PROBABILITY-FLOW DISTILLATION: DISTRIBUTION MATCHING IN PARAMETER SPACE

Anonymous authors
Paper under double-blind review

ABSTRACT

Score distillation methods use pretrained diffusion models as priors for optimizing parameters that are mapped to data by a differentiable forward model, most notably in text-to-3D generation. Yet the distribution they induce over those parameters is not well understood. Observing that existing distillation methods reduce to one of three—Score Distillation Sampling (SDS), Score Distillation via Inversion (SDI), and Variational Score Distillation (VSD)—we extend the particle variational inference view of VSD to the other two. We show that SDS collapses onto the modes of the target, while SDI converges to a contracted version of it, and explain why SDI needs a negative classifier-free guidance scale. Next, we observe that Tweedie's posterior mean equals a single Euler step of the probability-flow ODE (PF-ODE). Replacing this step in SDI with a full reverse solve makes the target a fixed point, but it requires solving two concatenated PF-ODEs. Dropping a Jacobian from the resulting gradient gives Probability-Flow Distillation (PFD), which requires solving only the forward PF-ODE. Experiments on synthetic targets, the CelebA dataset, and text-to-3D generation support our analysis and demonstrate the practical effectiveness of PFD.

1 INTRODUCTION

Diffusion and related generative models (Song & Ermon, 2019; Ho et al., 2020; Rombach et al., 2021; Saharia et al., 2022; Lipman et al., 2023) have demonstrated remarkable success in sampling from complex data distributions. However, in some applications the quantity of interest is not the data itself but the parameters that generate the data through a differentiable system, $\boldsymbol{x}_0 = \boldsymbol{g}(\boldsymbol{\theta}, \boldsymbol{\omega})$, where $\boldsymbol{\theta} \in \Theta$, $\boldsymbol{x}_0 \in \mathcal{X}$, $\boldsymbol{g} : \Theta \times \Omega \to \mathcal{X}$ is differentiable in $\boldsymbol{\theta}$, and $\boldsymbol{\omega} \sim \xi$ is the randomness of the forward model itself. The goal is to find a distribution π on Θ such that the pushforward of the independent coupling $\pi \otimes \xi$ under $\boldsymbol{g}$ is as close as possible to the data distribution, i.e.,

$$\pi^\star \in \arg\min_\pi \; D_{\mathrm{KL}}\big(\boldsymbol{g}_\#(\pi \otimes \xi) \,\big\|\, p_0\big), \tag{1}$$

so that $\boldsymbol{g}(\boldsymbol{\theta}, \boldsymbol{\omega})$ is approximately distributed as p_0. This objective has motivated a family of distillation[1] algorithms that use a pretrained diffusion model to sample in the parameter space Θ.

Poole et al. (2022) introduced this idea with Score Distillation Sampling (SDS) for text-to-3D generation, where $\boldsymbol{\theta}$ denotes the parameters of a 3D representation (Mildenhall et al., 2020; Müller et al., 2022; Shen et al., 2021; Kerbl et al., 2023), $\boldsymbol{g}$ a differentiable renderer, and $\boldsymbol{\omega}$ the camera pose it renders from. The motivation was the scarcity of high-quality 3D data: existing 3D datasets consist largely of synthetic assets and capture far less of the diversity and detail of real-world appearances than large-scale image datasets. Therefore, distilling 2D diffusion priors for 3D generation remains an active research direction. These methods, however, address a broader class of inverse and parameter-optimization problems. For example, Richter-Powell et al. (2025) use a pretrained audio diffusion model to optimize the parameters of an impact-synthesizer and also to separate sound sources from a text prompt. On the other hand, in Distribution Matching Distillation (Yin et al., 2024), $\boldsymbol{\theta}$ holds the weights of a one-step generator and $\boldsymbol{\omega} \sim \xi = \mathcal{N}(\mathbf{0}, \boldsymbol{I})$ is its input; writing $\boldsymbol{g}(\boldsymbol{\theta}, \boldsymbol{\omega}) = G_{\boldsymbol{\theta}}(\boldsymbol{\omega})$, the target is $\pi = \delta_{\boldsymbol{\theta}^\star}$ such that $(G_{\boldsymbol{\theta}^\star})_\# \xi = p_0$.

We observe that subsequent text-to-3D distillation methods (see Appendix G for a review and grouping) are instances—either directly or in approximation—of three representative methods: (i) SDS,

[1] We use the term in a broader sense than its common use—methods that accelerate sampling in $\mathcal{X}$.

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.

(ii) Score Distillation via Inversion (SDI) (Lukoianov et al., 2024), and (iii) Variational Score Distillation (VSD) (Wang et al., 2023). We therefore restrict our scope to these methods, which can all be written abstractly as

$$\frac{\mathrm{d}\boldsymbol{\theta}}{\mathrm{d}\tau} = -\mathbb{E}_{t,\boldsymbol{\omega},\boldsymbol{\epsilon}}\left[\left(\frac{\partial \boldsymbol{x}_0}{\partial \boldsymbol{\theta}}\right)^{\top} \boldsymbol{\Delta}(\boldsymbol{x}_0, t, \boldsymbol{\epsilon})\right], \tag{2}$$

where $\boldsymbol{\Delta}$ is the gradient estimator used by the method and $\boldsymbol{\epsilon}$ is the Gaussian noise it draws. If the gradient estimator is deterministic given $\boldsymbol{x}_0$, we drop the $\boldsymbol{\epsilon}$ argument and write $\boldsymbol{\Delta}(\boldsymbol{x}_0, t)$. The Jacobian $\partial \boldsymbol{x}_0/\partial \boldsymbol{\theta}$, taken at the sampled $\boldsymbol{\omega}$, transfers the update from the data space to the parameter space. Here τ is the optimization time of the distillation algorithm, distinct from the diffusion time t; at each step, a new $t \sim \mathcal{U}[0,1]$ is sampled while τ continues to evolve. Thus, the optimization is not tied to the finite horizon of the diffusion model's schedule. We view equation 2 as dynamics on a set of particles $\{\boldsymbol{\theta}_i\}_i$ and write $q_{0,\tau} = \boldsymbol{g}_{\#}(\pi_\tau \otimes \xi)$ for the induced distribution of $\boldsymbol{x}_0$ at time τ. Only VSD was explicitly derived in this particle variational inference (ParVI) formulation, with $q_{0,\tau} \to p_0$ as its goal. Its gradient estimator is given by:

$$\boldsymbol{\Delta}(\boldsymbol{x}_0, t, \boldsymbol{\epsilon}) = w(t)\left(\nabla_{\boldsymbol{x}} \log q_{t,\tau}(\boldsymbol{x}_t) - \nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}_t)\right), \qquad \boldsymbol{x}_t = \boldsymbol{x}_0 + \sigma_t \boldsymbol{\epsilon}, \tag{3}$$

under which the parameter distribution π_τ follows the Wasserstein gradient flow of the functional $\mathcal{F}[\boldsymbol{g}_{\#}(\pi \otimes \xi)]$, where $\mathcal{F}[q] = \mathbb{E}_t[w(t)\, D_{\mathrm{KL}}(q_t[q] \,\|\, p_t)]$. Here $q_t = q * \mathcal{N}(\mathbf{0}, \sigma_t^2 \boldsymbol{I}_d)$, and the same relates p_t to p_0. The other two are not derived in this way, so the question of which distribution results does not arise in their derivation.

In this work, we extend the study to SDS and SDI and characterize the distribution to which $q_{0,\tau}$ converges under each. In particular, we show that under SDS it collapses onto the modes of p_0, while under SDI it reaches a contracted version of p_0. This explains the qualitative differences between the 3D assets these methods produce. The ParVI view of SDI also explains the need for the negative classifier-free guidance (CFG) scale that the method requires. We then make an observation that Tweedie's posterior mean is a single Euler step of the reverse probability-flow ODE (PF-ODE), which suggests a distribution-matching extension of SDI. However, this requires solving two concatenated ODEs. We address this by introducing Probability-Flow Distillation (PFD), a simpler variant of the method which, surprisingly, converges faster than the extension itself. In our experiments, it is also on par with or better than VSD. We evaluate PFD against SDS, SDI, and VSD in settings of increasing complexity in both p_0 and $\boldsymbol{g}$. We begin with sampling from three synthetic targets: a one-dimensional Gaussian, a two-dimensional funnel, and a twenty-dimensional mixture of Gaussians. We then use a diffusion model trained on the CelebA dataset for four tasks: image generation, symmetric image generation, inpainting, and vector graphics generation. Finally, we turn to text-to-3D generation. Across these settings, the results agree with the analysis and show that PFD performs well in practice.

2 Preliminaries

2.1 Probability-Flow ODE

Consider the Itô SDE, $\mathrm{d}\boldsymbol{x}_t = \boldsymbol{f}(\boldsymbol{x}_t, t)\,\mathrm{d}t + g(t)\,\mathrm{d}\boldsymbol{w}_t$, where $t \in [0,1]$, $\boldsymbol{f}$ is the drift, g is the diffusion coefficient, and $\boldsymbol{w}_t$ is standard Brownian motion. With $p_t(\boldsymbol{x}_t)$ denoting the marginal density of $\boldsymbol{x}_t \in \mathbb{R}^d$, Anderson (1982) derived the corresponding reverse-time SDE:

$$\mathrm{d}\boldsymbol{x}_t = \left[\boldsymbol{f}(\boldsymbol{x}_t, t) - g(t)^2 \nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}_t)\right]\mathrm{d}t + g(t)\,\mathrm{d}\bar{\boldsymbol{w}}_t, \tag{4}$$

where $\bar{\boldsymbol{w}}_t$ is the reverse-time Brownian motion. The quantity $\nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}_t)$, known as the score function, is generally intractable but can be approximated via denoising score matching (DSM) (Vincent, 2011) by minimizing the following objective:

$$\mathcal{L}_{\mathrm{DSM}}(\boldsymbol{\phi}) = \mathbb{E}_{\boldsymbol{x}_0 \sim p_0, \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}_d), t}\left[w(t)\,\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t)\|^2\right], \quad \boldsymbol{x}_t \sim p(\cdot \mid \boldsymbol{x}_0), \quad w(t) > 0. \tag{5}$$

The score function is then recovered up to a time-dependent scaling factor, i.e., $\nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}_t) \propto -\boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t)$ (assuming $\boldsymbol{\epsilon}_{\boldsymbol{\phi}}$ is the exact minimizer of equation 5). Associated with the forward SDE is an ODE that preserves the same marginal densities $p_t(\boldsymbol{x}_t)$ for all t, known as the probability-flow ODE (Song et al., 2021b):

$$\frac{\mathrm{d}\boldsymbol{x}_t}{\mathrm{d}t} = \boldsymbol{f}(\boldsymbol{x}_t, t) - \frac{1}{2} g(t)^2 \nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}_t). \tag{6}$$

2.2 DIFFUSION MODELS

Diffusion models can be written as SDEs of the form given in Section 2.1. Two choices of drift and diffusion coefficients are standard, both drifts being linear in $\boldsymbol{x}_t$, so that the perturbation kernel $p(\boldsymbol{x}_t \mid \boldsymbol{x}_0)$ is Gaussian and available in closed form. This is what makes training tractable: a sample at any time t can be drawn directly from $\boldsymbol{x}_0$ without simulating the SDE. For the Variance-Exploding (VE) SDE, $\mathrm{d}\boldsymbol{x}_t = \sqrt{\mathrm{d}\sigma_t^2/\mathrm{d}t}\,\mathrm{d}\boldsymbol{w}_t$, the perturbation kernel admits the reparameterization $\boldsymbol{x}_t = \boldsymbol{x}_0 + \sigma_t\boldsymbol{\epsilon}$, and the score is recovered from the trained network as $\nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}_t) = -\boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t)/\sigma_t$. For the Variance-Preserving (VP) SDE, $\mathrm{d}\boldsymbol{x}_t = -\frac{1}{2}\beta_t\boldsymbol{x}_t\,\mathrm{d}t + \sqrt{\beta_t}\,\mathrm{d}\boldsymbol{w}_t$, where $\beta_t \geq 0$ is the noise schedule, the corresponding reparameterization is $\boldsymbol{x}_t = \sqrt{\alpha_t}\,\boldsymbol{x}_0 + \sqrt{1-\alpha_t}\,\boldsymbol{\epsilon}$ with $\alpha_t = \exp(-\int_0^t \beta_s\,\mathrm{d}s)$, and $\nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}_t) = -\boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t)/\sqrt{1-\alpha_t}$. For the VE-SDE, the PF-ODE is given by $\mathrm{d}\boldsymbol{x}_t/\mathrm{d}t = \boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t)\,\mathrm{d}\sigma_t/\mathrm{d}t$. A change of variables brings the VP case to the same form: with $\tilde{\sigma}_t \coloneqq \sqrt{(1-\alpha_t)/\alpha_t}$ and $\tilde{\boldsymbol{x}}_t \coloneqq \boldsymbol{x}_t/\sqrt{\alpha_t}$, the VP reparameterization becomes $\tilde{\boldsymbol{x}}_t = \boldsymbol{x}_0 + \tilde{\sigma}_t\boldsymbol{\epsilon}$, with the PF-ODE given by

$$\frac{\mathrm{d}\tilde{\boldsymbol{x}}_t}{\mathrm{d}t} = \boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t)\,\frac{\mathrm{d}\tilde{\sigma}_t}{\mathrm{d}t}, \qquad \boldsymbol{x}_t = \frac{\tilde{\boldsymbol{x}}_t}{\sqrt{1+\tilde{\sigma}_t^2}}, \tag{7}$$

which is exactly the DDIM-ODE (Song et al., 2021a). We therefore work in the VE parametrization throughout. Given the deterministic nature of the DDIM-ODE, it can be integrated from $0 \to t$, mapping $\boldsymbol{x}_0$ uniquely to a noisy state $\boldsymbol{x}_t$. This procedure is referred to as DDIM inversion.

2.3 CLASSIFIER-FREE GUIDANCE

Classifier-free guidance (Ho & Salimans, 2021) is the de facto technique for controlling the conditioning strength. During training, the conditioning signal c is randomly dropped so that the network learns both a conditional predictor $\boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t, c)$ and an unconditional predictor $\boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t, \varnothing)$. At inference, the two are linearly extrapolated to form the guided noise estimate,

$$\hat{\boldsymbol{\epsilon}}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t, c) = \boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t, \varnothing) + \gamma\big(\boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t, c) - \boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t, \varnothing)\big). \tag{8}$$

Since each predictor approximates the score of its respective marginal, i.e., $\boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t, \cdot) = -\sigma_t\nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}_t \mid \cdot)$, we have $\hat{\boldsymbol{\epsilon}}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t, c) = -\sigma_t\nabla_{\boldsymbol{x}} \log \tilde{p}_t(\boldsymbol{x}_t \mid c)$, where

$$\tilde{p}_t(\boldsymbol{x}_t \mid c) \;\propto\; p_t(\boldsymbol{x}_t \mid c)\left(\frac{p_t(\boldsymbol{x}_t \mid c)}{p_t(\boldsymbol{x}_t \mid \varnothing)}\right)^{\gamma-1}. \tag{9}$$

Therefore, CFG replaces the conditional distribution $p_t(\boldsymbol{x}_t \mid c)$ with a tilted version of it, reweighted by the likelihood ratio between the conditional and the unconditional marginals; sampling is driven away from $p_t(\cdot \mid \varnothing)$ and towards $p_t(\cdot \mid c)$. Setting $\gamma = 1$ recovers the conditional model, while $\gamma > 1$ amplifies alignment with c, typically at the expense of diversity.

3 ANALYSIS OF SDS AND SDI

3.1 SDS COLLAPSES ONTO THE MODES

SDS starts from the idea that $\boldsymbol{x}_0 = \boldsymbol{g}(\boldsymbol{\theta}, \boldsymbol{\omega})$ should be likely under the pretrained model for every draw of $\boldsymbol{\omega}$. Since the likelihood is intractable, the training loss in equation 5 is used as a surrogate, and the parameters are obtained as $\boldsymbol{\theta}^\star = \arg\min_{\boldsymbol{\theta}} \mathbb{E}_{\boldsymbol{\omega}}\big[\mathcal{L}_{\mathrm{DSM}}(\boldsymbol{\phi}, \boldsymbol{x}_0 = \boldsymbol{g}(\boldsymbol{\theta}, \boldsymbol{\omega}))\big]$, now minimized over $\boldsymbol{\theta}$ with $\boldsymbol{\phi}$ held fixed. Differentiating with respect to $\boldsymbol{\theta}$ gives

$$\nabla_{\boldsymbol{\theta}}\mathcal{L}_{\mathrm{DSM}} \;\propto\; \mathbb{E}_{t,\boldsymbol{\omega},\boldsymbol{\epsilon}}\left[w(t)\left(\frac{\partial \boldsymbol{x}_0}{\partial \boldsymbol{\theta}}\right)^{\top} \underbrace{\cancel{\left(\frac{\partial \boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t)}{\partial \boldsymbol{x}_t}\right)^{\top}}}_{\text{omitted in SDS}} (\boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t) - \boldsymbol{\epsilon})\right]. \tag{10}$$

The middle factor is the Jacobian of the score network, a scaled Hessian of the log-density, which is expensive to evaluate and ill-conditioned at small noise levels. SDS omits it and uses $\boldsymbol{\Delta} =$

$w(t)\big(\boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t) - \boldsymbol{\epsilon}\big)$ as the gradient estimator. Taking $w(t) = \sigma_t$, it can be written as $\boldsymbol{\Delta} = \boldsymbol{x}_0 - \hat{\boldsymbol{x}}_0(\boldsymbol{x}_t)$, where

$$\hat{\boldsymbol{x}}_0(\boldsymbol{x}_t) = \boldsymbol{x}_t - \sigma_t \boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t) = \boldsymbol{x}_t + \sigma_t^2 \nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}_t) = \mathbb{E}_p[\boldsymbol{x}_0 \,|\, \boldsymbol{x}_t] \tag{11}$$

is the posterior mean of $\boldsymbol{x}_0$ given the noisy observation $\boldsymbol{x}_t$, under the forward process of p_0. Unless stated otherwise, in this section and the next we take $\boldsymbol{g}$ to be the identity, $\boldsymbol{x}_0 = \boldsymbol{g}(\boldsymbol{\theta}, \boldsymbol{\omega}) = \boldsymbol{\theta}$, so that the induced distribution coincides with the particle distribution, $q_{0,\tau} = \pi_\tau$, and we refer to either as the particle distribution. This simplifies the exposition and the notation. We return to the general case in the experiments (Section 5).

The target is not a fixed point of SDS. To see this, suppose $q_{0,\tau} = p_0$ and assume p_0 is non-degenerate with finite second moments, i.e., $\mathrm{Cov}(\boldsymbol{x}_0) \succ 0$. The law of total covariance then gives

$$\mathrm{Cov}(\hat{\boldsymbol{x}}_0) = \mathrm{Cov}(\boldsymbol{x}_0) - \mathbb{E}[\mathrm{Cov}(\boldsymbol{x}_0 \,|\, \boldsymbol{x}_t)]\,. \tag{12}$$

The subtracted term is positive definite for $\sigma_t > 0$. If $\boldsymbol{v}^\top \mathbb{E}[\mathrm{Cov}(\boldsymbol{x}_0 \,|\, \boldsymbol{x}_t)]\boldsymbol{v} = 0$ for some $\boldsymbol{v} \neq \boldsymbol{0}$, then $\mathrm{Var}(\boldsymbol{v}^\top \boldsymbol{x}_0 \,|\, \boldsymbol{x}_t) = 0$ almost surely, so $\boldsymbol{v}^\top \boldsymbol{x}_0 = h(\boldsymbol{x}_t)$ for some measurable h, i.e., $\boldsymbol{v}^\top \boldsymbol{x}_0$ is completely determined by the noisy observation $\boldsymbol{x}_t$. Since $\boldsymbol{x}_t$ has full support given $\boldsymbol{x}_0$, h must be constant, so $\mathrm{Var}(\boldsymbol{v}^\top \boldsymbol{x}_0) = 0$, contradicting $\mathrm{Cov}(\boldsymbol{x}_0) \succ 0$. Hence $\mathrm{Cov}(\hat{\boldsymbol{x}}_0) \prec \mathrm{Cov}(\boldsymbol{x}_0)$, while $\mathbb{E}[\hat{\boldsymbol{x}}_0] = \mathbb{E}[\boldsymbol{x}_0]$ by the tower property. At $q_{0,\tau} = p_0$, SDS therefore moves the particles $\{\boldsymbol{x}_{0,i}\}_i$ toward a distribution with the same mean but a strictly smaller covariance, so p_0 is not a fixed point.

The particles instead converge onto the modes of p_0, which can be seen by writing out the dynamics. Substituting $\boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t) = -\sigma_t \nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}_t)$ gives $\boldsymbol{\Delta} = -\sigma_t^2 \nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}_t) - \sigma_t \boldsymbol{\epsilon}$. Since $\mathbb{E}_{\boldsymbol{\epsilon}}[\boldsymbol{\epsilon}] = \boldsymbol{0}$, the second term vanishes under expectation, so each particle evolves as

$$\frac{\mathrm{d}\boldsymbol{x}_0}{\mathrm{d}\tau} = -\mathbb{E}_{t,\boldsymbol{\epsilon}}\left[\boldsymbol{\Delta}(\boldsymbol{x}_0, t, \boldsymbol{\epsilon})\right] = \mathbb{E}_{t,\boldsymbol{\epsilon}}\left[\sigma_t^2\, \nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}_0 + \sigma_t \boldsymbol{\epsilon})\right]. \tag{13}$$

No term on the right-hand side accounts for $q_{0,\tau}$, so nothing in the update maintains its entropy. Therefore, each particle ascends $\mathbb{E}_{t,\boldsymbol{\epsilon}}[\sigma_t^2 \log p_t(\boldsymbol{x}_0 + \sigma_t \boldsymbol{\epsilon})]$ without reference to the others, and particles starting in different basins converge to different local maxima. To connect back to the general case, each particle $\boldsymbol{\theta}$ independently converges to a point at which $\boldsymbol{g}(\boldsymbol{\theta}, \boldsymbol{\omega})$ lies near a local maximum of p_0 for ξ-almost every $\boldsymbol{\omega}$, up to the constraints imposed by $\boldsymbol{g}$. The following proposition considers the case of isotropic Gaussians.

Proposition 1. *Let $p_0 = \mathcal{N}(\boldsymbol{\mu}, \rho^2 \boldsymbol{I}_d)$ be the target and take the VE schedule $\sigma_t = t$. Then the SDS update in equation 13 reduces to the following linear ODE:*

$$\frac{\mathrm{d}\boldsymbol{x}_0}{\mathrm{d}\tau} = \mathbb{E}_{t,\boldsymbol{\epsilon}}\left[t^2\, \nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}_0 + t\boldsymbol{\epsilon})\right] = -A(\rho)\,(\boldsymbol{x}_0 - \boldsymbol{\mu}), \tag{14a}$$

$$A(\rho) \coloneqq \mathbb{E}_t\left[\frac{t^2}{\rho^2 + t^2}\right] = 1 - \rho \arctan\frac{1}{\rho} \;>\; 0 \quad \forall \rho. \tag{14b}$$

Since equation 14a is affine in $\boldsymbol{x}_0$, an isotropic Gaussian initialization $q_{0,0} = \mathcal{N}(\boldsymbol{m}_0, \nu_0^2 \boldsymbol{I}_d)$ remains in that family for all τ. Writing $q_{0,\tau} = \mathcal{N}(\boldsymbol{m}_\tau, \nu_\tau^2 \boldsymbol{I}_d)$, we get $\boldsymbol{m}_\tau = \boldsymbol{\mu} + e^{-A(\rho)\tau}(\boldsymbol{m}_0 - \boldsymbol{\mu})$ and $\nu_\tau^2 = \nu_0^2\, e^{-2A(\rho)\tau}$. Thus, $q_{0,\tau}$ converges to the degenerate Gaussian $\delta_{\boldsymbol{\mu}}$.

3.2 SDI CONVERGES TO A CONTRACTED TARGET

SDI builds on the observation that SDS resembles a high-variance version of DDIM and replaces the randomly sampled noise $\boldsymbol{\epsilon}$ with the noise implied by DDIM inversion. The gradient estimator retains the form $\boldsymbol{\Delta}(\boldsymbol{x}_0, t) = \boldsymbol{x}_0 - \hat{\boldsymbol{x}}_0(\boldsymbol{x}_t)$, with $\boldsymbol{x}_t$ now obtained deterministically by DDIM inversion.

DDIM inversion as approximate forward PF-ODE. Instead of sampling $\boldsymbol{x}_t \sim \mathcal{N}(\boldsymbol{x}_0, \sigma_t^2 \boldsymbol{I}_d)$, we can map $\boldsymbol{x}_0$ to $\boldsymbol{x}_t$ through the PF-ODE[2] induced by $\{q_{t,\tau}\}_t$, solved in the $0 \to t$ direction—writing $\boldsymbol{\Phi}_{0\to t}^{q_\tau}$ for its flow map so that $\boldsymbol{x}_t = \boldsymbol{\Phi}_{0\to t}^{q_\tau}(\boldsymbol{x}_0)$. This state has the same marginal distribution as before, but the map is now driven by $\nabla_{\boldsymbol{x}} \log q_{t,\tau}$ and thus explicitly depends on $q_{0,\tau}$. DDIM inversion can be thought of as an approximation of $\boldsymbol{\Phi}_{0\to t}^{q_\tau}$, integrating the unconditional score of the pretrained model $\nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}_t \,|\, \varnothing)$ rather than the score of the particle distribution $\nabla_{\boldsymbol{x}} \log q_{t,\tau}(\boldsymbol{x}_t)$. It can be argued that this substitution is reasonable in the text-to-3D setting for which it was designed, as the renderings $\boldsymbol{g}(\boldsymbol{\theta}, \boldsymbol{\omega})$ of a 3D asset remain close to the manifold of natural images throughout optimization, so that $\nabla_{\boldsymbol{x}} \log q_{t,\tau}(\boldsymbol{x}_t) \approx \nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}_t \,|\, \varnothing)$.

[2]We call the PF-ODEs induced by $\{q_{t,\tau}\}_t$ and $\{p_t\}_t$ the forward and reverse PF-ODEs, respectively.

Explanation of negative CFG scale in SDI. The approximation above describes the inversion without guidance. However, in practice, the inversion is run with CFG, and we now ask what the approximation implies in that case. In image editing, DDIM inversion with a positive CFG scale under a source prompt describing the input maps the image to a latent, and DDIM sampling, again with a positive scale but under the target prompt, gives the edited image. In the text-to-3D setting, no prompt describes the current renderings, so SDI inverts under the target prompt itself. With $\nabla_{\boldsymbol{x}} \log p_t(\cdot \,|\, \varnothing) \approx \nabla_{\boldsymbol{x}} \log q_{t,\tau}$ as above and $\nabla_{\boldsymbol{x}} \log p_t(\cdot \,|\, c) = \nabla_{\boldsymbol{x}} \log p_t$, the guided score of equation 9 becomes

$$\nabla_{\boldsymbol{x}} \log \tilde{p}_t(\boldsymbol{x}_t \,|\, c) \;\approx\; \nabla_{\boldsymbol{x}} \log \left[p_t(\boldsymbol{x}_t) \left(\frac{p_t(\boldsymbol{x}_t)}{q_{t,\tau}(\boldsymbol{x}_t)} \right)^{\gamma-1} \right], \tag{15}$$

so CFG drives the reverse PF-ODE away from the source $q_{t,\tau}$ and towards the target p_t. The inversion, however, integrates the forward PF-ODE, so its guidance must tilt the other way, exchanging p_t and $q_{t,\tau}$ in equation 15. We therefore need $\boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t, c) + \gamma(\boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t, \varnothing) - \boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t, c))$ in place of the standard $\boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t, \varnothing) + \gamma(\boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t, c) - \boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t, \varnothing))$ of equation 8. Rewriting it in the form the implementation expects gives

$$\begin{aligned} &\boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t, c) + \gamma\big(\boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t, \varnothing) - \boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t, c)\big) \\ &\quad = \boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t, \varnothing) + (1-\gamma)\big(\boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t, c) - \boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t, \varnothing)\big), \end{aligned} \tag{16}$$

so realizing an effective guidance scale γ in inversion corresponds to a user-specified CFG scale $\gamma_{\text{cfg}} = 1-\gamma$. For $\gamma > 1$, this gives $\gamma_{\text{cfg}} < 0$, explaining the negative CFG scale needed in SDI. Note that $\gamma_{\text{cfg}} = 0$ recovers the unguided inversion of the previous paragraph, the approximation of $\boldsymbol{\Phi}_{0\to t}^{q_\tau}$ with no CFG. Inversion under the target prompt with a positive scale would tilt it the wrong way and Appendix F shows that this makes editing unstable, while the negative scale restores stability.

Under the ideal version of SDI—in which $\boldsymbol{x}_t$ is obtained through the forward PF-ODE rather than its DDIM inversion approximation—the particle dynamics are given by

$$\frac{\mathrm{d}\boldsymbol{x}_0}{\mathrm{d}\tau} = -\mathbb{E}_t\left[\boldsymbol{\Delta}(\boldsymbol{x}_0, t)\right] = \mathbb{E}_t[\hat{\boldsymbol{x}}_0(\boldsymbol{\Phi}_{0\to t}^{q_\tau}(\boldsymbol{x}_0))] - \boldsymbol{x}_0. \tag{17}$$

We use this version in all experiments except text-to-3D, where we keep the DDIM inversion of the original SDI to stay faithful to it and to confirm that the analysis carries over to the approximation. Unlike SDS, the right-hand side of equation 17 depends on the current particle distribution $q_{0,\tau}$, and this dependence acts as an entropy term that resists collapse: the forward PF-ODE has to carry $q_{0,\tau}$ to $q_{t,\tau}$, so the variance it must add is fixed by the schedule, while the factor it applies is relative to the current spread; the more concentrated the particles are, the more strongly this feedback pushes them apart. However, the target is still not a fixed point since the posterior mean contracts the spread here as well. The limiting distribution therefore does not collapse, but its spread is strictly smaller than that of p_0. The following proposition makes this explicit for isotropic Gaussian p_0.

Proposition 2. *With $p_0 = \mathcal{N}(\boldsymbol{\mu}, \rho^2 \boldsymbol{I}_d)$ and the VE schedule $\sigma_t = t$, both $\boldsymbol{\Phi}_{0\to t}^{q_\tau}$ and $\hat{\boldsymbol{x}}_0$ are affine, so an isotropic Gaussian initialization remains in that family for all τ. Under equation 17, the mean and the standard deviation follow*

$$\frac{\mathrm{d}\boldsymbol{m}_\tau}{\mathrm{d}\tau} = -A(\rho)\,(\boldsymbol{m}_\tau - \boldsymbol{\mu}), \tag{18a}$$

$$\frac{\mathrm{d}\nu_\tau}{\mathrm{d}\tau} = (B(\nu_\tau) - 1)\,\nu_\tau, \qquad B(\nu) \coloneqq \frac{\rho^2}{\nu}\,\mathbb{E}_t\left[\frac{\sqrt{\nu^2 + t^2}}{\rho^2 + t^2}\right], \tag{18b}$$

with $A(\rho)$ as in Proposition 1. Equation 18b has a unique stationary point $\nu_\star$, given by $B(\nu_\star) = 1$, which is globally attracting and satisfies $\nu_\star < \rho$. The particle distribution therefore converges to $\mathcal{N}(\boldsymbol{\mu}, \nu_\star^2 \boldsymbol{I}_d)$, which shares the mean of p_0 but has a strictly smaller variance.

3.3 Posterior mean as a single Euler step

Observation 3. *In terms of the noise level σ, the PF-ODE takes the form $\mathrm{d}\boldsymbol{x}/\mathrm{d}\sigma = \boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}, t(\sigma))$. A single explicit Euler step of this ODE from σ_t to 0 gives $\boldsymbol{x}_t - \sigma_t \boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t) = \hat{\boldsymbol{x}}_0(\boldsymbol{x}_t)$. The posterior mean is therefore exactly one Euler step of the reverse PF-ODE, from noise to data.*

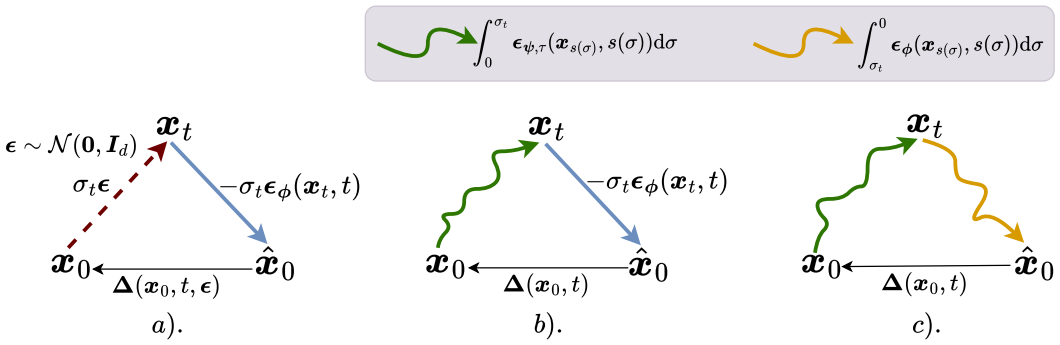


Figure 1: (a) SDS: $\boldsymbol{x}_t$ is obtained by Gaussian perturbation of $\boldsymbol{x}_0$, and $\hat{\boldsymbol{x}}_0$ by the posterior mean. (b) SDI: $\boldsymbol{x}_t$ is obtained by solving the forward PF-ODE, and $\hat{\boldsymbol{x}}_0$ still by the posterior mean. (c) Both are obtained by integrating the PF-ODE, removing the posterior-mean approximation. The green and orange curved arrows denote integration of the forward and reverse PF-ODEs, driven by $\boldsymbol{\epsilon}_{\psi,\tau} = -\sigma_t \nabla_{\boldsymbol{x}} \log q_{t,\tau}$ and $\boldsymbol{\epsilon}_{\boldsymbol{\phi}} = -\sigma_t \nabla_{\boldsymbol{x}} \log p_t$ respectively.

The observation above suggests a dual-bridge extension to SDI, illustrated in Figure 1, in which we solve the reverse PF-ODE rather than taking the posterior mean. Denoting its flow map as $\boldsymbol{\Phi}^p_{t\to 0}$, the new gradient estimator becomes

$$\boldsymbol{\Delta}(\boldsymbol{x}_0, t) = \boldsymbol{x}_0 - \boldsymbol{\Phi}^p_{t\to 0}(\boldsymbol{\Phi}^{q_\tau}_{0\to t}(\boldsymbol{x}_0))\,, \tag{19}$$

for which the target is now a fixed point, since when $q_{0,\tau} = p_0$, $\boldsymbol{\Phi}^p_{t\to 0}$ inverts $\boldsymbol{\Phi}^{q_\tau}_{0\to t}$, and $\boldsymbol{\Delta}$ vanishes for every particle.

Theorem 4. *For the estimator in equation 19, the following holds:*

$$\boldsymbol{\Delta}(\boldsymbol{x}_0, t) = \int_0^t \frac{1}{2} g_s^2 \, \frac{\partial \boldsymbol{\Phi}^p_{s\to 0}}{\partial \boldsymbol{x}}(\boldsymbol{x}_s) \left(\nabla_{\boldsymbol{x}} \log q_{s,\tau}(\boldsymbol{x}_s) - \nabla_{\boldsymbol{x}} \log p_s(\boldsymbol{x}_s)\right) \mathrm{d}s, \tag{20a}$$

$$\mathbb{E}_t[\boldsymbol{\Delta}(\boldsymbol{x}_0, t)] = \int_0^1 \frac{1}{2}(1-s)\, g_s^2 \, \frac{\partial \boldsymbol{\Phi}^p_{s\to 0}}{\partial \boldsymbol{x}}(\boldsymbol{x}_s) \left(\nabla_{\boldsymbol{x}} \log q_{s,\tau}(\boldsymbol{x}_s) - \nabla_{\boldsymbol{x}} \log p_s(\boldsymbol{x}_s)\right) \mathrm{d}s, \tag{20b}$$

where $\boldsymbol{x}_s = \boldsymbol{\Phi}^{q_\tau}_{0\to s}(\boldsymbol{x}_0)$, g_s is the diffusion coefficient of Section 2.1, and $\partial \boldsymbol{\Phi}^p_{s\to 0}/\partial \boldsymbol{x}$ is the Jacobian of the reverse PF-ODE started at time s.

Taking $t = 1$ gives $\boldsymbol{\Delta}(\boldsymbol{x}_0, 1)$, which is exactly the Dual Diffusion Implicit Bridge (DDIB) (Su et al., 2023) between the particle distribution and the target. The expected gradient in equation 20b computes the same integral but with a different weight, so each update moves the particles an infinitesimal step in a direction correlated with DDIB, explaining the tendency of $q_{0,\tau}$ towards p_0.

4 Probability-Flow Distillation

The dual-bridge method solves two concatenated PF-ODEs and therefore doubles the NFEs per step relative to SDI, which is already expensive compared to the one or two evaluations of SDS and VSD. Looking at equation 20b, were it not for the Jacobian $\partial \boldsymbol{\Phi}^p_{t\to 0}/\partial \boldsymbol{x}$, the gradient could be evaluated from the forward PF-ODE alone. We therefore define Probability-Flow Distillation, under which the particles evolve as:

$$\frac{\mathrm{d}\boldsymbol{x}_0}{\mathrm{d}\tau} = \int_0^1 w(t) \left(\nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}_t) - \nabla_{\boldsymbol{x}} \log q_{t,\tau}(\boldsymbol{x}_t)\right) \mathrm{d}t, \tag{21}$$

where $\boldsymbol{x}_t = \boldsymbol{\Phi}^{q_\tau}_{0\to t}(\boldsymbol{x}_0)$ and $w(t) = \frac{1}{2}(1-t)\, g_t^2$; this choice of $w(t)$ is not restrictive, as the results below hold for any $w(t) > 0$. Like the other methods, PFD could be implemented by sampling t at each step and using the gradient estimator in equation 20a without the Jacobian. However, equation 21 already gives the expression for the exact PFD gradient, so we can evaluate it directly, with the asymptotic cost per step unchanged. This avoids the Monte Carlo variance from sampling t, which can reduce the number of optimization steps needed for convergence in practice.

To keep the comparison fair, we replace the expectation over $t \sim \mathcal{U}[0,1]$ in the other baselines with an integral over $[0,1]$, evaluated on N quadrature nodes $\{t_i\}_{i=1}^N$: SDS integrates $\boldsymbol{x}_0 - \hat{\boldsymbol{x}}_0(\boldsymbol{x}_{t_i})$ with $\boldsymbol{x}_{t_i}$ given by a fresh Gaussian perturbation of $\boldsymbol{x}_0$ at each node, SDI integrates the same but with $\{\boldsymbol{x}_{t_i}\}_i$ obtained from a single solve of the forward PF-ODE, and VSD integrates the score difference in equation 21 with $\boldsymbol{x}_{t_i}$ obtained as in SDS. SDS requires one evaluation of the pretrained model $\boldsymbol{\epsilon}_{\boldsymbol{\phi}}$ per node, while SDI[3], VSD, and PFD require two: one of $\boldsymbol{\epsilon}_{\boldsymbol{\phi}}$ and one of $\boldsymbol{\epsilon}_{\boldsymbol{\psi}}$, an auxiliary model trained online to approximate the score $\nabla_{\boldsymbol{x}} \log q_{s,\tau}$. In all four cases, the NFEs per step grow as $\mathcal{O}(N)$. In contrast, implementing the dual-bridge method in the same way requires solving a reverse PF-ODE at every node t_i, and thus needs $\mathcal{O}(N^2)$ NFEs per step. We therefore include it only in the synthetic experiments, where it serves as a check that removing the posterior-mean approximation does promote distribution matching.

Comparison with VSD. (i) VSD has the same gradient form as PFD, but with an expectation over $\boldsymbol{\epsilon}$. The two differ only in the coupling $(\boldsymbol{x}_0, \boldsymbol{x}_t)$, which is stochastic in the former case and deterministic in the latter. Therefore, a Monte Carlo variance remains in realizing the VSD gradient, whereas the PFD gradient is deterministic, up to $\boldsymbol{\omega}$. (ii) VSD is the Wasserstein gradient of $\mathbb{E}_t[w(t)\, D_{\mathrm{KL}}(q_{t,\tau} \,\|\, p_t)]$ and therefore gives the steepest descent on that functional. The following theorem shows that when one looks instead at $D_{\mathrm{KL}}(q_{0,\tau} \,\|\, p_0)$, PFD decreases it faster than VSD when a small time range $[0,\eta]$ is considered.

Theorem 5. *Under the VE schedule $\sigma_t = t$ restricted to $[0,\eta]$, the particles under VSD and PFD evolve as follows:*

$$\left.\frac{\mathrm{d}\boldsymbol{x}_0}{\mathrm{d}\tau}\right|_{\mathrm{VSD}} = \int_0^\eta w(t)\, \mathbb{E}_{\boldsymbol{\epsilon}}[\nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}_t) - \nabla_{\boldsymbol{x}} \log q_{t,\tau}(\boldsymbol{x}_t)]\, \mathrm{d}t, \qquad \boldsymbol{x}_t = \boldsymbol{x}_0 + t\boldsymbol{\epsilon}, \tag{22a}$$

$$\left.\frac{\mathrm{d}\boldsymbol{x}_0}{\mathrm{d}\tau}\right|_{\mathrm{PFD}} = \int_0^\eta w(t)\, (\nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}_t) - \nabla_{\boldsymbol{x}} \log q_{t,\tau}(\boldsymbol{x}_t))\, \mathrm{d}t, \qquad \boldsymbol{x}_t = \boldsymbol{\Phi}_{0\to t}^{q_\tau}(\boldsymbol{x}_0). \tag{22b}$$

Let $\mathrm{d}D_{\mathrm{KL}}/\mathrm{d}\tau$ denote the rate of change of $D_{\mathrm{KL}}(q_{0,\tau} \,\|\, p_0)$. Then, assuming $q_{0,\tau}$ decays at infinity, an update from a common $q_{0,\tau}$ gives:

$$\left.\frac{\mathrm{d}D_{\mathrm{KL}}}{\mathrm{d}\tau}\right|_{\mathrm{VSD}} - \left.\frac{\mathrm{d}D_{\mathrm{KL}}}{\mathrm{d}\tau}\right|_{\mathrm{PFD}} = C_\eta \left(\mathbb{E}_{\boldsymbol{x}_0 \sim q_{0,\tau}}\left[\left\|\nabla^2 \log p_0(\boldsymbol{x}_0) - \nabla^2 \log q_{0,\tau}(\boldsymbol{x}_0)\right\|_F^2\right] + O(\eta^2)\right). \tag{23}$$

Here $C_\eta = \frac{1}{2}\int_0^\eta w(t)\, t^2\, \mathrm{d}t$, which is strictly positive for any $w(t) > 0$, and the leading coefficient is non-negative. PFD therefore decreases $D_{\mathrm{KL}}(q_{0,\tau} \,\|\, p_0)$ at least as fast as VSD for all sufficiently small η.

Although Theorem 5 concerns small η, PFD remains competitive with VSD for η close to 1. The following proposition establishes this for isotropic Gaussians, where PFD converges faster than both VSD and the dual-bridge (DB) method, and the experiments in the next section confirm it more broadly.

Proposition 6. *Set $\eta = 1$ in equation 22a and equation 22b for VSD and PFD. DB follows the latter with the extra Jacobian factor $\partial \boldsymbol{\Phi}_{t\to 0}^{p} / \partial \boldsymbol{x}$. All three keep the isotropic Gaussian family invariant and admit p_0 as their unique, globally attracting fixed point. Writing*

$$\left.\frac{\mathrm{d}\boldsymbol{m}_\tau}{\mathrm{d}\tau}\right|_{\mathrm{M}} = -A_{\mathrm{M}}\, (\boldsymbol{m}_\tau - \boldsymbol{\mu}), \qquad \left.\frac{\mathrm{d}\nu_\tau}{\mathrm{d}\tau}\right|_{\mathrm{M}} = (\rho^2 - \nu_\tau^2)\, B_{\mathrm{M}}(\nu_\tau), \tag{24}$$

for $\mathrm{M} \in \{\mathrm{PFD}, \mathrm{VSD}, \mathrm{DB}\}$, we have $A_{\mathrm{PFD}} = A_{\mathrm{VSD}} > A_{\mathrm{DB}}$ and $B_{\mathrm{PFD}} > B_{\mathrm{VSD}}, B_{\mathrm{DB}}$, with no ordering between B_{VSD} and B_{DB}. Therefore, PFD decreases $D_{\mathrm{KL}}(q_{0,\tau} \,\|\, p_0)$ strictly faster than DB, and strictly faster than VSD whenever $\nu_\tau \neq \rho$.

5 Experiments

5.1 Sampling from synthetic targets

We consider sampling ($\boldsymbol{g}(\boldsymbol{\theta}, \boldsymbol{\omega}) = \boldsymbol{\theta}$) from three synthetic targets of increasing difficulty. The targets are (i) a one-dimensional Gaussian $p_0 = \mathcal{N}(2, 0.5^2)$, (ii) Neal's two-dimensional funnel, $x_1 \sim$

[3] In text-to-3D, SDI substitutes $\boldsymbol{\epsilon}_{\boldsymbol{\phi}}$ under the null prompt for $\boldsymbol{\epsilon}_{\boldsymbol{\psi}}$.

$\mathcal{N}(0, 1.5^2)$ and $x_2 \,|\, x_1 \sim \mathcal{N}(0, e^{x_1})$, and (iii) a mixture of Gaussians (MoG) with 10 components in $\mathbb{R}^{20}$, $p_0(\boldsymbol{x}) = \sum_{k=1}^{10} \pi_k \mathcal{N}(\boldsymbol{x};\, \boldsymbol{\mu}_k,\, v_k \boldsymbol{I}_{20})$, whose parameters are drawn once as $\boldsymbol{\pi} \sim \mathrm{Dir}(8 \cdot \mathbf{1}_{10})$, $\boldsymbol{\mu}_k \sim \mathcal{N}(\mathbf{0}, 3^2 \boldsymbol{I}_{20})$, and $v_k \sim \mathcal{U}[0.35, 0.45]$. Figure 2 shows the results for the Gaussian target, where the propositions of Sections 3 and 4 can be checked directly, and Figure 3 shows the results for the funnel and the MoG. The same behavior appears on all three targets. SDS collapses onto the modes of p_0, SDI converges to a contracted version of p_0, and DB, VSD, and PFD match p_0. The left panel of Figure 2 shows that $D_{\mathrm{KL}}(q_{0,\tau} \,\|\, p_0)$ plateaus for SDS and SDI but keeps decreasing for DB, VSD, and PFD, fastest for PFD, followed by VSD and then DB. The right panel compares the gradients of PFD and VSD for a single particle. Since $\boldsymbol{g}$ is deterministic, the gradient of PFD is deterministic as well, whereas the gradient of VSD is noisy.

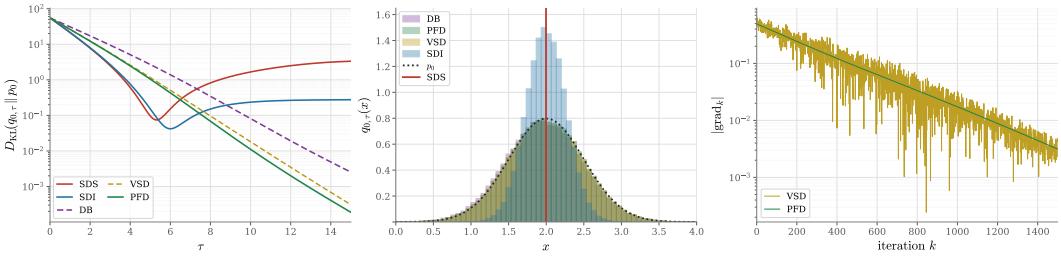


Figure 2: One-dimensional Gaussian target with 10^5 particles. (left) $D_{\mathrm{KL}}(q_{0,\tau} \,\|\, p_0)$ against optimization time. (middle) Final particle distributions compared with p_0. The vertical line marks the point mass reached by SDS. (right) Gradient magnitude of VSD and PFD for a single particle across iterations.

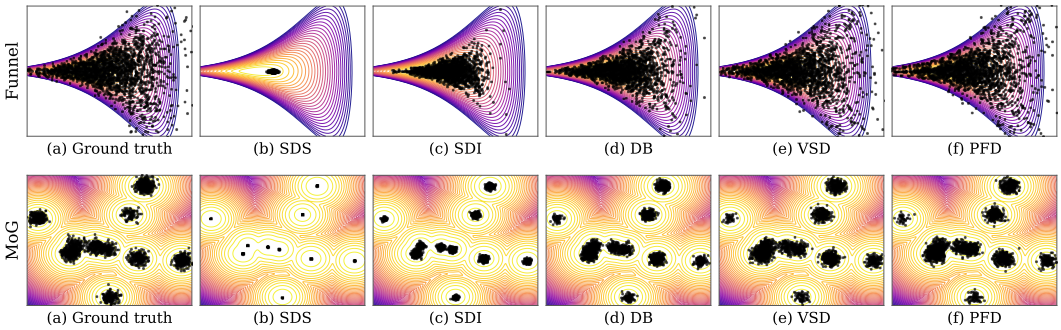


Figure 3: $2 \cdot 10^3$ particles after optimization on the two-dimensional funnel (top) and the twenty-dimensional MoG (bottom). The MoG samples are projected onto the two leading principal components of the ground-truth samples. The slight contraction of DB comes from discretization error, which is larger for DB because it solves $N + 1$ ODEs per step whereas the other methods solve one.

5.2 Experiments on the CelebA dataset

We train a VE diffusion model on the CelebA dataset (Liu et al., 2015) and use it as the pretrained model for four image tasks, all with a deterministic $\boldsymbol{g}$: (i) image generation, $\boldsymbol{g}(\boldsymbol{\theta}) = \boldsymbol{\theta}$, (ii) symmetric image generation, $\boldsymbol{g}(\boldsymbol{\theta}) = [\boldsymbol{\theta}, \mathrm{flip}(\boldsymbol{\theta})]$, which mirrors the left half $\boldsymbol{\theta}$ about the vertical axis, (iii) image inpainting, $\boldsymbol{g}(\boldsymbol{\theta}) = \boldsymbol{M} \odot \boldsymbol{\theta} + (\mathbf{1} - \boldsymbol{M}) \odot \boldsymbol{x}^{\mathrm{obs}}$, with a fixed binary mask $\boldsymbol{M}$ and the observed image $\boldsymbol{x}^{\mathrm{obs}}$, and (iv) vector graphics, where $\boldsymbol{g}$ is the differentiable rasterizer of `diffvg` (Li et al., 2020) and $\boldsymbol{\theta}$ holds the control points, colors, and widths of a set of Bézier strokes. Since $\boldsymbol{\omega}$ plays no role, $q_{0,\tau} = \boldsymbol{g}_{\#}\pi_\tau$, and matching p_0 therefore means spreading the particles so that $\boldsymbol{g}_{\#}\pi_\tau$ gets as close to p_0 as the range of $\boldsymbol{g}$ allows. In inpainting, that range is the set of images agreeing with $\boldsymbol{x}^{\mathrm{obs}}$ outside the mask, so the target is p_0 conditioned on the observed pixels, and the particles should cover the plausible completions rather than concentrating on a single one. For image generation, we run each method with four seeds and 2.5k particles per seed, giving 10k samples per method. Table 1 reports their FID, computed with `clean-fid` (Parmar et al., 2022), and CLIP-MMD (CMMD) (Jayasumana et al., 2024). As references, it also reports both metrics for the Euler

Table 1: FID and CMMD for image generation on CelebA, over 10k samples per method.

Reference			Distillation		
Method	FID ↓	CMMD ↓	Method	FID ↓	CMMD ↓
Dataset	2.89	< 0.001	SDS	236.82	6.531
Euler 50	25.54	0.377	SDI	108.78	3.289
Heun 50	24.43	0.368	VSD	**16.61**	**0.106**
			PFD	16.92	**0.106**

Table 2: Text-to-3D on 22 prompts, scored by CLIP R-precision and ImageReward (IR).

	CLIP R-precision ↑			
Method	ViT-L/14	ViT-B/16	ViT-B/32	IR ↑
SDS	0.796	0.820	0.802	-1.150
SDI	0.925	0.917	0.886	-0.257
VSD	0.958	**0.955**	**0.998**	0.680
PFD	**0.969**	**0.955**	**0.998**	**0.891**

and Heun samplers and for a disjoint 10k subset of the dataset. Both metrics give the same ordering. SDS is furthest from p_0, since each particle converges to a mode and the samples it produces are nearly identical. SDI is closer but still far. Its samples are varied but appear smooth, since contracting the spread of $q_{0,\tau}$ suppresses the high-frequency detail that varies most across p_0. VSD and PFD perform similarly and recover the spread of p_0 (see Appendices I.1–I.4).

5.3 Text-to-3D Generation

We build on the `threestudio` (Guo et al., 2023) implementations of SDS, SDI, and VSD, modified so that all methods have the same per-step cost, and implement PFD in the same framework. Since `threestudio` supports only a single representation, we are restricted to the single-particle regime, where $\pi_\tau = \delta_{\boldsymbol{\theta}_\tau}$ and $q_{0,\tau} = \boldsymbol{g}(\boldsymbol{\theta}_\tau, \cdot)_{\#}\xi$ is the distribution of renders over camera poses $\boldsymbol{\omega}$. With $c(\boldsymbol{\omega})$ the view-dependent prompt, matching p_0 therefore asks $\boldsymbol{g}(\boldsymbol{\theta}_\tau, \boldsymbol{\omega})$ to look like a sample from $p_0(\,\cdot \mid c(\boldsymbol{\omega}))$, while the shared $\boldsymbol{\theta}_\tau$ keeps nearby views consistent. The closer this holds, the finer the texture and geometry of the asset. Consistency between nearby views does not rule out global artifacts such as the Janus problem, which persist but are orthogonal to this work. We therefore evaluate the quality of the rendered views. Table 2 reports CLIP R-precision (Park et al., 2021) under three backbones, which measures how well the renders match their prompt, and ImageReward (Xu et al., 2023), which is trained to predict human preferences. Both metrics are averaged over 120 rendered views of the assets generated from 22 prompts. SDS pushes every view toward the same high-density image, resulting in oversaturated and oversmoothed assets. SDI avoids this collapse, but its contraction still leaves the views smooth. VSD and PFD match the spread of p_0 and recover fine texture and geometry (see Appendix I.5).

6 Conclusion

We studied score distillation through the objective of equation 1, asking how closely the induced distribution $q_{0,\tau}$ matches p_0 under each method and identified the distribution to which it converges. Because its update carries no dependence on $q_{0,\tau}$, SDS admits no term that maintains entropy, and the induced distribution collapses onto the modes of p_0. In SDI, DDIM inversion restores this dependence by approximating the forward PF-ODE of $q_{0,\tau}$, which prevents collapse but still leaves a spread that is strictly smaller than that of the target. The same view explains the sign flip in the CFG scale that SDI requires. Replacing the posterior mean with a reverse PF-ODE solve removes this bias at the cost of an additional solve from every node. We derived the gradient of this dual-bridge extension and used it to motivate PFD, which outperforms the extension and is competitive with VSD. Our experiments bear this out across image generation, symmetric generation, inpainting, and vector graphics on CelebA, and on text-to-3D, where a single representation confines us to one particle and the same ordering appears as the fidelity of the rendered views. For general targets, the superiority of PFD is established only in the small-time regime, and extending it, along with the multi-particle setting in text-to-3D, which would yield diverse as well as high-fidelity assets, is left for future work.

Reproducibility Statement

The experiments are readily reproducible from the details in Section 5 and Appendix H, and detailed proofs of all propositions and theorems appear in Appendices A–E. Full training details for

the model used in the CelebA experiments are given in Appendix H; all other experiments use publicly available pretrained models, and both `diffvg` and `threestudio` are open-source. We will release our code as well as model checkpoints publicly upon publication.

AI USE STATEMENT

In this work, we used generative AI tools to assist with correcting grammar, improving clarity, and verifying correctness. The ideas, theoretical results, proofs, experimental design, analysis, and conclusions were developed by the authors. All AI-assisted outputs were reviewed and verified by the authors, who take full responsibility for the final content.

ETHICS STATEMENT

This work does not involve human subjects, and we collect no new data. Our experiments use the publicly available CelebA dataset, which consists of face images of public figures, together with publicly available pretrained generative models, in accordance with their respective licenses and terms of use. CelebA is known to be demographically imbalanced, and the pretrained models inherit biases from their training data, so results produced by any of the methods we study may reflect those biases. This work is intended for research in generative modeling and should be used responsibly.

REFERENCES

Brian D.O. Anderson. Reverse-time diffusion equation models. *Stochastic Processes and their Applications*, 12(3):313–326, 1982. ISSN 0304-4149. doi: https://doi.org/10.1016/0304-4149(82)90051-5. URL https://www.sciencedirect.com/science/article/pii/0304414982900515.

Rui Chen, Yongwei Chen, Ningxin Jiao, and Kui Jia. Fantasia3d: Disentangling geometry and appearance for high-quality text-to-3d content creation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 22246–22256, October 2023.

Yuan-Chen Guo, Ying-Tian Liu, Ruizhi Shao, Christian Laforte, Vikram Voleti, Guan Luo, Chia-Hao Chen, Zi-Xin Zou, Chen Wang, Yan-Pei Cao, and Song-Hai Zhang. threestudio: A unified framework for 3d content generation. https://github.com/threestudio-project/threestudio, 2023.

Philip Hartman. *Ordinary Differential Equations*. Society for Industrial and Applied Mathematics, second edition, 2002. doi: 10.1137/1.9780898719222. URL https://epubs.siam.org/doi/abs/10.1137/1.9780898719222.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021. URL https://openreview.net/forum?id=qw8AKxfYbI.

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

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

Sadeep Jayasumana, Srikumar Ramalingam, Andreas Veit, Daniel Glasner, Ayan Chakrabarti, and Sanjiv Kumar. Rethinking fid: Towards a better evaluation metric for image generation. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9307–9315, 2024. doi: 10.1109/CVPR52733.2024.00889.

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=k7FuTOWMOc7.

Oren Katzir, Or Patashnik, Daniel Cohen-Or, and Dani Lischinski. Noise-free score distillation, 2023. URL https://arxiv.org/abs/2310.17590.

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4), July 2023. URL https://repo-sam.inria.fr/fungraph/3d-gaussian-splatting/.

Tzu-Mao Li, Michal Lukáč, Gharbi Michaël, and Jonathan Ragan-Kelley. Differentiable vector graphics rasterization for editing and learning. *ACM Trans. Graph. (Proc. SIGGRAPH Asia)*, 39 (6):193:1–193:15, 2020.

Yixun Liang, Xin Yang, Jiantao Lin, Haodong Li, Xiaogang Xu, and Yingcong Chen. Luciddreamer: Towards high-fidelity text-to-3d generation via interval score matching. *arXiv preprint arXiv:2311.11284*, 2023.

Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. Magic3d: High-resolution text-to-3d content creation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 300–309, June 2023.

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. URL https://openreview.net/forum?id=PqvMRDCJT9t.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

Artem Lukoianov, Haitz S'aez de Oc'ariz Borde, Kristjan Greenewald, Vitor Guizilini, Timur Bagautdinov, Vincent Sitzmann, and Justin M Solomon. Score distillation via reparametrized ddim. *Advances in Neural Information Processing Systems*, 37:26011–26044, 2024.

David McAllister, Songwei Ge, Jia-Bin Huang, David W. Jacobs, Alexei A Efros, Aleksander Holynski, and Angjoo Kanazawa. Rethinking score distillation as a bridge between image distributions. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=I8PkICj9kM.

Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, 2020.

Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text inversion for editing real images using guided diffusion models. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6038–6047, 2023. doi: 10.1109/CVPR52729.2023.00585.

Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Trans. Graph.*, 41(4):102:1–102:15, July 2022. doi: 10.1145/3528223.3530127. URL https://doi.org/10.1145/3528223.3530127.

Dong Huk Park, Samaneh Azadi, Xihui Liu, Trevor Darrell, and Anna Rohrbach. Benchmark for compositional text-to-image synthesis. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*, 2021. URL https://openreview.net/forum?id=bKBhQhPeKaF.

Gaurav Parmar, Richard Zhang, and Jun-Yan Zhu. On aliased resizing and surprising subtleties in gan evaluation. In *CVPR*, 2022.

Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv*, 2022.

Jessie Richter-Powell, Antonio Torralba, and Jonathan Lorraine. Score distillation sampling for audio: Source separation, synthesis, and beyond. *arXiv preprint arXiv:2505.04621*, 2025.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. *CoRR*, abs/2112.10752, 2021. URL https://arxiv.org/abs/2112.10752.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Raphael Gontijo-Lopes, Burcu Karagol Ayan, Tim Salimans, Jonathan Ho, David J. Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=08Yk-n5l2Al.

Tianchang Shen, Jun Gao, Kangxue Yin, Ming-Yu Liu, and Sanja Fidler. Deep marching tetrahedra: a hybrid representation for high-resolution 3d shape synthesis. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021a. URL https://openreview.net/forum?id=St1giarCHLP.

Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. In *Advances in Neural Information Processing Systems*, pp. 11895–11907, 2019.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021b. URL https://openreview.net/forum?id=PxTIG12RRHS.

Xuan Su, Jiaming Song, Chenlin Meng, and Stefano Ermon. Dual diffusion implicit bridges for image-to-image translation. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=5HLoTvVGDe.

Pascal Vincent. A connection between score matching and denoising autoencoders. *Neural Computation*, 23(7):1661–1674, 2011. doi: 10.1162/NECO_a_00142.

Zhengyi Wang, Cheng Lu, Yikai Wang, Fan Bao, Chongxuan Li, Hang Su, and Jun Zhu. Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

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 *Proceedings of the 37th International Conference on Neural Information Processing Systems*, pp. 15903–15935, 2023.

Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Frédo Durand, William T Freeman, and Taesung Park. One-step diffusion with distribution matching distillation. In *CVPR*, 2024.

Xin Yu, Yuan-Chen Guo, Yangguang Li, Ding Liang, Song-Hai Zhang, and XIAOJUAN QI. Text-to-3d with classifier score distillation. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=ktG8Tun1Cy.

Junzhe Zhu, Peiye Zhuang, and Sanmi Koyejo. HIFA: High-fidelity text-to-3d generation with advanced diffusion guidance. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=IZMPWmcS3H.

A Proof of Proposition 1

With $p_0 = \mathcal{N}(\boldsymbol{\mu}, \rho^2 \boldsymbol{I}_d)$ and the VE schedule $\sigma_t = t$, the perturbation is given by $\boldsymbol{x}_t = \boldsymbol{x}_0 + t\boldsymbol{\epsilon}$. Convolution of two Gaussians gives $p_t = \mathcal{N}(\boldsymbol{\mu}, (\rho^2 + t^2)\boldsymbol{I}_d)$, whose score is

$$\nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}_t) = -\frac{\boldsymbol{x}_t - \boldsymbol{\mu}}{\rho^2 + t^2}. \tag{25}$$

Substituting equation 25 into equation 13, we get

$$\frac{\mathrm{d}\boldsymbol{x}_0}{\mathrm{d}\tau} = -\,\mathbb{E}_{t,\boldsymbol{\epsilon}}\left[\frac{t^2\,(\boldsymbol{x}_0 + t\boldsymbol{\epsilon} - \boldsymbol{\mu})}{\rho^2 + t^2}\right]. \tag{26}$$

The inner expectation is $\mathbb{E}_{\boldsymbol{\epsilon}}[\boldsymbol{x}_0 + t\boldsymbol{\epsilon} - \boldsymbol{\mu}] = \boldsymbol{x}_0 - \boldsymbol{\mu}$, and since the remaining factor depends only on t, equation 26 reduces to

$$\frac{\mathrm{d}\boldsymbol{x}_0}{\mathrm{d}\tau} = -A(\rho)\,(\boldsymbol{x}_0 - \boldsymbol{\mu}), \qquad A(\rho) = \mathbb{E}_t\left[\frac{t^2}{\rho^2 + t^2}\right]. \tag{27}$$

With $t \sim \mathcal{U}[0,1]$, writing $\frac{t^2}{\rho^2+t^2} = 1 - \frac{\rho^2}{\rho^2+t^2}$, and using $\int_0^1 (\rho^2 + t^2)^{-1}\mathrm{d}t = \rho^{-1}\arctan(1/\rho)$ gives

$$A(\rho) = 1 - \rho^2 \cdot \frac{1}{\rho}\arctan\frac{1}{\rho} = 1 - \rho\arctan\frac{1}{\rho}, \tag{28}$$

which is > 0 and ≤ 1 for all $\rho > 0$. With $q_{0,\tau} = \mathcal{N}(\boldsymbol{m}_\tau, \nu_\tau^2 \boldsymbol{I}_d)$, taking the expectation with respect to $\boldsymbol{x}_0$ gives $\mathrm{d}\boldsymbol{m}_\tau/\mathrm{d}\tau = -A(\rho)(\boldsymbol{m}_\tau - \boldsymbol{\mu})$, and subtracting this from equation 27 gives $\mathrm{d}(\boldsymbol{x}_0 - \boldsymbol{m}_\tau)/\mathrm{d}\tau = -A(\rho)(\boldsymbol{x}_0 - \boldsymbol{m}_\tau)$. Thus, the deviation from the mean contracts at the same rate, giving $\mathrm{d}\nu_\tau/\mathrm{d}\tau = -A(\rho)\nu_\tau$. Both are linear, with solutions

$$\boldsymbol{m}_\tau = \boldsymbol{\mu} + e^{-A(\rho)\tau}(\boldsymbol{m}_0 - \boldsymbol{\mu}), \qquad \nu_\tau^2 = \nu_0^2\, e^{-2A(\rho)\tau}. \tag{29}$$

Since $A(\rho) > 0$, both $\|\boldsymbol{m}_\tau - \boldsymbol{\mu}\|$ and ν_τ decay exponentially to zero, so $q_{0,\tau} \to \delta_{\boldsymbol{\mu}}$ as $\tau \to \infty$. □

B Proof of Proposition 2

We write ν for ν_τ and $\boldsymbol{m}$ for $\boldsymbol{m}_\tau$ where no confusion arises. The PF-ODE associated with $\{q_{t,\tau}\}_t$ is $\mathrm{d}\boldsymbol{x}_t/\mathrm{d}t = -t\,\nabla_{\boldsymbol{x}} \log q_{t,\tau}(\boldsymbol{x}_t) = t(\boldsymbol{x}_t - \boldsymbol{m})/(\nu^2 + t^2)$, whose solution is given by

$$\boldsymbol{\Phi}_{0\to t}^{q_\tau}(\boldsymbol{x}_0) = \boldsymbol{m} + c_t(\nu)\,(\boldsymbol{x}_0 - \boldsymbol{m}), \qquad c_t(\nu) \coloneqq \frac{\sqrt{\nu^2 + t^2}}{\nu}. \tag{30}$$

Substituting the score of p_t (equation 25) into the posterior mean $\hat{\boldsymbol{x}}_0$ gives

$$\hat{\boldsymbol{x}}_0(\boldsymbol{x}_t) = \boldsymbol{x}_t - \frac{t^2(\boldsymbol{x}_t - \boldsymbol{\mu})}{\rho^2 + t^2} = \boldsymbol{\mu} + \frac{\rho^2}{\rho^2 + t^2}\,(\boldsymbol{x}_t - \boldsymbol{\mu}), \tag{31}$$

which is affine, as is equation 30. Defining $\boldsymbol{u} \coloneqq \boldsymbol{x}_0 - \boldsymbol{m}$ and combining equation 30 with equation 31,

$$\hat{\boldsymbol{x}}_0(\boldsymbol{\Phi}_{0\to t}^{q_\tau}(\boldsymbol{x}_0)) - \boldsymbol{x}_0 = -\frac{t^2}{\rho^2 + t^2}\,(\boldsymbol{m} - \boldsymbol{\mu}) + \left(\frac{\rho^2\, c_t(\nu)}{\rho^2 + t^2} - 1\right)\boldsymbol{u}. \tag{32}$$

This, when substituted in equation 17, gives

$$\frac{\mathrm{d}\boldsymbol{x}_0}{\mathrm{d}\tau} = -A(\rho)\,(\boldsymbol{m} - \boldsymbol{\mu}) + (B(\nu) - 1)\,\boldsymbol{u}, \tag{33}$$

with $A(\rho) = \mathbb{E}_t[t^2/(\rho^2 + t^2)]$ and $B(\nu) = (\rho^2/\nu)\,\mathbb{E}_t[\sqrt{\nu^2 + t^2}/(\rho^2 + t^2)]$. Since $\mathbb{E}[\boldsymbol{u}] = \boldsymbol{0}$, taking the expectation with respect to $\boldsymbol{x}_0$ in equation 33 gives equation 18a, and subtracting this from equation 33 leaves $\mathrm{d}\boldsymbol{u}/\mathrm{d}\tau = (B(\nu) - 1)\boldsymbol{u}$. Thus, the deviation from the mean contracts at a rate of $1 - B(\nu)$, and equation 18b follows.

Writing

$$B(\nu) = \rho^2\,\mathbb{E}_t\left[\frac{1}{\rho^2 + t^2}\sqrt{1 + \frac{t^2}{\nu^2}}\right] = \rho^2 \int_0^1 \frac{1}{\rho^2 + t^2}\sqrt{1 + \frac{t^2}{\nu^2}}\mathrm{d}t, \tag{34}$$

the integrand is strictly decreasing in ν for every $t > 0$, so B is strictly decreasing on $(0, \infty)$. It diverges as $\nu \to 0^+$ and tends to $\rho^2\,\mathbb{E}_t[(\rho^2 + t^2)^{-1}] = 1 - A(\rho)$ as $\nu \to \infty$. Since $1 - A(\rho) < 1$, there is a unique $\nu_\star$ with $B(\nu_\star) = 1$. For $\nu < \nu_\star$, we have $B(\nu) > 1$ and $\mathrm{d}\nu_\tau/\mathrm{d}\tau > 0$, and for $\nu > \nu_\star$ the reverse, so $\nu_\star$ is globally attracting. Finally, evaluating equation 34 at $\nu = \rho$ gives $B(\rho) = \rho\,\mathbb{E}_t[(\rho^2 + t^2)^{-1/2}] < 1$, because $(\rho^2 + t^2)^{-1/2} < \rho^{-1}$ for $t > 0$. As B is decreasing and $B(\nu_\star) = 1$, it follows that $\nu_\star < \rho$, and the particle distribution converges to $\mathcal{N}(\boldsymbol{\mu}, \nu_\star^2 \boldsymbol{I}_d)$. □

C Proof of Proposition 6

Evaluating the score difference at a point $\boldsymbol{x}_t$, we obtain

$$\nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}_t) - \nabla_{\boldsymbol{x}} \log q_{t,\tau}(\boldsymbol{x}_t) = -\frac{\boldsymbol{m} - \boldsymbol{\mu}}{\rho^2 + t^2} + \frac{\rho^2 - \nu^2}{(\nu^2 + t^2)(\rho^2 + t^2)} (\boldsymbol{x}_t - \boldsymbol{m}). \tag{35}$$

Taking $\eta = 1$ in equation 22a and equation 22b, and appending the Jacobian factor for DB, the three methods share the same form.

$$\left.\frac{\mathrm{d}\boldsymbol{x}_0}{\mathrm{d}\tau}\right|_{\mathrm{M}} = \int_0^1 w(t)\, \Lambda_t^{\mathrm{M}} \left(\nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}_t^{\mathrm{M}}) - \nabla_{\boldsymbol{x}} \log q_{t,\tau}(\boldsymbol{x}_t^{\mathrm{M}})\right) \mathrm{d}t, \tag{36}$$

and differ only in Λ_t^{M} and in the coupling $(\boldsymbol{x}_0, \boldsymbol{x}_t^{\mathrm{M}})$. PFD has $\Lambda_t = 1$ with $\boldsymbol{x}_t = \boldsymbol{\Phi}_{0\to t}^{q_\tau}(\boldsymbol{x}_0)$, so $\boldsymbol{x}_t - \boldsymbol{m} = c_t(\nu)\, \boldsymbol{u}$ by equation 30, VSD has $\Lambda_t = 1$ with $\boldsymbol{x}_t = \boldsymbol{x}_0 + t\boldsymbol{\epsilon}$, for which $\mathbb{E}_{\boldsymbol{\epsilon}}[\boldsymbol{x}_t - \boldsymbol{m}] = \boldsymbol{u}$, and DB has the same coupling as PFD with

$$\Lambda_t = \frac{\partial \boldsymbol{\Phi}_{t\to 0}^{p}}{\partial \boldsymbol{x}} = \frac{\rho}{\sqrt{\rho^2 + t^2}}, \tag{37}$$

obtained as in equation 30 but for the reverse PF-ODE, for which we get $\boldsymbol{\Phi}_{0\to t}^{p}(\boldsymbol{x}) - \boldsymbol{\mu} = (\sqrt{\rho^2 + t^2}/\rho)\,(\boldsymbol{x} - \boldsymbol{\mu})$. Substituting equation 35 into equation 36, every update is affine in $\boldsymbol{x}_0$, so the isotropic Gaussian family is closed under all three. Since $\mathbb{E}[\boldsymbol{u}] = \boldsymbol{0}$, taking the expectation with respect to $\boldsymbol{x}_0$ gives the left-hand side of equation 24 with

$$A_{\mathrm{PFD}} = A_{\mathrm{VSD}} = \int_0^1 \frac{w(t)}{\rho^2 + t^2}\, \mathrm{d}t, \qquad A_{\mathrm{DB}} = \rho \int_0^1 \frac{w(t)}{(\rho^2 + t^2)^{3/2}}\, \mathrm{d}t, \tag{38}$$

and subtracting it from equation 36 leaves $\mathrm{d}\boldsymbol{u}/\mathrm{d}\tau$, whose coefficient gives $\mathrm{d}\nu_\tau/\mathrm{d}\tau$. The factor $\rho^2 - \nu^2$ in equation 35 does not depend on t. Pulling it out and using $\nu\, c_t(\nu) = \sqrt{\nu^2 + t^2}$ gives the right-hand side of equation 24 with

$$B_{\mathrm{PFD}}(\nu) = \int_0^1 \frac{w(t)}{\sqrt{\nu^2 + t^2}\,(\rho^2 + t^2)}\, \mathrm{d}t, \tag{39a}$$

$$B_{\mathrm{VSD}}(\nu) = \int_0^1 \frac{w(t)\,\nu}{(\nu^2 + t^2)(\rho^2 + t^2)}\, \mathrm{d}t, \tag{39b}$$

$$B_{\mathrm{DB}}(\nu) = \int_0^1 \frac{w(t)\,\rho}{\sqrt{\nu^2 + t^2}\,(\rho^2 + t^2)^{3/2}}\, \mathrm{d}t. \tag{39c}$$

Since $w > 0$, all three integrands are positive on $(0, 1)$ for every $\nu > 0$, so $B_{\mathrm{M}}(\nu) > 0$ and the sign of $\mathrm{d}\nu_\tau/\mathrm{d}\tau$ is that of $\rho^2 - \nu_\tau^2$: the variance increases below ρ and decreases above it, so $\nu_\star = \rho$ is the unique stationary point and is globally attracting. Likewise $A_{\mathrm{M}} > 0$, so $\boldsymbol{m}_\tau \to \boldsymbol{\mu}$ exponentially, and p_0 is the unique, globally attracting fixed point of all three.

It remains to order the constants. The integrands of equation 38 differ by $\rho/\sqrt{\rho^2 + t^2} < 1$, so

$$A_{\mathrm{DB}} < A_{\mathrm{PFD}} = A_{\mathrm{VSD}}. \tag{40}$$

Those of equation 39a, equation 39b and equation 39c differ pointwise in t by

$$\frac{B_{\mathrm{PFD}}}{B_{\mathrm{VSD}}} : \frac{\sqrt{\nu^2 + t^2}}{\nu}, \qquad \frac{B_{\mathrm{PFD}}}{B_{\mathrm{DB}}} : \frac{\sqrt{\rho^2 + t^2}}{\rho}, \qquad \frac{B_{\mathrm{DB}}}{B_{\mathrm{VSD}}} : \frac{\rho\sqrt{\nu^2 + t^2}}{\nu\sqrt{\rho^2 + t^2}}, \tag{41}$$

where the first two exceed one for every $t > 0$ and every $\nu > 0$, giving $B_{\mathrm{PFD}}(\nu) > B_{\mathrm{VSD}}(\nu)$ and $B_{\mathrm{PFD}}(\nu) > B_{\mathrm{DB}}(\nu)$. The third exceeds one for $\nu < \rho$ and falls below one for $\nu > \rho$, the two coinciding only at $\nu = \rho$. Therefore, there is no ordering between B_{VSD} and B_{DB}. For isotropic Gaussians, we have

$$D_{\mathrm{KL}}(q_{0,\tau} \,\|\, p_0) = \frac{\|\boldsymbol{m}_\tau - \boldsymbol{\mu}\|^2}{2\rho^2} + \frac{d}{2}\left(\frac{\nu_\tau^2}{\rho^2} - 1 - 2\log\frac{\nu_\tau}{\rho}\right), \tag{42}$$

which increases as $\boldsymbol{m}_\tau$ moves away from $\boldsymbol{\mu}$ and as ν_τ moves away from ρ in either direction. PFD drives $\boldsymbol{m}_\tau$ to $\boldsymbol{\mu}$ at least as fast as the other two by equation 40 and ν_τ to ρ strictly faster by equation 41, so it decreases $D_{\mathrm{KL}}(q_{0,\tau} \,\|\, p_0)$ strictly faster than both VSD and DB. □

D Proof of Theorem 4

We write $\boldsymbol{v}^q(\boldsymbol{x}_t, t) \coloneqq \boldsymbol{f}(\boldsymbol{x}_t, t) - \frac{1}{2} g_t^2 \nabla_{\boldsymbol{x}} \log q_{t,\tau}(\boldsymbol{x}_t)$ and $\boldsymbol{v}^p(\boldsymbol{x}_t, t) \coloneqq \boldsymbol{f}(\boldsymbol{x}_t, t) - \frac{1}{2} g_t^2 \nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}_t)$ for the velocity fields of the forward and reverse PF-ODEs, respectively. Now, $\boldsymbol{\Delta}(\boldsymbol{x}_0, s) = \boldsymbol{x}_0 - \boldsymbol{\Phi}^p_{s\to 0}(\boldsymbol{x}_s)$, where $\boldsymbol{x}_s = \boldsymbol{\Phi}^{q_\tau}_{0\to s}(\boldsymbol{x}_0)$. Taking the derivative with respect to s gives

$$\frac{\mathrm{d}\boldsymbol{\Delta}}{\mathrm{d}s} = -\frac{\mathrm{d}}{\mathrm{d}s}\, \boldsymbol{\Phi}^p_{s\to 0}(\boldsymbol{x}_s) = -\left(\frac{\partial \boldsymbol{\Phi}^p_{s\to 0}}{\partial \boldsymbol{x}}(\boldsymbol{x}_s)\, \boldsymbol{v}^q(\boldsymbol{x}_s, s) + \partial_s \boldsymbol{\Phi}^p_{s\to 0}(\boldsymbol{x}_s) \right), \tag{43}$$

where the first term comes from the dependence on $\boldsymbol{x}_s$, with $\mathrm{d}\boldsymbol{x}_s/\mathrm{d}s = \boldsymbol{v}^q(\boldsymbol{x}_s, s)$, and the second from the s-dependence of $\boldsymbol{\Phi}^p_{s\to 0}$.

Lemma 1. *Let $\boldsymbol{\Phi}_{s\to s'}(\boldsymbol{x})$ be the flow map of $\mathrm{d}\boldsymbol{x}/\mathrm{d}t = \boldsymbol{F}(\boldsymbol{x}, t)$, carrying a state $\boldsymbol{x}$ at time s to time s'. Then we have the following:*

$$\partial_s \boldsymbol{\Phi}_{s\to s'}(\boldsymbol{x}) = -\frac{\partial \boldsymbol{\Phi}_{s\to s'}}{\partial \boldsymbol{x}}\, \boldsymbol{F}(\boldsymbol{x}, s), \tag{44}$$

where $\partial \boldsymbol{\Phi}_{s\to s'}/\partial \boldsymbol{x}$ is the Jacobian of the flow map with respect to its initial condition, evaluated at the starting point $\boldsymbol{x}$. See Chapter 5 of Hartman (2002) for a proof.

Applying Lemma 1 with $\boldsymbol{F} = \boldsymbol{v}^p$ to the second term of equation 43, we get

$$\begin{aligned} \frac{\mathrm{d}\boldsymbol{\Delta}}{\mathrm{d}s} &= -\frac{\partial \boldsymbol{\Phi}^p_{s\to 0}}{\partial \boldsymbol{x}}(\boldsymbol{x}_s)\, (\boldsymbol{v}^q(\boldsymbol{x}_s, s) - \boldsymbol{v}^p(\boldsymbol{x}_s, s)) \\ &= \frac{1}{2} g_s^2\, \frac{\partial \boldsymbol{\Phi}^p_{s\to 0}}{\partial \boldsymbol{x}}(\boldsymbol{x}_s)\, (\nabla_{\boldsymbol{x}} \log q_{s,\tau}(\boldsymbol{x}_s) - \nabla_{\boldsymbol{x}} \log p_s(\boldsymbol{x}_s)), \end{aligned} \tag{45}$$

where the drift $\boldsymbol{f}$ cancels out in the difference. Integrating equation 45 from 0 to t subject to the boundary condition $\boldsymbol{\Delta}(\boldsymbol{x}_0, 0) = \boldsymbol{0}$ gives equation 20a. By Fubini's theorem, exchanging the order of integration with $t \sim \mathcal{U}[0, 1]$ gives

$$\mathbb{E}_t[\boldsymbol{\Delta}(\boldsymbol{x}_0, t)] = \int_0^1 \int_0^t (\cdot)_s \,\mathrm{d}s\,\mathrm{d}t = \int_0^1 (\cdot)_s \int_s^1 \mathrm{d}t\,\mathrm{d}s = \int_0^1 (1 - s)\,(\cdot)_s \,\mathrm{d}s, \tag{46}$$

where $(\cdot)_s$ denotes the integrand of equation 20a, which gives equation 20b. □

E Proof of Theorem 5

Define $\boldsymbol{G}_t(\boldsymbol{x}) \coloneqq \nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}) - \nabla_{\boldsymbol{x}} \log q_{t,\tau}(\boldsymbol{x})$, then the two velocity fields are given by:

$$\boldsymbol{v}_{\mathrm{VSD}}(\boldsymbol{x}_0) = \int_0^\eta w(t)\, \mathbb{E}_{\boldsymbol{\epsilon}}[\boldsymbol{G}_t(\boldsymbol{x}_0 + t\boldsymbol{\epsilon})]\,\mathrm{d}t, \qquad \boldsymbol{v}_{\mathrm{PFD}}(\boldsymbol{x}_0) = \int_0^\eta w(t)\, \boldsymbol{G}_t(\Phi^{q_\tau}_{0\to t}(\boldsymbol{x}_0))\,\mathrm{d}t. \tag{47}$$

If the particles evolve under a field $\boldsymbol{v}$, the density satisfies the continuity equation $\partial_\tau q_{0,\tau} = -\nabla \cdot (q_{0,\tau} \boldsymbol{v})$. Hence, differentiating $D_{\mathrm{KL}}(q_{0,\tau} \| p_0)$ and integrating by parts gives

$$\frac{\mathrm{d}D_{\mathrm{KL}}}{\mathrm{d}\tau} = \int \log \frac{q_{0,\tau}(\boldsymbol{x}_0)}{p_0(\boldsymbol{x}_0)}\, \partial_\tau q_{0,\tau}(\boldsymbol{x}_0)\,\mathrm{d}\boldsymbol{x}_0 = -\,\mathbb{E}_{\boldsymbol{x}_0 \sim q_{0,\tau}}[\langle \boldsymbol{v}(\boldsymbol{x}_0),\, \boldsymbol{G}_0(\boldsymbol{x}_0)\rangle]\,. \tag{48}$$

Thus, it suffices to expand $\boldsymbol{v}_{\mathrm{VSD}} - \boldsymbol{v}_{\mathrm{PFD}}$ and pair the result with $\boldsymbol{G}_0$. Since $q_{t,\tau}$ and p_t are convolutions with $\mathcal{N}(\boldsymbol{0}, t^2 \boldsymbol{I}_d)$, they depend on t only through t^2, so $\boldsymbol{G}_t = \boldsymbol{G}_0 + O(t^2)$, together with its spatial derivatives. For the first field, a second-order Taylor expansion of $\boldsymbol{G}_t$ about $\boldsymbol{x}_0$ gives

$$\boldsymbol{G}_t(\boldsymbol{x}_0 + t\boldsymbol{\epsilon}) = \boldsymbol{G}_t(\boldsymbol{x}_0) + t\, \frac{\partial \boldsymbol{G}_t}{\partial \boldsymbol{x}}(\boldsymbol{x}_0)\, \boldsymbol{\epsilon} + \frac{t^2}{2}\, \boldsymbol{Q}_t(\boldsymbol{x}_0)[\boldsymbol{\epsilon}] + O(t^3), \tag{49}$$

where $\boldsymbol{Q}_t(\boldsymbol{x}_0)[\boldsymbol{\epsilon}]$ is the vector whose j-th entry is $\boldsymbol{\epsilon}^\top \boldsymbol{H}_j \boldsymbol{\epsilon}$, with $\boldsymbol{H}_j$ being the Hessian of the j-th component of $\boldsymbol{G}_t$ at $\boldsymbol{x}_0$. The first-order term vanishes under $\mathbb{E}_{\boldsymbol{\epsilon}}[\boldsymbol{\epsilon}] = \boldsymbol{0}$, and since $\mathbb{E}_{\boldsymbol{\epsilon}}[\boldsymbol{\epsilon}^\top \boldsymbol{H}_j \boldsymbol{\epsilon}] = \operatorname{tr} \boldsymbol{H}_j = \Delta\,[\boldsymbol{G}_t]_j$, the quadratic term contracts into a component-wise Laplacian $\Delta \boldsymbol{G}_t(\boldsymbol{x}_0)$. The odd orders vanish by the symmetry of the Gaussian, so

$$\mathbb{E}_{\boldsymbol{\epsilon}}[\boldsymbol{G}_t(\boldsymbol{x}_0 + t\boldsymbol{\epsilon})] = \boldsymbol{G}_t(\boldsymbol{x}_0) + \frac{t^2}{2}\, \Delta \boldsymbol{G}_0(\boldsymbol{x}_0) + O(t^4). \tag{50}$$

For the second field, the forward PF-ODE reads $\mathrm{d}\boldsymbol{x}/\mathrm{d}s = -s\,\nabla_{\boldsymbol{x}} \log q_{s,\tau}(\boldsymbol{x})$, so $\boldsymbol{\Phi}^{q_\tau}_{0\to t}(\boldsymbol{x}_0) = \boldsymbol{x}_0 - \int_0^t s\,\nabla_{\boldsymbol{x}} \log q_{s,\tau}(\boldsymbol{x}_s)\,\mathrm{d}s$, giving

$$\boldsymbol{\Phi}^{q_\tau}_{0\to t}(\boldsymbol{x}_0) = \boldsymbol{x}_0 - \frac{t^2}{2}\,\nabla_{\boldsymbol{x}} \log q_{0,\tau}(\boldsymbol{x}_0) + O(t^4), \tag{51}$$

and a first-order expansion of $\boldsymbol{G}_t$ about $\boldsymbol{x}_0$ gives

$$\boldsymbol{G}_t(\boldsymbol{\Phi}^{q_\tau}_{0\to t}(\boldsymbol{x}_0)) = \boldsymbol{G}_t(\boldsymbol{x}_0) - \frac{t^2}{2}\,\frac{\partial \boldsymbol{G}_0}{\partial \boldsymbol{x}}(\boldsymbol{x}_0)\,\nabla_{\boldsymbol{x}} \log q_{0,\tau}(\boldsymbol{x}_0) + O(t^4). \tag{52}$$

In both expansions, $\boldsymbol{G}_t$ has been replaced by $\boldsymbol{G}_0$ in the t^2 terms at a cost of $O(t^4)$. Both share the zeroth-order term $\boldsymbol{G}_t(\boldsymbol{x}_0)$, which cancels exactly in the difference, and what survives is the difference of the two t^2 terms, with a remainder of $O(t^4)$. Subtracting and integrating against $w(t)$, with $C_\eta = \frac{1}{2}\int_0^\eta w(t)\,t^2\,\mathrm{d}t$ and $\int_0^\eta w(t)\,t^4\,\mathrm{d}t \le \eta^2 \int_0^\eta w(t)\,t^2\,\mathrm{d}t$, results in

$$\boldsymbol{v}_{\mathrm{VSD}} - \boldsymbol{v}_{\mathrm{PFD}} = C_\eta \left(\Delta \boldsymbol{G}_0 + \frac{\partial \boldsymbol{G}_0}{\partial \boldsymbol{x}}\,\nabla_{\boldsymbol{x}} \log q_{0,\tau} + O(\eta^2) \right). \tag{53}$$

Write $\varphi \coloneqq \log p_0 - \log q_{0,\tau}$, so that $\boldsymbol{G}_0 = \nabla\varphi$ and $\partial \boldsymbol{G}_0/\partial \boldsymbol{x} = \nabla^2\varphi$, and let $\mathcal{L}$ denote the infinitesimal Langevin operator of $q_{0,\tau}$,

$$\mathcal{L}f = \Delta f + \nabla f^\top \nabla \log q_{0,\tau}. \tag{54}$$

The bracket in equation 53 is then $\mathcal{L}$ applied to each component of $\boldsymbol{G}_0$, since its j-th entry is $\Delta\,\partial_j\varphi + \nabla(\partial_j\varphi)^\top \nabla \log q_{0,\tau} = \mathcal{L}\,\partial_j\varphi$. Note that this result can be generalized to any sufficiently smooth test function. Thus, we state it as a result in its own right.

Proposition 7. *Under the VE schedule $\sigma_t = t$, let $f : \mathbb{R}^d \to \mathbb{R}$ be C^4. Then*

$$\mathbb{E}_{\boldsymbol{\epsilon}}\big[f(\boldsymbol{x}_0 + t\boldsymbol{\epsilon})\big] - f\big(\boldsymbol{\Phi}^{q_\tau}_{0\to t}(\boldsymbol{x}_0)\big) = \frac{t^2}{2}\,\mathcal{L}f(\boldsymbol{x}_0) + O(t^4), \tag{55}$$

where $\mathcal{L}f = \Delta f + \langle \nabla_{\boldsymbol{x}} \log q_{0,\tau}, \nabla_{\boldsymbol{x}} f\rangle$ is the infinitesimal generator of the Langevin dynamics $\mathrm{d}\boldsymbol{x}_s = \nabla_{\boldsymbol{x}} \log q_{0,\tau}(\boldsymbol{x}_s)\,\mathrm{d}s + \sqrt{2}\,\mathrm{d}\boldsymbol{w}_s$, which has invariant density $q_{0,\tau}$.

Multiplying equation 54 by $q_{0,\tau}$ and using $q_{0,\tau}\nabla \log q_{0,\tau} = \nabla q_{0,\tau}$ gives $q_{0,\tau}\mathcal{L}f = q_{0,\tau}\Delta f + \nabla f^\top \nabla q_{0,\tau}$, which, by the product rule, is

$$q_{0,\tau}\,\mathcal{L}f = \nabla \cdot (q_{0,\tau}\nabla f)\,, \tag{56}$$

so that integrating by parts and using the decay of $q_{0,\tau}$ at infinity to discard the boundary term,

$$-\,\mathbb{E}_{\boldsymbol{x}_0\sim q_{0,\tau}}[f\,\mathcal{L}f] = -\int f\,\nabla \cdot (q_{0,\tau}\nabla f) = \int q_{0,\tau}\,\|\nabla f\|^2 = \mathbb{E}_{\boldsymbol{x}_0\sim q_{0,\tau}}\big[\|\nabla f\|^2\big]\,, \tag{57}$$

the Dirichlet energy of f associated with $\mathcal{L}$. Applying equation 57 with $f = \partial_j\varphi$, summing over j, and using $\sum_j \|\nabla\partial_j\varphi\|^2 = \|\nabla^2\varphi\|_F^2$,

$$\mathbb{E}_{\boldsymbol{x}_0\sim q_{0,\tau}}\left[\left\langle \Delta \boldsymbol{G}_0 + \frac{\partial \boldsymbol{G}_0}{\partial \boldsymbol{x}}\,\nabla \log q_{0,\tau},\ \boldsymbol{G}_0 \right\rangle\right] = -\,\mathbb{E}_{\boldsymbol{x}_0\sim q_{0,\tau}}\left[\left\|\nabla^2\varphi\right\|_F^2\right]. \tag{58}$$

Substituting equation 53 into equation 48 and applying equation 58,

$$\left.\frac{\mathrm{d}D_{\mathrm{KL}}}{\mathrm{d}\tau}\right|_{\mathrm{VSD}} - \left.\frac{\mathrm{d}D_{\mathrm{KL}}}{\mathrm{d}\tau}\right|_{\mathrm{PFD}} = C_\eta \left(\mathbb{E}_{\boldsymbol{x}_0\sim q_{0,\tau}}\left[\left\|\nabla^2\varphi\right\|_F^2\right] + O(\eta^2)\right), \tag{59}$$

Since $C_\eta > 0$ and the leading term is non-negative, the right-hand side is non-negative for η small enough. PFD therefore decreases $D_{\mathrm{KL}}(q_{0,\tau}\,\|\,p_0)$ at least as fast as VSD from the same $q_{0,\tau}$, and strictly faster unless $\nabla^2\varphi$ vanishes $q_{0,\tau}$-almost everywhere. □

F NEGATIVE CFG SCALE IN IMAGE EDITING

Figure 4 compares three ways of running the inversion, each followed by DDIM sampling under the target prompt. Method 1 is standard DDIM inversion editing, which inverts under a source prompt describing the input at a positive CFG scale. Once guidance is applied, it no longer preserves the input faithfully, which is what motivates methods such as Null-Text Inversion (Mokady et al., 2023). Method 2 replaces the source prompt with the target prompt at the same positive scale. As discussed in Section 3.2, this tilts the inversion towards p_t rather than away from it, and the edits become unstable. Method 3 inverts under the target prompt with the sign reversed, as SDI does. By the argument of Section 3.2, this approximates the forward PF-ODE, so the edit becomes an approximate DDIB between the input and target distributions. This restores the stability lost in Method 2 at the cost of some over-saturation and smoothing stemming from the approximation. This supports reading DDIM inversion under the target prompt with a negative CFG scale as an approximation of $\mathbf{\Phi}^{q_\tau}_{0\to t}$.

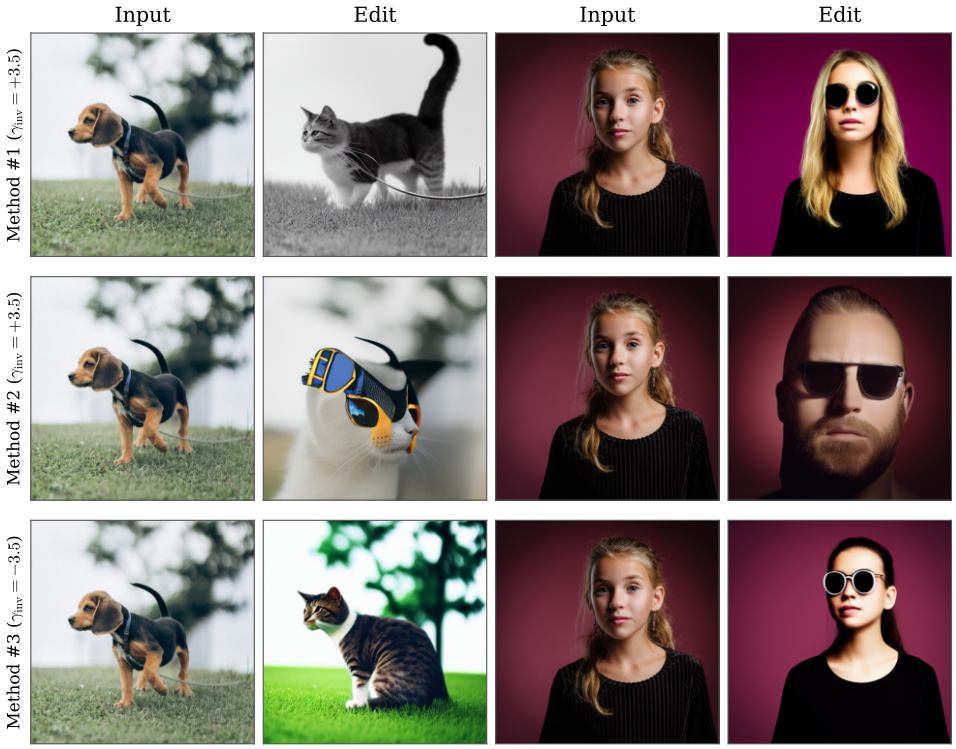


Figure 4: Editing under three different inversion methods. Method 1 inverts under a source prompt describing the input at $\gamma_{\text{inv}} = +3.5$, Method 2 inverts under the target prompt at the same positive scale, Method 3 inverts under the target prompt at $\gamma_{\text{inv}} = -3.5$. The source prompts are *"photograph of a puppy on the grass"* and *"a photograph of a face"*, and the target prompts are *"photograph of a cat on the grass"* and *"a photograph of a face with sunglasses"*. The sources for the two input images can be found at `https://huggingface.co/learn/diffusion-course/en/unit4/2` and the edits use the `stable-diffusion-v1-5` checkpoint.

G GROUPING OF EXISTING DISTILLATION METHODS

Figure 5 shows the classification of some text-to-3D distillation works as instances of the three methods. VSD uses $\boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t, c) - \boldsymbol{\epsilon}_{\boldsymbol{\psi},\tau}(\boldsymbol{x}_t, t)$ as the gradient estimator. Classifier-Score Distillation (CSD) (Yu et al., 2024) approximates this in the same way that SDI approximates its ideal implementation—replacing $\boldsymbol{\epsilon}_{\boldsymbol{\psi},\tau}(\boldsymbol{x}_t, t)$ with the pretrained model's unconditional predictor—so that its gradient estimator is $\boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t, c) - \boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t, \varnothing)$. Noise-Free Score Distillation (NFSD) can likewise be viewed as a VSD approximation (see Section 5 of Katzir et al. (2023)). SDS-Bridge (McAllister et al., 2024) alludes to the dual-bridge construction of Section 3.3 but avoids it in practice on the grounds of cost. Its algorithm instead runs vanilla SDS for an initial phase and then switches to the gradient estimator $\boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t, c) - \boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t, c_{\text{src}})$, where the source prompt c_{src} (e.g., `blurry, over-saturated`) is intended as a closer description of the current induced distribution $q_{0,\tau}$ than the null prompt. Hence, it is an approximation of VSD as well.

SDS-based methods use $\hat{\boldsymbol{\epsilon}}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t, c) - \boldsymbol{\epsilon}$ as their gradient estimator. Their official implementations, however, use a large CFG scale γ, as this provides better results. Expanding the guided prediction $\hat{\boldsymbol{\epsilon}}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t, c)$ gives

$$(\boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t, \varnothing) - \boldsymbol{\epsilon}) + \gamma\,(\boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t, c) - \boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t, \varnothing))\,, \tag{60}$$

in which the first term does not grow with γ, while the second, which is the CSD approximation of VSD, dominates at large γ, making these methods approximations of VSD as well. To stay closer to SDS as originally conceived, we use a low CFG value (7.5) in our text-to-3D experiments.

Interval Score Matching (ISM) (Liang et al., 2023), like SDI, constructs $\boldsymbol{x}_t$ by DDIM inversion, but it runs that inversion differently. ISM inverts under the null prompt alone, giving a CFG-free approximation of the forward PF-ODE $\boldsymbol{\Phi}^{q_\tau}_{0\to t}$, whereas SDI uses both the conditional and the unconditional predictors and therefore needs a negative CFG scale (see Section 3.2). The two also differ in what the denoised prediction $\hat{\boldsymbol{x}}_0(\boldsymbol{x}_t)$ is compared against, with SDI using $\boldsymbol{x}_0$ itself and ISM using $\hat{\boldsymbol{x}}_0(\boldsymbol{x}_{t-\Delta t})$, the prediction from the penultimate inversion iterate.

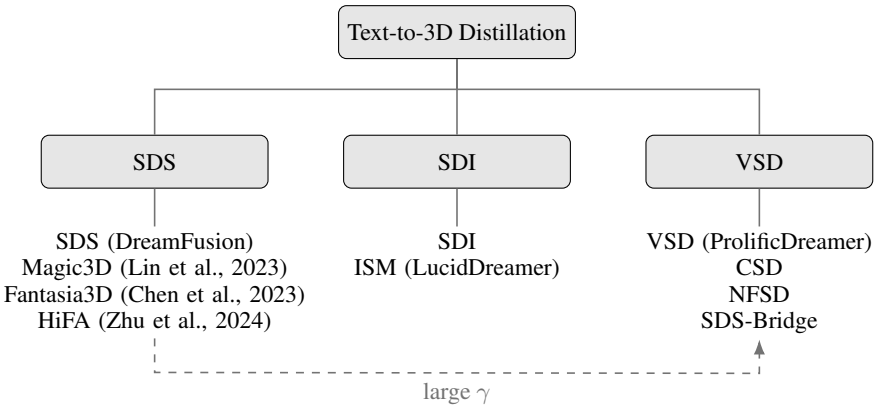


Figure 5: Grouping of text-to-3D distillation approaches into three methods. The dashed edge indicates the VSD approximation of SDS at large CFG scales.

H Implementation Details

All experiments are run on a single NVIDIA A100 GPU with 80 GB of memory.

H.1 Sampling from synthetic targets

For the Gaussian and the mixture, the scores $\nabla_{\boldsymbol{x}} \log p_t$ are available in closed form. For the funnel, we train a score network with DSM on 10^5 samples from p_0. All integrals over t are evaluated with 8 Gauss–Legendre nodes, and every run takes 1500 Euler steps in τ. For the Gaussian target, the score $\nabla_{\boldsymbol{x}} \log q_{t,\tau}$ and both flow maps are available in closed form. We evaluate them at the empirical mean and standard deviation of the particles. For the funnel and the mixture, $\boldsymbol{\epsilon}_{\boldsymbol{\psi},\tau}$ is an MLP trained online with DSM on $\{\boldsymbol{x}_{0,i}\}_i$, with 1000 warm-up Adam steps followed by 4 steps before each update. The forward PF-ODE is solved with one Euler step between consecutive quadrature nodes, and DB solves each reverse PF-ODE with 7 Euler steps.

H.2 Experiments on the CelebA dataset

The pretrained model is a U-Net with base width 128, channel multipliers $(1, 2, 2, 2)$, two residual blocks per resolution, and self-attention at 16×16. It uses the preconditioning, log-normal noise distribution ($P_{\text{mean}} = -1.2$, $P_{\text{std}} = 1.2$), and $\sigma \in [0.002, 80]$ of Karras et al. (2022), and is trained on 64×64 images for 200k steps with AdamW (learning rate 2×10^{-4}, batch size 64, EMA decay 0.9999). The Euler and Heun baselines use the $\rho = 7$ schedule of Karras et al. (2022). For distillation, the integral over noise levels is evaluated using the trapezoidal rule on 32 nodes of the same schedule over $\sigma \in [0.002, 5]$. The forward PF-ODE of SDI and PFD is solved with one Euler step between consecutive nodes. $\boldsymbol{\epsilon}_{\boldsymbol{\psi},\tau}$ is initialized from the pretrained model and trained with Adam (learning rate 10^{-5}) for 4 steps on batches of 32 before each update. Particles are initialized from $\mathcal{N}(\mathbf{0}, \boldsymbol{I})$ and optimized for 500 steps with SGD (learning rate 0.1), using 128 particles per task for the qualitative results in Appendices I.1–I.4. For the quantitative evaluation of image generation, we instead use 2.5k particles per run, and $\boldsymbol{\epsilon}_{\boldsymbol{\psi},\tau}$ takes 8 steps on batches of 128. For vector graphics, each $\boldsymbol{\theta}$ holds 256 open paths of 1–3 cubic Bézier segments with RGBA stroke colors, rendered at 128×128 over a gray background. The render is downsampled to 64×64 with antialiased bilinear interpolation to give $\boldsymbol{x}_0$. The control points, stroke colors, and stroke widths (clamped to $[1, 4]$ pixels) are optimized with Adam at learning rates 2.0, 0.01, and 0.1.

H.3 Text-to-3D Generation

All methods share the same scene representation, renderer, losses, and optimizer. The assets may therefore differ from those reported in the original works, which tune these choices per method. The characteristic effects nonetheless carry over, with SDS oversmoothing and SDI lacking fine-grained detail. The scene is an implicit volume with an Instant-NGP (Müller et al., 2022) hash-grid encoding (16 levels, 2 features per level, a 2^{19} table, and resolutions from 16 to 4096), a softplus density initialized with the blob of Lin et al. (2023), and a neural environment-map background. It is rendered at 512×512 with 512 samples per ray, from random cameras with distance in $[1, 1.5]$, field of view in $[40°, 70°]$, and elevation in $[-10°, 45°]$. We add z-variance regularization (Zhu et al., 2024) with weight 1 and optimize for 5k steps with AdamW ($\beta = (0.9, 0.99)$) in mixed precision, using learning rates of 0.01 for the hash grid and 10^{-3} for the density, color, and background networks. SDI, VSD, and PFD use Stable Diffusion 2.1 base (`stabilityai/stable-diffusion-2-1-base`) with a CFG scale of 7.5. For VSD and PFD, $\boldsymbol{\epsilon}_{\boldsymbol{\psi},\tau}$ is the frozen pretrained model together with a trainable LoRA adapter (Hu et al., 2022), trained with a learning rate of 10^{-4} on the renders of the asset at the current iteration. SDI follows its original implementation, with DDIM inversion under the target prompt and a negative CFG scale of -6.5. SDS with Stable Diffusion at this resolution and a CFG scale of 7.5 collapses to empty space within 500 steps, so we use DeepFloyd IF (`DeepFloyd/IF-I-XL-v1.0`) instead. The integral is evaluated on 10 quadrature nodes placed with the `trailing` timestep spacing of the DDIM scheduler, and the forward PF-ODE is solved with one Euler step between consecutive nodes. A run of 5k steps takes about 1 hour for SDS and about 2 hours for SDI, VSD, and PFD. Evaluation renders 120 views on a circle at distance 1.5 with a 70° field of view.

I Qualitative Results

I.1 Image Generation



(a) SDS

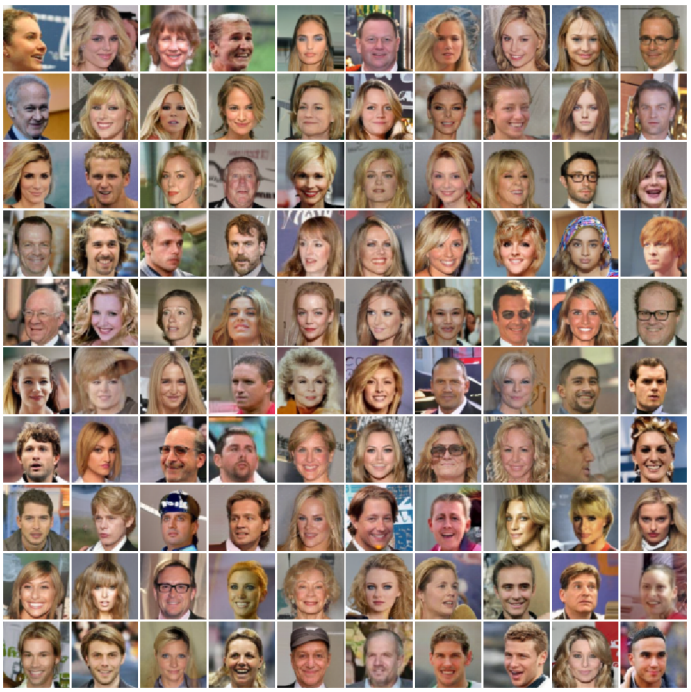


(b) SDI

Figure 6: Uncurated image generation on CelebA, showing $\boldsymbol{x}_0 = \boldsymbol{\theta}$ for 100 of the 128 particles optimized per method.



(c) VSD



(d) PFD

Figure 6: Uncurated image generation on CelebA, showing $\boldsymbol{x}_0 = \boldsymbol{\theta}$ for 100 of the 128 particles optimized per method (continued).

I.2 SYMMETRIC IMAGE GENERATION

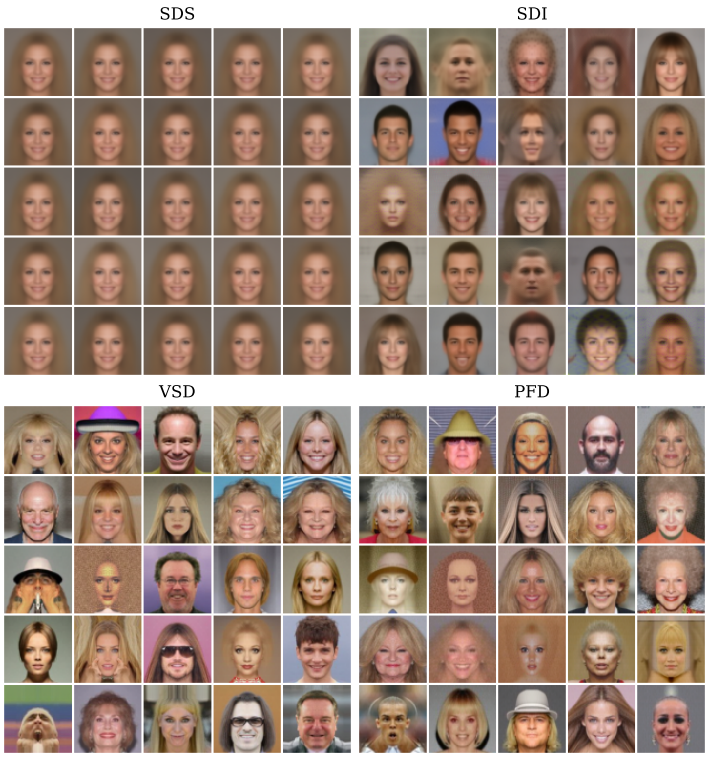


Figure 7: Uncurated symmetric image generation on CelebA, showing $\boldsymbol{x}_0 = [\boldsymbol{\theta}, \text{flip}(\boldsymbol{\theta})]$ for 25 of the 128 particles optimized per method.

I.3 VECTOR GRAPHICS GENERATION

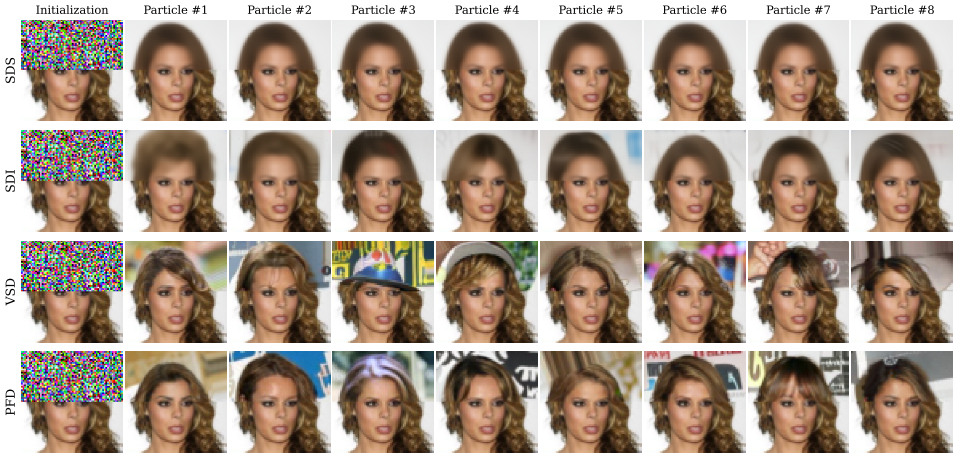


Figure 8: Uncurated vector graphics generation on CelebA, showing $\boldsymbol{x}_0 = \boldsymbol{g}(\boldsymbol{\theta})$, the rasterized render, for 25 of the 128 particles optimized per method.

I.4 IMAGE INPAINTING



(a) Left half masked, right half observed.



(b) Top half masked, bottom half observed.



(c) Upper-right triangle masked, lower-left observed.

Figure 9: Image inpainting on CelebA. In each panel, the first column shows the initialization, with noise in the masked region, and the remaining columns show $\boldsymbol{x}_0 = \boldsymbol{M} \odot \boldsymbol{\theta} + (\mathbf{1} - \boldsymbol{M}) \odot \boldsymbol{x}^{\text{obs}}$ for 8 uncurated particles of the 128 optimized per method.

I.5 TEXT-TO-3D GENERATION



Figure 10: Text-to-3D generation with SDS, SDI, VSD, and PFD.

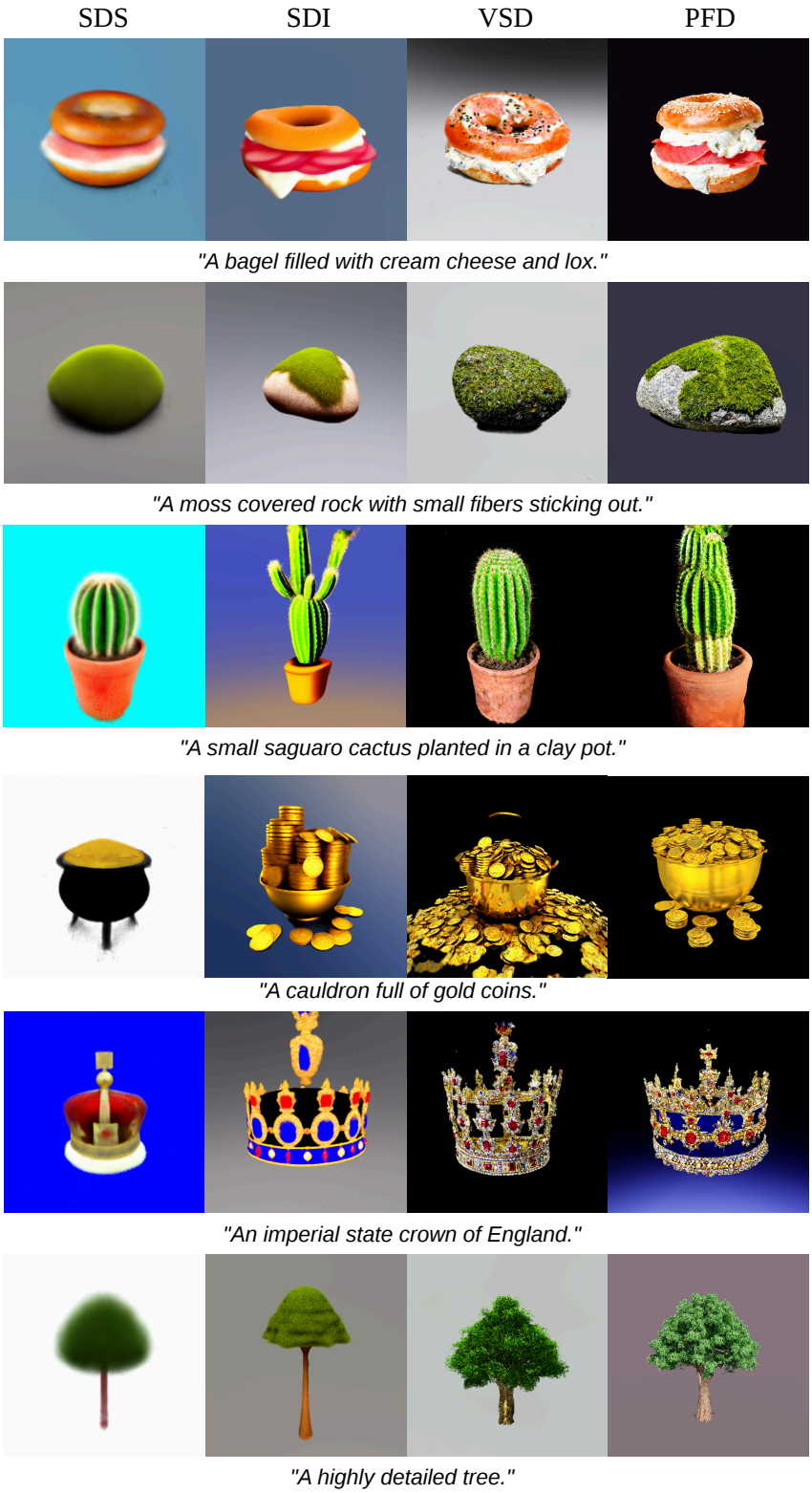


Figure 11: Text-to-3D generation with SDS, SDI, VSD, and PFD (continued).

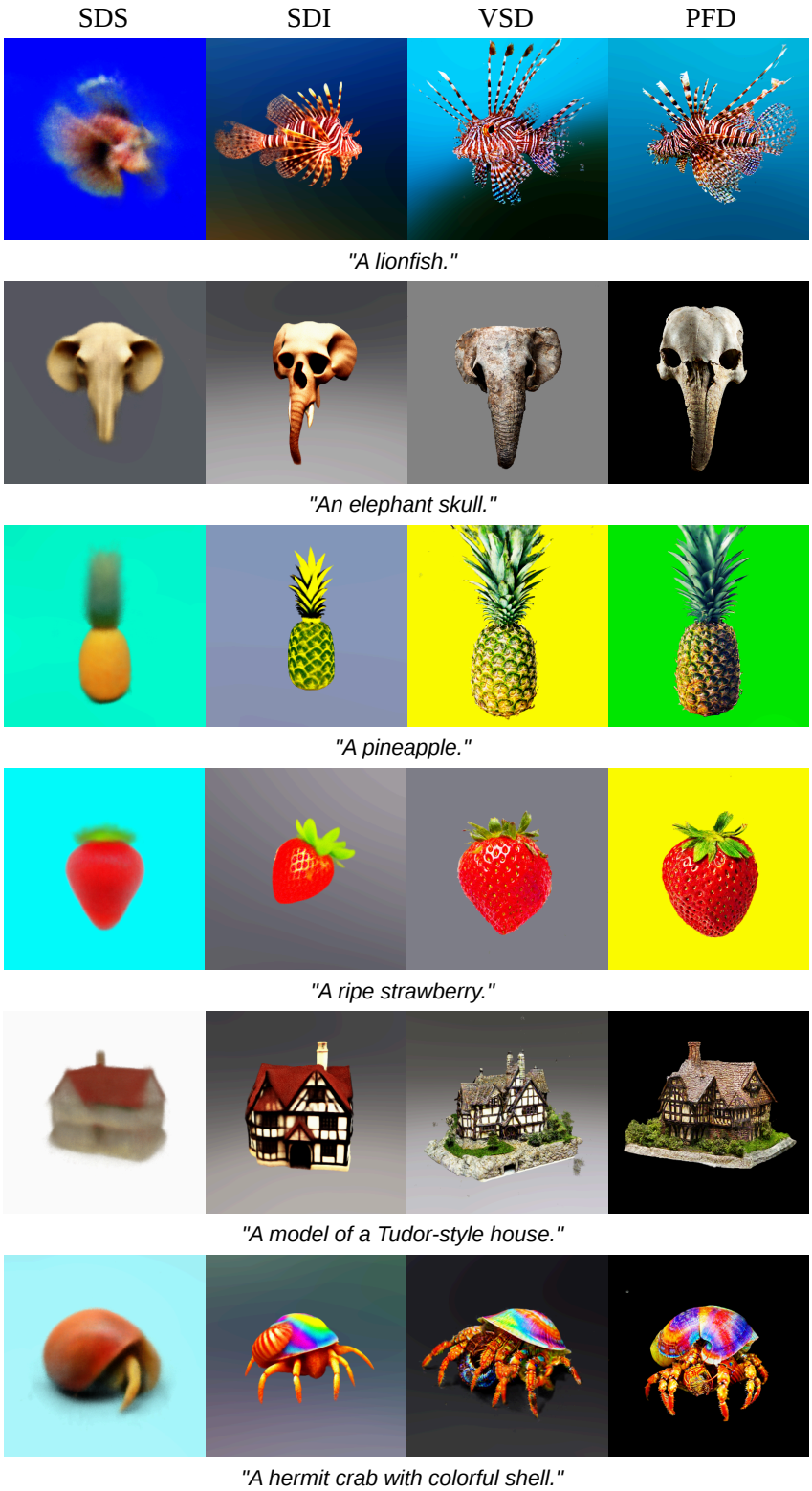


Figure 12: Text-to-3D generation with SDS, SDI, VSD, and PFD (continued).