GRADIENT-FREE NOISE OPTIMIZATION FOR REWARD ALIGNMENT IN GENERATIVE MODELS

Anonymous authors
Paper under double-blind review



Figure 1: Representative results of ZeNO across diverse one-step generators and reward functions. Our noise optimization consistently improves image quality and prompt alignment without generator modification or backpropagation. Generator specification and the corresponding text prompts are reported in Appendix I.

ABSTRACT

Existing reward alignment methods for diffusion and flow models rely on multi-step stochastic trajectories, making them difficult to extend to deterministic generators. A natural alternative is noise-space optimization, but existing approaches require backpropagation through the generator and reward pipeline, limiting applicability to differentiable settings. To address this, here we present ZeNO (**Ze**roth-order **No**ise **O**ptimization), a gradient-free framework that formulates noise optimization as a path-integral control problem, estimable from zeroth-order reward evaluations. When instantiated with an Ornstein–Uhlenbeck reference process and a one-step receding horizon, the repeated update connects to Langevin dynamics targeting a reward-tilted distribution. ZeNO enables effective inference-time scaling and demonstrates strong performance across diverse generators and reward functions, including a protein structure generation task where backpropagation is infeasible.

1 INTRODUCTION

Post-training aims to align a pre-trained generative model with desired objectives, often specified by a reward function such as prompt fidelity, aesthetic quality, human preference, etc. A broad class of recent post-training methods improves diffusion or flow models by optimizing them toward such rewards while keeping them close to a reference model. For example, DDPO (Black et al., 2024), Flow-GRPO (Liu et al., 2026a), PCPO (Lee & Ye, 2026), and SQDF (Kang et al., 2026) formulate

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.

alignment from a sequential Markov decision process perspective, whereas VGG-Flow (Liu et al., 2026b) and Residual ∇-DB (Liu et al., 2025) are motivated by optimal control and GFlowNet-style principles.

Despite their different motivations, these methods are primarily developed for settings in which generation is represented as a multi-step stochastic trajectory. This trajectory-level structure is central to their formulation, and consequently, extending these approaches to deterministic or one-step generators is not straightforward, since such generators do not naturally expose the same stochastic trajectory structure. Recent advances in generative modeling have led to the development of few-step or one-step deterministic generators, such as consistency models (Song et al., 2023; Luo et al., 2023) and distilled diffusion models (Yin et al., 2024b;a; Sauer et al., 2024b; Geng et al., 2025), which achieve high-quality generation at a fraction of the computational cost of their multi-step counterparts (Yin et al., 2024b). Aligning these models with reward functions is therefore of growing practical importance, yet existing post-training methods are not directly applicable to this.

A natural alternative is to optimize the input noise of a deterministic generator directly (Eyring et al., 2024; Min et al., 2026). Given a generator G_θ and a reward function r, one could backpropagate the terminal reward through the generator and perform gradient ascent in the noise space. However, this approach requires differentiating through the generator and reward pipeline, and can be unreliable when gradients reflect spurious local sensitivities rather than meaningful semantic improvement. Moreover, it is inapplicable when the reward is non-differentiable or when the generator is too large to support efficient backpropagation. For example, ReNO (Eyring et al., 2024) and ORIGEN (Min et al., 2026) operate in the noise space and connect noise optimization to gradient ascent or Langevin dynamics targeting a reward-tilted distribution, but still require backpropagation through the reward-generator composition, limiting their applicability to differentiable pipelines.

In this paper, we retain the noise-space perspective but recast noise optimization as a path-integral control (PIC) problem on the initial noise space of a deterministic generator, in which a control steers the noise, evolving under a reference process, toward high-reward samples. We call the resulting framework **ZeNO** (**Ze**roth-order **No**ise **O**ptimization), which brings three properties. First, the optimal control depends only on reward evaluations, requiring no gradient of the generator or the reward. Second, since the generator maps noise to a clean sample in one step, the terminal cost is always evaluated on a clean sample, avoiding the Tweedie-type approximation of PIC-style guidance along diffusion trajectories. Third, an Ornstein–Uhlenbeck (OU) reference process keeps the noise compatible with the pre-trained generator, and repeating the one-step receding-horizon control amounts to Langevin sampling of a reward-tilted distribution. In practice, we replace the exponential path-integral estimator, which degenerates with finitely many samples, with a linearized estimator whose error is provably bounded and which yields significantly higher reward under the same compute budget.

Empirically, ZeNO achieves competitive or superior performance compared to gradient-based methods across a range of generative models and reward functions, including non-differentiable reward, and enables effective inference-time scaling as more computation is allocated. We further validate the broad applicability of ZeNO on a protein structure generation,the designability reward is a non-differentiable pipeline of inverse folding, refolding, and structural alignment, so that gradient-based methods are inapplicable, demonstrating strong performance even under this challenging constraint.

2 BACKGROUND

Stochastic Differential Equation. For a stochastic process $\boldsymbol{x}_t$, a stochastic differential equation (SDE) is written as

$$d\boldsymbol{x}_t = f(\boldsymbol{x}_t, t)dt + g(t)d\boldsymbol{w}_t \tag{1}$$

where $f(\boldsymbol{x}_t, t)$ and $g(t)$ denote the drift and diffusion coefficients, and $\boldsymbol{w}_t$ denotes a Wiener process. A representative example is the Ornstein–Uhlenbeck (OU) process, obtained when $f(\boldsymbol{x}_t, t) = -a\boldsymbol{x}_t$ and $g(t) = b$ for $a > 0$ and $b > 0$. This process has the Gaussian stationary distribution $\mathcal{N}(0, b^2/2a)$ (Øksendal, 2003). In particular, when $2a = b^2$, the stationary distribution becomes $\mathcal{N}(0, \mathbf{I})$.

Reward Tilted Distribution. In post-training generative models to maximize rewards, the alignment problem is often formulated as an entropy-regularized reinforcement learning objective over the

generative trajectory. Let P^{ref} denote the trajectory distribution induced by the pretrained generative model, and let Q^θ denote the trajectory distribution induced by the finetuned model parameterized by θ. The objective can then be written as

$$\mathcal{L}(\theta) = \mathbb{E}_{\boldsymbol{x}_0 \sim q_0^\theta}\left[r(\boldsymbol{x}_0)\right] - \lambda D_{\mathrm{KL}}\left(Q^\theta \,\|\, P^{\mathrm{ref}}\right), \tag{2}$$

where the first term encourages the model to generate high-reward samples, while the KL regularization term penalizes excessive deviation from the pretrained model. The coefficient $\lambda > 0$ controls the trade-off between reward maximization and staying close to the reference generative process. Equivalently, this KL term can often be decomposed into a sum or integral of per-step policy divergences along the sampling trajectory, depending on the specific generative process. The marginal distribution induced by the optimal policy of Eq. (2) is given by (Uehara et al., 2024b)

$$p_t^\star(\boldsymbol{x}_t) \propto p_t^{\mathrm{ref}}(\boldsymbol{x}_t) e^{V(\boldsymbol{x}_t)}, \quad V(\boldsymbol{x}_t) := \log \mathbb{E}_{P^{\mathrm{ref}}}\left[\exp\left(\frac{r(\boldsymbol{x}_0)}{\lambda}\right)\middle|\boldsymbol{x}_t\right], \tag{3}$$

for $t \in [0,1]$, where $\boldsymbol{x}_0$ refers to the clean sample. This shows that the optimal marginal is obtained by tilting the reference marginal according to the conditional desirability of future high-reward samples. Notably, this tilted form also applies at $t = 1$, corresponding to the initial noise distribution, which is particularly relevant to settings where optimization is performed directly in the initial noise space.

Path Integral Control. Let $\boldsymbol{u}(\boldsymbol{x}_t, t)$ denote a control input that guides the reference SDE

$$d\boldsymbol{x}_t = f(\boldsymbol{x}_t, t)dt + g(t)\left(\boldsymbol{u}(\boldsymbol{x}_t, t)dt + d\boldsymbol{w}_t\right). \tag{4}$$

We seek a control that steers the trajectory toward high-reward terminal states while regularizing control effort. To this end, we consider the terminal-cost optimal control objective

$$J_{\boldsymbol{u}}(\boldsymbol{x}_t, t) = \mathbb{E}\left[m(\boldsymbol{x}_T) + \lambda \int_t^T \frac{1}{2}\|\boldsymbol{u}(\boldsymbol{x}_\tau, \tau)\|^2 d\tau\right], \tag{5}$$

where $m(\boldsymbol{x}_T)$ denotes the terminal cost and the running cost penalizes the control energy. Weighted by λ, this running cost equals the KL divergence between the controlled and reference path measures, so Eq. (5) is the control form of the KL-regularized objective in Eq. (2). Since the objective contains no intermediate-state cost, the optimal policy is driven solely by the terminal desirability of the trajectory subject to minimum control effort. Under this formulation, the optimal control has the path-integral form (Kappen, 2005; Huang et al., 2024; Uehara et al., 2024a).

$$\boldsymbol{u}^\star(\boldsymbol{x}_t, t) = \frac{\mathbb{E}_{\boldsymbol{x}_{t:T}}\left[\exp\left(-m(\boldsymbol{x}_T)/\lambda\right) d\boldsymbol{w}_t\right]}{\mathbb{E}_{\boldsymbol{x}_{t:T}}\left[\exp\left(-m(\boldsymbol{x}_T)/\lambda\right)\right]}, \tag{6}$$

where the expectation is taken over trajectories of the uncontrolled reference SDE in Eq. (4). This shows that the control can be estimated from reward-weighted trajectory samples and depends only on terminal-cost evaluations, without requiring differentiation through the reward or the generator.

A practical difficulty is that evaluating $\boldsymbol{u}^\star$ over the full horizon to T requires repeated rollout of Eq. (4), which becomes costly in long-horizon settings. A standard approximation is to truncate the horizon to $H < T$ and replace $m(\boldsymbol{x}_T)$ with a surrogate terminal cost evaluated after a shorter rollout. Although this reduces computation, it is valid only when the intermediate state X_{t+H} remains in the domain on which m is defined. In diffusion models, however, X_{t+H} is generally a noisy latent state, whereas reward models are typically defined on clean samples. This mismatch makes naive truncation inappropriate, and prior approaches therefore rely on Tweedie-type estimators to map noisy intermediate states to an estimate of the underlying clean sample (Chung et al., 2023; Kim et al., 2025b;c).

In contrast, ZeNO avoids this entirely by operating directly in the initial noise space, where the generator maps noise to clean samples in a single deterministic step, ensuring the terminal cost is always evaluated on a valid clean sample.

3 METHOD

We aim to align a deterministic generator with a reward function that may be non-differentiable. Since one-step deterministic generators do not fit the multi-step policy-learning framework of diffusion post-training (Black et al., 2024; Liu et al., 2025; 2026b;a), and their output is fully determined by the initial noise, we formulate alignment as a stochastic control problem in the input noise space whose optimal update can be estimated without gradients.

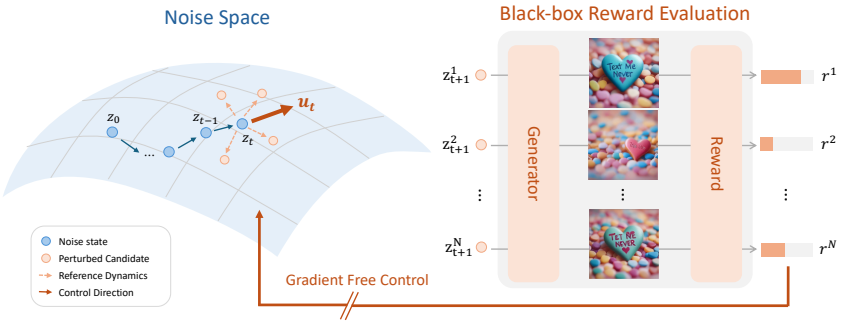


Figure 2: Overview of the ZeNO framework. At each iteration, N perturbed noise candidates $\boldsymbol{z}_{t+1}^{1:N}$ are sampled from the reference dynamics around the current noise state $\boldsymbol{z}_t$ and evaluated by the reward model through the generator G_θ. The reward-weighted perturbations are aggregated to estimate the control direction $\boldsymbol{u}_t$, which steers the noise toward higher-reward regions without requiring gradients through the generator or the reward model.

3.1 Problem Definition

Let $\boldsymbol{z}_0 \sim \mathcal{N}(0, \mathbf{I})$ denote the initial noise of a deterministic generator $G_\theta : \mathbb{R}^d \times \mathcal{C} \to \mathbb{R}^d$ with conditions $\mathcal{C}$ such as text embeddings, which we write as $G_\theta(\boldsymbol{z})$ for notational simplicity. Given a reward function $r : \mathbb{R}^d \to \mathbb{R}$, our goal is to construct a noise trajectory $\{\boldsymbol{z}_t\}_{t=0}^T$ from $\boldsymbol{z}_0$ whose terminal state $\boldsymbol{z}_T$ achieves a high reward. We formalize this by introducing a controlled SDE on the noise space and defining the terminal cost as $m(\boldsymbol{z}_T) := -r(G_\theta(\boldsymbol{z}_T))$.

3.2 Derivation

When we solve the path-integral control problem and obtain a control vector $\boldsymbol{u}(\boldsymbol{z}, t)$, the initial noise is updated by the controlled SDE:

$$d\boldsymbol{z} = f(\boldsymbol{z}, t)dt + g(t)\left(\boldsymbol{u}(\boldsymbol{z}, t)dt + d\boldsymbol{w}_t\right), \tag{7}$$

where $f(\boldsymbol{z}, t)$ and $g(t)$ denote the drift and diffusion coefficients, respectively, $d\boldsymbol{w}_t$ is a Wiener process, and $\boldsymbol{u}(\boldsymbol{z}, t)$ is the control vector. In the absence of control, i.e., $\boldsymbol{u}(\boldsymbol{z}, t) = 0$, this reduces to the uncontrolled reference dynamics. Given the terminal cost $m(\boldsymbol{z}_T)$, the optimal control becomes:

$$\boldsymbol{u}^\star(\boldsymbol{z}, t) = \frac{\mathbb{E}_{\boldsymbol{z}_{t:T}}\left[\exp\left(r(G_\theta(\boldsymbol{z}_T))/\lambda\right) d\boldsymbol{w}_t\right]}{\mathbb{E}_{\boldsymbol{z}_{t:T}}\left[\exp\left(r(G_\theta(\boldsymbol{z}_T))/\lambda\right)\right]}, \tag{8}$$

where the expectation is over trajectories induced by the uncontrolled reference dynamics, and $\boldsymbol{u}^\star$ here denotes the control increment over dt, whose scale is absorbed into η.

The path-integral control suffers from high variance due to the exponential weighting, and its Monte Carlo estimate is self-normalized, biased, and loses effective samples (Owen, 2013). As a remedy, we subtract the mean reward as a baseline and linearize the exponential weight around it, which removes the self-normalization and yields the following estimator and error bound (proof in Appendix B).

Proposition 1. *Let $r := r(G_\theta(\boldsymbol{z}_T))$ and $\bar{r} := \mathbb{E}[r]$, and assume $|r - \bar{r}| \le R$ almost surely. With $\bar{r}$ as the baseline, Eq. (8) can be linearized as*

$$\boldsymbol{u}_{\text{lin}} := \mathbb{E}[(r - \bar{r})d\boldsymbol{w}_t]/\lambda \tag{9}$$

with the error bounded by

$$\|\boldsymbol{u}^\star - \boldsymbol{u}_{\text{lin}}\| \le \frac{e^{R/\lambda}}{2} \frac{R\sqrt{\text{Var}(r)}}{\lambda^2} \left(1 + \frac{\sqrt{\text{Var}(r)}}{\lambda}\right) \sqrt{dt}, \tag{10}$$

which is at most $eR\sqrt{\text{Var}(r)}\sqrt{dt}/\lambda^2$ for $\lambda \ge R$.

The expectation of Eq. (9) is over the uncontrolled reference dynamics. The resulting estimator emphasizes perturbations that produce above-average reward while suppressing those that do not. By Proposition 1, the linearization error is small relative to the control scale once $\lambda \gg R$, requiring only a bounded reward. The PIC formulation further places this estimator inside a controlled SDE, whose reference dynamics we choose next.

Choice of Reference SDE. The reference dynamics determine how the noise evolves in the absence of control, and thus directly affect the compatibility of the updated noise $\boldsymbol{z}_T$ with the pre-trained generator. Among common choices, variance-exploding (VE) SDEs lead to uncontrolled growth in noise variance, making the resulting noise incompatible with pre-trained generative models. We therefore adopt the Ornstein–Uhlenbeck (OU) process:

$$f(\boldsymbol{z}, t) := -\frac{a_t}{2}\boldsymbol{z}, \quad g(t) := \sqrt{a_t} \tag{11}$$

which corresponds to a variance-preserving (VP) SDE with time-dependent variable a_t. Then, the discretized controlled OU process is given by

$$\boldsymbol{z} \leftarrow \sqrt{1-\beta}\boldsymbol{z} + \sqrt{\beta}\boldsymbol{\epsilon} + \eta\boldsymbol{u}^\star, \tag{12}$$

where $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I})$ and $\beta := a_t \Delta t$ is treated as a constant discretization coefficient. We use a small β throughout, so that Eq. (12) is a fine discretization of the controlled SDE ($\beta = 0.01$ in all experiments). Here, the control step size η absorbs the coefficients arising from the Euler–Maruyama discretization and the linear approximation introduced in Proposition 1. The uncontrolled OU process preserves $\mathcal{N}(0, \mathbf{I})$, so perturbed noises remain compatible with the pre-trained generator. With control, the update becomes Langevin sampling of a reward-tilted distribution (Proposition 2). For protein frames, Appendix I.2 uses an analogous reference process on $SE(3)^N$.

Algorithm 1 Algorithm of ZeNO

Require: Generator G_θ, reward r, initial noise $\boldsymbol{z}$, reference coefficient β, step size η, number of particles N, number of iterations M

$\boldsymbol{z}_0 \leftarrow \boldsymbol{z}$
for $m = 0, \dots, M-1$ **do**
  for $n = 1, \dots, N$ **do** ▷ in parallel
    $\boldsymbol{\epsilon}^{(n)} \sim \mathcal{N}(0, \mathbf{I})$
    $\boldsymbol{z}_m^{(n)} \leftarrow \sqrt{1-\beta}\boldsymbol{z}_m + \sqrt{\beta}\boldsymbol{\epsilon}^{(n)}$
    $r_m^{(n)} \leftarrow r(G_\theta(\boldsymbol{z}_m^{(n)}))$
  end for
  $\bar{r}_m \leftarrow \frac{1}{N}\sum_n r_m^{(n)}$
  $\boldsymbol{u}_m \leftarrow \frac{1}{N}\sum_n (r_m^{(n)} - \bar{r}_m)\boldsymbol{\epsilon}^{(n)}$
  $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I})$
  $\boldsymbol{z}_{m+1} \leftarrow \sqrt{1-\beta}\boldsymbol{z}_m + \sqrt{\beta}\boldsymbol{\epsilon} + \eta\boldsymbol{u}_m$
  $\boldsymbol{z}_{m+1} \leftarrow \boldsymbol{z}_{m+1}\sqrt{d}/\|\boldsymbol{z}_{m+1}\|$
end for
return $G_\theta(\boldsymbol{z}_M)$

One-step receding-horizon. Since the reference dynamics is an SDE on the noise space, the noise injected after step t accumulates in $\boldsymbol{z}_{t+H}$, so the relative contribution of $d\boldsymbol{w}_t$ to $\boldsymbol{z}_{t+H}$ decreases as H grows (Lemma 1). This motivates the one-step receding-horizon approximation ($H = 1$), where we replace $\boldsymbol{z}_T$ with a one-step proposal $\boldsymbol{z}' = \sqrt{1-\beta}\boldsymbol{z} + \sqrt{\beta}\boldsymbol{\epsilon}$ and estimate $\boldsymbol{u}_{\text{lin}}$ from N proposals by $\frac{1}{N}\sum_n (r^{(n)} - \bar{r})\,\boldsymbol{\epsilon}^{(n)}$ up to a constant absorbed into η, with $\bar{r}$ the mean reward.

Connection to Langevin dynamics. With $H = 1$, the controlled OU update turns out to be a discretized Langevin dynamics, as formalized below.

Proposition 2. *When $\beta \ll 1$ and the control is computed with $H = 1$, the discretized controlled OU process Eq. (12) is the Langevin step*

$$\boldsymbol{z} \leftarrow \boldsymbol{z} + \frac{\beta}{2}\nabla_{\boldsymbol{z}} \log p_\kappa^\star(\boldsymbol{z}) + \sqrt{\beta}\boldsymbol{\epsilon}, \tag{13}$$

where $p_\kappa^\star(\boldsymbol{z}) \propto p(\boldsymbol{z}) \exp\big(\kappa V(\boldsymbol{z})\big)$ with $p(\boldsymbol{z}) = \mathcal{N}(0, \mathbf{I})$, $V(\boldsymbol{z}) = \log \mathbb{E}_{\boldsymbol{\epsilon}}[\exp(r(G_\theta(\sqrt{1-\beta}\boldsymbol{z} + \sqrt{\beta}\boldsymbol{\epsilon}))/\lambda)]$ is the value of the one-step problem, and $\kappa = 2\eta/\beta$.

In the noise-space PIC problem, the value at the terminal time is $r(G_\theta(\boldsymbol{z}))/\lambda$, so for a horizon long enough for the reference dynamics to forget the initial noise, the full-horizon problem approximately targets $p^\star(\boldsymbol{z}) \propto p(\boldsymbol{z}) \exp(r(G_\theta(\boldsymbol{z}))/\lambda)$ at the terminal time. Repeating the one-step receding-horizon update, which Proposition 2 identifies as a Langevin step, instead moves the noise toward the stationary distribution $p_\kappa^\star$, which approaches $p^\star$ as $\beta \to 0$ with fixed κ, up to the temperature scaled by κ, coinciding with it for $\kappa = 1$. Since Algorithm 1 uses $\boldsymbol{u}_{\text{lin}}$ instead of $\boldsymbol{u}^\star$, this interpretation holds for the algorithm up to the linearization error bounded in Proposition 1. In the full-horizon control, however, the terminal state is dominated by later perturbations (Lemma 1), whose effect on the reward remains as noise in the control estimate, whereas with $H = 1$ the terminal state varies only through the current perturbation.

Relation to existing methods. The most closely related method is ORIGEN (Min et al., 2026), which also runs Langevin dynamics in the initial noise space toward a reward-tilted distribution. ORIGEN obtains the score by backpropagating through the reward–generator composition, which requires differentiable rewards and reliable gradient flow. ZeNO targets a distribution of the same form, $p_\kappa^\star$ in Proposition 2, but the PIC formulation supplies the drift as an expectation of reward-weighted noise

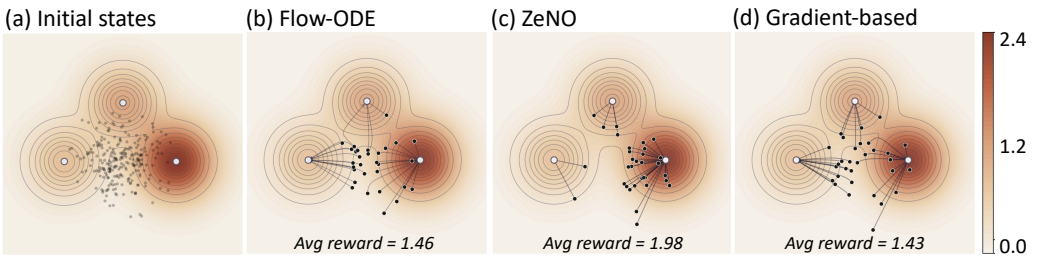


Figure 3: Toy experiment. (a) A 2D Gaussian mixture distribution with reward landscape (red: higher reward). Black dots denote initial noise samples. (b) Probability flow ODE trajectories from initial noises following the score field. (c) Trajectories from updated noises using zeroth-order path integral control (ours), consistently steering samples toward the global high-reward mode. (d) Trajectories from updated noises using first-order gradient-based updates, which tend to be guided toward nearby modes due to their local nature, yielding suboptimal reward.

estimated from one-step reward evaluations alone, so it applies to non-differentiable and black-box rewards.

The proposed method with the OU process is described in Algorithm 1 and depicted in Figure 2. At each of M iterations, N one-step proposals are drawn around the current noise, their rewards are evaluated, and the resulting estimate of $\boldsymbol{u}_{\text{lin}}$ replaces $\boldsymbol{u}^\star$ in Eq. (12) to update the noise. The number of particles N and the number of iterations M thus serve as inference-time scaling knobs, and increasing either allocates more computation toward reward improvement without modifying the generator or reward model (see Section 4.2).

Practical design choice. Although the uncontrolled OU process preserves $\mathcal{N}(0, \mathbf{I})$, the control term can shift the norm of the noise away from $\sqrt{d}$, around which the norm of a standard Gaussian vector in $\mathbb{R}^d$ concentrates. We therefore re-normalize $\boldsymbol{z}$ to norm $\sqrt{d}$ after each update, which keeps the noise compatible with the pre-trained generator. We treat this projection as a practical choice outside Proposition 2.

Table 1: Comparison of empirical mode probabilities and KL divergence from the target reward-tilted distribution.

Method	Mode probability	KL ↓
Target	0.139 / 0.673 / 0.188	–
Ours	0.140 / 0.700 / 0.160	**0.0026**
Grad	0.412 / 0.384 / 0.204	0.2481

4 Experiments

4.1 Toy Experiment

We begin with a two-dimensional toy experiment to illustrate the core behavior of the proposed method. Specifically, we consider a Gaussian mixture model as the target data distribution and define a deterministic generative process G_ϕ via an ODE whose velocity field follows the score of this distribution. Each mode is assigned a distinct reward value, inducing a non-uniform reward landscape over the data space, and the goal is to steer generated samples toward higher-reward modes. Figure 3a illustrates the reward landscape, where brown colors indicate higher reward. Without reward guidance, G_ϕ causes each initial noise to move toward its nearest mode (Figure 3b). When updated using ZeNO, the noise distribution shifts toward high-reward regions without collapsing to a single mode, reflecting the Langevin sampling implicitly induced by the control (Figure 3c). In contrast, first-order gradient updates yield a lower average reward than even the uncontrolled baseline (Figure 3d)[1], as local gradient information can guide noise toward nearby but suboptimal modes in flat or ambiguous regions of the reward landscape. Table 1 further confirms these observations: ZeNO closely matches the target tilted distribution, whereas gradient-based updates deviate significantly, concentrating mass on a subset of modes. While ZeNO is primarily motivated by non-differentiable or black-box reward settings, this result suggests an additional advantage over gradient-based methods in terms of robustness to local sensitivity issues.

[1]Step sizes are set so that the norms of noise updates are matched to those of ZeNO; the gradient-based reward improves only marginally with larger step sizes, confirming the issue is directional.



Figure 4: Qualitative comparison of ZeNO against baselines on SDXL-Turbo, DMD2, and DFGAN with PickScore (rows 1, 2) and aesthetic score (row 3) as reward functions.

4.2 Image Generation Experiments

Generators and Rewards. We consider four generators covering distilled diffusion models, consistency models, and GANs. DMD2 and SDXL-Turbo are distilled diffusion models based on SDXL, via distribution matching distillation (Yin et al., 2024a) and adversarial diffusion distillation (Sauer et al., 2024b;a), respectively; we use their one-step variants. LCM is a consistency model based on SDXL, enabling one-step generation via consistency distillation (Song et al., 2023; Luo et al., 2023). DFGAN (Tao et al., 2022) is a GAN-based generator included to demonstrate that ZeNO is applicable beyond diffusion-based architectures. For reward functions, we use aesthetic score (Schuhmann, 2022) and PickScore (Kirstain et al., 2023) as differentiable rewards, and GenEval (Ghosh et al., 2023) as a non-differentiable reward based on compositional text-image alignment. For DMD2, SDXL-Turbo, and LCM, we evaluate on aesthetic score using simple-animal prompts following Black et al. (2024); Clark et al. (2024) and on PickScore (Kirstain et al., 2023) using 200 evaluation prompts. For DFGAN, we use 500 CUB (Wah et al., 2011) evaluation prompts.

Baselines. We compare ZeNO against the following baselines. **Best-of-N** independently samples N noise vectors and selects the one yielding the highest reward, serving as the simplest inference-time scaling baseline applicable to any reward function. **ORIGEN** (Min et al., 2026) formulates noise optimization as Langevin dynamics targeting the reward-tilted distribution, representing the strongest gradient-based baseline in the noise space optimization setting. We note that ReNO (Eyring et al., 2024) similarly applies gradient ascent in the initial noise space; as ORIGEN subsumes this approach within a more principled formulation, we use ORIGEN as the representative gradient-based baseline. Both methods require differentiable rewards and reliable gradient flow through the generator. **Ψ-Sampler** (Yoon et al., 2026) applies pCNL-based Langevin dynamics in the initial noise space to sample from the reward-aware posterior. We adapt Ψ-Sampler by applying the pCNL update directly in the initial noise space with the deterministic generators. For gradient-based methods (ORIGEN and Ψ-Sampler), we match the number of generator calls under the same budget; note that these methods additionally require backpropagation through the generator and reward, incurring additional computational overhead.

Table 2: Results on differentiable rewards. We report aesthetic score and PickScore across four generators. **Bold** indicates the best, and underline indicates the second-best result. † Evaluated on 500 CUB prompts (Wah et al., 2011).

	DMD2		SDXL-Turbo		LCM		DFGAN†	
Method	Aes.	Pick.	Aes.	Pick.	Aes.	Pick.	Aes.	Pick.
Uncontrolled	4.42	20.49	5.43	21.65	4.59	18.53	4.26	19.48
Best-of-N	4.73	22.29	4.65	23.08	4.80	20.65	**5.65**	21.33
ORIGEN	5.97	21.83	5.93	21.50	4.45	18.83	5.34	20.98
Ψ-Sampler	6.22	22.56	**6.00**	23.35	5.90	21.11	5.20	20.81
ZeNO (ours)	**6.31**	**22.68**	**6.00**	**23.43**	**6.15**	**21.53**	5.63	**21.34**

Table 3: Results on non-differentiable rewards. We report GenEval scores for count and position, where gradient-based methods are inapplicable.

	DMD2		SDXL-Turbo	
Method	Count	Pos.	Count	Pos.
Uncontrolled	0.163	0.090	0.513	0.250
Best-of-N	0.838	**0.840**	0.925	**0.900**
ZeNO (ours)	**0.950**	0.760	**0.963**	0.680

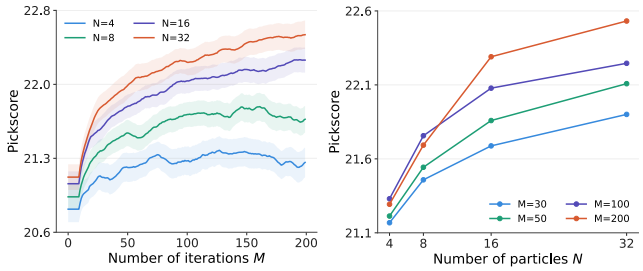


Figure 5: Inference-time scaling ablation for N and M.

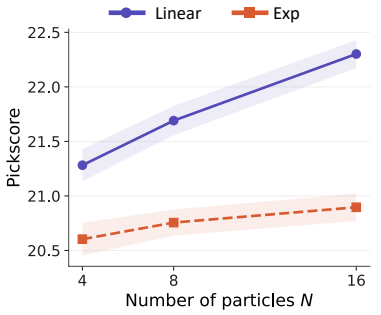


Figure 6: Estimator comparison.

Hyperparameters. For all experiments, we use $\beta = 0.01$, and control step size $\eta = 1.5$. For the main experiments, we run each algorithm with $N = 16$ and $M = 200$ to match the compute budget, and report the best reward among noise states. See Appendix E.1 for an ablation on β.

Comparison with Baselines. Table 2 reports results on aesthetic score and PickScore across all generators. ZeNO consistently improves reward across all generator and reward combinations. Compared to ORIGEN and Ψ-Sampler, ZeNO achieves competitive or superior performance despite requiring no gradients through the generator or reward model, demonstrating that zeroth-order noise optimization is a viable alternative to gradient-based methods. In particular, ORIGEN degrades below the uncontrolled baseline on LCM, consistent with our toy experiment findings that local gradient information can be unreliable, and further suggesting that this issue is exacerbated when backpropagating through complex generator-reward compositions. Notably, DFGAN results confirm that ZeNO generalizes beyond diffusion-based architectures, supporting its generator-agnostic applicability.

Table 3 reports results on GenEval, where the reward is computed by an object detection-based evaluation pipeline and is non-differentiable. In this setting, gradient-based methods including ORIGEN and Ψ-Sampler are inapplicable, and we compare ZeNO against Best-of-N under the same total generator call budget. We evaluate on counting and position subsets of GenEval, where uncontrolled generation consistently fails, providing meaningful room for reward-guided improvement. ZeNO consistently outperforms Best-of-N on counting tasks across all generators, demonstrating that iterative noise optimization extracts more reward improvement from the same computational budget than independent sampling. On position tasks, however, Best-of-N is better in some settings. We attribute this to the initial point dependency inherent in our one-step VP SDE formulation: since $\boldsymbol{z}_{t+1}$ is sampled in a single step from $\boldsymbol{z}_t$, the optimization heavily depends on the initial noise state, which is particularly limiting for position tasks where small spatial differences lead to large reward changes. Nevertheless, ZeNO outperforms Best-of-N across other rewards and counting tasks, demonstrating broad effectiveness as an inference-time optimization framework.

Inference-Time Scaling Analysis. Figure 5 shows PickScore of DMD2 as a function of the number of update iterations M (left) and the number of particles N (right), confirming that both serve as effective inference-time scaling knobs. Both sweeps confirm steady reward improvement: larger N consistently achieves higher reward at any given M, and larger M yields higher reward at any given N. These results demonstrate that ZeNO enables flexible inference-time scaling, allowing a trade-off between the number of particles and iterations under a given compute budget.

Estimator Comparison. Figure 6 compares the linearized estimator adopted in ZeNO (Proposition 1) with the exponential weighting in Eq. (8) across the number of particles N. Both estimators improve

Table 4: Comparison of designability and diversity across different NFE regimes. With ZeNO, RMF achieves stronger designability even at a single NFE than baselines.

Model	NFE	Designability		Diversity	
		< 2Å (↑)	scRMSD (↓)	Max. Cluster (↑)	Pairwise scTM (↓)
FrameFlow	5	0.04	6.53	0.68	0.22
FrameDiff	5	0.09	6.19	0.54	0.24
FrameFlow	2	0.00	N/A	N/A	N/A
RMF/S	1	0.03	7.93	0.84	0.33
+ **ZeNO**	1	**0.15**	**5.67**	**0.88**	0.34
RMF/M	1	0.14	4.78	0.60	0.33
+ **ZeNO**	1	**0.32**	**3.54**	**0.85**	0.35

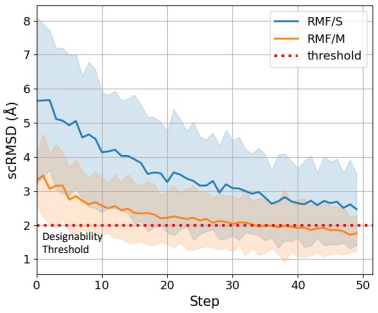


Figure 8: Evolution of designability with ZeNO on RMF-generated protein backbones.

as N increases, consistent with the variance reduction effect of using more particles. However, the linearized estimator substantially outperforms the exponential weighting, with the gap widening as N increases, consistent with the self-normalization of the latter discussed in Section 3.2.

4.3 Protein Structure Generation Experiments

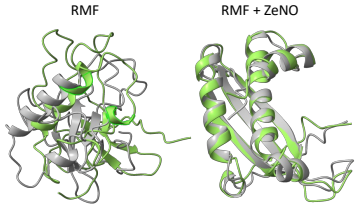


Figure 7: Structural overlay of the generated protein backbone and the ESMFold-refolded structure.

Most of the practically relevant objectives in protein design are non-differentiable or only available through external black-box pipelines. To demonstrate the applicability of ZeNO, we utilize the one-step Riemannian MeanFlow (RMF) (Woo et al., 2026) protein backbone generator. Following the representation used in frame-based protein generative models such as FrameDiff and FrameFlow, RMF represents protein backbones as residue-wise $\mathrm{SE}(3)$ frames and generates structures by updating these frames through a learned flow map. We use scRMSD as a black-box reward, measured by a self-consistency protocol based on inverse folding (ProteinMPNN (Dauparas et al., 2022)) and refolding (ESMFold (Lin et al., 2023)). A backbone is considered designable when scRMSD < 2Å. Further details are provided in Appendix I.2. Table 4 and Figure 7 show that applying ZeNO to RMF substantially improves backbone designability. For both RMF/S and RMF/M model, ZeNO increases the fraction of designable samples and reduces the average scRMSD. This results indicate that test-time optimization of the initial frame noise can effectively improve sample quality without modifying the generative model. More notably, with only a single flow evaluation, RMF+ZeNO achieves substantially higher designability than FrameFlow and FrameDiff with 5 NFEs, suggesting that ZeNO can partially compensate for reduced sampling steps by exploiting reward-guided optimization in the initial noise space. At the same time, the number of structural clusters increases for both models, while the pairwise scTM changes only marginally. This shows that the improvement in designability does not come from collapsing to a narrow set of similar structures. Figure 8 further supports this interpretation. As the ZeNO iterations progress, the scRMSD steadily decreases, demonstrating that the control signal consistently guides the initial frames toward more designable protein backbones.

5 Conclusion

We proposed ZeNO, a zeroth-order noise optimization framework that aligns deterministic generators with reward functions without requiring gradients, formulating noise updates as a path-integral control problem to yield a Langevin-like update rule targeting the reward-tilted distribution while remaining compatible with pre-trained generators via Ornstein–Uhlenbeck reference dynamics. Empirically, ZeNO enables effective inference-time scaling and demonstrates strong performance across a range of generators and reward functions, including settings where gradient-based methods are fundamentally inapplicable. A current limitation is that the zeroth-order estimator requires multiple reward evaluations per update step, which can be computationally demanding when the generator or reward model is expensive to evaluate.

AI USE STATEMENT

In this work, we used generative AI tools for sentence-level editing of texts written by authors to improve readability. We have not used generative AI tools for other aspects, such as generating synthetic data, developing theoretical models, formulating mathematical claims or proposing research hypothesis. We have reviewed all AI-assisted work. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work proposes an inference-time optimization method for aligning pre-trained generative models with reward functions. As with other reward alignment methods, ZeNO may inherit biases of the underlying generators and reward models, and could in principle be used to steer generators toward undesirable content if paired with a harmful reward. Our experiments use only publicly available models, prompts, and datasets, and do not involve human subjects.

REPRODUCIBILITY STATEMENT

The full algorithm is given in Algorithm 1, and all hyperparameters for image generation are reported in Section 4.2 and Appendix I.1. Complete proofs of Propositions 1, 2 and Lemma 1 are provided in Appendices B-D. The protein reference dynamics, control estimation, and reward are described in detail in Appendix I.2. Code will be made publicly available upon publication.

REFERENCES

Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=YCWjhGrJFD.

Hyungjin Chung, Jeongsol Kim, Michael Thompson Mccann, Marc Louis Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=OnD9zGAGT0k.

Kevin Clark, Paul Vicol, Kevin Swersky, and David J. Fleet. Directly fine-tuning diffusion models on differentiable rewards. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=1vmSEVL19f.

J. Dauparas, I. Anishchenko, N. Bennett, H. Bai, R. J. Ragotte, L. F. Milles, B. I. M. Wicky, A. Courbet, R. J. de Haas, N. Bethel, P. J. Y. Leung, T. F. Huddy, S. Pellock, D. Tischer, F. Chan, B. Koepnick, H. Nguyen, A. Kang, B. Sankaran, A. K. Bera, N. P. King, and D. Baker. Robust deep learning–based protein sequence design using proteinmpnn. *Science*, 378(6615):49–56, 2022. doi: 10.1126/science.add2187. URL https://www.science.org/doi/abs/10.1126/science.add2187.

Luca Eyring, Shyamgopal Karthik, Karsten Roth, Alexey Dosovitskiy, and Zeynep Akata. ReNO: Enhancing one-step text-to-image models through reward-based noise optimization. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=MXY0qsGgeO.

Dan Friedman and Adji Bousso Dieng. The vendi score: A diversity evaluation metric for machine learning. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856.

Zhengyang Geng, Mingyang Deng, Xingjian Bai, J Zico Kolter, and Kaiming He. Mean flows for one-step generative modeling. *arXiv preprint arXiv:2505.13447*, 2025.

Dhruba Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. Geneval: An object-focused framework for evaluating text-to-image alignment. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023. URL https://openreview.net/forum?id=Wbr51vK331.

Yujia Huang, Adishree Ghatare, Yuanzhe Liu, Ziniu Hu, Qinsheng Zhang, Chandramouli Shama Sastry, Siddharth Gururani, Sageev Oore, and Yisong Yue. Symbolic music generation with non-differentiable rule guided diffusion. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 19772–19797. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/huang24g.html.

Hyeongyu Kang, Jaewoo Lee, Woocheol Shin, Kiyoung Om, and Jinkyoo Park. Diffusion finetuning via reparameterized policy gradient of the soft q-function. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=8zoxC9e23q.

H. Kappen. Path integrals and symmetry breaking for optimal control theory. *J Stat Mech- Theory E*, 2005, 05 2005. doi: 10.1088/1742-5468/2005/11/P11011.

Byungjun Kim, Soobin Um, and Jong Chul Ye. Diverse text-to-image generation via contrastive noise optimization. *arXiv preprint arXiv:2510.03813*, 2025a.

Jaemin Kim, Bryan Sangwoo Kim, and Jong Chul Ye. Free2guide: Training-free text-to-video alignment using image lvlm. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 17920–17929, October 2025b.

Sunwoo Kim, Minkyu Kim, and Dongmin Park. Test-time alignment of diffusion models without reward over-optimization. In *The Thirteenth International Conference on Learning Representations*, 2025c. URL https://openreview.net/forum?id=vi3DjUhFVm.

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 *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=G5RwHpBUv0.

Jeongjae Lee and Jong Chul Ye. PCPO: Proportionate credit policy optimization for preference alignment of image generation models. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=alY08iknli.

Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Robert Verkuil, Ori Kabeli, Yaniv Shmueli, Allan dos Santos Costa, Maryam Fazel-Zarandi, Tom Sercu, Salvatore Candido, and Alexander Rives. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, 2023. doi: 10.1126/science.ade2574. URL https://www.science.org/doi/abs/10.1126/science.ade2574.

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 *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2026a. URL https://openreview.net/forum?id=oCBKGw5HNf.

Zhen Liu, Tim Z. Xiao, Weiyang Liu, Yoshua Bengio, and Dinghuai Zhang. Efficient diversity-preserving diffusion alignment via gradient-informed GFlownets. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=Aye5wL6TCn.

Zhen Liu, Tim Z. Xiao, Carles Domingo-Enrich, Weiyang Liu, and Dinghuai Zhang. Value gradient guidance for flow matching alignment. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2026b. URL https://openreview.net/forum?id=6MmOy2Ji8V.

Simian Luo, Yiqin Tan, Longbo Huang, Jian Li, and Hang Zhao. Latent consistency models: Synthesizing high-resolution images with few-step inference. *arXiv preprint arXiv:2310.04378*, 2023.

Yuhang Ma, Xiaoshi Wu, Keqiang Sun, and Hongsheng Li. Hpsv3: Towards wide-spectrum human preference score, 2025. URL https://arxiv.org/abs/2508.03789.

Yunhong Min, Daehyeon Choi, Kyeongmin Yeo, Jihyun Lee, and Minhyuk Sung. ORIGEN: Zero-shot 3d orientation grounding in text-to-image generation. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2026. URL https://openreview.net/forum?id=CKOyN2HShu.

Bernt Øksendal. Stochastic differential equations. In *Stochastic differential equations: an introduction with applications*, pp. 38–50. Springer, 2003.

Art B. Owen. *Monte Carlo theory, methods and examples*. https://artowen.su.domains/mc/, 2013.

Axel Sauer, Frederic Boesel, Tim Dockhorn, Andreas Blattmann, Patrick Esser, and Robin Rombach. Fast high-resolution image synthesis with latent adversarial diffusion distillation. In *SIGGRAPH Asia 2024 Conference Papers*, pp. 1–11, 2024a.

Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial diffusion distillation. In *European Conference on Computer Vision*, pp. 87–103. Springer, 2024b.

Christoph Schuhmann. Improved aesthetic predictor. https://github.com/christophschuhmann/improved-aesthetic-predictor, 2022. GitHub repository.

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. *arXiv preprint arXiv:2303.01469*, 2023.

Ming Tao, Hao Tang, Fei Wu, Xiao-Yuan Jing, Bing-Kun Bao, and Changsheng Xu. Df-gan: A simple and effective baseline for text-to-image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16515–16525, 2022.

Masatoshi Uehara, Yulai Zhao, Tommaso Biancalani, and Sergey Levine. Understanding reinforcement learning-based fine-tuning of diffusion models: A tutorial and review. *arXiv preprint arXiv:2407.13734*, 2024a.

Masatoshi Uehara, Yulai Zhao, Kevin Black, Ehsan Hajiramezanali, Gabriele Scalia, Nathaniel Lee Diamant, Alex M Tseng, Tommaso Biancalani, and Sergey Levine. Fine-tuning of continuous-time diffusion models as entropy-regularized control. *arXiv preprint arXiv:2402.15194*, 2024b.

Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 7 2011.

Dongyeop Woo, Marta Skreta, Seonghyun Park, Kirill Neklyudov, and Sungsoo Ahn. Riemannian meanflow, 2026. URL https://arxiv.org/abs/2602.07744.

Tianwei Yin, Michaël Gharbi, Taesung Park, Richard Zhang, Eli Shechtman, Fredo Durand, and William T Freeman. Improved distribution matching distillation for fast image synthesis. *Advances in neural information processing systems*, 37:47455–47487, 2024a.

Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Fredo 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, 2024b.

TaeHoon Yoon, Yunhong Min, Kyeongmin Yeo, and Minhyuk Sung. ψ-sampler: Initial particle sampling for SMC-based inference-time reward alignment in score models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2026. URL https://openreview.net/forum?id=slVqJAI5sT.

A Path-Integral Control Formulation

We briefly derive the optimal control form used in the main text, following Kappen (2005). Consider the controlled SDE

$$d\boldsymbol{z} = f(\boldsymbol{z}, t)dt + g(t)(\boldsymbol{u}(\boldsymbol{z}, t)dt + d\boldsymbol{w}_t), \tag{14}$$

with terminal cost

$$m(\boldsymbol{z}_T) = -r(G_\theta(\boldsymbol{z}_T)). \tag{15}$$

The path-integral control problem seeks the optimal control that minimizes

$$J(\boldsymbol{u}) = \mathbb{E}\left[m(\boldsymbol{z}_T) + \lambda \int_0^T \frac{1}{2}\|\boldsymbol{u}(\boldsymbol{z}_t, t)\|^2 dt\right], \tag{16}$$

where the quadratic control cost regularizes the deviation from the uncontrolled reference dynamics.

Define the desirability function

$$\Psi(\boldsymbol{z}, t) = \mathbb{E}\left[\exp(r(G_\theta(\boldsymbol{z}_T))/\lambda) \mid \boldsymbol{z}_t = \boldsymbol{z}\right], \tag{17}$$

and the value function

$$V(\boldsymbol{z}, t) = \log \Psi(\boldsymbol{z}, t), \tag{18}$$

consistent with Eq. (3), so that the optimal cost-to-go is $-\lambda V$. By the Feynman-Kac theorem, the value function satisfies the corresponding Hamilton-Jacobi-Bellman equation after the logarithmic transformation. The optimal control is given by the score of the desirability function:

$$\boldsymbol{u}^\star(\boldsymbol{z}, t) = g(t)\nabla_{\boldsymbol{z}} V(\boldsymbol{z}, t) = g(t)\nabla_{\boldsymbol{z}} \log \Psi(\boldsymbol{z}, t). \tag{19}$$

In our implementation, the scalar factor $g(t)$ is absorbed into the update step size. Thus, we use the following normalized control:

$$\boldsymbol{u}^\star(\boldsymbol{z}, t) = \nabla_{\boldsymbol{z}} \log \mathbb{E}\left[\exp(r(G_\theta(\boldsymbol{z}_T))/\lambda) \mid \boldsymbol{z}_t = \boldsymbol{z}\right]. \tag{20}$$

Equivalently, the control can be written in path-integral form under the uncontrolled reference dynamics, where, as in Eq. (8), $\boldsymbol{u}^\star$ denotes the control increment and the time and diffusion coefficients are absorbed into the step size. Let $d\boldsymbol{w}_t$ denote the Brownian increment at time t. Then,

$$\boldsymbol{u}^\star(\boldsymbol{z}, t) := \frac{\mathbb{E}_{\boldsymbol{z}_{t:T}}\left[\exp(r(G_\theta(\boldsymbol{z}_T))/\lambda) d\boldsymbol{w}_t\right]}{\mathbb{E}_{\boldsymbol{z}_{t:T}}\left[\exp(r(G_\theta(\boldsymbol{z}_T))/\lambda)\right]}. \tag{21}$$

This expression shows that the optimal control can be estimated using only zeroth-order reward evaluations along uncontrolled trajectories.

B Proof of Proposition 1

Proposition 1. *Let $r := r(G_\theta(\boldsymbol{z}_T))$ and $\bar{r} := \mathbb{E}[r]$, and assume $|r - \bar{r}| \le R$ almost surely. With $\bar{r}$ as the baseline, Eq. (8) can be linearized as*

$$\boldsymbol{u}_{\text{lin}} := \mathbb{E}[(r - \bar{r})d\boldsymbol{w}_t]/\lambda \tag{9}$$

with the error bounded by

$$\|\boldsymbol{u}^\star - \boldsymbol{u}_{\text{lin}}\| \le \frac{e^{R/\lambda}}{2} \frac{R\sqrt{\text{Var}(r)}}{\lambda^2} \left(1 + \frac{\sqrt{\text{Var}(r)}}{\lambda}\right) \sqrt{dt}, \tag{10}$$

which is at most $eR\sqrt{\text{Var}(r)}\sqrt{dt}/\lambda^2$ for $\lambda \ge R$.

Proof. Let $\delta := (r - \bar{r})/\lambda$. Then, $|\delta| \le R/\lambda$, $\mathbb{E}[\delta] = 0$, and $\boldsymbol{u}_{\text{lin}} = \mathbb{E}[\delta\, d\boldsymbol{w}_t]$. We apply the Taylor expansion with the remainder as $e^\delta = 1 + \delta + \rho$, where $\rho = e^\zeta \delta^2/2 \ge 0$ for some ζ between 0 and δ. Also, since $|\zeta| \le |\delta| \le R/\lambda$, we have $e^\zeta \le e^{R/\lambda}$, which gives

$$0 \le \rho \le \tfrac{1}{2} e^{R/\lambda} \delta^2. \tag{22}$$

Since $e^{r/\lambda} = e^{\bar{r}/\lambda} e^{\delta}$, the factor $e^{\bar{r}/\lambda}$ cancels in Eq. (8). Then, the denominator of the optimal control is

$$\mathbb{E}[e^{\delta}] = 1 + \mathbb{E}[\rho], \tag{23}$$

where the equality follows from $\mathbb{E}[\delta] = 0$, and the numerator is

$$\mathbb{E}[e^{\delta} d\boldsymbol{w}_t] = \boldsymbol{u}_{\mathrm{lin}} + \mathbb{E}[\rho \, d\boldsymbol{w}_t], \tag{24}$$

where the equality holds as $\mathbb{E}[d\boldsymbol{w}_t] = 0$. By combining them, we get

$$\boldsymbol{u}^{\star} - \boldsymbol{u}_{\mathrm{lin}} = \frac{\mathbb{E}[\rho \, d\boldsymbol{w}_t] - \mathbb{E}[\rho] \, \boldsymbol{u}_{\mathrm{lin}}}{1 + \mathbb{E}[\rho]}, \tag{25}$$

which gives $\boldsymbol{u}^{\star} = \boldsymbol{u}_{\mathrm{lin}}$ when $\rho = 0$. Since $\rho \geq 0$, the denominator is at least 1, and thus

$$\|\boldsymbol{u}^{\star} - \boldsymbol{u}_{\mathrm{lin}}\| \leq \|\mathbb{E}[\rho \, d\boldsymbol{w}_t]\| + \mathbb{E}[\rho] \, \|\boldsymbol{u}_{\mathrm{lin}}\|. \tag{26}$$

By the Cauchy–Schwarz inequality, using $\boldsymbol{v}^{\top} d\boldsymbol{w}_t \sim \mathcal{N}(0, dt)$ for any unit vector $\boldsymbol{v}$, we have

$$\|\mathbb{E}[\rho \, d\boldsymbol{w}_t]\| \leq \sqrt{\mathbb{E}[\rho^2]} \, \sqrt{dt}, \qquad \|\boldsymbol{u}_{\mathrm{lin}}\| \leq \frac{\sqrt{\mathrm{Var}(r)} \, \sqrt{dt}}{\lambda}. \tag{27}$$

Also, since $|r - \bar{r}| \leq R$, the bound on ρ gives

$$\mathbb{E}[\rho^2] \leq \frac{e^{2R/\lambda} R^2 \, \mathrm{Var}(r)}{4\lambda^4}, \qquad \mathbb{E}[\rho] \leq \frac{e^{R/\lambda} \, \mathrm{Var}(r)}{2\lambda^2}. \tag{28}$$

By combining them, we get

$$\|\boldsymbol{u}^{\star} - \boldsymbol{u}_{\mathrm{lin}}\| \leq \frac{e^{R/\lambda}}{2} \left(\frac{R\sqrt{\mathrm{Var}(r)}}{\lambda^2} + \frac{\mathrm{Var}(r)^{3/2}}{\lambda^3} \right) \sqrt{dt}. \tag{29}$$

Since $\sqrt{\mathrm{Var}(r)} \leq R$, we have

$$\frac{\mathrm{Var}(r)^{3/2}}{\lambda^3} \leq R \frac{\mathrm{Var}(r)}{\lambda^3}, \tag{30}$$

which gives Eq. (10). When $\lambda \geq R$, $e^{R/\lambda} \leq e$ and $\sqrt{\mathrm{Var}(r)} \leq R \leq \lambda$, which gives $e \, R\sqrt{\mathrm{Var}(r)}\sqrt{dt}/\lambda^2$. □

Interpretation. Since $\lambda \log \mathbb{E}[e^{r/\lambda}] = \mathbb{E}[r] + \mathrm{Var}(r)/(2\lambda) + O(\lambda^{-2})$, the linearized control keeps only the average reward, i.e., the risk-neutral part of the risk-sensitive PIC objective. Both $\boldsymbol{u}^{\star}$ and $\boldsymbol{u}_{\mathrm{lin}}$ scale as $1/\lambda$, which is absorbed into η, while their difference scales as $1/\lambda^2$, so the linearization error becomes small relative to the scale of the control as λ grows. Moreover, λ does not appear in Algorithm 1, and to first order the target of Proposition 2 depends on λ and κ only through λ/κ, so λ can be taken large relative to R without changing the target.

C Proof of Proposition 2

Proposition 2. *When $\beta \ll 1$ and the control is computed with $H = 1$, the discretized controlled OU process Eq. (12) is the Langevin step*

$$\boldsymbol{z} \leftarrow \boldsymbol{z} + \frac{\beta}{2} \nabla_{\boldsymbol{z}} \log p^{\star}_{\kappa}(\boldsymbol{z}) + \sqrt{\beta}\boldsymbol{\epsilon}, \tag{13}$$

where $p^{\star}_{\kappa}(\boldsymbol{z}) \propto p(\boldsymbol{z}) \exp\big(\kappa V(\boldsymbol{z})\big)$ with $p(\boldsymbol{z}) = \mathcal{N}(0, \mathbf{I})$, $V(\boldsymbol{z}) = \log \mathbb{E}_{\boldsymbol{\epsilon}}[\exp(r(G_{\theta}(\sqrt{1-\beta}\boldsymbol{z} + \sqrt{\beta}\boldsymbol{\epsilon}))/\lambda)]$ is the value of the one-step problem, and $\kappa = 2\eta/\beta$.

Proof. A Langevin step with step size s toward a distribution π reads $\boldsymbol{z} \leftarrow \boldsymbol{z} + s \, \nabla_{\boldsymbol{z}} \log \pi(\boldsymbol{z}) + \sqrt{2s} \, \boldsymbol{\epsilon}$. We show that Eq. (12) takes this form with $s = \beta/2$ and $\pi = p^{\star}_{\kappa}$.

When $\beta \ll 1$, we use $\sqrt{1-\beta} = 1 - \beta/2 + O(\beta^2)$, which gives, to first order in β,

$$\boldsymbol{z} \leftarrow \boldsymbol{z} + \frac{\beta}{2} \left(-\boldsymbol{z} + \kappa \boldsymbol{u}^{\star}\right) + \sqrt{\beta} \, \boldsymbol{\epsilon}, \qquad \kappa = 2\eta/\beta, \tag{31}$$

where we keep κ fixed. The noise term already matches, since $\sqrt{2 \cdot \beta/2} = \sqrt{\beta}$, so it remains to show that the drift $-\boldsymbol{z} + \kappa \boldsymbol{u}^\star$ equals $\nabla_{\boldsymbol{z}} \log p^\star_\kappa(\boldsymbol{z})$.

For the first term, since $p(\boldsymbol{z}) = \mathcal{N}(0, \mathbf{I})$, we have $-\boldsymbol{z} = \nabla_{\boldsymbol{z}} \log p(\boldsymbol{z})$. For the second term, the control with $H = 1$ is the optimal control of the problem whose terminal time is one step ahead. Applying Eq. (20) to this problem, i.e., with $\boldsymbol{z}_T$ replaced by the one-step proposal $\boldsymbol{z}' = \sqrt{1-\beta}\boldsymbol{z} + \sqrt{\beta}\boldsymbol{\epsilon}$, gives

$$\boldsymbol{u}^\star = \nabla_{\boldsymbol{z}} \log \mathbb{E}_{\boldsymbol{\epsilon}} \left[\exp \left(r(G_\theta(\boldsymbol{z}'))/\lambda \right) \right] = \nabla_{\boldsymbol{z}} V(\boldsymbol{z}). \tag{32}$$

Then, the drift becomes

$$-\boldsymbol{z} + \kappa \nabla_{\boldsymbol{z}} V(\boldsymbol{z}) = \nabla_{\boldsymbol{z}} \log \left(p(\boldsymbol{z}) \exp(\kappa V(\boldsymbol{z})) \right) = \nabla_{\boldsymbol{z}} \log p^\star_\kappa(\boldsymbol{z}), \tag{33}$$

which proves the claim. Since V is the same at every iteration, iterating this step gives a Langevin chain whose stationary distribution approximates $p^\star_\kappa$ for small β. □

Optimal noise distribution. For $p^\star(\boldsymbol{z}) \propto p(\boldsymbol{z}) \exp(r(G_\theta(\boldsymbol{z}))/\lambda)$ with normalizing constant Z, we have, for any distribution q,

$$\mathbb{E}_{\boldsymbol{z} \sim q}[r(G_\theta(\boldsymbol{z}))] - \lambda \, \mathrm{KL}(q \,\|\, p) = \lambda \log Z - \lambda \, \mathrm{KL}(q \,\|\, p^\star). \tag{34}$$

Thus, $p^\star$ is the unique maximizer of the KL-regularized objective.

Relation to the full-horizon problem. Given the initial noise $\boldsymbol{z}_0$, the optimal path measure is the reference path measure from $\boldsymbol{z}_0$ tilted by $\exp(r(G_\theta(\boldsymbol{z}_T))/\lambda)$. As this weight depends only on the endpoint, the terminal marginal is proportional to $p^{\mathrm{ref}}_T(\boldsymbol{z} \mid \boldsymbol{z}_0) \exp(r(G_\theta(\boldsymbol{z}))/\lambda)$. For a horizon long enough for the reference dynamics to forget $\boldsymbol{z}_0$, $p^{\mathrm{ref}}_T(\cdot \mid \boldsymbol{z}_0)$ approaches p, and thus the terminal marginal approximately becomes $p^\star$.

Limit of the one-step target. The one-step control does not plan the remaining trajectory, and instead moves the noise toward the stationary distribution $p^\star_\kappa$ of the iterated Langevin step. As $\beta \to 0$ with fixed κ, the one-step proposal $\boldsymbol{z}'$ concentrates at $\boldsymbol{z}$, so $V(\boldsymbol{z})$ approaches $r(G_\theta(\boldsymbol{z}))/\lambda$, and $p^\star_\kappa$ approaches $p^\star$ up to the temperature scaled by κ, coinciding with it for $\kappa = 1$.

D Proof of Lemma 1

Lemma 1. *Under the uncontrolled process Eq. (12), the fraction of the variance of $\boldsymbol{z}_{t+H}$ given $\boldsymbol{z}_t$ that is due to $\boldsymbol{\epsilon}_t$ is*

$$s_H = \frac{\beta(1-\beta)^{H-1}}{1-(1-\beta)^H}, \tag{35}$$

which equals 1 at $H = 1$ and decreases monotonically in H.

Proof. Setting $\boldsymbol{u}^\star = 0$ in Eq. (12), the uncontrolled process gives

$$\boldsymbol{z}_{t+1} = \sqrt{1-\beta}\, \boldsymbol{z}_t + \sqrt{\beta}\, \boldsymbol{\epsilon}_t. \tag{36}$$

Applying this recursively for H steps,

$$\begin{aligned}
\boldsymbol{z}_{t+H} &= \sqrt{1-\beta}\, \boldsymbol{z}_{t+H-1} + \sqrt{\beta}\, \boldsymbol{\epsilon}_{t+H-1} && (37)\\
&= (1-\beta)\, \boldsymbol{z}_{t+H-2} + \sqrt{\beta}(1-\beta)^{1/2} \boldsymbol{\epsilon}_{t+H-2} + \sqrt{\beta}\, \boldsymbol{\epsilon}_{t+H-1} && (38)\\
&= \cdots && (39)\\
&= (1-\beta)^{H/2} \boldsymbol{z}_t + \sqrt{\beta}(1-\beta)^{(H-1)/2} \boldsymbol{\epsilon}_t + \sqrt{\beta} \sum_{k=0}^{H-2} (1-\beta)^{k/2} \boldsymbol{\epsilon}_{t+H-1-k}. && (40)
\end{aligned}$$

Given $\boldsymbol{z}_t$, the noise terms are independent, so the variance of $\boldsymbol{z}_{t+H}$ per coordinate is the sum of the squared coefficients,

$$\underbrace{\beta(1-\beta)^{H-1}}_{\text{from } \boldsymbol{\epsilon}_t} + \beta \sum_{k=0}^{H-2} (1-\beta)^k = \beta \sum_{k=0}^{H-1} (1-\beta)^k = 1 - (1-\beta)^H, \tag{41}$$

and the fraction due to $\boldsymbol{\epsilon}_t$ is s_H. Let $q := (1-\beta)^{H-1} \in (0, 1]$. Then, $s_H = \beta q/(1-(1-\beta)q)$ increases in q, and q decreases in H, so s_H decreases in H with $s_1 = 1$. □

E Ablation Study

E.1 β ablation

We conduct an ablation study on the SDE variance parameter β with SDXL-Turbo. As shown in Appendix C, the Langevin-like sampling interpretation targeting the tilted prior distribution holds in the small-variance regime, i.e., when $\beta \ll 1$. As shown in Table 5, $\beta = 0.01$ achieves the highest aesthetic score by a large margin, whereas all larger values of β yield nearly identical and substantially lower scores.

Table 5: Ablation study on the effect of β on aesthetic score.

β	Aesthetic Score (↑)
0.01	**6.000**
0.05	4.585
0.10	4.582
0.30	4.586

E.2 Perturbation Ablation

The OU reference process is motivated by preserving the Gaussian prior, while the practical algorithm combines one-step discrete perturbations with an explicit norm projection (Algorithm 1). To examine whether the proposed control update contributes beyond simpler noise-space search schemes, we compare ZeNO with the following baselines on PickScore with SDXL-Turbo. All variants use the same evaluation budget and hyperparameters as ZeNO ($\beta = 0.01$, $\eta = 1.5$, $N = 16$, $M = 200$).

- **Gaussian Random Walk** The noise evolves under the uncontrolled OU reference dynamics with the same norm projection. This baseline isolates the effect of random exploration without reward-driven control. It corresponds to the "Uncontrolled" baseline in the main paper.
- **ZeNO + Tangent Space** Instead of explicit normalization, both the Gaussian perturbations and the reward-weighted control are restricted to the sphere $\mathcal{S} = \{\boldsymbol{z} : \|\boldsymbol{z}\| = \sqrt{d}\}$ around which $\mathcal{N}(0, \mathbf{I})$ concentrates. Specifically, each vector $\boldsymbol{v}$ is projected onto the tangent space at the current state $\boldsymbol{z}$ via
$$\mathrm{P}_{\boldsymbol{z}}(\boldsymbol{v}) = \boldsymbol{v} - \frac{\langle \boldsymbol{v}, \boldsymbol{z} \rangle}{\|\boldsymbol{z}\|^2} \boldsymbol{z}, \tag{42}$$
and the update is applied through a single exponential-map step on $\mathcal{S}$,
$$\mathrm{Exp}_{\boldsymbol{z}}(\boldsymbol{v}) = \cos\left(\frac{\|\boldsymbol{v}\|}{\sqrt{d}}\right) \boldsymbol{z} + \sqrt{d} \sin\left(\frac{\|\boldsymbol{v}\|}{\sqrt{d}}\right) \frac{\boldsymbol{v}}{\|\boldsymbol{v}\|}. \tag{43}$$
- **ZeNO w/o normalization** ZeNO with the norm projection (line 12 of Algorithm 1) removed, so that the noise evolves under the controlled OU update in Eq. (12) alone.

Table 6: Comparison with simpler noise-space search schemes on PickScore with SDXL-Turbo under the same evaluation budget. **Bold** indicates the best, and underline indicates the second-best result.

Method	Gaussian random walk	ZeNO (tangent space)	ZeNO w/o normalization	ZeNO
PickScore (↑)	21.65	<u>23.40</u>	23.01	**23.43**

The Gaussian random walk performs substantially worse than all ZeNO variants, showing that the reward improvement of ZeNO does not arise from random exploration of the noise space alone. Rather, it comes from the reward-weighted control estimated from zeroth-order evaluations.

Removing the normalization still improves over the random walk, which again confirms the benefit of the control update. However, during optimization, the letent gradually drifts away from the Gaussian shell without norm-preserving. This supports the practical design choice, where renormalization keeps the noise compatible with the pre-trained generator throughout optimization.

F Results on Additional Human Preference Reward

To examine whether ZeNO remains effective under more recent and stronger human preference models, we additionally evaluate on HPSv3 (Ma et al., 2025), a wide-spectrum human preference score trained on large-scale preference data spanning diverse generative models and image sources. We use SDXL-Turbo as the one-step generator and follow the same experimental protocol as in Section 4.2, with $\beta = 0.01$, $\eta = 1.5$, $N = 16$, and $M = 200$. For the gradient-based baselines (ORIGEN and Ψ-Sampler), we match the number of generator calls, as in the main experiments.

Table 7: Results on HPSv3 with SDXL-Turbo. **Bold** indicates the best, and underline indicates the second-best result.

Method	Uncontrolled	Best-of-N	ORIGEN	Ψ-Sampler	ZeNO (ours)
HPSv3 (↑)	9.804	13.693	9.570	**14.271**	<u>14.186</u>

As shown in Table 7, ZeNO substantially improves HPSv3 over the uncontrolled baseline and outperforms Best-of-N under the same generator call budget, indicating that iterative noise optimization extracts more reward improvement than independent sampling. ZeNO achieves performance comparable to Ψ-Sampler, with a gap of less than 0.1, even though Ψ-Sampler additionally requires backpropagation through the generator and the reward model. In contrast, ORIGEN falls below the uncontrolled baseline, consistent with our observations on LCM in Table 2. Overall, these results confirm that ZeNO generalizes to recent human preference models and remains competitive with gradient-based methods while using only zeroth-order reward evaluations.

G Diversity Analysis

ZeNO is designed to improve reward while preserving the diversity of the reference generator. From the Langevin interpretation in Proposition 2, the controlled noise dynamics does not aim to collapse all samples to a single reward-maximizing point. Instead, it moves the noise toward a reward-tilted distribution,

$$p_\kappa^\star(\boldsymbol{z}) \propto p(\boldsymbol{z}) \exp\big(\kappa V(\boldsymbol{z})\big), \tag{44}$$

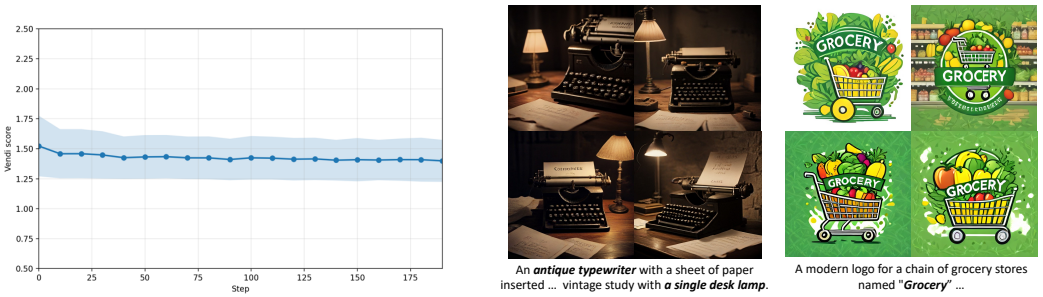
where the reference prior $p(\boldsymbol{z})$ preserves the support and diversity of the original generator, while the exponential value tilt increases the probability of high-reward regions. Therefore, although the generated reward distribution is expected to shift toward higher reward samples, the resulting sample distribution should still retain diversity as long as κ is not too large. This behavior is consistent with our toy experiment in Figure 3, where ZeNO closely matches the target tilted distribution, while gradient-based updates concentrate probability mass on a subset of modes.

To empirically verify this behavior in high-dimensional image generation, we measure diversity throughout ZeNO optimization using the Vendi Score (Friedman & Dieng, 2023) following Kim et al. (2025a). Given a set of generated images $\{x_i\}_{i=1}^n$, we compute image embeddings and form a normalized pairwise similarity matrix K. The Vendi Score is defined as the exponential entropy of the eigenvalue distribution of K,

$$\mathrm{VS}(\{x_i\}_{i=1}^n) = \exp\left(-\sum_j \lambda_j \log \lambda_j\right), \tag{45}$$

where $\{\lambda_j\}$ are the normalized eigenvalues of K. A higher Vendi Score indicates a larger effective number of distinct samples, while a decreasing score indicates reduced diversity.

Figure 9 shows the Vendi Score over ZeNO iterations. We observe a small decrease at the early stage of optimization, which is expected because the reward tilt shifts the initial distribution toward higher-reward regions. Importantly, however, the diversity does not continue to decrease. Instead, the Vendi



(a) Vendi Score (b) Generated examples

Figure 9: Diversity analysis of ZeNO. Left: Vendi Score over ZeNO iterations. The score slightly decreases during the initial reward-guided optimization phase, but quickly saturates and remains stable across subsequent iterations. Right: representative generated samples during optimization. These results indicate that ZeNO preserves sample diversity and does not collapse to a narrow set of modes while shifting the generation distribution toward higher-reward regions. Shaded regions indicate variability across samples.

Score quickly saturates and remains stable over the rest of the optimization trajectory. This suggests that ZeNO improves reward by redistributing probability mass according to the tilted distribution rather than collapsing the generator output to a narrow set of samples. In other words, ZeNO preserves meaningful sample diversity while steering the noise distribution toward higher-reward regions.

H Wall-Clock Analysis and Complementarity with Best-of-N

While the main experiments compare methods under a fixed number of generator/reward evaluations, Best-of-N (BoN) is fully parallelizable, whereas ZeNO performs M sequential update rounds. We therefore compare the two under a fixed wall-clock budget on a single RTX 4090, using SDXL-Turbo with PickScore.

Table 8: PickScore under a fixed wall-clock budget (single RTX 4090, SDXL-Turbo).

Wall-clock (s)	0	50	100	150	200	250	300
Best-of-N	22.81	23.05	23.08	**23.20**	23.20	23.20	23.20
ZeNO	22.50	23.08	23.15	23.31	23.38	**23.52**	23.52

Under a fixed hardware budget, the per-round cost of ZeNO is comparable to that of BoN, since each round evaluates N particles in parallel. BoN starts higher because it selects the best of many random samples, whereas ZeNO starts from a single uncontrolled sample. ZeNO quickly catches up, and after roughly 150 s BoN saturates while ZeNO continues to improve, reaching 23.52 within the same budget. With more memory or multiple GPUs, BoN could reduce its latency further; however, ZeNO's particle evaluations are equally parallelizable.

I Implementation Details

All experiments are conducted using NVIDIA RTX 3090 and RTX 4090 GPUs with 24GB of memory.

I.1 Image Generation

Figure captions The images in Figure 1 were generated using SDXL-Turbo, and the corresponding text prompts are listed below, starting from the upper-left and proceeding counter-clockwise.

1. "A teddy bear wearing holding a whiteboard, 'ZeNO' written on it."

2. "A beautifully decorated birthday cake with smooth blue icing, the letters 'Happy 30th Jake' elegantly spelled out on top, surrounded by colorful candles and sparkling decorations, set on a white tablecloth in a cozy living room."
3. "Studio shot of intricate shoe sculptures crafted from vibrant colored wires, with the text 'rte' prominently displayed on a clean, white background."
4. "A campfire in a dense forest, with smoke curling upwards and forming the words 'Send Help' against a twilight sky, partially obscured by the treetops."
5. "An astronaut proudly displays the 'Mission Success' patch on their spacesuit, standing against the backdrop of a stunning Earthrise over the lunar horizon, with the vast cosmos stretching beyond."

CFG scale We used a CFG scale of 0.0 for SDXL-Turbo and DMD2, and 8.0 for LCM, following the default recommended settings for each model.

I.2 Protein Structure Generation

Initial-frame representation. For protein structure generation, we apply ZeNO to a one-step Riemannian MeanFlow (RMF) backbone generator. Following frame-based protein generative models, each backbone of length N is represented as residue-wise rigid frames $x = \{(R_i, t_i)\}_{i=1}^N \in \mathrm{SE}(3)^N$, where $R_i \in \mathrm{SO}(3)$ and $t_i \in \mathbb{R}^3$. The RMF generator maps an initial frame sample x to a backbone $y = \Phi_\theta(x)$, and Φ_θ is kept frozen throughout ZeNO optimization.

Reference dynamics on protein frames. Since $\mathrm{SO}(3)$ is not closed under addition, we replace the Euclidean OU step in Eq. (12) with a prior-preserving counterpart on frames. For step sizes $\beta, \beta_R \in (0,1)$, one step of the reference kernel reads

$$t_i^{k+1} = \sqrt{1-\beta}\, t_i^k + \sqrt{\beta}\, \sigma_t\, \epsilon_i^k, \qquad \epsilon^k \sim \mathcal{N}_0(0, I), \tag{46}$$

$$R_i^{k+1} = \exp\big(\sqrt{\beta_R}\, \widehat{\omega_i^k}\big) R_i^k, \qquad \omega_i^k \sim \mathcal{N}(0, I_3), \tag{47}$$

where σ_t is the standard deviation of the translation prior, $\mathcal{N}_0(0, I)$ denotes the standard Gaussian projected onto the zero-center-of-mass subspace of $\mathbb{R}^{3N}$, and $\widehat{\cdot}$ maps a vector in $\mathbb{R}^3$ to its skew-symmetric matrix in $\mathfrak{so}(3)$. The translation step preserves the centered Gaussian prior since $(\sqrt{1-\beta})^2 + \beta = 1$, and the rotation step preserves the uniform (Haar) prior on $\mathrm{SO}(3)^N$. Hence, as in the image setting, the uncontrolled kernel leaves the RMF source distribution invariant.

Control estimation and update. At each iteration k, we draw K one-step proposals from the reference kernel,

$$\tilde{t}_i^{(j)} = \sqrt{1-\beta}\, t_i^k + \sqrt{\beta}\, \sigma_t\, v_i^{(j)}, \qquad \tilde{R}_i^{(j)} = \exp\big(\sqrt{\beta_R}\, \widehat{\omega_i^{(j)}}\big) R_i^k, \tag{48}$$

with $v^{(j)} \sim \mathcal{N}_0(0, I)$ and $\omega_i^{(j)} \sim \mathcal{N}(0, I_3)$, evaluate $r_j = r(\Phi_\theta(\tilde{x}^{(j)}))$, where r is the black-box designability reward defined below, and form advantages $a_j = r_j - \frac{1}{K}\sum_{j'} r_{j'}$. As in Algorithm 1, we average the advantage-weighted perturbations, normalizing the translation and rotation components by their respective noise scales:

$$u_{t,i} = \frac{1}{\sqrt{\beta}\, \sigma_t K} \sum_{j=1}^K a_j\, v_i^{(j)}, \qquad u_{R,i} = \frac{1}{\sqrt{\beta_R} K} \sum_{j=1}^K a_j\, \omega_i^{(j)}. \tag{49}$$

The frames are then updated by the controlled reference kernel,

$$t_i^{k+1} = \sqrt{1-\beta}\, t_i^k + \sqrt{\beta}\, \sigma_t\, \epsilon_i^k + \eta\, \bar{u}_{t,i}, \tag{50}$$

$$R_i^{k+1} = \exp\big(\eta\, \widehat{\bar{u}_{R,i}} + \sqrt{\beta_R}\, \widehat{\omega_i^k}\big) R_i^k, \tag{51}$$

with fresh noise $\epsilon^k \sim \mathcal{N}_0(0, I)$ and $\omega_i^k \sim \mathcal{N}(0, I_3)$, followed by re-centering the translations to zero center of mass. Here, η is the step size, and $\bar{u}_{t,i}$ and $\bar{u}_{R,i}$ denote the controls clipped so that each per-residue update $\eta\, \bar{u}_{\cdot,i}$ has norm at most τ_t and τ_R, respectively, for numerical stability.

Designability reward. We use a black-box designability reward based on an inverse-folding and refolding self-consistency protocol. Given a generated backbone $y = \Phi_\theta(x)$, we first use ProteinMPNN to design M amino-acid sequences conditioned on y:

$$s_m \sim p_{\mathrm{MPNN}}(s \mid y), \qquad m = 1, \ldots, M. \tag{52}$$

Each designed sequence is then refolded using ESMFold:

$$\hat{y}_m = \mathrm{ESMFold}(s_m). \tag{53}$$

We measure the self-consistency between the generated backbone y and the refolded structure $\hat{y}_m$ using the self-consistency RMSD,

$$d_m = \mathrm{scRMSD}(y, \hat{y}_m). \tag{54}$$

Following standard designability evaluation, we summarize the backbone-level self-consistency by the best refolded sequence,

$$d_{\mathrm{sc}}(y) = \min_{m=1,\ldots,M} \mathrm{scRMSD}(y, \hat{y}_m). \tag{55}$$

A generated backbone is considered designable if

$$d_{\mathrm{sc}}(y) < 2\text{Å}. \tag{56}$$

For reward-guided optimization, we convert this self-consistency score into a scalar reward. In the simplest form, we use the negative self-consistency RMSD,

$$r(y) = -d_{\mathrm{sc}}(y), \tag{57}$$

so that higher reward corresponds to lower refolding error and thus higher designability. This reward is non-differentiable with respect to the RMF initial frames because it involves ProteinMPNN sequence sampling, ESMFold refolding, structural alignment, and RMSD computation. ZeNO only requires reward evaluations and therefore can directly optimize this designability objective without backpropagating through ProteinMPNN, ESMFold, or the self-consistency metric.

J Additional Qualitative Results

J.1 Image Results



Figure 10: Images generated by 1-step SDXL-Turbo with ZeNO, using PickScore as the reward signal.



... advertisement that reads "***Try Our New Burger***" with an appetizing image ...



... cursive text that reads "***Wish You Were Here***" against a backdrop of a serene...



A close-up of a ***wristwatch face*** with intricate, vintage-style engravings, ...



... boldly displays the phrase "***Skate or Die***" in a vibrant ...



... surrounded by intricate ***illustrations of mountains***, forests, and a distant coastline...



... road sign near a rugged cliff, reading "***Danger Edge Unstable***", with the vast,...



A vibrant movie poster with the tagline "***The Adventure Begins*** ...



A protestor holds a hand-painted sign demanding "***Equal Rights Now***"...



... a weathered wooden sign that reads "***Path To Peace***", set against a backdrop ...



A close-up of a spacesuit arm patch that reads "***Mission to Europa***"...

Figure 11: Images generated by 1-step SDXL-DMD2 with ZeNO, using PickScore as the reward signal.

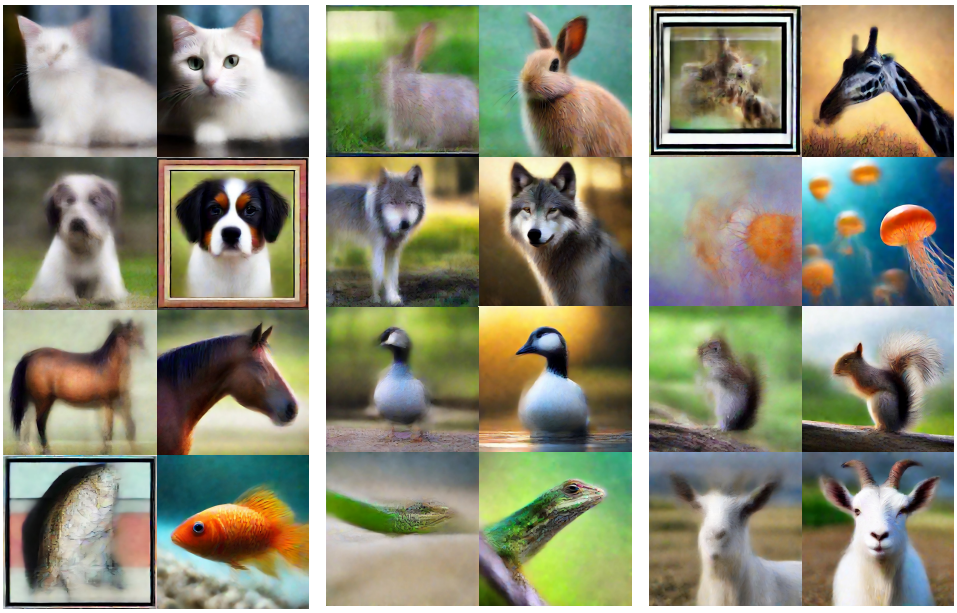


Figure 12: Images generated by 1-step LCM with ZeNO, using aesthetic score as the reward signal.



Figure 13: Images generated by DF-GAN with ZeNO, using aesthetic score as the reward signal.

J.2 PROTEIN RESULTS

We provide additional uncurated qualitative results for 30 protein samples, showing that the per-sample scRMSD decreases over ZeNO iterations.

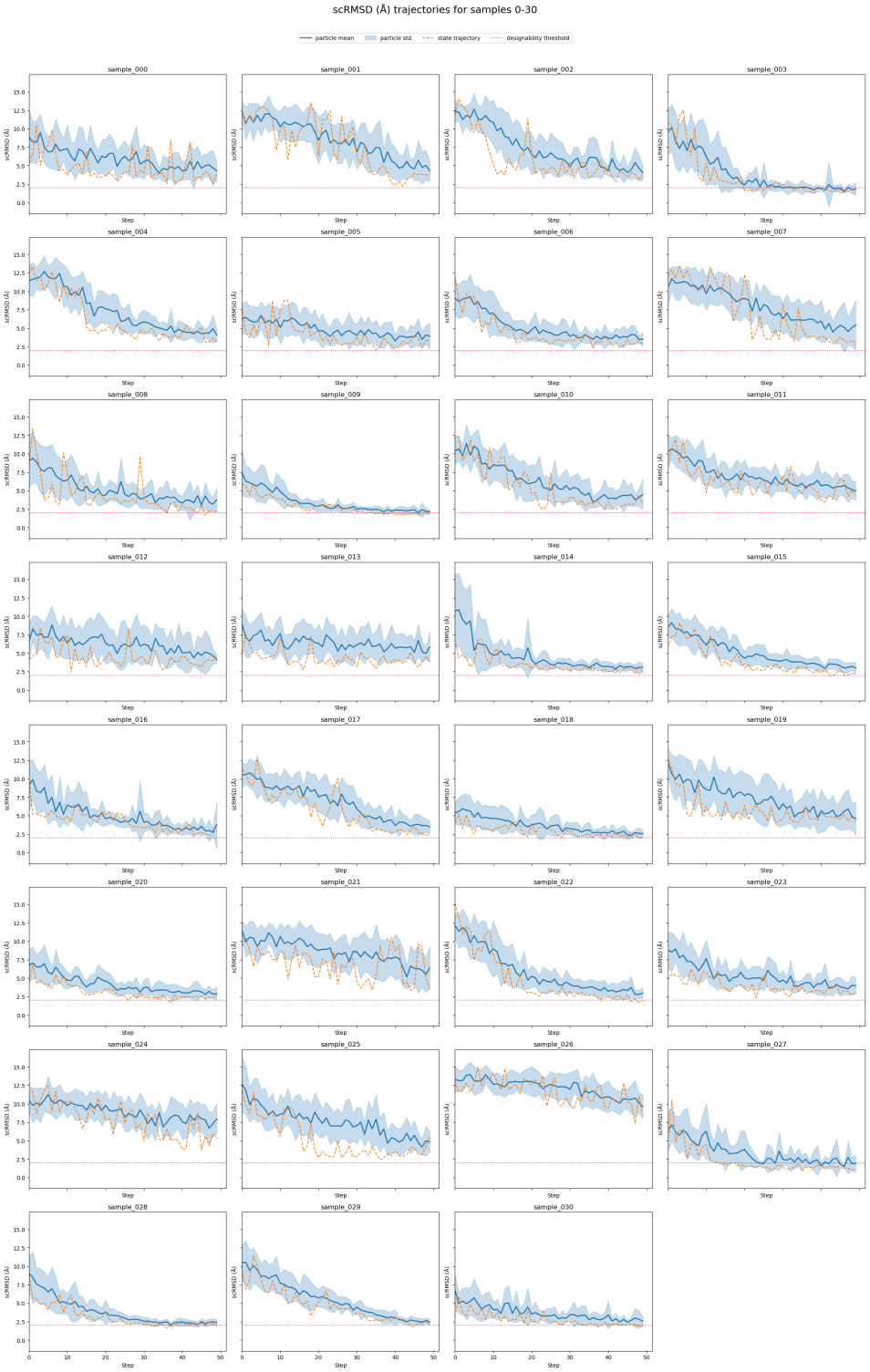


Figure 14: Step-wise scRMSD trajectories for each sample generated by the RMF/S model.



Figure 15: Step-wise scRMSD trajectories for each sample generated by the RMF/M model.