_t^* + \frac{\nu_t^2 \chi_t}{\tau}\Sigma_t^\dagger(\nabla_{\mathbf{x}} V_t - \nabla_{\mathbf{x}} W_t). \tag{29}$$

Thus scale-corrected preconditioning retains the ordinary/soft-value residual. Differentiating the conditional exponential moment gives

$$\mathbf{k}_\ell = \frac{\mathrm{Cov}(\mathbf{Z}, e^{R(\mathbf{Z})/\tau} \mid \mathbf{y})}{\chi\, \mathbb{E}[e^{R(\mathbf{Z})/\tau} \mid \mathbf{y}]}, \qquad \nabla_{\mathbf{y}} V_\ell = \chi^{-1} \Sigma_\ell \mathbf{G}_\ell. \tag{30}$$

These identities yield the two-term discrepancy in Eq. (25); neither residual is discarded.

Uniform endpoint assumptions. On a stable covariance support with orthogonal projector P, assume, uniformly in $\mathbf{y}$ as $\chi \downarrow 0$,

$$\begin{aligned} &\mathrm{Range}(\Sigma) = \mathrm{Range}(P), \qquad \Sigma = \chi P + O(\chi^2), \qquad \mathbf{c} = \chi \mathbf{w} + O(\chi^2), \qquad P\mathbf{w} = \mathbf{w}, \\ &\frac{\mathrm{Cov}(\mathbf{Z}, e^{R(\mathbf{Z})/\tau} \mid \mathbf{y})}{\mathbb{E}[e^{R(\mathbf{Z})/\tau} \mid \mathbf{y}]} = \tau^{-1} \chi \mathbf{w} + O(\chi^2), \qquad \sup_{\mathbf{y},\ell} \|\mathbf{w}\| < \infty. \end{aligned} \tag{31}$$

The support has fixed rank and the remainders are uniform in vector and operator norms. These assumptions on the conditional covariance suffice for the mean-square error criterion in Theorem 3; the noisy laws $p_\ell = P_\chi \rho$ and $\pi_\ell^* = P_\chi \pi^*$ have ambient densities for every $\chi > 0$, even when ρ is singular. In the Gaussian-subspace model of Eq. (22), the covariance condition holds explicitly:

$$\Sigma_\ell = \frac{\chi}{1 + \chi/\sigma_{\mathrm{data}}^2} P = \chi P - \frac{\chi^2}{\sigma_{\mathrm{data}}^2} P + O(\chi^3), \qquad \chi \Sigma_\ell^\dagger = (1 + \chi/\sigma_{\mathrm{data}}^2) P.$$

On this support, $\lambda_{\min}^+(\Sigma) \geq \chi/2$ for sufficiently small χ. Hence

$$\mathbf{G}_\ell - \mathbf{w} = \Sigma^\dagger(\mathbf{c} - \Sigma \mathbf{w}) = O(\chi), \qquad \mathbf{k}_\ell = \tau^{-1}\mathbf{w} + O(\chi), \qquad \mathbf{d}_\ell = O(\chi).$$

This proves the scale-consistency assertion. In the original coordinates,

$$\frac{\tau}{\nu_t^2}\|\widetilde{\mathbf{u}}_t - \mathbf{u}_t^*\| = \|\chi_t \mathbf{g}_t - \nabla_{\mathbf{x}} W_t\| = \frac{\tau}{\alpha_t}\|\mathbf{d}_\ell\| = O(\chi_t).$$

An unnormalized control-drift bound additionally depends on the schedule ν_t^2.

Scope and regularity of the conditional guarantee. Assume $\tau > 0$, $0 < Z_\tau = \int e^{R/\tau} \mathrm{d}\rho < \infty$, well-posed corrected dynamics, positive smooth noisy densities, and finite initial relative entropy at a finite ℓ_0. Assume the differentiations and integrations by parts below are justified, and the score-error energy is locally integrable. Reference scores and the displayed guidance field are realized exactly. For $\mathcal{H}_\ell = \mathrm{KL}(\mu_\ell \| \pi_\ell^*)$ and $\mathcal{I}_\ell = \mathbb{E}_{\mu_\ell}\|\nabla \log(\mu_\ell/\pi_\ell^*)\|^2$, assume $\mathcal{I}_\ell \geq 2\kappa \mathcal{H}_\ell$ for some $\kappa > 0$ independent of $\ell \geq \ell_0$.

Entropy proof. Gaussian convolution and $\chi = e^{-2\ell}$ give $\partial_\ell \pi_\ell^* = -\chi \Delta \pi_\ell^*$. Hence the oracle drift $(\chi + a_0)\nabla \log \pi_\ell^*$ with diffusion $\sqrt{2a_0}I$ exactly transports π_ℓ^*. The approximate drift differs by $\mathbf{D}_\ell = (\chi + a_0)\mathbf{d}_\ell$. Writing $f_\ell = \log(\mu_\ell/\pi_\ell^*)$, the relative-entropy identity and Young's inequality imply

$$\mathcal{H}_\ell' = -a_0 \mathcal{I}_\ell + \mathbb{E}_{\mu_\ell}\langle \mathbf{D}_\ell, \nabla f_\ell \rangle \leq -a_0 \kappa \mathcal{H}_\ell + \varepsilon_\ell, \qquad \varepsilon_\ell = \frac{(\chi + a_0)^2}{2a_0}\mathbb{E}_{\mu_\ell}\|\mathbf{d}_\ell\|^2. \tag{32}$$

Consequently,

$$\mathcal{H}_\ell \le e^{-a_0\kappa(\ell-\ell_0)}\mathcal{H}_{\ell_0} + \int_{\ell_0}^{\ell} e^{-a_0\kappa(\ell-r)}\varepsilon_r \mathrm{d}r. \tag{33}$$

Since $\varepsilon_\ell \to 0$, split the integral at a sufficiently large fixed time to see that both terms vanish. This proves relative-entropy tracking. Pinsker's inequality and $\pi^*_\ell \Rightarrow \pi^*$ then give the weak endpoint limit. For a singular clean target, the endpoint conclusion is weak convergence, not total-variation convergence to that singular measure. More generally, the same proof needs only that the exponentially weighted error integral in Eq. (33) tends to zero.

Discrete rewards and a smooth reference. For bounded measurable R, the positive-noise covariance identity requires no reward derivative, and the convergence theorem applies to an explicitly specified bounded smooth reference R_ε at fixed $\varepsilon > 0$, whenever its assumptions hold. If $R_\varepsilon \to R$ in $L^1(\rho)$ with a common bound, then

$$\pi^{*,\varepsilon}(\mathrm{d}\mathbf{z}) = \frac{e^{R_\varepsilon(\mathbf{z})/\tau}\rho(\mathrm{d}\mathbf{z})}{\int e^{R_\varepsilon/\tau}\mathrm{d}\rho} \longrightarrow \pi^*(\mathrm{d}\mathbf{z}) \quad \text{in total variation,}$$

because the exponential is Lipschitz on the bounded reward range and the normalizers converge.

D Numerical adjoint used in the transport ablation

Let F_t be a deterministic ODE completion with differentiable velocity. Transporting a fitted clean-image direction $\mathbf{b}$ to the current state requires the Jacobian product $DF_t^\top \mathbf{b}$. For a smooth path, the continuous adjoint solves

$$\dot{\boldsymbol{\lambda}}_s = -D_{\mathbf{x}}\mathbf{v}(\mathbf{x}_s, s)^\top \boldsymbol{\lambda}_s, \qquad \boldsymbol{\lambda}_0 = \mathbf{b}, \qquad \boldsymbol{\lambda}_t = DF_t(\mathbf{x}_t)^\top \mathbf{b}. \tag{34}$$

Indeed, a state perturbation satisfies $\dot{\boldsymbol{\xi}}_s = D_{\mathbf{x}}\mathbf{v}\,\boldsymbol{\xi}_s$, so $\boldsymbol{\lambda}_s^\top \boldsymbol{\xi}_s$ is constant. For an Euler completion $\mathbf{z}_{k+1} = \mathbf{z}_k + \Delta t_k \mathbf{v}(\mathbf{z}_k, t_k)$, define $J_k = I + \Delta t_k D_{\mathbf{x}}\mathbf{v}(\mathbf{z}_k, t_k)$. A velocity perturbation at step q gives

$$\delta R = \Delta t_q \langle \boldsymbol{\lambda}_{q+1}, \delta \mathbf{v}_q \rangle, \qquad \boldsymbol{\lambda}_{q+1} = J_{q+1}^\top \cdots J_{N-1}^\top \mathbf{b}, \qquad \mathbf{b} = \nabla R(\mathbf{z}_N). \tag{35}$$

The ablation replaces $\mathbf{b}$ by $\widehat{\mathbf{u}}_q$, evaluates this suffix along the query's own completion by vector–Jacobian products, and rescales it to the original coefficient's norm. An empty suffix is the identity. This keeps per-step update magnitudes fixed while testing the cost and benefit of transporting directions. For a coefficient error $\mathbf{e}$, the continuous pullback error is bounded by $\|DF_t\|_{\mathrm{op}}\|\mathbf{e}\|$. Direct transfer avoids these Jacobian products; its training performance is evaluated in Section 6.2.

E Complete reward and weighting ablations

Every training arm uses $8\times$A800-40GB. We show both compute and update count because the training choices change update cost. Budget-matched scores are linearly interpolated within the recorded curves.

E.1 Reward-dependent actor weights

All six arms use the same mean-centered ODE-group clean-image regression and 48×24 collected trajectories. We change only the query weights: all 24 ($a_i = 1$); $a_i = 1 - r_i$; $a_i = \mathbf{1}\{r_i < 1\}$; random six; highest six; lowest six. Random selection uses the first six independent-seed rollouts, so selection is independent of reward. The scalar functions consider all 24 queries; zero-weight queries need neither a fit for their own actor nor a backward pass. Their clean images still serve as sibling observations. The denominator is the sum of actor weights, as in Eq. (42), not the number of candidates.

At 28 GPU-hours the two ceiling-aware arms have completed approximately 81–84 updates, versus about 110 for the lowest-six arm.

The saturation motivation also has a frozen-policy check: at an early checkpoint, a one-sided perturbation of 3% of state RMS along the group fit changes the reward of failed samples by $+0.064$ and successful samples by -0.019, with random-direction controls around ± 0.003. During updates 28–65, roughly 74% of the all-24 arm's actor samples already have maximum reward.

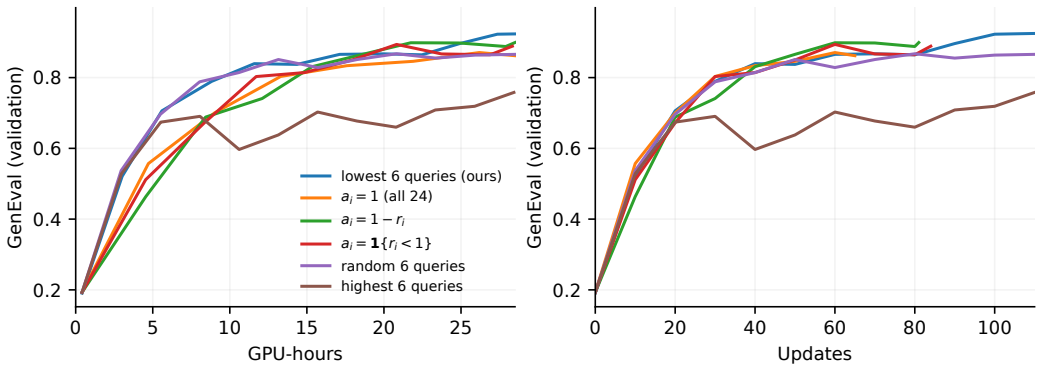


Figure 5: GenEval query weights at a 28-GPU-hour budget, shown against compute and updates.

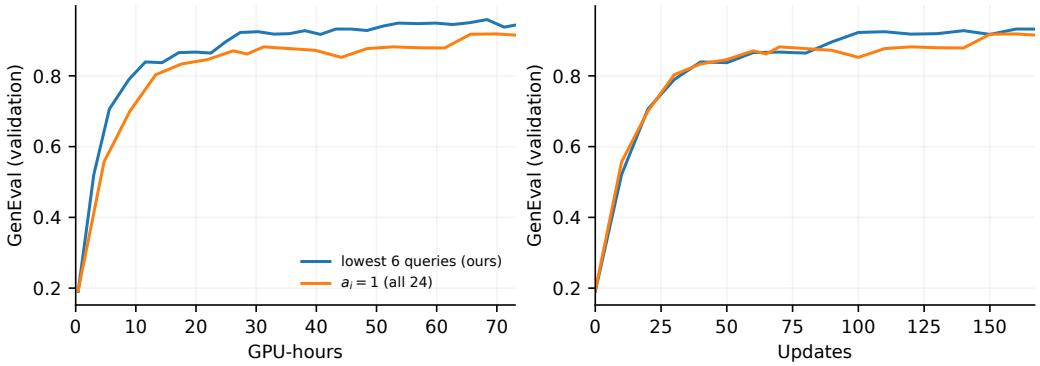


Figure 6: The all-24 continuation. The lowest-six advantage persists beyond the initial budget but narrows near 72 GPU-hours.

E.2 Norm-matched adjoint transport of the clean-image estimate

This arm keeps the PickScore 24×24 bank recipe and replaces each fitted clean-image direction by the suffix pullback in Eq. (35). The Jacobians are evaluated along each sample's own Euler completion from $\mathbf{z}_{q+1} = \mathbf{x}_q + \Delta t_q \mathbf{v}(\mathbf{x}_q, t_q)$, the completion that produced its clean image, and applied by vector–Jacobian products without a dense Jacobian. Each transported direction is then rescaled to the root-mean-square of the direction it replaces, separately for every sample and step, so the update keeps the recipe's magnitudes across samples and times and changes only the orientation. Over the 69 updates, the transported direction has median cosine 0.18 with the clean-image direction at the first step (range 0.15–0.41), rising monotonically to 0.79 at step 7 and 1 at the last two. The transport runs in the training precision (bfloat16); on eight probe samples its median cosine with a float32 reference is 0.71, 0.95, 0.97, 0.99, and 1.00 at steps 0, 2, 4, 6, and 8.

At 12, 24, 35, 47, 59, 70, and 81 GPU-hours (updates $10, 20, \ldots, 60$ and 69), the transported arm scores 22.23, 22.68, 22.92, 22.96, 23.04, 23.09, and 23.18. At the same compute the direct arm scores 22.97, 23.42, 23.43, 23.63, 23.66, 23.75, and 23.65; at the same updates it scores 22.21, 22.94, 23.36, 23.45, 23.31, 23.40, and 23.48. From update 20 onward the per-update deficit (0.26–0.49) exceeds the difference between the two training seeds at matched updates (at most 0.14 through update 70; Appendix E.4).

E.3 PickScore directional-alignment probe

We compare noisy-state and clean-image directions from matched PickScore perturbations at the initial model and an update-200 checkpoint. We measure their cosine with the real reward gradi-

Table 5: Longer GenEval comparison at fixed compute, using interpolation within the saved curves.

GPU-hours	28	35	44	52	61
Lowest 6	0.92	0.92	0.93	0.94	0.95
All 24	0.86	0.88	0.85	0.88	0.88

ent $\mathbf{g}_{t_q}^{\mathrm{ODE}} = \nabla_{\mathbf{x}}(R \circ F_{t_q})(\mathbf{x}_q)$, differentiated through the numerical ODE completion, VAE, and PickScore (Figure 2).

We average signed cosines within prompts with time weights proportional to t_q and use prompt-bootstrap resampling for confidence intervals. The clean-image direction has higher cosine with the real gradient, although its advantage diminishes near the last query (Figure 2).

Using the actual trajectory bank, the direct estimate also has positive cosine with $\mathbf{g}_{t_q}^{\mathrm{ODE}}$ at update 200.

E.4 PickScore training-seed replication

We test the PickScore training recipe with multiple seeds. Figure 7 shows their learning curves.

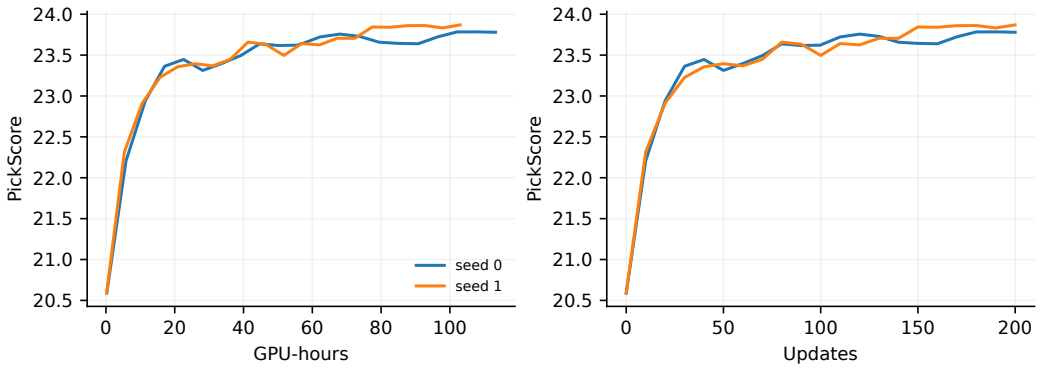


Figure 7: PickScore training across seeds, shown against GPU-hours (left) and updates (right).

F Implementation and reproducibility protocol

Reward-adaptive rollouts, one clean-image regression. Table 6 gives the configurations for Algorithm 1. All use SD3.5-Medium, the ten-step shifted SD3 grid, CFG $= 1$, and one parameter update per collection. The reference rows and update normalization for the sampling schemes in Section 5.1 are specified below.

Exploration transition. With $\Delta t_q = t_{q+1} - t_q < 0$ and time-dependent diffusion coefficient γ_q, its interior-step discretization is

$$\begin{aligned} \mathbf{x}_{i,q+1} &= \mathbf{x}_{i,q} + \Delta t_q \left[\mathbf{v}_{\boldsymbol{\theta}_{\mathrm{old}}}(\mathbf{x}_{i,q}, t_q, c) + \frac{\gamma_q^2}{2t_q} \widehat{\boldsymbol{\epsilon}}_{i,q} \right] + \gamma_q \sqrt{-\Delta t_q}\, \boldsymbol{\xi}_{i,q}, \\ \widehat{\boldsymbol{\epsilon}}_{i,q} &= \mathbf{x}_{i,q} + (1 - t_q) \mathbf{v}_{\boldsymbol{\theta}_{\mathrm{old}}}(\mathbf{x}_{i,q}, t_q, c), \qquad \boldsymbol{\xi}_{i,q} \sim \mathcal{N}(0, I). \end{aligned} \tag{36}$$

Noise coefficient and boundaries. The configuration's constant noise mode holds the radius parameter η fixed, not the per-transition standard deviation. The recorded coefficient is

$$\gamma_q^2 = \frac{\eta^2 t_q}{1 - \widetilde{t}_q}, \qquad \widetilde{t}_q = \begin{cases} t_1, & q = 0, \\ t_q, & q > 0. \end{cases} \tag{37}$$

The factor $\sqrt{-\Delta t_q}$ makes the perturbation step-dependent. Using $1 - t_1$ at $t_0 = 1$ avoids the singular denominator.

Table 6: Configurations at the reported checkpoints. OCR floor uses Eq. (43) with $\lambda_{\text{floor}} = 10$ and the margin schedule $m = 0 \to 0.03 \to 0.06$ at updates 40 and 100.

Setting	PickScore	OCR	GenEval	OCR floor
Estimator	bank	bank	group	group
Prompts / update	24	8	48	48
Trajectories / prompt	24	12	24	24
Trained trajectories / prompt	24	12	6 lowest	6 lowest
Rollout	SDE, $\eta = 0.03$	SDE, $\eta = 0.5$	ODE	ODE
Reward evaluations / trajectory	11	11	1	1
Regression rows	head-8	head-8	23 siblings	23 siblings
Ridge λ_{sum}	0.3	0.3	0.3	0.3
Time weights	$\omega_q \propto t_q$	$\omega_q \propto t_q$	$\omega_q \propto t_q$	$\omega_q \propto t_q$
Initial learning rate	3×10^{-4}	3×10^{-4}	3×10^{-4}	3×10^{-4}
Reported update	200	120	400	150
GPU-hours (8×A800)	110	19.5	129	41

Algorithm 1 One DiffusionRVGE update with centered clean-image regression

Require: Collection parameters $\boldsymbol{\theta}_{\text{old}}$, reward R, grid $t_0 > \cdots > t_N = 0$, ridge λ_{sum}
Require: Rollout and query-weight rules (Section 5.1)
Sample prompts c_i and collect query trajectories $\{\mathbf{x}_{i,q}\}_{q=0}^{N}$.
Complete rollout states to clean images $\mathbf{z}_j$ and evaluate $r_j = R(\mathbf{z}_j, c_j)$.
Assign actor weights a_i from the collected rewards; set $Z_a = \sum_i a_i$.
for all query trajectories i with $a_i > 0$ and times $q = 0, \dots, N-1$ **do**
    Form same-prompt references $\mathcal{B}_{i,q}$ and set the rollout-dependent image and reward centers $(\widehat{\mathbf{m}}_{i,q}, \widehat{\bar{r}}_{i,q})$.
    for all $j \in \mathcal{B}_{i,q}$ **do**
        $\mathbf{d}_{ij} \leftarrow \mathbf{z}_j - \widehat{\mathbf{m}}_{i,q}$; $b_{ij} \leftarrow r_j - \widehat{\bar{r}}_{i,q}$.
    end for
    Discard zero displacements; set $\ell_{ij} = \|\mathbf{d}_{ij}\|_2$ for each retained row.
    Stack $\mathbf{d}_{ij}^{\top}/\ell_{ij}$ into $Q_{i,q}$ and b_{ij}/ℓ_{ij} into $\mathbf{b}_{i,q}$.
    $\widehat{\mathbf{u}}_{i,q} \leftarrow Q_{i,q}^{\top}(Q_{i,q}Q_{i,q}^{\top} + \lambda_{\text{sum}} I)^{-1}\mathbf{b}_{i,q}$; use 0 for an empty fit.
end for
Set $\omega_q = t_q / (N^{-1} \sum_{p=0}^{N-1} t_p)$; detach directions, weights, and Z_a.
if $Z_a > 0$ **then**
    $\delta\mathbf{v}_{\boldsymbol{\theta},i,q} \leftarrow \mathbf{v}_{\boldsymbol{\theta}}(\mathbf{x}_{i,q}, t_q, c_i) - \text{sg}[\mathbf{v}_{\boldsymbol{\theta}_{\text{old}}}(\mathbf{x}_{i,q}, t_q, c_i)]$.
    $\mathcal{L} \leftarrow \dfrac{2}{NZ_a} \sum_{i:a_i>0} \sum_{q=0}^{N-1} a_i \omega_q \langle \delta\mathbf{v}_{\boldsymbol{\theta},i,q}, \text{sg}[\widehat{\mathbf{u}}_{i,q}] \rangle$.
    Starting from $\boldsymbol{\theta} = \boldsymbol{\theta}_{\text{old}}$, take one parameter update to minimize $\mathcal{L}$.
end if

Regression rows and solve. Equations (10)–(12) state the fit for a generic query. Here i indexes trajectories in a prompt group, q indexes query time, and $j \in \mathcal{B}_{i,q}$ indexes reference completions; $(\widehat{\mathbf{m}}_{i,q}, \widehat{\bar{r}}_{i,q})$ denote the centers in Eq. (10) at query time t_q. For each retained nonzero displacement, define

$$\boldsymbol{\phi}_{i,q,j} = \frac{\mathbf{z}_j - \widehat{\mathbf{m}}_{i,q}}{\|\mathbf{z}_j - \widehat{\mathbf{m}}_{i,q}\|_2}, \qquad \widetilde{b}_{i,q,j} = \frac{r_j - \widehat{\bar{r}}_{i,q}}{\|\mathbf{z}_j - \widehat{\mathbf{m}}_{i,q}\|_2}. \tag{38}$$

Stack $\boldsymbol{\phi}_{i,q,j}^{\top}$ into $Q_{i,q}$ and the scalar targets $\widetilde{b}_{i,q,j}$ into $\mathbf{b}_{i,q}$. The fitted coefficient is

$$\widehat{\mathbf{u}}_{i,q} = \arg\min_{\mathbf{u}} \left\{ \|Q_{i,q}\mathbf{u} - \mathbf{b}_{i,q}\|_2^2 + \lambda_{\text{sum}}\|\mathbf{u}\|_2^2 \right\} = Q_{i,q}^{\top}(Q_{i,q}Q_{i,q}^{\top} + \lambda_{\text{sum}} I)^{-1}\mathbf{b}_{i,q}. \tag{39}$$

This is Eq. (12) with $\ell_{ij} = \|\mathbf{d}_{ij}\|$, before scaling the direction by κ_t; the sum-loss ridge coefficient is $\lambda_{\text{sum}} = 0.3$. The dual solve scales with the number of observations. The fitted coefficient $\widehat{\mathbf{u}}$ summarizes the observed reward changes across the chosen clean-image contrasts.

Trajectory-bank samples. For the trajectory bank, use $\mathcal{B}_{i,q} = \{(i,m) : 0 \le m < 8,\ m \ne q\}$ on the ten-step grid, with

$$\mathbf{z}_{i,q} = F_{t_q}^{\boldsymbol{\theta}_{\text{old}}}(\mathbf{x}_{i,q}), \qquad \widehat{\mathbf{m}}_{i,q} = \mathbf{z}_{i,q}, \qquad \widehat{\bar{r}}_{i,q} = R(\mathbf{z}_{i,q}). \tag{40}$$

The numerators in Eq. (38) become $\mathbf{z}_{i,m} - \mathbf{z}_{i,q}$ and $r_{i,m} - r_{i,q}$. This head-8 reference set avoids the shortest late-stage chords. All retained completions enter one regression with the same rule.

Independent-seed group samples. For independent-seed groups, use $\mathcal{B}_{i,q} = \{j : j \neq i\}$, with

$$\widehat{\mathbf{m}}_{i,q} = \bar{\mathbf{z}}_{-i} = \frac{1}{K-1}\sum_{j\neq i}\mathbf{z}_j, \qquad \widehat{\bar{r}}_{i,q} = \bar{r}_{-i} = \frac{1}{K-1}\sum_{j\neq i} r_j. \tag{41}$$

The raw rows satisfy $\sum_{j\neq i}\mathbf{d}_{ij} = 0$ and $\sum_{j\neq i} b_{ij} = 0$. Normalize these rows as in Eq. (38), solve Eq. (39), and reuse the resulting $\widehat{\mathbf{u}}_i$ at every query time on trajectory i.

Discrete velocity loss and normalization. The reported updates implement Eq. (14) on the N-step grid. For B collected trajectories with stopped query weights a_i and $Z_a = \sum_{i=1}^{B} a_i > 0$,

$$\mathcal{L}_{\text{linear}}(\boldsymbol{\theta};\boldsymbol{\theta}_{\text{old}}) = \frac{1}{N\,\text{sg}[Z_a]}\sum_{i=1}^{B}\sum_{q=0}^{N-1} 2\omega_q\,\text{sg}[a_i]\,\langle \mathbf{v}_{\boldsymbol{\theta}}(\mathbf{x}_{i,q},t_q,c_i) - \text{sg}[\mathbf{v}_{\boldsymbol{\theta}_{\text{old}}}(\mathbf{x}_{i,q},t_q,c_i)], \text{sg}[\widehat{\mathbf{u}}_{i,q}]\rangle\,, \tag{42}$$

and the loss is zero when $Z_a = 0$. Here $\omega_q = t_q/(N^{-1}\sum_p t_p)$ is the normalized empirical product $\omega_{t_q} = w_{t_q}\chi_{t_q}\kappa_{t_q}$ from Section 4. Equation (42) is the discrete form of Eq. (14), with t normalized to unit mean over the grid and the average taken over queries with $a_i > 0$. We set $a_i = 1$ for trajectory banks and $a_i = \mathbf{1}\{i \in \text{Lowest}_6(\mathbf{r}_{c_i})\}$ for GenEval, breaking ties randomly. Both binary selection and the continuous weights compared in Section 6.4 act on the stopped actor term after the fit.

Both weights and denominator are stopped, and a zero denominator gives zero loss. Setting all $a_i = 1$ recovers the unweighted batch average; binary lowest-six weights recover the selected-sample average exactly. Zero-weight queries require no actor backward pass. Every reported run uses these unit-mean linear time weights. All-equal rewards give a zero mean-centered group direction even if rank selection retains six queries. The stricter combination $\mathbf{1}\{r_i < 1\}\mathbf{1}\{i \in \text{Lowest}_6\}$ is distinct from the ungated lowest-six reference and is not silently substituted for it.

F.1 Five-reward joint training

The joint model of Table 1 alternates the two update types above on DiffusionNFT's five training rewards. With $q = \text{PickScore}/26 + \text{ClipScore} + \text{HPSv2.1}$, Pick-a-Pic prompts receive $r = 10q$, and GenEval and OCR prompts receive their task reward with the floor of Eq. (43), $r = R_{\text{task}} + 10\min(q - \bar{q}_c + m, 0)$, where $m = 0$ before update 700 and $m = 0.03$ afterwards. A group update draws 48 prompts from the listed sources, rolls out 24 ODE trajectories per prompt, and fits Algorithm 1 with the centers in Eq. (41). It trains the six lowest-reward trajectories of each prompt, restricted on GenEval and OCR prompts to trajectories with task reward below 0.99. A bank update uses the PickScore configuration of Table 6 with reward $10q$. The GenEval task reward is the partial-credit score up to update 450 and the mean of the partial-credit and strict judgements afterwards. Updates 500–600 use 40 sampling steps with actor loss on every fourth state; all other stages use 10 steps. The learning rate for updates 700–850 is reached after a 20-update warm-up. Table 7 lists the stages.

Table 7: Stages of the five-reward joint model. The prompt mix gives the fractions of Pick-a-Pic, GenEval, and OCR prompts per update; the first stage is the first training seed of the single-reward PickScore run in Table 2.

Updates	Update type	Prompt mix	Learning rate
0–200	bank, PickScore	1 / 0 / 0	3×10^{-4}
200–500	group	0.20 / 0.45 / 0.35	3×10^{-4}
500–600	group	0.20 / 0.60 / 0.20	1.35×10^{-4}
600–700	bank	1 / 0 / 0	7.25×10^{-5}
700–850	group	0 / 0 / 1	7.25×10^{-5}
850–1000	group	0.20 / 0.60 / 0.20	8.75×10^{-5}

Compute. The full joint model uses 404 GPU-hours from SD3.5-Medium. DiffusionNFT does not report the cost of its joint model. Its described stages total $800+300+200+200+100 = 1{,}600$ iterations with 40 sampling steps; its Table 1 lists 1.7k, and we use the smaller count. Timing its official code on our hardware and accounting for the described schedule imply at least $2{,}000$ GPU-hours in total, or at least $5\times$ our budget.

G Measurement provenance

Our single-reward and five-reward evaluations use the same protocols and datasets as DiffusionNFT, enabling direct comparison (Section 5.1). Figure 1 uses one image per DrawBench prompt for the PickScore curves, alongside the OCR and GenEval tests. Sampling counts can differ between checkpoint evaluations and learning-curve readouts, so their terminal values need not coincide.

Each curve retains its recorded cumulative GPU time. The source includes saved CSVs, a checksum manifest, and figure-generation scripts. Budget comparisons interpolate within observed runs; no missing result is extrapolated.

H Reward exploitation

H.1 Scheduled quality floor for OCR

OCR-only training replaces requested scenes with large text on a flat background; on the main DrawBench evaluation, its PickScore is 20.32 versus 22.34 for the base model (Table 2). We counter this with a group-relative quality floor, using $q = \mathrm{PickScore}/26 + \mathrm{ClipScore} + \mathrm{HPSv2.1}$ and regressing

$$r = \mathrm{OCR} + \lambda_{\mathrm{floor}} \min(q - \bar{q}_c + m, 0), \qquad \lambda_{\mathrm{floor}} = 10, \tag{43}$$

where $\bar{q}_c$ is the prompt-group mean. A tight initial floor establishes scene content before relaxing the OCR tradeoff: $m = 0$ until update 40, 0.03 until 100, then 0.06. After 150 updates (41 GPU-hours), this run reaches OCR 0.97 and DrawBench PickScore 22.17 while retaining scenes in the inspected samples (Figure 9). The floor mitigates the observed flat-page exploit.

OCR layout comparison. We quantify templating by the mean cross-prompt correlation of 8×8 luminance thumbnails (base 0.05; a fixed layout approaches one). At update 40, the trajectory-bank and ODE-group runs in Section 6.3 have correlations 0.93 and 0.52, respectively, and both produce flat, text-dominated layouts. The scheduled quality floor retains the requested scenes in the inspected samples, with a shared central-sign composition (correlation 0.51). Its DrawBench Aesthetic score is 5.64, compared with 4.65 for the OCR-only checkpoint (Table 2).

Why schedule the margin m? A high-noise reference penalty prevents the template but also removes the OCR signal. Training only failing trajectories still produces the flat-page layout, and a linear PickScore term preserves content but stalls OCR near 0.85. A floor with a fixed reference (PickScore two points above the base model) preserves scenes and reaches 0.93 after 120 updates but stops there, because once the model's typical quality exceeds the reference the floor is silent; the relative floor of Eq. (43) follows the model. Starting from initialization with the loose margin $m = 0.06$ and $\lambda_{\mathrm{floor}} = 10$ reaches OCR 0.95 within 40 updates, but on a cartoon-signboard template whose PickScore stays at the base level, so the quality rewards do not see it. The schedule starts with a tight margin to establish scene content, then relaxes it to improve OCR. The scheduled floor reaches 0.97 with scenes preserved; its remaining errors are mostly single glyphs in long strings, and the surviving style is a shared neon rendering of the text.

Prompt: A pink colored car.

RVGE



Flow-GRPO



Prompt: Three cars on the street.

RVGE



Flow-GRPO



Figure 8: PickScore visual results comparing the DiffusionRVGE PickScore checkpoint with the public Flow-GRPO PickScore checkpoint on DrawBench prompts. Within each pair, models use the same prompt, seed, and 40-step ODE sampler.

Prompt: Hotel key card sleeve with “Room 1408 Access”.

RVGE + floor



Flow-GRPO



Prompt: Botanical garden map titled “Tropical Zone Ahead”.

RVGE + floor



Flow-GRPO



Figure 9: OCR visual results comparing the DiffusionRVGE quality-floor checkpoint with the public Flow-GRPO Text checkpoint on OCR test prompts. Within each pair, models use the same prompt, seed, and 40-step ODE sampler; panel labels abbreviate the prompts.

Prompt: A yellow hair dryer and a green apple.

RVGE GenEval



Flow-GRPO GenEval



Prompt: A white suitcase.

RVGE GenEval



Flow-GRPO GenEval



Figure 10: GenEval visual results comparing the DiffusionRVGE single-reward GenEval checkpoint with the public Flow-GRPO GenEval checkpoint on GenEval prompts. Within each pair, models use the same prompt, seed, and 40-step ODE sampler.

Prompt: A triangular purple flower pot.

RVGE joint



DiffusionNFT



Prompt: A medieval painting of the wifi not working.

RVGE joint



DiffusionNFT



Figure 11: DrawBench visual results comparing the five-reward joint DiffusionRVGE checkpoint with the public DiffusionNFT multi-reward checkpoint. Each pair uses the same full prompt and seed with 40-step ODE sampling; panel labels abbreviate the prompts.

Prompt: A street sign reading "wowomen".

RVGE joint



DiffusionNFT



Prompt: A fire-truck door reading "Engine Company 8".

RVGE joint



DiffusionNFT



Figure 12: OCR test visual results comparing the five-reward joint DiffusionRVGE checkpoint with the public DiffusionNFT multi-reward checkpoint. Each pair uses the same full prompt and seed with 40-step ODE sampling; panel labels abbreviate the prompts.