METADMD: STABLE AND SCALABLE TRAINING OF POSTERIOR SAMPLERS FOR TEST-TIME STEERING

Anonymous authors
Paper under double-blind review

ABSTRACT

Test-time steering aligns pretrained diffusion and flow models with arbitrary rewards without fine-tuning, but it requires sampling clean data from the posterior given each intermediate noisy state. Exact posterior sampling via SDE or ODE simulation is computationally expensive, and repeating it at every step makes the cost prohibitive. Stochastic flow maps reduce this cost by learning a flow map that transports Gaussian noise to the posterior in a few steps. However, they rely on consistency objectives that distill the exact integration of this trajectory, which makes training unstable and slow to converge. We observe that steering requires only samples that follow the posterior, not the trajectory that reaches them. Based on this, we propose MetaDMD, which learns a few-step posterior sampler by matching posterior distributions instead of individual trajectories. MetaDMD requires no consistency loss, or pretrained flow map, thus trains stably even on large text-to-image (T2I) models. On class-conditional ImageNet with SiT-XL/2, MetaDMD outperforms stochastic flow maps in test-time steering with reward gradients and performs comparably without them. On SD3.5-M, it also outperforms them across various rewards on DrawBench and on GenEval, while converging approximately $6\times$ faster.

1 INTRODUCTION

Diffusion and flow models (Ho et al., 2020; Lipman et al., 2023; Liu et al., 2023; Albergo et al., 2025) are the standard approach to image generation, and most state-of-the-art text-to-image (T2I) models are built on them (Esser et al., 2024; Black Forest Labs, 2024). However, they are trained only to reproduce the training data distribution, which does not directly reflect task-specific objectives such as prompt fidelity, aesthetic quality, and human preference (Xu et al., 2023; Kirstain et al., 2023; Wu et al., 2023b; Ghosh et al., 2023). It is therefore necessary to align these models with reward functions that measure such properties. This is commonly formulated as sampling from the reward-tilted distribution (Potaptchik et al., 2026b)

$$p^\star(x) \propto e^{r(x)}\, p^{\text{model}}(x), \tag{1}$$

where p^{model} over $x \in \mathbb{R}^d$ is the output distribution induced by the pretrained model and r is a given reward function. One way to sample from $p^\star$ is *test-time steering*, which keeps the pretrained model fixed and only modifies the sampling process to target $p^\star$, allowing a single model can serve any reward (Potaptchik et al., 2026b; Holderrieth et al., 2026a).

Exact reward steering requires the estimation of the value function (Uehara et al., 2025):

$$V_t(x_t) = \log \mathbb{E}_{x_1 \sim p^{\text{model}}_{1|t}(\cdot|x_t)}[e^{r(x_1)}], \tag{2}$$

where $p^{\text{model}}_{1|t}(\cdot \mid x_t)$ is the posterior of clean data given the noisy state x_t at time t. Since V_t in equation 2 involves an expectation over the posterior, estimating it requires multiple posterior samples $x_1 \sim p^{\text{model}}_{1|t}(\cdot \mid x_t)$ at every sampling step. Efficient posterior sampling from $p^{\text{model}}_{1|t}(\cdot \mid x_t)$ is therefore the core problem of test-time steering.

The simplest approach to posterior sampling is to simulate the reverse-time stochastic differential equation (SDE) from the current noisy state x_t to the end of the trajectory, which requires large number of function evaluations (NFEs). GLASS Flows (Holderrieth et al., 2026b) reduces the NFEs

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

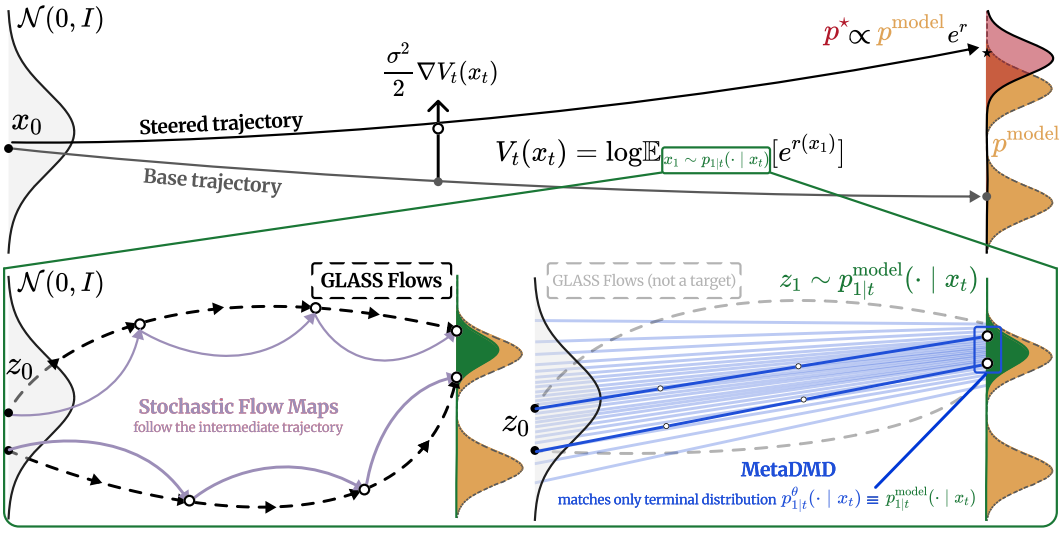

Figure 1: **Test-time steering and posterior sampling.** *(Top)* The outer sampling process carries $x_0 \sim p^{\text{prior}}$ to the reward-tilted distribution $p^\star \propto p^{\text{model}} e^r$. Steering adds the correction $\frac{\sigma_t^2}{2}\nabla V_t(x_t)$ at every outer state x_t, whose estimation requires posterior samples from $p^{\text{model}}_{1|t}(\cdot \mid x_t)$. *(Bottom)* The inner sampling process carries $z_0 \sim \mathcal{N}(0, I)$ to $z_1 \sim p^{\text{model}}_{1|t}(\cdot \mid x_t)$. GLASS Flows realizes it as a multi-step ODE of the frozen teacher, and stochastic flow maps distill its exact integration. **MetaDMD** imposes no such trajectory-level constraint and only matches $p^\theta_{1|t}(\cdot \mid x_t)$ to $p^{\text{model}}_{1|t}(\cdot \mid x_t)$.

by replacing the SDE with an ODE that transports Gaussian noise to the posterior. This ODE is built from the pretrained model and needs no training, but it still requires multiple steps. Stochastic flow maps (Potaptchik et al., 2026b; Holderrieth et al., 2026a) bring this down to a few steps by distilling the integration of this ODE into a flow map (Boffi et al., 2024; 2025), which directly carries Gaussian noise to a posterior sample given x_t.

However, stochastic flow maps face challenge in stability and scalability. They inherit the consistency-based training of standard flow maps, which constructs regression targets from the network itself rather than using fixed targets (Song et al., 2023; Boffi et al., 2024; 2025). These network-dependent targets can introduce bias, destabilize training and increase sensitivity to hyperparameters (Hu et al., 2026; Geng et al., 2026). Some variants additionally require Jacobian-vector products (JVPs), increasing memory and computational costs (Lu & Song, 2025; Geng et al., 2025; Zheng et al., 2026). These challenges hinder scaling; to the best of our knowledge, stochastic flow maps have not yet been trained on large-scale diffusion models such as T2I models (Esser et al., 2024).

To address these issues, we propose **MetaDMD**, a stable and scalable few-step posterior sampler. Our key insight is that steering requires only samples that follow the posterior $p^{\text{model}}_{1|t}(\cdot \mid x_t)$, rather than samples reached by following a specific trajectory, such as the GLASS Flows trajectory that stochastic flow maps are trained to reproduce. A posterior sampler can therefore be trained by distribution matching alone, just as Distribution Matching Distillation (DMD) (Yin et al., 2024b;a) trains a few-step student generator to match the marginal output distribution p^{model} of a pretrained model. MetaDMD extends this principle to posterior sampling by matching the student's posterior distribution $p^\theta_{1|t}(\cdot \mid x_t)$ to that of the pretrained model $p^{\text{model}}_{1|t}(\cdot \mid x_t)$. This ***posterior distribution matching*** enables few-step posterior sampling without the trajectory-level consistency constraints used by stochastic flow maps.

We evaluate MetaDMD on test-time steering with various rewards, both with and without reward gradients. We compare mainly with stochastic flow maps (Potaptchik et al., 2026b; Holderrieth et al., 2026a), using Meta Flow Maps (MFM) (Potaptchik et al., 2026b) as a representative baseline. Most importantly, MetaDMD scales to large T2I models: on SD3.5-M (Esser et al., 2024), a 2.5B-parameter model, it outperforms MFM in both standard reward scores and GenEval (Ghosh et al., 2023), while converging approximately $6\times$ faster. On class-conditional ImageNet 256×256 with SiT-XL/2 (Ma

et al., 2024), a standard benchmark for reward steering, it achieves comparable or better results than MFM. These results demonstrate that MetaDMD provides more stable and scalable posterior sampling for test-time steering than stochastic flow maps.

Our contributions are as follows:

- Motivated by the observation that steering does not require trajectory-level constraints on posterior sampling, we propose **MetaDMD**, a few-step posterior sampler trained via *posterior distribution matching*.
- By matching only its output distribution to the posterior, MetaDMD avoids the trajectory-level constraints that destabilize training, making it stable and scalable to large T2I models.
- In test-time steering both with and without reward gradients, MetaDMD outperforms MFM on SD3.5-M in standard reward scores and GenEval while converging approximately $6\times$ faster, and achieves comparable or better results on ImageNet with SiT-XL/2.

2 PRELIMINARY

Our goal is to build a few-step posterior sampler by distilling a pretrained model, the *teacher*, into a *student* generator. We review distillation methods in Sec. 2.1 and posterior samplers in Sec. 2.2.

2.1 DISTILLATION

Flow Matching. Let $v_\phi(x_t, t) : \mathbb{R}^d \times [0,1] \to \mathbb{R}^d$ be a flow matching velocity field (Liu et al., 2023; Lipman et al., 2023; Albergo et al., 2025) that generates p^{model}, an approximation of the data distribution p^{data}. Solving the ODE

$$\frac{\mathrm{d}x_t}{\mathrm{d}t} = v_\phi(x_t, t), \quad x_0 \sim \mathcal{N}(0, I), \tag{3}$$

from $t = 0$ to $t = 1$ yields samples $x_1 \sim p^{\text{model}}$. The velocity field can be equivalently expressed as the denoiser $D_\phi(x_t, t) = \mathbb{E}[x_1 \mid x_t]$ (Lipman et al., 2024), the posterior mean of clean data given the noisy state x_t:

$$D_\phi(x_t, t) = x_t + (1-t)\, v_\phi(x_t, t). \tag{4}$$

Throughout this paper, we refer to the pretrained model v_ϕ, or equivalently D_ϕ, as the *teacher*.

Flow Maps. A flow map learns the exact integration of the velocity field v_ϕ in Eq. (3) between any two times, so that a single evaluation replaces numerical simulation of the ODE (Boffi et al., 2024). Flow map learning is therefore a *trajectory distillation*, in which the student inherits the coupling between noise and samples induced by v_ϕ. Flow maps are trained with consistency objectives that compute their targets from the network itself instead of from a fixed regression target (Song et al., 2023; Boffi et al., 2025). These network-dependent targets introduce bias and make training unstable and sensitive to hyperparameters (Hu et al., 2026; Geng et al., 2026).

Distribution Matching Distillation (DMD). DMD (Yin et al., 2024b;a) distills the teacher D_ϕ into a few-step student $G_\theta(x_t, t)$, which maps a state x_t at time t directly to a clean sample $\hat{x} = G_\theta(x_t, t)$, by matching marginal distributions rather than exact individual trajectories. Specifically, DMD minimizes the KL divergence between the student's output distribution p^θ and the teacher's distribution p^{model},

$$\min_\theta \; D_{\mathrm{KL}}\big(p^\theta \,\|\, p^{\text{model}}\big). \tag{5}$$

The gradient of this KL divergence involves the scores of p^θ and p^{model} (Yin et al., 2024b), which are ill-defined where p^{model} vanishes. DMD therefore matches the two distributions in the noised space. By Tweedie's formula (Efron, 2011), this reduces to supervising the student output $\hat{x}$ with the difference between the teacher denoiser D_ϕ and the fake denoiser D_ψ at the noised state $\hat{x}_t = (1-t)\epsilon + t\,\hat{x}$, where D_ψ tracks the student's own output distribution p^θ. This yields an alternating training of the student G_θ and the fake denoiser D_ψ, or equivalently the fake velocity network v_ψ,

$$\mathcal{L}_{\text{DMD}}(\theta) = \mathbb{E}_{\hat{x}\sim p^\theta,t,\epsilon}\left[\frac{1}{2}\left\|\hat{x} - \text{sg}\left[\hat{x} - w\left(D_\psi(\hat{x}_t,t) - D_\phi(\hat{x}_t,t)\right)\right]\right\|_2^2\right], \tag{6}$$

$$\mathcal{L}_{\text{fake}}(\psi) = \mathbb{E}_{\hat{x}\sim p^\theta,\,t,\,\epsilon}\left[\left\|v_\psi(\hat{x}_t,t) - \left(\text{sg}[\hat{x}] - \epsilon\right)\right\|_2^2\right], \tag{7}$$

where $\text{sg}[\cdot]$ denotes stop-gradient and w is the scaling term.

2.2 Test-Time Steering via Posterior Sampling

Test-Time Steering. Test-time steering keeps the teacher frozen and steers its base trajectory so that the samples follow the reward-tilted distribution $p^\star$ in Eq. (1). This requires the gradient of the value function V_t in Eq. (2), which is added to the velocity field for steering :

$$x_t = x_0 + \int_0^t \left[v_\phi(x_\tau,\tau) + \frac{\sigma_\tau^2}{2}\nabla V_\tau(x_\tau)\right]\mathrm{d}\tau, \tag{8}$$

where $\sigma_t^2 = 2(1-t)/t$ is the diffusion coefficient of the linear trajectory. Since V_t is an expectation over the posterior $p_{1|t}^{\text{model}}(\cdot \mid x_t)$, estimating its gradient requires multiple posterior samples at every denoising step (see Sec. A for the estimators used in practice), which urges the need for an efficient few-step posterior sampler.

Posterior Sampling for Value Function Estimation. GLASS Flows (Holderrieth et al., 2026b) performs posterior sampling with an ODE that maps the prior p^{prior} to the posterior $p_{1|t}^{\text{model}}(\cdot \mid x_t)$, only using the teacher v_ϕ without any training. Since this ODE still requires multiple steps for each posterior sampling, stochastic flow maps (Holderrieth et al., 2026a; Potaptchik et al., 2026b) distill its integration into a student, thereby reducing posterior sampling to a few steps. However, they inherit the disadvantages of flow maps, whose unstable training hinders their scale-up to large models. In practice, this instability makes them rely on a pretrained flow map for initialization, incurring the cost of an additional pretraining stage.

3 MetaDMD

Our goal is to steer the sampling process toward the reward-tilted distribution $p^\star$. We refer to the state x_t introduced above as the *outer state*, whose evolution follows the steered trajectories in Eq. (8). At each outer state, steering requires estimating the value gradient $\nabla V_t(x_t)$, which depends on samples from the posterior distribution $p_{1|t}^{\text{model}}(\cdot \mid x_t)$.

To obtain these samples, we introduce an *inner state* z_s, with its own time $s \in [0,1]$. For each fixed outer state-time pair (x_t,t), the inner sampling process transports fresh Gaussian noise $z_0 \sim \mathcal{N}(0,I)$ to a clean posterior sample $z_1 \sim p_{1|t}^{\text{model}}(\cdot \mid x_t)$. GLASS Flows (Holderrieth et al., 2026b) constructs such a process through an ODE whose terminal distribution is $p_{1|t}^{\text{model}}(\cdot \mid x_t)$. Repeating this inner process with independent noise provides the posterior samples needed to estimate the value function and its gradient at x_t, as illustrated in Fig. 1

Crucially, this estimation requires the inner endpoints to follow the correct posterior, without requiring a particular correspondence between each initial noise z_0 and its endpoint z_1. We can therefore learn a few-step posterior sampler by matching its posterior output distribution, without reproducing the individual trajectories of the GLASS Flows. This motivates **MetaDMD**, which extends DMD to match $p_{1|t}^{\text{model}}(\cdot \mid x_t)$, conditioned on the outer state-time pair (x_t,t).

3.1 Training MetaDMD: posterior distribution matching

MetaDMD learns a few-step posterior sampler for test-time steering through *posterior distribution matching*. Given an outer state (x_t,t), it trains a student $\bar{G}_\theta(z_s,s \mid x_t,t)$ whose output distribution matches the posterior $p_{1|t}^{\text{model}}(\cdot \mid x_t)$, without constraining the individual inner trajectories. This amounts to three substitutions in DMD, summarized in Tab. 1: the state (x_t,t) is replaced by the inner state (z_s,s) conditioned on (x_t,t), the teacher denoiser is replaced by a *posterior denoiser*, and the student becomes a few-step *posterior sampler*.

Table 1: Comparison between DMD and our MetaDMD.

Method	**DMD** (Yin et al., 2024a)	**MetaDMD (Ours)**
States	(x_t, t)	inner (z_s, s) conditioned on outer (x_t, t)
Objective	$\min_\theta D_{\mathrm{KL}}\big(p^\theta \,\|\, p^{\text{model}}\big)$	$\min_\theta \mathbb{E}_{t,x_t}\big[D_{\mathrm{KL}}\big(p^\theta_{1\|t}(\cdot \mid x_t) \,\|\, p^{\text{model}}_{1\|t}(\cdot \mid x_t)\big)\big]$
Student	$\hat{x} = G_\theta(x_t, t)$	$\hat{z} = \bar{G}_\theta(z_s, s \mid x_t, t)$
Teacher	$D_\phi(x_t, t) = \mathbb{E}[x_1 \mid x_t]$	$\bar{D}_\phi(z_s, s \mid x_t, t) = \mathbb{E}[x_1 \mid x_t, z_s]$
$\mathcal{L}_{\text{DMD}}(\theta)$	Eq. (6)	Eq. (12)
$\mathcal{L}_{\text{fake}}(\psi)$	Eq. (7)	Eq. (13)

Specifically, our goal is to learn the student $\bar{G}_\theta(z_s, s \mid x_t, t)$ that maps an inner state z_s to a clean posterior sample $\hat{z}$ in a few steps, conditioned on the outer state (x_t, t). Let $p^\theta_{1|t}(\cdot \mid x_t)$ denote the distribution of the student's outputs. MetaDMD minimizes the KL divergence between this distribution and the teacher's posterior, averaged over outer states:

$$\min_\theta \; \mathbb{E}_{t,x_t}\Big[D_{\mathrm{KL}}\big(p^\theta_{1|t}(\cdot \mid x_t) \,\|\, p^{\text{model}}_{1|t}(\cdot \mid x_t)\big)\Big]. \tag{9}$$

Note that this objective matches only the distribution of the student's outputs for each outer state. It leaves the correspondence between the inner noise $z_0 \sim \mathcal{N}(0, I)$ and its endpoint $z_1 \sim p^{\text{model}}_{1|t}(\cdot \mid x_t)$ unconstrained. This is sufficient because the value function in Eq. (2) depends only on samples from the posterior, not on the trajectory along which they are obtained.

As in DMD, the KL divergence in Eq. (9) is matched in the noised space, which requires the denoiser of the teacher's distribution. For the posterior, this is the *posterior denoiser* $\bar{D}_\phi(z_s, s \mid x_t, t) = \mathbb{E}[x_1 \mid x_t, z_s]$, the conditional mean of x_1 given both the outer and inner states. It can be derived from the teacher D_ϕ in closed form without training (Holderrieth et al., 2026b):

$$\bar{D}_\phi(z_s, s \mid x_t, t) = D_\phi\big(\tilde{x}(x_t, z_s),\, \tilde{t}(t, s)\big), \tag{10}$$

where $\tilde{x}$ and $\tilde{t}$ are given in Eqs. (19) and (20) of Sec. B.

Training Objective. Each training iteration proceeds in three stages: (i) we construct an outer state x_t, (ii) roll out the student to obtain an output $\hat{z}$, and (iii) update the student and the fake denoiser with posterior distribution matching losses conditioned on x_t. We obtain the outer state x_t by noising clean data, $x_t = (1-t)x_0 + t x_1$ with $x_0 \sim \mathcal{N}(0, I)$, $x_1 \sim p^{\text{data}}$, and $t \sim \mathcal{U}(0, 1)$, as in stochastic flow maps (Potaptchik et al., 2026b; Holderrieth et al., 2026a).

Given (x_t, t), the student $\bar{G}_\theta$ generates a posterior sample $\hat{z}$ through a K-step rollout from $z_{s_0} \sim \mathcal{N}(0, I)$ over the discretized inner times $0 = s_0 < \cdots < s_K = 1$. Each inner step $k = 0, \ldots, K-1$ is an Euler step of the inner ODE with the posterior denoiser $\bar{D}_\phi$ replaced by the student (cf. Eq. (22)):

$$z_{s_{k+1}} = \frac{1 - s_{k+1}}{1 - s_k}\, z_{s_k} + \frac{s_{k+1} - s_k}{1 - s_k}\, \bar{G}_\theta(z_{s_k}, s_k \mid x_t, t). \tag{11}$$

Then, the student output $\hat{z}$ is diffused along the inner trajectory, $\hat{z}_s = (1-s)\,\epsilon + s\,\hat{z}$ with $\epsilon \sim \mathcal{N}(0, I)$ and $s \sim \mathcal{U}(0, 1)$, and the MetaDMD loss is the DMD loss in Eq. (6) with both denoisers conditioned on the outer state x_t:

$$\mathcal{L}_{\text{MetaDMD}}(\theta) = \mathbb{E}_{t,x_t,\hat{z},s,\epsilon}\Big[\tfrac{1}{2}\big\|\hat{z} - \mathrm{sg}\big[\hat{z} - w\,\big(\bar{D}_\psi(\hat{z}_s, s \mid x_t, t) - \bar{D}_\phi(\hat{z}_s, s \mid x_t, t)\big)\big]\big\|_2^2\Big], \tag{12}$$

where w normalizes the update per sample (Yin et al., 2024a). The difference of the two denoisers is the difference of the scores of $p^\theta_{1|t}$ and $p^{\text{model}}_{1|t}$ diffused to inner time s, so the loss moves $\hat{z}$ toward the posterior; its gradient is a descent direction for the KL objective in Eq. (9) along the inner path (Sec. C). Here, $\bar{D}_\psi(z_s, s \mid x_t, t) = z_s + (1-s)\,\bar{v}_\psi(z_s, s \mid x_t, t)$ is the fake denoiser, parameterized as the teacher denoiser in Eq. (10), and its inner velocity $\bar{v}_\psi$ is trained with the flow matching loss on the diffused student outputs:

Algorithm 1 MetaDMD training

Require: Teacher v_ϕ, inner steps K, student update period M

$\theta \leftarrow \phi$, $\psi \leftarrow \phi$
for $n = 0, 1, 2, \ldots$ **do**
  Sample $x_1 \sim p^{\text{data}}$, $x_0 \sim \mathcal{N}(0, I)$, $t \sim \mathcal{U}(0, 1)$ and set $x_t \leftarrow (1-t)\, x_0 + t\, x_1$ ▷ outer state
  Sample $z_{s_0} \sim \mathcal{N}(0, I)$ and rollout Eq. (11) without gradients
  Sample $k \sim \mathcal{U}\{0, \ldots, K-1\}$ and set $\hat{z} \leftarrow \bar{G}_\theta(z_{s_k}, s_k \mid x_t, t)$ ▷ with gradients
  if $n \bmod M = 0$ **then**
    Sample $s \sim \mathcal{U}(0, 1)$, $\epsilon \sim \mathcal{N}(0, I)$ and set $\hat{z}_s \leftarrow (1-s)\, \epsilon + s\, \hat{z}$
    Update θ with $\nabla_\theta \mathcal{L}_{\text{MetaDMD}}$ in Eq. (12)
  end if
  Sample $s \sim \mathcal{U}(0, 1)$, $\epsilon \sim \mathcal{N}(0, I)$ and set $\hat{z}_s \leftarrow (1-s)\, \epsilon + s\, \text{sg}[\hat{z}]$
  Update ψ with $\nabla_\psi \mathcal{L}_{\text{Metafake}}$ in Eq. (13)
end for

Algorithm 2 Steering with MetaDMD

Require: Teacher v_ϕ, student G_θ, reward r, outer grid $0 = t_0 < \cdots < t_J = 1$, number of posterior samples N

Sample $x_{t_0} \sim \mathcal{N}(0, I)$
for $j = 0, \ldots, J-1$ **do**
  Sample $z_{s_0}^{(i)} \sim \mathcal{N}(0, I)$, rollout Eq. (11) with $G_\theta(\cdot \mid x_{t_j}, t_j)$ and set $x_1^{(i)} \leftarrow z_{s_K}^{(i)}$ for $i = 1, \ldots, N$
  $g \leftarrow$ Estimate of $\nabla V_{t_j}(x_{t_j})$ with Eq. (15) or Eq. (16)
  $v \leftarrow v_\phi(x_{t_j}, t_j)$ ▷ or $\bar{v}(x_{t_j}, t_j)$ in Eq. (14)
  $x_{t_{j+1}} \leftarrow x_{t_j} + (t_{j+1} - t_j)\big[v + \frac{\sigma_{t_j}^2}{2}\, g\big]$
end for
return x_{t_J}

$$\mathcal{L}_{\text{Metafake}}(\psi) = \mathbb{E}_{t, x_t, \hat{z}, s, \epsilon}\Big[\big\|\bar{v}_\psi(\hat{z}_s, s \mid x_t, t) - \big(\text{sg}[\hat{z}] - \epsilon\big)\big\|_2^2\Big]. \tag{13}$$

Both losses share the same outer state, so the fake denoiser tracks the posterior distribution $p_{1|t}^\theta(\cdot \mid x_t)$.

In summary, MetaDMD requires only forward noising of data, the student's own rollout, and evaluations of the teacher's posterior denoiser. Unlike stochastic flow maps (Potaptchik et al., 2026b; Holderrieth et al., 2026a), which constrain the entire inner trajectory and derive regression targets from the student itself, our objective supervises *only the output posterior distribution* against a fixed teacher. It therefore involves no consistency loss, no JVP, and no pretrained flow map, which is the source of its stability and scalability. algorithm 1 summarizes the training procedure.

3.2 Steering with MetaDMD

Once trained, the MetaDMD student $\bar{G}_\theta$ replaces the multi-step inner sampling process at every outer state x_t of the steered trajectory. Steering proceeds along the outer trajectory in Eq. (8), where the frozen teacher provides the velocity v_ϕ at x_t and the student provides the posterior samples from $p_{1|t}^{\text{model}}(\cdot \mid x_t)$ for estimating ∇V_t.

Steered Sampler. Given the current outer state x_t, we draw N posterior samples $\{x_1^{(i)}\}_{i=1}^N$ by running the rollout in Eq. (11) with $\bar{G}_\theta$ from independent noises $z_{s_0}^{(i)} \sim \mathcal{N}(0, I)$, and use them to estimate $\nabla V_t(x_t)$. We consider two estimators (App. A): a gradient-free estimator that reweights the posterior samples by their exponentiated rewards (Potaptchik et al., 2026a;b), and a gradient-based estimator that backpropagates the reward through the samples (Potaptchik et al., 2026b). The latter is available because the rollout in Eq. (11) is differentiable with respect to x_t. The outer state then advances along the steered trajectories in Eq. (8), with the velocity v_ϕ provided by the frozen teacher.



Figure 2: Qualitative comparison of test-time steering with the GenEval on SD3.5-M at 512×512.

Teacher-Free Variant. At inference, the teacher only provides the velocity $v_\phi(x_t, t) = \big(D_\phi(x_t, t) - x_t\big)/(1-t)$ in Eq. (4), where $D_\phi(x_t, t)$ is the mean of the posterior $p_{1|t}^{\text{model}}(\cdot \mid x_t)$. We estimate this mean with the posterior samples already drawn for ∇V_t:

$$\bar{v}(x_t, t) = \frac{\bar{x}_1 - x_t}{1-t}, \qquad \bar{x}_1 = \frac{1}{N}\sum_{i=1}^{N} x_1^{(i)}. \tag{14}$$

Using $\bar{v}$ in place of v_ϕ, the teacher need not be evaluated or kept in memory at inference. Algorithm 2 summarizes the sampler.

4 Experiments

4.1 Experimental Setup

Baselines. We compare MetaDMD with Meta Flow Map (MFM) (Potaptchik et al., 2026b), a representative stochastic flow map, on class-conditional ImageNet 256×256 with SiT-XL/2 (Ma et al., 2024) and on text-to-image generation with Stable Diffusion 3.5 Medium (SD3.5-M) (Esser et al., 2024). On ImageNet, we use the official MFM-XL/2 checkpoint, initialized from Decoupled MeanFlow (DMF-XL/2+) (Lee et al., 2026). On SD3.5-M, MFM is initialized from a public MeanFlow checkpoint[1] and trained with the MeanFlow objective (Geng et al., 2025). For vanilla SD3.5-M, we use a CFG scale of 4.0.

Evaluations. We use four human preference rewards: ImageReward (Xu et al., 2023), PickScore (Kirstain et al., 2023), HPSv2 (Wu et al., 2023b), and the LAION aesthetic predictor (Schuhmann, 2022). On SD3.5-M, we use DrawBench prompts (Saharia et al., 2022) and additionally steer toward GenEval (Ghosh et al., 2023) with the gradient-free estimator. Each run is steered with a single reward and evaluated with the same reward.

Sampling. We use $J = 28$ outer steps and $K = 4$ inner steps for both MetaDMD and MFM unless otherwise stated, so the same N corresponds to the same number of posterior sampler evaluations for both methods. Further implementation details are provided in App. D.

[1] https://huggingface.co/Yuanzhi/sd3.5-meanflow

Table 2: **Test-time steering on SD3.5-M** with the gradient-based estimator in Eq. (16) (top) and the gradient-free estimator in Eq. (15) (bottom). Each column steers with the listed reward and evaluates the same reward on DrawBench, except GenEval, which is evaluated on the GenEval benchmark. GenEval is non-differentiable and is therefore used only with the gradient-free estimator. The Teacher row reports SD3.5-M with CFG scale 4.0 and no steering. All results use $J = 28$ outer steps. **Bold** and underline denote the best and second-best values for each reward within each estimator.

	ImageReward↑		PickScore↑		HPSv2↑		Aesthetic↑		GenEval↑	
	MFM	**MetaDMD**	MFM	**MetaDMD**	MFM	**MetaDMD**	MFM	**MetaDMD**	MFM	**MetaDMD**
Teacher	0.79		22.27		0.276		5.48		0.672	
Gradient-based estimator										
$N = 1$	0.74	0.90	22.22	22.11	0.273	0.277	5.49	5.53	n/a	
$N = 4$	0.74	0.93	22.22	22.25	0.273	0.281	5.50	5.58	n/a	
$N = 8$	0.74	**0.99**	22.23	**22.29**	0.274	**0.285**	5.50	**5.62**	n/a	
Gradient-free estimator										
$N = 1$	0.74	0.78	22.21	22.28	0.273	0.276	5.47	5.53	–	–
$N = 4$	0.80	1.20	22.70	22.73	0.274	0.291	5.49	5.71	0.633	0.723
$N = 8$	0.82	**1.33**	22.12	**22.93**	0.275	**0.297**	5.52	**5.83**	0.672	**0.776**

4.2 Text-to-Image Generation

Training Efficiency. Since MetaDMD starts directly from the teacher, it requires far less training than MFM. The MFM baseline is obtained by training a MeanFlow model on SD3.5-M for 10K iterations and then training MFM for another 2K iterations, which amounts to 12K iterations in total. MetaDMD requires $6\times$ fewer iterations, reaching the results in Tab. 2 after only 2K iterations.

Steering Results. Tab. 2 reports steering on SD3.5-M with both estimators. With the gradient-based estimator, MetaDMD improves ImageReward, HPSv2 and the aesthetic score over the unsteered teacher, and these gains grow with N. The difference from MFM is clearest with the gradient-free estimator. At $N = 8$ MetaDMD raises ImageReward from 0.79 to 1.33, whereas MFM stays within 0.04 of the teacher on ImageReward, HPSv2 and the aesthetic score. MetaDMD also steers SD3.5-M toward the non-differentiable GenEval score and raises it from 0.672 to 0.723 at $N = 4$.

4.3 Class-Conditional ImageNet

Training Efficiency. The efficiency gap is even larger on ImageNet. The official MFM-XL/2 checkpoint is initialized from DMF-XL/2+, which is distilled from SiT-XL/2 for 400K iterations (Lee et al., 2026), and is then trained for another 100K iterations (Potaptchik et al., 2026b). MetaDMD is trained from SiT-XL/2 for only 2K iterations, which is $250\times$ fewer than the 500K iterations required for MFM.

Steering Results. Tab. 3 reports steering with gradient-based estimator for $N \in \{1, 4, 8, 16\}$ posterior samples per outer step. MetaDMD achieves higher ImageReward, PickScore and HPSv2 than MFM at every N. Fig. 3 shows qualitative samples steered with HPSv2.

Teacher-Free Steering. Since the teacher provides only the base velocity during steering, we examine whether the MetaDMD student can also fulfill this role. Tab. 4 replaces the teacher velocity v_ϕ with the estimate $\bar{v}$ in Eq. (14), which is computed from the MetaDMD posterior samples alone. This teacher-free variant matches or improves ImageReward, PickScore and HPSv2 at every N and lowers the aesthetic score by at most 0.12.

5 Related Work

Test-time steering keeps the pretrained model frozen and changes only its sampling process so that the samples follow the reward-tilted distribution in Eq. (1) (Uehara et al., 2025). Exact steering requires the value function V_t in Eq. (2), an expectation over the posterior $p^{\text{model}}_{1|t}(\cdot \mid x_t)$, and existing methods differ mainly in how they approximate this posterior. MetaDMD belongs to the family of methods that estimate V_t by Monte Carlo averaging over samples drawn from the posterior. Within this family, GLASS Flows (Holderrieth et al., 2026b) obtain these samples with an ODE built from the pretrained model, and stochastic flow maps (Potaptchik et al., 2026b; Holderrieth et al., 2026a) distill this

Table 3: **Test-time steering on SiT-XL/2** for ImageNet 256×256 with the gradient-based estimator for different numbers of MC samples N. All results use $J = 28$ outer steps and $K = 4$ inner steps per posterior sample. The Teacher row reports the teacher without steering. **Bold** and underline denote the best and second-best values for each reward.

	ImageReward↑		PickScore↑		HPSv2↑		Aesthetic↑	
	MFM	**MetaDMD**	MFM	**MetaDMD**	MFM	**MetaDMD**	MFM	**MetaDMD**
Teacher	0.19		20.80		0.240		4.72	
$N = 1$	1.49	1.62	22.48	23.99	0.320	0.345	7.17	7.40
$N = 4$	1.63	1.71	23.83	24.76	0.354	0.363	7.68	7.73
$N = 8$	1.61	**1.74**	24.23	25.02	0.360	0.368	7.84	7.81
$N = 16$	1.68	1.71	24.60	**25.32**	0.371	**0.377**	**8.02**	7.92

Table 4: **Teacher-free steering.** Teacher-based and teacher-free steering with MetaDMD and the gradient-based estimator. v_ϕ uses the frozen teacher for the base velocity as in our main results, and $\bar{v}$ replaces it with the posterior sample mean in Eq. (14). All results use $J = 28$ outer steps and $K = 4$ inner steps per posterior sample. **Bold** and underline denote the best and second-best values for each reward.

	ImageReward↑		PickScore↑		HPSv2↑		Aesthetic↑	
	v_ϕ	$\bar{v}$	v_ϕ	$\bar{v}$	v_ϕ	$\bar{v}$	v_ϕ	$\bar{v}$
$N = 1$	1.62	1.67	23.99	24.14	0.345	0.350	7.40	7.36
$N = 4$	1.71	1.74	24.76	24.76	0.363	0.363	7.73	7.66
$N = 8$	1.74	**1.76**	25.02	25.18	0.368	0.371	7.81	7.69
$N = 16$	1.71	1.75	**25.32**	**25.32**	**0.377**	**0.377**	**7.92**	7.81

ODE into a few-step sampler. Stochastic flow maps, however, rely on consistency objectives that are unstable to train and hard to scale. Our observation is that steering needs only the distribution of the inner endpoints, not the trajectory that reaches them Therefore, MetaDMD trains the posterior sampler by matching its output distribution to the teacher posterior, which requires no consistency loss and trains stably at scale.

Other methods avoid posterior sampling and replace the posterior with a single look-ahead prediction $x_1^{\mathrm{pt}}(x_t)$ of the clean sample, which approximates the value function by $V_t(x_t) \approx r\big(x_1^{\mathrm{pt}}(x_t)\big)$. We refer to these as *point estimates*, since they implicitly approximate the posterior by a Dirac delta at $x_1^{\mathrm{pt}}(x_t)$. Guidance methods take this point to be the denoiser output $D_\phi(x_t, t)$ (Chung et al., 2023), which is the posterior mean and thus a blurry average of plausible outcomes. Flow map look-ahead sharpens this point estimate by mapping x_t onto the data manifold with a few-step generator (Sabour et al., 2025). Both choices reduce the posterior to a single point, so the value estimate remains biased unless the posterior is concentrated around that point. Sequential Monte Carlo and importance sampling methods correct this bias by reweighting and resampling a population of particles whose weights are computed from such point estimates (Wu et al., 2023a; Singhal et al., 2025; Kim et al., 2025; Sabour et al., 2025). These corrections become exact only as the number of particles grows, so they remain costly in high dimensions.

6 Conclusion

We present MetaDMD, a few-step posterior sampler for test-time steering trained by posterior distribution matching. Our key observation is that value function estimation depends only on samples from the posterior, not on the trajectory that produces them. Building on this, MetaDMD extends DMD to match the student's posterior distribution to that of the teacher, conditioned on the outer state. This removes the consistency losses, JVPs and pretrained flow map initialization that stochastic flow maps rely on, so MetaDMD can be trained directly from the teacher. As a result, it trains stably on large T2I models. On SD3.5-M, it outperforms MFM across rewards on DrawBench and on GenEval while converging approximately 6× faster, and on ImageNet with SiT-XL/2 it achieves comparable or better steering performance. Moreover, the mean of its posterior samples can replace the teacher velocity, which removes the teacher from inference. We hope that separating posterior sampling from trajectory distillation makes reward steering practical for larger diffusion and flow models.

AI USE STATEMENT

We did not use generative AI tools for any tasks with required disclosure under the ICLR 2027 policy. We used generative AI tools only to improve the readability of the manuscript, such as polishing sentences and checking grammar. All AI-assisted edits were reviewed by the authors, and we take full responsibility for the content of this work.

REFERENCES

Michael Albergo, Nicholas M Boffi, and Eric Vanden-Eijnden. Stochastic interpolants: A unifying framework for flows and diffusions. *Journal of Machine Learning Research*, 2025.

Black Forest Labs. FLUX. https://github.com/black-forest-labs/flux, 2024.

Nicholas M. Boffi, Michael S. Albergo, and Eric Vanden-Eijnden. Flow map matching with stochastic interpolants: A mathematical framework for consistency models. *arXiv preprint arXiv:2406.07507*, 2024.

Nicholas M. Boffi, Michael S. Albergo, and Eric Vanden-Eijnden. How to build a consistency model: Learning flow maps via self-distillation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. arXiv:2505.18825.

Hyungjin Chung, Jeongsol Kim, Michael T. Mccann, Marc L. Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems. In *International Conference on Learning Representations (ICLR)*, 2023.

Bradley Efron. Tweedie's formula and selection bias. *Journal of the American Statistical Association*, 106(496):1602–1614, 2011.

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. In *International Conference on Machine Learning (ICML)*, 2024.

Zhengyang Geng, Mingyang Deng, Xingjian Bai, J. Zico Kolter, and Kaiming He. Mean flows for one-step generative modeling. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Zhengyang Geng, Yiyang Lu, Zongze Wu, Eli Shechtman, J. Zico Kolter, and Kaiming He. Improved mean flows: On the challenges of fastforward generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

Dhruba Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. GenEval: An object-focused framework for evaluating text-to-image alignment. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Peter Holderrieth, Douglas Chen, Luca Eyring, Ishin Shah, Giri Anantharaman, Yutong He, Zeynep Akata, Tommi Jaakkola, Nicholas Matthew Boffi, and Max Simchowitz. Diamond maps: Efficient reward alignment via stochastic flow maps. In *Forty-third International Conference on Machine Learning (ICML)*, 2026a.

Peter Holderrieth, Uriel Singer, Tommi Jaakkola, Ricky T. Q. Chen, Yaron Lipman, and Brian Karrer. GLASS flows: Transition sampling for alignment of flow and diffusion models. In *International Conference on Learning Representations (ICLR)*, 2026b.

Zheyuan Hu, Chieh-Hsin Lai, Yuki Mitsufuji, and Stefano Ermon. CMT: Mid-training for efficient learning of consistency, mean flow, and flow map models. In *International Conference on Learning Representations (ICLR)*, 2026. arXiv:2509.24526.

Sunwoo Kim, Minkyu Kim, and Dongmin Park. Test-time alignment of diffusion models without reward over-optimization. In *International Conference on Learning Representations (ICLR)*, 2025.

Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiana, Joe Penna, and Omer Levy. Pick-a-pic: An open dataset of user preferences for text-to-image generation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Kyungmin Lee, Sihyun Yu, and Jinwoo Shin. Decoupled meanflow: Turning flow models into flow maps for accelerated sampling. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=P4VsQexEPe.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations*, 2023.

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

Xingchao Liu, Chengyue Gong, and qiang liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *The Eleventh International Conference on Learning Representations*, 2023.

Cheng Lu and Yang Song. Simplifying, stabilizing and scaling continuous-time consistency models. In *International Conference on Learning Representations (ICLR)*, 2025.

Nanye Ma, Mark Goldstein, Michael S. Albergo, Nicholas M. Boffi, Eric Vanden-Eijnden, and Saining Xie. SiT: Exploring flow and diffusion-based generative models with scalable interpolant transformers. In *European Conference on Computer Vision (ECCV)*, 2024.

Peter Potaptchik, Lee Cheuk Kit, and Michael Samuel Albergo. Tilt matching for scalable sampling and fine-tuning. In *Forty-third International Conference on Machine Learning*, 2026a.

Peter Potaptchik, Adhi Saravanan, Abbas Mammadov, Alvaro Prat, Michael Samuel Albergo, and Yee Whye Teh. Meta flow maps enable scalable reward alignment. In *Forty-third International Conference on Machine Learning*, 2026b. URL https://openreview.net/forum?id=K5gV8Yptne.

Amirmojtaba Sabour, Michael S. Albergo, Carles Domingo-Enrich, Nicholas M. Boffi, Sanja Fidler, Karsten Kreis, and Eric Vanden-Eijnden. Test-time scaling of diffusions with flow maps. *arXiv preprint arXiv:2511.22688*, 2025.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, Jonathan Ho, David Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Christoph Schuhmann. LAION-AESTHETICS, August 2022. https://laion.ai/blog/laion-aesthetics/. Accessed: 2026-09-20.

Raghav Singhal, Zachary Horvitz, Ryan Teehan, Mengye Ren, Zhou Yu, Kathleen McKeown, and Rajesh Ranganath. A general framework for inference-time scaling and steering of diffusion models. In *International Conference on Machine Learning (ICML)*, 2025.

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. In *International Conference on Machine Learning (ICML)*, 2023.

Masatoshi Uehara, Yulai Zhao, Chenyu Wang, Xiner Li, Aviv Regev, Sergey Levine, and Tommaso Biancalani. Inference-time alignment in diffusion models with reward-guided generation: Tutorial and review. *arXiv preprint arXiv:2501.09685*, 2025.

Luhuan Wu, Brian L. Trippe, Christian A. Naesseth, David M. Blei, and John P. Cunningham. Practical and asymptotically exact conditional sampling in diffusion models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023a.

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

Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. ImageReward: Learning and evaluating human preferences for text-to-image generation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Tianwei Yin, Michaël Gharbi, Taesung Park, Richard Zhang, Eli Shechtman, Frédo Durand, and William T. Freeman. Improved distribution matching distillation for fast image synthesis. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024a.

Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Frédo Durand, William T. Freeman, and Taesung Park. One-step diffusion with distribution matching distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024b.

Kaiwen Zheng, Yuji Wang, Qianli Ma, Huayu Chen, Jintao Zhang, Yogesh Balaji, Jianfei Chen, Ming-Yu Liu, Jun Zhu, and Qinsheng Zhang. Large scale diffusion distillation via score-regularized continuous-time consistency. In *The Fourteenth International Conference on Learning Representations (ICLR)*, 2026.

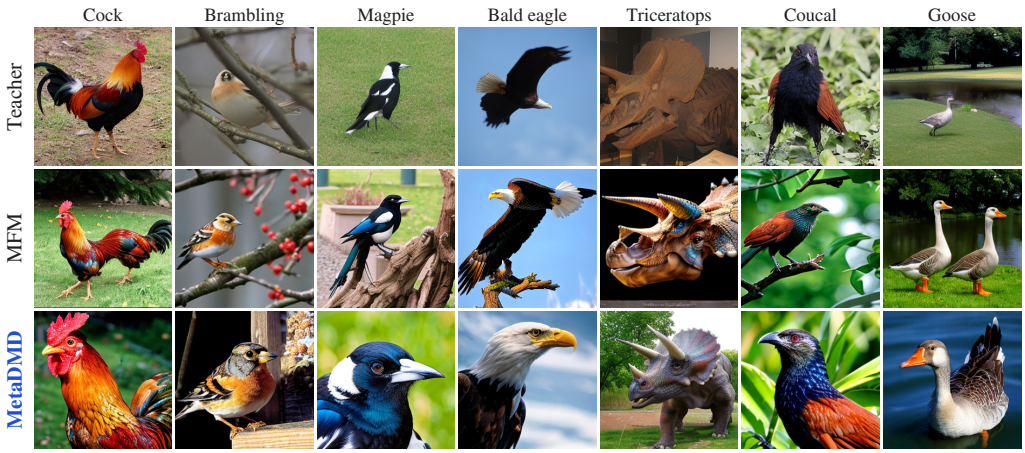

Figure 3: Qualitative comparison of test-time steering with the HPSv2 reward on class-conditional ImageNet 256×256. From top to bottom, the rows show samples from the teacher SiT-XL/2 without steering, MFM (Potaptchik et al., 2026b) and MetaDMD.

A Estimators of the Value Gradient

Since ∇V_t in equation 8 has no closed form, we estimate it at every outer step from N posterior samples $x_1^{(i)} \sim p_{1|t}^{\text{model}}(\cdot \mid x_t)$. We use one of the two estimators below.

Gradient-Free Estimator. Under the linear path the posterior score is $\nabla_{x_t} \log p_{1|t}^{\text{model}}(x_1 \mid x_t) = t\big(x_1 - D_\phi(x_t, t)\big)/(1-t)^2$. Differentiating V_t in equation 2 therefore gives ∇V_t as $t/(1-t)^2$ times the gap between the reward-tilted posterior mean and the denoiser D_ϕ in equation 4. Estimating the tilted mean with self-normalized weights gives an estimator that needs no gradient of the reward r Potaptchik et al. (2026a;b),

$$\widehat{\nabla V_t}(x_t) = \frac{t}{(1-t)^2}\Big(\sum_{i=1}^{N} \omega_i\, x_1^{(i)} - D_\phi(x_t, t)\Big), \quad \omega_i = \frac{\exp\big(r(x_1^{(i)})\big)}{\sum_{i'=1}^{N} \exp\big(r(x_1^{(i')})\big)}. \tag{15}$$

Gradient-Based Estimator. If r is differentiable and the posterior samples come from a differentiable sampler $x_1^{(i)} = \Phi(\epsilon^{(i)}; x_t, t)$ with $\epsilon^{(i)} \sim \mathcal{N}(0, I)$, we can instead differentiate a Monte Carlo estimate of V_t with respect to x_t Potaptchik et al. (2026b),

$$\widehat{\nabla V_t}(x_t) = \nabla_{x_t} \log \frac{1}{N} \sum_{i=1}^{N} \exp\Big(r\big(\Phi(\epsilon^{(i)}; x_t, t)\big)\Big). \tag{16}$$

For MetaDMD the sampler Φ is the student rollout in equation 11 with $\epsilon^{(i)} = z_{s_0}^{(i)}$, so the gradient flows through every evaluation of $\bar{G}_\theta$ while the inner noises stay fixed.

B GLASS Flows for Posterior Sampling

We derive the posterior denoiser in equation 10 and the inner velocity of the GLASS ODE for the case used throughout this paper. GLASS Flows Holderrieth et al. (2026b) sample a general transition $p_{t'|t}^{\text{model}}$ from a pretrained flow model. We only need the posterior $p_{1|t}^{\text{model}}(\cdot \mid x_t)$, which is the special case $t' = 1$. In this case the construction reduces to a short computation that we give in full.

Inner path. Fix an outer state $x_t = (1-t)\, x_0 + t\, x_1$ on the linear path with $x_0 \sim \mathcal{N}(0, I)$ and $x_1 \sim p^{\text{model}}$, so that x_1 given x_t follows the teacher's posterior $p_{1|t}^{\text{model}}(\cdot \mid x_t)$. For $s \in [0, 1]$ we

define the inner state as a second noisy observation of the same clean sample x_1

$$z_s = (1-s)\,\epsilon + s\,x_1, \qquad \epsilon \sim \mathcal{N}(0, I) \text{ independent of } x_0. \tag{17}$$

Conditioned on x_t the inner path starts from $z_0 = \epsilon \sim \mathcal{N}(0, I)$ and ends at $z_1 = x_1 \sim p^{\text{model}}_{1|t}(\cdot \mid x_t)$. This is the GLASS transition with $t' = 1$ and a linear inner schedule.

Inner velocity. Given x_1 the inner path equation 17 has the conditional velocity $\frac{\mathrm{d}z_s}{\mathrm{d}s} = x_1 - \epsilon = (x_1 - z_s)/(1-s)$. Here x_1 is random given (x_t, z_s) with law $p(x_1 \mid x_t, z_s)$. Marginalizing the conditional velocity over this law gives the inner ODE

$$\frac{\mathrm{d}z_s}{\mathrm{d}s} = \frac{\mathbb{E}[x_1 \mid x_t, z_s] - z_s}{1-s}. \tag{18}$$

By the marginalization argument of flow matching Lipman et al. (2023) with all distributions conditioned on x_t, integrating equation 18 from $z_0 \sim \mathcal{N}(0, I)$ transports z_s along the law of equation 17. Hence $z_1 \sim p^{\text{model}}_{1|t}(\cdot \mid x_t)$. The only unknown in equation 18 is the two-observation denoiser $\mathbb{E}[x_1 \mid x_t, z_s]$. The next result shows that it equals the one-observation denoiser of the teacher at a reparameterized state and time.

Sufficient statistic. Let $\mathrm{snr}(t) = t/(1-t)$ denote the signal-to-noise ratio of the linear path. For $t, s \in [0, 1)$ define

$$\lambda(t, s) = \mathrm{snr}(t)^2 + \mathrm{snr}(s)^2, \qquad \tilde{t}(t, s) = \frac{\sqrt{\lambda}}{1 + \sqrt{\lambda}}, \tag{19}$$

$$\tilde{x}(x_t, z_s) = \frac{1}{\sqrt{\lambda}\,(1 + \sqrt{\lambda})}\Big(\frac{t}{(1-t)^2}\,x_t + \frac{s}{(1-s)^2}\,z_s\Big). \tag{20}$$

Proposition 1. *The two observations (x_t, z_s) and the single observation $x_{\tilde{t}} = \tilde{x}$ on the same linear path as x_t carry the same information about x_1. That is $p(x_1 \mid x_t, z_s) = p^{\text{model}}_{1|\tilde{t}}(x_1 \mid \tilde{x})$ and in particular*

$$\mathbb{E}[x_1 \mid x_t, z_s] = D\big(\tilde{x}, \tilde{t}\big), \tag{21}$$

where $D(x, t) = \mathbb{E}[x_1 \mid x_t = x]$ is the exact denoiser of this linear path.

Proof. By Bayes' rule $p(x_1 \mid x_t, z_s) \propto p^{\text{model}}(x_1)\,p(x_t \mid x_1)\,p(z_s \mid x_1)$ since x_0 and ϵ are independent. Write $a = \mathrm{snr}(t)$ and $b = \mathrm{snr}(s)$. Dropping terms that do not depend on x_1 the log-likelihood of the two observations is

$$\begin{aligned}\log p(x_t \mid x_1) + \log p(z_s \mid x_1) &= -\frac{\|x_t - t\,x_1\|^2}{2(1-t)^2} - \frac{\|z_s - s\,x_1\|^2}{2(1-s)^2} \\ &= -\frac{a^2 + b^2}{2}\,\|x_1\|^2 + x_1^\top\Big(\frac{t}{(1-t)^2}\,x_t + \frac{s}{(1-s)^2}\,z_s\Big) + \text{const}.\end{aligned}$$

For a single observation $x_{\tilde{t}} = \tilde{x}$ the same expansion gives

$$\log p(x_{\tilde{t}} = \tilde{x} \mid x_1) = -\frac{\mathrm{snr}(\tilde{t})^2}{2}\,\|x_1\|^2 + x_1^\top\,\frac{\tilde{t}}{(1-\tilde{t})^2}\,\tilde{x} + \text{const}.$$

The quadratic terms agree because $\mathrm{snr}(\tilde{t}) = \sqrt{\lambda}$ by equation 19. The linear terms agree because $\tilde{t}/(1-\tilde{t})^2 = \sqrt{\lambda}\,(1 + \sqrt{\lambda})$ and by equation 20. Both likelihoods are therefore equal up to a factor independent of x_1 and so are the posteriors. The argument never uses the form of p^{model}, so the result holds for any distribution of x_1. □

Replacing the exact denoiser D by the teacher denoiser D_ϕ in equation 4 within Proposition 1 gives the posterior denoiser $\bar{D}_\phi$ in equation 10, and substituting it into equation 18 gives the inner velocity of the GLASS ODE

$$\bar{v}_\phi(z_s, s \mid x_t, t) = \frac{\bar{D}_\phi(z_s, s \mid x_t, t) - z_s}{1-s} = \frac{D_\phi\big(\tilde{x}(x_t, z_s),\, \tilde{t}(t, s)\big) - z_s}{1-s}.$$

Each inner step thus costs a single teacher evaluation and no retraining is required. We apply classifier-free guidance to D_ϕ before this substitution and treat the guided denoiser as exact as in Holderrieth et al. (2026b).

Relation to the general GLASS velocity. GLASS Flows Holderrieth et al. (2026b) write the inner velocity for a general inner schedule as $c_1(s)\, z_s + c_2(s)\, D_\phi + c_3(s)\, x_t$. With $t' = 1$ the inner path starts from pure noise, so $c_3 = 0$. With the linear inner schedule we get $c_1 = -1/(1-s)$ and $c_2 = 1/(1-s)$, which recovers equation 18. Their sufficient statistic and reparameterized time correspond to $\tilde{x}/\tilde{t}$ and $\tilde{t}$ in our notation.

Sampling. GLASS Flows integrate the inner ODE with Euler steps on a grid $0 = s_0 < \cdots < s_K = 1$. Since the grid ends at $s_K = 1$ the velocity is never evaluated at $s = 1$ and the factor $1/(1-s)$ stays finite. One Euler step from s_k to s_{k+1} reads

$$z_{s_{k+1}} = \frac{1 - s_{k+1}}{1 - s_k}\, z_{s_k} + \frac{s_{k+1} - s_k}{1 - s_k}\, \bar{D}_\phi(z_{s_k}, s_k \mid x_t, t). \tag{22}$$

This has the same form as the student rollout equation 11 with the student $\bar{G}_\theta$ replaced by the posterior denoiser $\bar{D}_\phi$.

C Distribution Matching Loss and the KL Divergence

We show that the MetaDMD loss equation 12 follows the gradient of the KL objective equation 9 applied along the inner probability path, following the argument of DMD Yin et al. (2024b). The standard DMD loss equation 6 is the special case in which the outer state is pure noise ($t = 0$). Then equation 19 and equation 20 give $\tilde{t} = s$ and $\tilde{x} = z_s$, so the posterior denoiser reduces to the teacher denoiser and the inner time s takes the role of t in equation 6.

Setup. We fix an outer state x_t and drop it from the notation of the distributions below. The student output $\hat{z}$ follows $p_1^\theta = p_{1|t}^\theta(\cdot \mid x_t)$ and the teacher's posterior is $p_1 = p_{1|t}^{\text{model}}(\cdot \mid x_t)$. Diffusing both along the inner path gives

$$\hat{z}_s = (1-s)\,\epsilon + s\,\hat{z} \sim p_s^\theta, \qquad z_s = (1-s)\,\epsilon + s\,x_1 \sim p_s \text{ with } x_1 \sim p_1, \tag{23}$$

where $\epsilon \sim \mathcal{N}(0, I)$ is independent of $\hat{z}$ and x_1.

Scores from denoisers. By Tweedie's formula Efron (2011) the score of a distribution diffused along the linear path is affine in its denoiser, $\nabla_z \log q_s(z) = \big(s\,\mathbb{E}[x_1 \mid z_s = z] - z\big)/(1-s)^2$ for $z_s = (1-s)\,\epsilon + s\,x_1$ with $x_1 \sim q$. On the teacher side this denoiser is $\mathbb{E}[x_1 \mid x_t, z_s]$, which equals the posterior denoiser $\bar{D}_\phi(z_s, s \mid x_t, t)$ by Proposition 1. On the student side the fake loss equation 13 is a squared error with target $\hat{z} - \epsilon$, so its minimizer is $\bar{v}_{\psi^\star}(z, s \mid x_t, t) = \mathbb{E}[\hat{z} - \epsilon \mid \hat{z}_s = z]$. Since $\hat{z}_s + (1-s)(\hat{z} - \epsilon) = \hat{z}$ by equation 23, the optimal fake denoiser $\bar{D}_{\psi^\star}(z, s \mid x_t, t) = z + (1-s)\,\bar{v}_{\psi^\star}(z, s \mid x_t, t)$ equals $\mathbb{E}[\hat{z} \mid \hat{z}_s = z]$. For an optimal fake network the score difference thus reduces to a denoiser difference

$$\nabla_z \log p_s^\theta(z) - \nabla_z \log p_s(z) = \frac{s}{(1-s)^2}\Big(\bar{D}_\psi(z, s \mid x_t, t) - \bar{D}_\phi(z, s \mid x_t, t)\Big). \tag{24}$$

Gradient of the KL divergence. We write the student output as $\hat{z} = \bar{G}_\theta(\xi)$, where ξ collects the student inputs and does not depend on θ. Then $D_{\mathrm{KL}}(p_s^\theta \| p_s) = \mathbb{E}_{\xi,\epsilon}\big[\log p_s^\theta(\hat{z}_s) - \log p_s(\hat{z}_s)\big]$ and differentiating under the expectation gives two terms. The term from the explicit dependence of the density p_s^θ on θ vanishes because $\mathbb{E}_{z\sim p_s^\theta}\big[\nabla_\theta \log p_s^\theta(z)\big] = \nabla_\theta \int p_s^\theta(z)\,\mathrm{d}z = 0$. The remaining term comes from the dependence of the sample $\hat{z}_s$ on θ. Substituting $\partial \hat{z}_s/\partial\theta = s\,\partial\hat{z}/\partial\theta$ and equation 24 into it gives

$$\nabla_\theta D_{\mathrm{KL}}\big(p_s^\theta \,\|\, p_s\big) = \mathbb{E}_{\hat{z},\epsilon}\Big[\frac{s^2}{(1-s)^2}\Big(\frac{\partial \hat{z}}{\partial \theta}\Big)^\top \Delta\Big], \qquad \Delta = \bar{D}_\psi(\hat{z}_s, s \mid x_t, t) - \bar{D}_\phi(\hat{z}_s, s \mid x_t, t). \tag{25}$$

The scores enter only through their values at $\hat{z}_s$ and are not differentiated with respect to θ, which is what allows the stop-gradient in equation 12.

Table 5: Test-time steering with the gradient-based estimator using $K = 2$ inner steps per posterior sample. MetaDMD is trained with $K = 2$ inner steps, and MFM (Potaptchik et al., 2026b) is also sampled with $K = 2$ inner steps. All results use $J = 28$ outer steps. Bold marks the better of MFM and MetaDMD at each N.

	ImageReward↑		PickScore↑		HPSv2↑		Aesthetic↑	
	MFM	MetaDMD	MFM	MetaDMD	MFM	MetaDMD	MFM	MetaDMD
$N = 8$	1.67	**1.74**	24.22	**24.85**	0.356	**0.366**	**8.01**	7.77
$N = 16$	1.71	**1.77**	24.69	**25.16**	0.368	**0.371**	**8.07**	7.84

Gradient of the MetaDMD loss. The stop-gradient in equation 12 makes the target $\hat{z} - w\Delta$ constant, so the loss has the gradient

$$\nabla_\theta \mathcal{L}_{\text{MetaDMD}}(\theta) = \mathbb{E}_{t,x_t,\hat{z},s,\epsilon}\Big[w\left(\frac{\partial \hat{z}}{\partial \theta}\right)^\top \Delta\Big]. \tag{26}$$

Comparing equation 26 with equation 25 shows that for a weight $w(s)$ depending only on s the MetaDMD loss has the same gradient as the path KL objective

$$\mathbb{E}_{t,x_t}\Big[\mathbb{E}_s\Big[\frac{(1-s)^2}{s^2}\, w(s)\, D_{\text{KL}}\big(p_s^\theta(\cdot \mid x_t) \,\big\|\, p_s(\cdot \mid x_t)\big)\Big]\Big].$$

Since adding Gaussian noise does not lose information about a distribution, this objective is minimized when the student's output distribution matches the teacher's posterior. In practice w is the per-sample normalization of DMD2 Yin et al. (2024a), which rescales the contribution of each sample by a positive factor. The update therefore remains a descent direction for the path KL objective with the same stationary point $\Delta = 0$.

Stagewise matching. The student output $\hat{z} = \bar{G}_\theta(z_{s_k}, s_k \mid x_t, t)$ is taken at a random stage k of the rollout equation 11 whose input z_{s_k} is computed without gradients. The argument above then applies with $\xi = z_{s_k}$, so every stage is trained to output posterior samples and the final state z_{s_K} follows $p_{1|t}^{\text{model}}(\cdot \mid x_t)$ at the optimum.

D Implementation Details

The MetaDMD student and the fake network share the same learning rate, which is 10^{-4} on SiT-XL/2 and 10^{-5} on SD3.5-M. Following the two time-scale update rule (TTUR) of DMD2 (Yin et al., 2024a), we update the student once every $M = 5$ fake network updates on both models.

E Additional Experiments

Steering with Fewer Inner Steps. To further reduce the cost of posterior sampling, we train MetaDMD with $K = 2$ inner steps and sample MFM with $K = 2$ as well, which halves the cost of each posterior sample. All other settings follow the ImageNet experiments in Sec. 4.3. Tab. 5 shows that MetaDMD keeps its advantage on ImageReward, PickScore and HPSv2 at both $N = 8$ and $N = 16$, while MFM remains higher on the aesthetic score. Compared with $K = 4$ in Tab. 3, halving the inner steps degrades MetaDMD only marginally. The largest decrease is 0.17 in PickScore at $N = 8$, and ImageReward at $N = 16$ even increases from 1.71 to 1.77.

Table 6: Test-time steering on class-conditional ImageNet 256×256 with SiT-XL/2 using the gradient-based estimator in Eq. (16) (top) and the gradient-free estimator in Eq. (15) (bottom) for different numbers of MC samples N. All results use $J = 28$ outer steps and $K = 4$ inner steps per posterior sample. The Teacher row reports the teacher without steering. Bold marks the best result for each reward within each estimator, compared across MFM (Potaptchik et al., 2026b) and MetaDMD and all N.

	ImageReward↑		PickScore↑		HPSv2↑		Aesthetic↑	
	MFM	**MetaDMD**	MFM	**MetaDMD**	MFM	**MetaDMD**	MFM	**MetaDMD**
Teacher	0.19		20.80		0.240		4.72	
Gradient-based estimator								
$N = 1$	1.49	1.62	22.48	23.99	0.320	0.345	7.17	7.40
$N = 4$	1.63	1.71	23.83	24.76	0.354	0.363	7.68	7.73
$N = 8$	1.61	**1.74**	24.23	25.02	0.360	0.368	7.84	7.81
$N = 16$	1.68	1.71	24.60	**25.32**	0.371	**0.377**	**8.02**	7.92
Gradient-free estimator								
$N = 1$	0.47	0.77	19.86	20.57	0.246	0.257	5.38	5.43
$N = 4$	0.85	1.00	20.50	20.98	0.268	0.272	5.69	5.64
$N = 8$	1.17	1.16	21.01	21.46	0.289	0.281	6.03	5.89
$N = 16$	**1.41**	1.30	21.61	**21.83**	**0.306**	0.287	**6.38**	6.15