Diffusion-based Annealed Boltzmann Generators : benefits, pitfalls and hopes

Anonymous authors
Paper under double-blind review

Abstract

Sampling configurations at thermodynamic equilibrium is a central challenge in statistical physics. Boltzmann Generators (BGs) address this problem by pairing a generative model with a Monte Carlo (MC) correction scheme, yielding asymptotically consistent samples from an unnormalized target density. However, most existing BGs rely on classic MC mechanisms such as importance sampling, which (i) impose strong constraints on the backbone model (typically requiring exact and efficient likelihood evaluation) and (ii) suffer from severe scalability issues in high-dimensional, multi-modal settings. This work investigates BGs built around annealed Monte Carlo (aMC) schemes, which mitigate the limitations of classic MC by bridging a simple reference distribution to the target through a sequence of intermediate densities. In this context, diffusion models (DMs) are particularly appealing backbones: they are powerful generative models and naturally induce density paths that have been leveraged in prior aMC-based methods. We provide an empirical meta-analysis of this DM-based aMC-BG design choice on controlled yet challenging synthetic benchmarks based on multi-modal Gaussian mixtures, varying inter-mode separation, number of modes, and dimensionality. To disentangle learning effects from inference effects, we first study an idealized setting in which the DM is perfectly learned, and then turn to realistic settings where the DM is trained from data. Even in the idealized regime, we find that standard aMC integrations of DMs that rely only on first-order stochastic denoising kernels systematically fail in the proposed scenarios. In contrast, incorporating second-order denoising kernels can substantially improve performance when the required covariance information is available. Motivated by this gap, we propose an alternative aMC integration based on deterministic first-order transport maps derived from DMs; empirically, this approach consistently outperforms its stochastic first-order counterpart, albeit at increased computational cost. Overall, while results in the perfect-learning regime suggest that exploiting DM-induced dynamics within aMC is a promising route to building effective BGs, our experiments with learned DMs show that DM–aMC combinations still struggle to produce accurate BGs in practice. We attribute this limitation primarily to inaccuracies in DM log-density estimation.

1 Introduction

Sampling configurations from the Boltzmann distribution of a system $\pi(x) \propto \exp(-\mathcal{E}(x))$, where $\mathcal{E}(x)$ denotes the potential energy of configuration x, is a foundational and long-standing challenge. Reliable access to samples from π underpins the estimation of many key observables which, in turn, govern macroscopic behavior. Hence, efficient Boltzmann sampling is central to a broad range of applications, from characterizing biomolecular function and accelerating drug discovery to materials design and the study of complex statistical-physics models (Liu, 2001; Krauth, 2006; Stoltz et al., 2010; Ohno et al., 2018; Frenkel & Smit, 2023).

The core difficulty of sampling stems from the geometry of realistic energy landscapes. In many practical settings, the energy $\mathcal{E}$ is high-dimensional, non-smooth, and highly rugged, with numerous metastable basins (referred to as "modes") separated by high barriers. This structure severely challenges classical simulation-based approaches such as Molecular Dynamics (MD) and Markov Chain Monte Carlo (MCMC), whose generated samples follow dynamics prone to trapping in local minima, thus requiring a computationally prohibitive number of successive steps to mix across modes. The resulting samples are strongly correlated, leading to large statistical inefficiencies.

Boltzmann Generators (BGs) (Noé et al., 2019) address this bottleneck by amortizing sampling cost through training a generative model p^θ to approximate π, followed by a correction step that turns proposals from p^θ into samples from the target π. Modern BGs predominantly rely on normalizing flows (NFs), either discrete (DNFs) (Rezende & Mohamed, 2015; Papamakarios et al., 2021) or the more expressive continuous variant (CNFs) (Chen et al., 2018; Grathwohl et al., 2019), because they support efficient sampling and (in principle) tractable density evaluations. For NFs, the natural correction mechanism is to embed proposals into Monte Carlo (MC) schemes, most prominently Importance Sampling (IS) (Müller et al., 2019; Noé et al., 2019; Köhler et al., 2020; Klein et al., 2023; Klein & Noé, 2024) and MCMC (Albergo et al., 2019; Gabrié et al., 2022; Del Debbio et al., 2022; Brofos et al., 2022; Samsonov et al., 2022; Cabezas et al., 2024). However, these strategies are highly sensitive to the overlap between p^θ and π (Agapiou et al., 2017; Grenioux et al., 2023): in high dimension or for highly multi-modal targets, even small modeling errors can yield extremely poor correction capabilities. Moreover, in the CNF setting, evaluating $p^\theta(x)$ accurately is itself expensive, as it requires solving a neural ODE.

Recently introduced Diffusion Models (DMs) (Sohl-Dickstein et al., 2015; Ho et al., 2020; Song et al., 2021) are generative models that have achieved state-of-the-art performance across many data modalities (Kong et al., 2021; Ho et al., 2022; Karras et al., 2024; Abramson et al., 2024), and thus provide a natural alternative to NFs as the backbone of Boltzmann generators. We review DMs in detail in Section 2.1. Their core principle is to learn how to remove noise from corrupted samples; training across many noise levels yields a sequential generation procedure that maps pure noise to structured data. While DMs often produce higher-fidelity samples than NFs on complex distributions, their inference mechanism does not integrate directly into classical BG pipelines, most notably because their likelihood is typically not available in a tractable form.

This work reviews and extends approaches that turn DMs into BGs by leveraging annealed Monte Carlo (aMC) methods, introduced in Section 2.2 as refinements of classical MC schemes. The key idea of aMC is to replace a hard sampling problem by a sequence of easier ones, relying on a user-defined path of intermediate densities that bridges a simple base distribution to the target π. While many such paths are possible, several recent works have shown that DMs suggest a particularly natural construction; we unify and review these strategies in Section 3. Our overarching objective is to address the following question:

How can Diffusion Models yield accurate and efficient Boltzmann Generators?

To explore this question, we examine two complementary experimental regimes:

- **(A) Idealized regime:** we assume that the DM is perfectly learned, thus isolating the statistical inference errors induced by aMC from errors due to imperfect training;
- **(B) Realistic regime:** the DM is trained from biased data, reflecting practical settings.

Our main contributions are the following:

- We present a unified review of existing approaches that integrate Diffusion Models into annealed Monte Carlo to build Boltzmann Generators. These methods exploit the sequence of marginal distributions induced by the DM's denoising process as intermediate densities in aMC. In idealized regime **(A)**, we show that such DM-informed constructions consistently outperform traditional aMC designs.

- We further analyze strategies that leverage the conditional structure of the denoising process, which is naturally available from DMs. In practice, this is achieved through Gaussian approximations of the conditional distributions between consecutive noise levels. We distinguish *first-order* approximations, which match only the conditional mean, from *second-order* approximations, which also incorporate covariance information. In idealized setting **(A)**, we find that first-order approximations offer no improvement over a naive, correlation-free baseline (i.e., using marginal densities alone), despite additional access to exact knowledge of conditional means, whereas second-order approximations yield substantial performance gains.

- We propose a complementary alternative to Gaussian approximations by introducing deterministic transport maps. Importantly, these maps integrate seamlessly into the aMC framework and require only access to the previously mentioned conditional mean. In idealized regime **(A)**, this deterministic approach achieves performance comparable to second-order stochastic methods, at the cost of a small computational overhead, but without requiring covariance estimates.

- In realistic regime **(B)**, where all DM's components are learned from data, we observe a significant performance degradation across all DM-based aMC-BG methods compared to idealized regime **(A)**. Our empirical results indicate that this gap could primarily be due to inaccuracies the approximation of intermediate DM's densities.

Although BGs are often benchmarked on molecular systems, we instead focus on controlled yet challenging Gaussian mixture distributions. These widely used targets enable systematic comparison under precisely controlled levels of difficulty (Grenioux et al., 2025; Noble et al., 2025), and crucially allow exact computation of the quantities required in idealized setting **(A)**.

Multi-modal target distributions under consideration

We consider: *(i)* the bimodal distribution of Grenioux et al. (2025), denoted *TwoModes*, which allows one to control both the system's dimensionality, denoted dim, and the separation between imbalanced modes through a parameter $a > 0$ (larger a implies a larger gap); and *(ii)* the multi-modal target of (Noble et al., 2025, Appendix H.1), denoted *ManyModes*, which features a variable number of modes with non-uniform weights. Formal definitions are recalled in Appendix D.1.

For each target family, we select three representative "edge-case" configurations that combine high dimensionality with strong multi-modality, and are therefore particularly challenging. For *TwoModes*, we consider: close modes in high dimension ($a = 1.0$, dim $= 128$), distant modes in low dimension ($a = 10.0$, dim $= 16$), and an intermediate case ($a = 5.0$, dim $= 64$). For *ManyModes*, we use 16, 32, and 64 modes with dimension fixed to 32. To improve numerical stability and avoid target-specific hyperparameter tuning, all targets are standardized to have zero mean and unit covariance. We report Sliced Wasserstein Distance (Bonneel et al., 2015) as the primary metric in the main paper; additional metrics and ablations are provided in Appendix D.3.

Notation. For any measurable space $(\mathrm{X}, \mathcal{X})$, we denote by $\mathcal{P}(\mathrm{X})$ the space of probability measures defined on $(\mathrm{X}, \mathcal{X})$. Unless specified, if X is a topological space, then $\mathcal{X}$ is defined as the Borel σ-field of X. For simplicity, we use the same notation to refer both to a probability distribution and its density wrt the Lebesgue measure when it is defined. In our paper, π^{base} denotes a simple distribution that is easy to sample from (for instance, Gaussian), and is referred to as the "base" distribution. We denote $\mathrm{N}(\mu, \Sigma)$ with $\mu \in \mathbb{R}^d$ and $\Sigma \in \mathrm{S}_d^{++}$ the multivariate Gaussian distribution with mean μ and covariance Σ. For any Markov kernel $Q : \mathcal{B}(\mathbb{R}^d) \times \mathbb{R}^d \to [0, 1]$, we denote its conditional density $q(y|x) = Q(\mathrm{d}y, x)/\mathrm{d}y$ for any $(x, y) \in \mathbb{R}^d \times \mathbb{R}^d$.

Moreover, for any probability distribution $\mu \in \mathcal{P}(\mathbb{R}^d)$, we denote by $\mu Q \in \mathcal{P}(\mathbb{R}^d)$ the distribution obtained by applying the kernel Q to μ, defined by

$$(\mu Q)(\mathrm{d}y) = \int_{\mathbb{R}^d} Q(\mathrm{d}y, x)\mathrm{d}\mu(x) \ .$$

For ease of reading, we may use the same notation for $Q(y, x)$ and $q(y|x)$ throughout the paper. For any C^1-diffeomorphism $T : \mathbb{R}^d \to \mathbb{R}^d$, we denote by $J_{\mathrm{T}}(x)$ the Jacobian matrix of T evaluated at x, and by $\mathrm{T}_{\#}\mu$ the pushforward of the distribution μ by T. Hence, if $X \sim \mu$, then $\mathrm{T}(X) \sim \mathrm{T}_{\#}\mu$. By the change-of-variable formula, the density of $\mathrm{T}_{\#}\mu$ wrt the Lebesgue measure is given by

$$\mathrm{T}_{\#}\mu(x) = \mu(\mathrm{T}^{-1}(x)) \left|\det J_{\mathrm{T}^{-1}}(x)\right| \ . \tag{1}$$

2 Background

Before detailing existing DM-based aMC-BG methods (Section 3) and presenting our deterministic version (Section 4), we first review the key ingredients that underpin these approaches: diffusion models (Section 2.1) and annealed sampling techniques (Section 2.2). Throughout this section, for both generative and sampling frameworks, π and π^{base} will respectively refer to the target and the base distributions.

2.1 Diffusion models

Forward process. The stochastic "noising" process of DMs that gradually corrupts the data with increasing Gaussian noise is described by a linear SDE of the form

$$\mathrm{d}X_t = f(t)X_t\mathrm{d}t + g(t)\mathrm{d}W_t, \quad X_0 \sim \pi, \quad t \in [0, T] \ , \tag{2}$$

where $(W_t)_{t\geq 0}$ is a standard Brownian motion, and $f : [0, T] \to \mathbb{R}$ and $g : [0, T] \to (0, \infty)$ are given schedule functions. Marginally, this forward diffusion process can be explicitly defined by

$$X_t \overset{\mathrm{d}}{=} S(t)X_0 + S(t)\sigma(t)Z, \quad X_0 \sim \pi, \quad Z \sim \mathrm{N}(0, \mathrm{I}_d) \ , \tag{3}$$

where $S(t) = \exp(\int_0^t f(u)\mathrm{d}u)$ and $\sigma^2(t) = \int_0^t g^2(u)/S^2(u)\mathrm{d}u$. As a result, the marginal density of X_t, denoted by p_t, is a convolution of π with a Gaussian kernel that writes as

$$p_t(x) = \int_{\mathbb{R}^d} \mathrm{N}(x; S(t)x_0, S(t)^2\sigma^2(t)\mathrm{I}_d)\mathrm{d}\pi(x_0) \ . \tag{4}$$

With an appropriate choice of schedules f and g (or equivalently, S and σ), the forward process interpolates between $p_0 = \pi$ and $p_T = \pi^{\mathrm{base}}$, where π^{base} is a Gaussian distribution independent of π. We refer to Song et al. (2021) and Karras et al. (2022) for a detailed presentation of commonly chosen noising schemes. In Appendix B, we detail computations related to the widely used Variance Preserving (VP) and Variance Exploding (VE) settings. In practice, the integral in (4) generally cannot be computed in closed form, rendering the marginal density p_t intractable for an arbitrary target distribution π.

Backward process. To generate new data, the idea is to reverse time in SDE (2) so as to denoise samples from π^{base} into samples from π. Under mild regularity conditions on f, g, and π, it can be shown (Anderson, 1982) that the reverse-time dynamics of the noising SDE is itself governed by another SDE, commonly referred to as the reverse-time or denoising SDE

$$\mathrm{d}X_t = \left[f(t)X_t - g^2(t)\nabla \log p_t(X_t)\right]\mathrm{d}t + g(t)\mathrm{d}\tilde{B}_t, \quad X_T \sim \pi^{\mathrm{base}} \ , \tag{5}$$

where $(\tilde{B}_t)_{t\geq 0}$ is a reverse-time standard Brownian motion. Interestingly, the stochastic process induced by the denoising SDE has the same marginal distributions $(p_t)_{t\in[0,T]}$ as the stochastic process induced by its deterministic counterpart, called the probability flow ODE (PF-ODE) (Song et al., 2021)

$$\mathrm{d}X_t = \left[f(t)X_t - \frac{g^2(t)}{2}\nabla \log p_t(X_t)\right]\mathrm{d}t, \quad X_T \sim \pi^{\mathrm{base}} \ . \tag{6}$$

Thus, to obtain samples from π at inference, one needs to either solve the SDE (5) or the ODE (6) backward in time (*i.e.*, from $t = T$ to $t = 0$), starting from noise samples drawn from π^{base}. Below, we detail the denoising transition kernels and transport maps associated with approximate numerical solvers for, respectively, the SDE (5) and the ODE (6).

Stochastic transition kernels. For $0 \leq s < t \leq T$, the conditional distribution of X_t given $X_s = x_s$ is a tractable Gaussian distribution $q_{t|s}(\cdot|x_s)$, called *noising* transition kernel, that writes as

$$q_{t|s}(\cdot \mid x_s) = \mathrm{N}\left(\alpha_{t|s}\, x_s, \sigma^2_{t|s}\, \mathrm{I}_d\right) , \tag{7}$$

where $\alpha_{t|s} = S(t)/S(s)$ and $\sigma^2_{t|s} = S^2(t)[\sigma^2(t) - \sigma^2(s)]$. In contrast, the conditional distribution of X_s given $X_t = x_t$ induced by the denoising SDE (5), denoted $q_{s|t}(\cdot|x_t)$ and called *denoising* transition kernel, does not have a closed-form expression in general. and is usually approximated by a Gaussian distribution.

A natural way to approximate the denoising transition kernel $q_{s|t}$ is to numerically solve the denoising SDE (5) using integration schemes such as Euler-Maruyama (EM). This yields the approximate kernel

$$q^{\mathrm{EM}}_{s|t}(\cdot|x_t) = \mathrm{N}\left(x_t + (t-s)\{-f(t)x_t + g^2(t)\nabla \log p_t(x_t)\}, g^2(t)(t-s)\mathrm{I}_d\right) , \tag{8}$$

which requires access only to the score function $\nabla \log p_t$. More advanced schemes such as Exponential Integration (EI) can be employed for improved accuracy which is the approach adopted in this work (see Appendix B). These methods are coined "first-order" because they rely solely on the score.

In contrast to first-order methods, "second-order" approximations of the denoising kernel $q_{s|t}(\cdot|x_t)$ require access to the Hessian $\nabla^2 \log p_t(x_t)$ in addition to the score $\nabla \log p_t(x_t)$. A natural approach is to approximate $q_{s|t}$ with a Gaussian distribution whose mean and covariance are given by a second-order expansion of Tweedie's formula (Grenioux et al., 2024, Appendix A, Lemma 4)

$$\begin{aligned} m_{s|t}(x_t) &= \mathbb{E}[X_s|X_t = x_t] = \frac{x_t + \sigma^2_{t|s}\nabla \log p_t(x_t)}{\alpha_{t|s}} , \\ \Sigma_{s|t}(x_t) &= \mathrm{Cov}[X_s|X_t = x_t] = \sigma^2_{t|s}\frac{\mathrm{I}_d + \sigma^2_{t|s}\nabla^2 \log p_t(x_t)}{\alpha^2_{t|s}} . \end{aligned} \tag{9}$$

This leads to the second-order Gaussian approximation

$$q^{\mathrm{DDPM}}_{s|t}(\cdot|x_t) = \mathrm{N}(m_{s|t}(x_t), \Sigma_{s|t}(x_t)) . \tag{10}$$

We refer to this second-order Gaussian approximation as the Skip-Step Denoising Diffusion Probabilistic Models (DDPM) kernel, following the terminology introduced by Ou et al. (2025).

Deterministic transition maps. In the case of the PF-ODE (6), $q_{s|t}$ degenerates to a Dirac mass, *i.e.*, $X_s = \mathrm{T}_{s|t}(x_t)$ where $\mathrm{T}_{s|t} : \mathbb{R}^d \to \mathbb{R}^d$ is the deterministic map that solves the ODE (6) backward in time on $[s, t]$. In practice, $\mathrm{T}_{s|t}$ is intractable too, but may be approximated via first-order integration methods. For instance, using the Euler scheme leads to

$$\mathrm{T}^{\mathrm{EM}}_{s|t}(x_t) = x_t - f(t)(t-s)x_t + \frac{g^2(t)}{2}(t-s)\nabla \log p_t(x_t) . \tag{11}$$

Similarly to the stochastic setting, EI versions of such transition maps can be derived to reduce discretization error, see Appendix B for more details.

Training DMs. In practice, the score functions $\{\nabla \log p_t\}_{t\in[0,T]}$ and, for second-order methods, the corresponding Hessians $\{\nabla^2 \log p_t\}_{t\in[0,T]}$, are not available in closed form for general target distributions and must therefore be estimated. As a result, data generation relies on approximate dynamics: first, the SDE (5) or ODE (6) is approximated through estimated scores (yielding an *estimation error*); second, these approximate dynamics are numerically solved using the tools described above (yielding a *discretization error*).

Score functions are typically learned from data using score-matching approaches (Hyvärinen, 2005; Vincent, 2011; Song et al., 2021; Bortoli et al., 2024). While Hessians can in principle be obtained by differentiating the learned score network, this is computationally prohibitive in practice. Early methods therefore relied on scalar, state-independent approximations (Ho et al., 2020), whereas more recent works have proposed more accurate diagonal or full-matrix approximations—possibly state-dependent—learned via dedicated objectives using a pre-trained score model (Nichol & Dhariwal, 2021; Bao et al., 2022b;a). A recent overview is provided in Ou et al. (2025).

Another line of research aims at rather approximating the log-densities $\{\log p_t\}_{t\in[0,T]}$ with neural networks, and then taking the derivative with respect to the input to obtain score or Hessian approximations. Various related objectives have been recently designed, either based on maximum likelihood (Gao et al., 2021; Zhang et al., 2023; Zhu et al., 2024; Noble et al., 2025), consistency via Fokker-Planck equation (Shi et al., 2024; Plainer et al., 2025), consistency via Bayes's rule (He et al., 2025) or multi-label classification (Yadin et al., 2024). In practice, the dominant strategy remains the score matching approach, which indirectly approximates the DM log-densities by training a neural network to match their gradient (Song & Kingma, 2021; Salimans & Ho, 2021; Du et al., 2023; Phillips et al., 2024; Thornton et al., 2025) or their time derivative (Guth et al., 2025b; Yu et al., 2025) : for the latter, we will refer to it as "time" score matching.

Diffusion model under consideration

In all experiments presented below, the noising diffusion process is chosen to be the linear Variance Preserving (VP) diffusion path (Song et al., 2021) with hyperparameters $(\beta_{\min}, \beta_{\max}, T) = (0.1, 20, 1)$, whose exact noising kernel (7) is computed in Lemma 14 (Appendix B.2). When using stochastic denoising kernels, we consider the EI scheme given in Lemma 15 (Appendix B.2) for first-order approaches and the DDPM scheme given in (9) for second-order approaches; when using noising and denoising transport maps, we consider the EI-based ODE integration schemes detailed in Lemmas 16 and 17 (Appendix B.2). Moreover, we set the time discretization $\{t_k\}_{k=0}^K \subset [0,T]$ so as to be constant in log-SNR increments (see Appendix A.2 for more details). We will refer to the induced sequence of densities $\{p_{t_k}\}_{k=0}^{K-1}$ (also denoted $\{p_k\}_{k=0}^{K-1}$), marginally defined by (4), as the "diffusion" path.

2.2 Standard Monte Carlo & Annealed sampling

This section presents the Monte Carlo tools that are central to all BG methods presented below. We recall that the original purpose of these methods is to generate samples from π, with only access to its energy function $\mathcal{E}$ up to an additive constant. We begin by reviewing classic techniques, which serve as foundation for the aMC methods introduced afterwards.

Importance Sampling. Importance Sampling (IS) is a fundamental Monte Carlo method that approximates expectations taken under π using samples drawn from a proposal distribution ρ whose density is tractable. Assuming that $\mathrm{Supp}(\rho) \subset \mathrm{Supp}(\pi)$, any π-integrable function ϕ satisfies

$$\mathbb{E}_\pi[\phi(X)] = \mathbb{E}_\rho\left[w(X)\phi(X)\right], \quad \text{where } w(x) = \frac{\pi(x)}{\rho(x)} \text{ is the importance weight.}$$

In practice, this means that sampling from π via IS reduces to (i) sample N particles $\{x^i\}_{i=1}^N$ from ρ and (ii) reweight them using the importance weights $\{w(x^i)\}_{i=1}^N$ [1]. Although IS is simple to implement, its accuracy critically depends on how well ρ matches π. In particular, the variance of the importance weights can grow rapidly, potentially exponentially with the dimension, when the mismatch is large (Agapiou et al., 2017).

Markov Chain Monte Carlo. Markov Chain Monte Carlo (MCMC) methods are designed to simulate a Markov chain whose stationary distribution is π, hence generating asymptotically accurate samples.

[1] When the density π is only known up to a normalizing constant, as it is often the case in practice, one turns to the *self-normalized* weights $\bar{w}(x^i) = w(x^i)/\sum_{j=1}^N w(x^j)$, which however leads to a biased estimator.

MCMC methods typically construct their transition mechanism using a proposal distribution $q(y|x)$, which suggests a new state y from the current state x. The Metropolis-Hastings (MH) algorithm then corrects this proposal via an acceptance-rejection step to ensure that the chain targets the desired distribution π. Specifically, given x, the proposed $y \sim q(\cdot|x)$ is accepted with probability

$$\alpha(x, y) = \min\left(1, \frac{q(x|y)\pi(y)}{q(y|x)\pi(x)}\right) = \min\left(1, \frac{q(x|y)\exp(-\mathcal{E}(y))}{q(y|x)\exp(-\mathcal{E}(x))}\right) , \tag{12}$$

otherwise the new state is set as x. Note that the MH algorithm can be extended to the deterministic case, when $q(\cdot|x) = \delta_{\mathrm{T}(x)}$ for a diffeomorphism $\mathrm{T} : \mathbb{R}^d \to \mathbb{R}^d$ that is required to be involutive, *i.e.*, $\mathrm{T} \circ \mathrm{T} = \mathrm{I}_d$. In this case, the acceptance probability only depends on the previous state x and writes

$$\alpha(x) = \min\left(1, \frac{\mathrm{T}_{\#}\pi(x)}{\pi(x)}\right) = \min\left(1, \frac{\exp(-\mathcal{E}(\mathrm{T}(x))\,|\det J_T(x)|}{\exp(-\mathcal{E}(x))}\right) . \tag{13}$$

This deterministic formulation encompasses the popular Hamiltonian Monte Carlo (HMC) algorithm (Neal, 2012). As with IS, the performance of such MH-based samplers hinges on the quality of the proposal. For instance, independent proposals scale poorly with dimension (Grenioux et al., 2023), and multi-modal targets pose additional challenges, as proposals must efficiently explore both within and across the modes. Modern MH variants (Metropolis et al., 1953; Duane et al., 1987), including the Metropolis-Adjusted Langevin Algorithm (MALA) (Roberts & Tweedie, 1996), leverage gradient information to improve local mixing but still struggle with global exploration.

While IS and MCMC are fundamental sampling tools, they often fail in high-dimensional or multi-modal settings. *Annealed* sampling specifically addresses this limitation by breaking the original sampling problem into K sampling problems with gradual complexity, by introducing a sequence of distributions $\{p_k\}_{k=0}^K$ that smoothly bridge between a simple base distribution $p_K = \pi^{\text{base}}$ and the target $p_0 = \pi$. We consider such sequence in the rest of this section. By leveraging correlations across this sequence, it is possible to gradually transform samples from π^{base} into samples from π while avoiding the pitfalls of standard MC methods.

Annealed Importance Sampling. Annealed Importance Sampling (AIS) (Neal, 2001) extends classic IS by defining a joint target distribution $\pi_{0:K}$ over a sequence of variables $(x_0, \dots, x_K)$ such that its 0-th marginal is the target distribution π. Similarly, a joint proposal distribution $\rho_{0:K}$ is built such that its K-th marginal is the base distribution π_{prior}. Both of these joint distributions are designed recursively as follows

$$\pi_{0:K}(x_{0:K}) = \pi(x_0)\prod_{k=0}^{K-1} q_{k+1|k}(x_{k+1}|x_k), \quad \rho_{0:K}(x_{0:K}) = \pi^{\text{base}}(x_K)\prod_{k=0}^{K-1} q_{k|k+1}(x_k|x_{k+1}) , \tag{14}$$

where $q_{k+1|k}$ and $q_{k|k+1}$ respectively denote *forward* and *backward* Markov transition kernels. In this case, the importance weights are defined by

$$w^{\text{AIS}}(x_{0:K}) = \frac{\pi_{0:K}(x_{0:K})}{\rho_{0:K}(x_{0:K})} \tag{15}$$

Analogously to IS, sampling from π reduces to (i) sample N trajectories of particles $\{x^i_{0:K}\}_{i=1}^N$ from $\rho_{0:K}$ and (ii) reweight the particles $\{x^i_0\}_{i=1}^N$ with the importance weights $w^{\text{AIS}}(x_{0:K})$ [2]. However, while easier to achieve than classic IS, the efficiency of AIS also depends on how closely $\rho_{0:K}$ matches $\pi_{0:K}$. In particular, if there exists a sequence of bridging distributions $\{p_k\}_{k=0}^K$ (*i.e.*, such that $p_0 = \pi$ and $p_K = \pi^{\text{base}}$) for which the forward and backward kernels satisfy the Bayes rule defined as

$$p_k(x_k)q_{k+1|k}(x_{k+1}|x_k) = p_{k+1}(x_{k+1})q_{k|k+1}(x_k|x_{k+1}), \quad \forall k \in \{0, \dots, K-1\} , \tag{16}$$

then it holds exactly that $\pi_{0:K} = \rho_{0:K}$, *i.e.*, the estimator has minimal variance.

In standard AIS (Neal, 2001), the forward and backward kernels are typically chosen to be identical reversible MCMC kernels with respect to a given density path $\{p_k\}_{k=0}^K$ interpolating π to π^{base}, which simplifies the importance weights given in (15) but violates the Bayes consistency condition (16).

[2] In practice, these weights are also self-normalized as in classic IS.

Sequential Monte Carlo. Sequential Monte Carlo (SMC) methods (Doucet et al., 2001; Del Moral et al., 2006) address a major limitation of AIS, namely weight degeneracy, where importance weights progressively concentrate on a few particles—an effect that is particularly severe in high-dimensional settings. While SMC relies on the same forward and backward kernels as AIS, it introduces intermediate resampling steps that effectively decompose a single long AIS trajectory from p_K to p_0 into two consecutive AIS procedures. Concretely, an initial AIS run propagates particles from p_K to an intermediate distribution p_k for some $k \in \{1, \dots, K-1\}$; particles are then resampled according to their importance weights to obtain a population representative of p_k. A second AIS run, initialized from these resampled particles, subsequently propagates the system from p_k to p_0. This mid-trajectory realignment prevents particle collapse, maintains diversity, and significantly reduces weight degeneracy. The construction naturally extends to multiple resampling points by partitioning the path between π^{base} and π into shorter AIS segments, which substantially reduces the variance of the AIS estimator without increasing the cost of importance-weight evaluations. In practice, SMC methods are often further augmented with MCMC rejuvenation steps at each stage to better align particles with the intermediate distributions, at the expense of additional computational cost.

Replica Exchange. Replica Exchange (RE) (Swendsen & Wang, 1986; Geyer et al., 1991; Hukushima & Nemoto, 1996) is an annealed sampling method that predates AIS and SMC. Unlike these sequential methods, RE correlates the distributions $\{p_k\}_{k=0}^K$ in parallel, rather than through a recursion. The goal is to construct a MCMC algorithm targeting the extended distribution $\bar{\pi}_{0:K}(x_{0:K}) = p_0(x_0)p_1(x_1)\dots p_K(x_K)$. Its transition kernel is composed of two parts: (i) an exploration kernel that independently applies standard MCMC updates to each p_k in parallel, and (ii) a communication kernel that correlates the different marginals. A basic communication move consists of a deterministic "swap" between two consecutive levels k and $k+1$, mapping $(x_0, \dots, x_k, x_{k+1}, \dots, x_K)$ to $(x_0, \dots, x_{k+1}, x_k, \dots, x_K)$. Since this mapping is involutive, it can be used within the Metropolis–Hastings correction to ensure that the joint distribution $\bar{\pi}_{0:K}$ is stationary. The corresponding acceptance rate obtained from (13) is given by

$$\alpha_k^{\text{RE}}(x_{0:K}) = \min\left(1, \frac{p_{k+1}(x_k)p_k(x_{k+1})}{p_k(x_k)p_{k+1}(x_{k+1})}\right). \tag{17}$$

By applying these MH-calibrated swaps in parallel between even or odd pairs of indices in $\{0, \dots, K\}$, one defines the even and odd communication kernels, respectively. These are commonly combined using a uniform mixture to build the full communication kernel. However, recent work suggests that deterministically alternating between even and odd kernels is more effective (Okabe et al., 2001; Syed et al., 2022). We adopt this so-called *non-reversible* strategy in the rest of the paper.

Standard designs of interpolation density paths. A central component of all aMC methods is the design of the interpolation density path. This path is critical to ensure good performance: in AIS and SMC, it governs the overlap between consecutive distributions, which directly affects the variance of the estimators; in RE, the consecutive overlap controls the probability of accepting swap moves between adjacent levels. When only the unnormalized density of π is available, a common choice is the geometric interpolation path (Neal, 2001; Gelman & Meng, 1998), defined for all $x \in \mathbb{R}^d$ by

$$p_k(x) \propto \pi(x)^{\beta_k}\pi^{\text{base}}(x)^{1-\beta_k}, \tag{18}$$

where the annealing schedule $\{\beta_k\}_{k=0}^K$ is decreasing, and satisfies $(\beta_0, \beta_K) = (1, 0)$. We will refer to the collection of unnormalized densities obtained via (18) as the "tempering" path. The major benefit of the tempering paths is their computational efficiency, as they allow for simple evaluations of the scores $\{\nabla \log p_k\}_{k=0}^K$, which are frequently required in MCMC transition kernels, via a linear combination of $\nabla \log \pi$ and $\nabla \log \pi^{\text{base}}$.

However, these paths are usually pathological for multi-modal targets, as they suffer from mass teleportation (also referred to as mode switching), which reflects sudden shifts in probability mass between modes along the interpolation path (Máté & Fleuret, 2023). In practice, such sudden shifts undermine the assumed proximity between bridging densities, leading to instability in aMC. Mitigating this issue usually requires either carefully tuning the annealing schedule $\{\beta_k\}_{k=0}^K$ for each target or using a large number of intermediate levels K, which can incur significant computational cost.

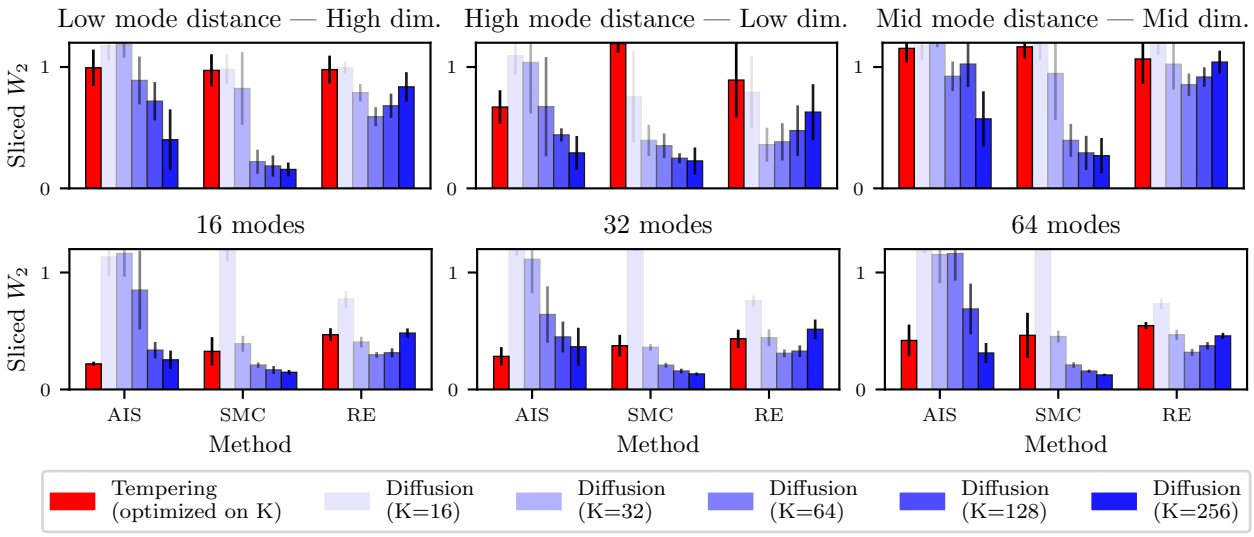

Figure 1: **Sampling results for classic annealed samplers with diffusion (blue) and tempering (red) density paths**, when targeting *TwoModes* **(Top)** and *ManyModes* **(Bottom)** distributions in idealized setting **(A)**. For tempering paths, we display the best-performing result among all values of K. For diffusion paths, we display the results for all values of K : the darker the bar, the higher K. In particular, these configurations do not share the same computational budget. Each result is averaged over 8 runs with 8,192 samples per run. We observe that for almost all target settings and samplers, there exists a number of annealing levels K for which the noising density path outperforms the best tempering baseline.

3 Diffusion-based aMC as a Boltzmann Generator backbone : benefits and pitfalls

Diffusion models are a natural fit for aMC schemes, as they inherently define a sequence of intermediate densities that can be leveraged in sampling algorithms such as AIS, SMC, or RE. In Section 3.1, we show that even a naive integration, simply using the sequence of DM densities as a direct replacement for the classic tempering sequence, can already deliver strong performance, thanks to the favorable properties of the Gaussian convolution paths induced by DMs. In Section 3.2, we review related methodologies, that additionally propose to "enhance" standard aMC tools using DM stochastic transition kernels. However, we demonstrate in Section 3.3 that those designs are fundamentally limited in challenging multi-modal scenarios.

We emphasize that, although the presented methods involve different hyperparameters, we focus our numerical evaluation solely on the effect of the number of annealing levels (defined as $K \in \{16, 32, 64, 128, 256\}$), common to all methods, because it directly controls the overlap between consecutive distributions along the annealing path, a factor highlighted as crucial to the performance of aMC. We leave investigation of automatic tuning methods (Syed et al., 2022; 2025) for future work.

3.1 Of the interest of diffusion-based density paths

As noted by Máté & Fleuret (2023), diffusion paths are typically well conditioned and avoid common pitfalls of tempering paths, such as abrupt mode switching. In particular, DM-induced paths preserve the relative mass of different modes throughout the annealing process, leading to more stable sampling dynamics. This explains why diffusion paths consistently outperform tempering paths in aMC, as illustrated in idealized setting **(A)** by Figure 1. Across all standard aMC methods considered (see Section 2.2), we observe that there exists a number of annealing levels K for which a perfectly learned diffusion path outperforms an optimally tuned tempering path. For AIS and SMC, performance generally improves with increasing K, with best results attained at the largest value tested ($K = 256$). For RE-based samplers, the dependence on K is less monotonic: while larger K improves local overlap and facilitates swaps, it can also hinder long-range communication between levels, leading to degraded performance beyond a certain point. Overall, these results highlight the strength of diffusion over tempering paths, motivating their use when a learned DM is available.

3.2 Review of existing diffusion-based aMC-BGs

Interestingly, DMs provide more than a sequence of intermediate densities: they also grant access to noising and denoising stochastic transition kernels (see, e.g., (7) and (8), (10)), which can be strategically exploited to improve both efficiency and robustness. In this section, we review existing extensions of aMC that leverage this additional structure. These approaches assume access to a DM defined on a discrete time grid $\{t_k\}_{k=0}^K \subset [0, T]$, enabling the additional evaluation of the associated noising kernels $\{q_{k+1|k}\}_{k=0}^{K-1}$ and denoising kernels $\{q_{k|k+1}\}_{k=0}^{K-1}$.

Diffusion-based AIS. DMs have been successfully integrated into AIS frameworks in recent work (Zhang et al., 2024; 2025a). The core idea consists in using the exact noising transition kernels (7) as forward kernels, and *first-order* denoising transition kernels[3], similar to (8), as backward kernels, to respectively define the extended target and proposal distributions, see (14). By doing so, only the score functions are needed, not the log-densities. A key advantage of this approach is that, when the backward kernels match the exact denoising kernels, the forward and backward transitions satisfy the optimal Bayes condition (16), which ensures that the importance weights exhibit minimal variance.

Diffusion-based SMC. The exact same use of DM transition kernels has recently been extended to the SMC setting through the *Particle Denoising Diffusion Sampler* (PDDS) (Phillips et al., 2024). In contrast to AIS, however, the SMC formulation additionally requires the intermediate log-densities, up to normalizing constants, in order to perform resampling.

Diffusion-based RE. In the spirit of PDDS, Zhang et al. (2025b) lately explored the use of DM transition kernels within the RE framework to propose the *Diffusion-based Accelerated Parallel Tempering* (Diff-APT) sampler. In Diff-APT, the traditional RE swaps between adjacent levels are combined with stochastic refinements inherited from those kernels. Given current states x_k and x_{k+1} at levels k and $k+1$, Diff-APT first samples proposal states $y_{k+1} \sim q_{k+1|k}(\cdot|x_k)$ and $y_k \sim q_{k|k+1}(\cdot|x_{k+1})$, where $q_{k+1|k}$ and $q_{k|k+1}$ respectively denote the exact noising (forward) kernel, see (7), and a *first-order* denoising (backward) kernel, see (8) for instance, between times t_k and t_{k+1}. By exploiting the underlying correlation between noise levels, each chain is moved closer to its corresponding target distribution, respectively p_{k+1} and p_k. Then, this stochastic-based swap is calibrated using the MH correction, resulting in the following acceptance probability

$$\alpha_k^{\mathrm{RE}}(x_{0:K}, y_{0:K}) = \min\left(1, \frac{p_k(y_k)p_{k+1}(y_{k+1})q_{k+1|k}(x_{k+1}|y_k)q_{k|k+1}(x_k|y_{k+1})}{p_k(x_k)p_{k+1}(x_{k+1})q_{k+1|k}(y_{k+1}|x_k)q_{k|k+1}(y_k|x_{k+1})}\right), \tag{19}$$

defined for any $(x_{0:K}, y_{0,K}) \in \mathbb{R}^{(K+1)d} \times \mathbb{R}^{(K+1)d}$. Compared to the standard RE acceptance ratio (17), this novel expression features four additional terms, which correspond to symmetric evaluations of forward and backward kernels. As in AIS and SMC, if the forward and backward kernels satisfy the Bayes condition (16) the proposed swap is systematically accepted, *i.e.*, the acceptance probability (19) always equals one.

3.3 Current approaches fail due to first-order approximations of DM stochastic dynamics

Although theoretically well motivated, the existing DM-based aMC-BGs reviewed in Section 3.2 do not yield noticeable improvements over the standard baseline studied in Section 3.1, in idealized setting **(A)** where both log-densities and score functions are assumed to be perfectly known. In practice, this deficiency is expected to be further exacerbated by the additional errors introduced by learning approximations. In Figure 3, we evaluate the aforementioned methods in this perfect-learning regime on target distributions of increasing difficulty and report the resulting errors. The figure shows that this unfavorable behavior persists across all considered targets, corresponding to the different rows. We argue that this systematically poor performance stems from the use of *first-order* denoising kernels as backward kernels within aMC schemes.

[3]Although Zhang et al. (2025a) propose to adjust the covariance of the denoising kernels via additional learning, we still consider this approach as 'first-order' as it does not rely on the Hessian functions $\{\nabla^2 \log p_k\}_{k=0}^K$.

Indeed, if we leverage access to Hessian functions [4] to build *second-order* denoising kernels such as (9), we consistently obtain significant gains over both the baseline and their first-order counterparts. This additional result highlights the value of higher-order information for guiding transitions along the diffusion density path. However, using such second-order kernels in practice requires additional covariance estimations (Ou et al., 2025), which is beyond the scope of most of DM training methods, where only approximations of log-densities and/or scores are available.

4 Exploiting deterministic transitions of DMs in aMC methods : a new hope ?

In this section, we propose investigating the design of a deterministic diffusion-based aMC-BG. We first describe its general principle in Section 4.1 and detail in Section 4.2 how to instantiate it concretely for DMs. We demonstrate that, in idealized setting **(A)**, our method outperforms previous first-order approaches, while being on par with the second-order stochastic ones. Similarly to Section 3.2, we assume throughout this section that we have access to scores and log-densities from a DM associated to a certain time discretization $\{t_k\}_{k=0}^K \subset [0, T]$.

4.1 General methodology

From stochastic to deterministic DM dynamics. To further exploit the potential of aMC sampling methods, we propose to use *deterministic* kernels, by replacing stochastic transition kernels with their deterministic counterparts, which approximate the PF-ODE (6) rather than the denoising SDE (5). Below, we explain how the aMC framework presented in Section 2.2 naturally extends to this setting.

Annealed samplers with deterministic transitions. In this paragraph, we consider K pairs of candidate transport maps, divided between *forward* maps $\{\mathrm{T}_{k+1|k}\}_{k=0}^{K-1}$ and *backward* maps $\{\mathrm{T}_{k|k+1}\}_{k=0}^{K-1}$. Moreover, we assume that **(a)** these maps are C^1-diffeomorphisms, and **(b)** verify the *per-level mutual invertibility* property, defined for any $k \in \{0, \dots, K-1\}$ by

$$\mathrm{T}_{k+1|k} \circ \mathrm{T}_{k|k+1} = \mathrm{T}_{k|k+1} \circ \mathrm{T}_{k+1|k} = \mathrm{Id} \, . \tag{20}$$

To exploit the use of these transport maps into aMC samplers, we simply propose to set the forward Markov kernels $\{q_{k+1|k}\}_{k=0}^{K-1}$ and backward Markov kernels $\{q_{k|k+1}\}_{k=0}^{K-1}$ (used as transition kernels between adjacent levels in aMC methods) as Dirac masses defined for any $k \in \{0, \dots, K-1\}$ by $q_{k+1|k} = \delta_{\mathrm{T}_{k+1|k}}$ and $q_{k|k+1} = \delta_{\mathrm{T}_{k|k+1}}$ respectively.

Adaptation to AIS/SMC instance. Under this setting, the AIS framework boils down to standard IS targeting π with the push-forward of π^{base} through all backward maps as proposal. Using the change-of-variables formula, the AIS weight (15) admits the following deterministic version, solely depending on the state x_0 :

$$w^{\mathrm{AIS}}(x_0) = \frac{\pi(x_0)}{(\mathrm{T}_{0:K})_{\#}\pi^{\mathrm{base}}(x_0)}, \qquad \text{with } \mathrm{T}_{0:K} = \mathrm{T}_{0|1} \circ \mathrm{T}_{1|2} \dots \circ \mathrm{T}_{K-1|K}. \tag{21}$$

Using the chain rule, the determinant of the Jacobian of the full map $\mathrm{T}_{0:K}$ appearing in $(\mathrm{T}_{0:K})_{\#}\pi^{\mathrm{base}}$ (see (1)) can be written as a product of the determinants of Jacobian of the individual maps $\mathrm{T}_{k|k+1}$ for $k \in \{0, K-1\}$.

Adaptation to RE instance. By substituting Markov kernels with Dirac masses, the resulting swap in RE sampling procedure defines a deterministic map on the full extended space

$$\bar{\mathrm{T}}_k(x_{0:K}) = (x_0, \dots, \mathrm{T}_{k|k+1}(x_{k+1}), \mathrm{T}_{k+1|k}(x_k), \dots, x_K) \, ,$$

which is guaranteed to be involutive due to assumption **(b)**. In particular, this property ensures that $\bar{\mathrm{T}}_k$ can effectively be integrated within the Metropolis–Hastings algorithm with deterministic proposal, see (13).

[4] In our experiments, we only exploit the diagonal of the exact Hessians to ensure a good compromise between accuracy and computational efficiency in high dimension.

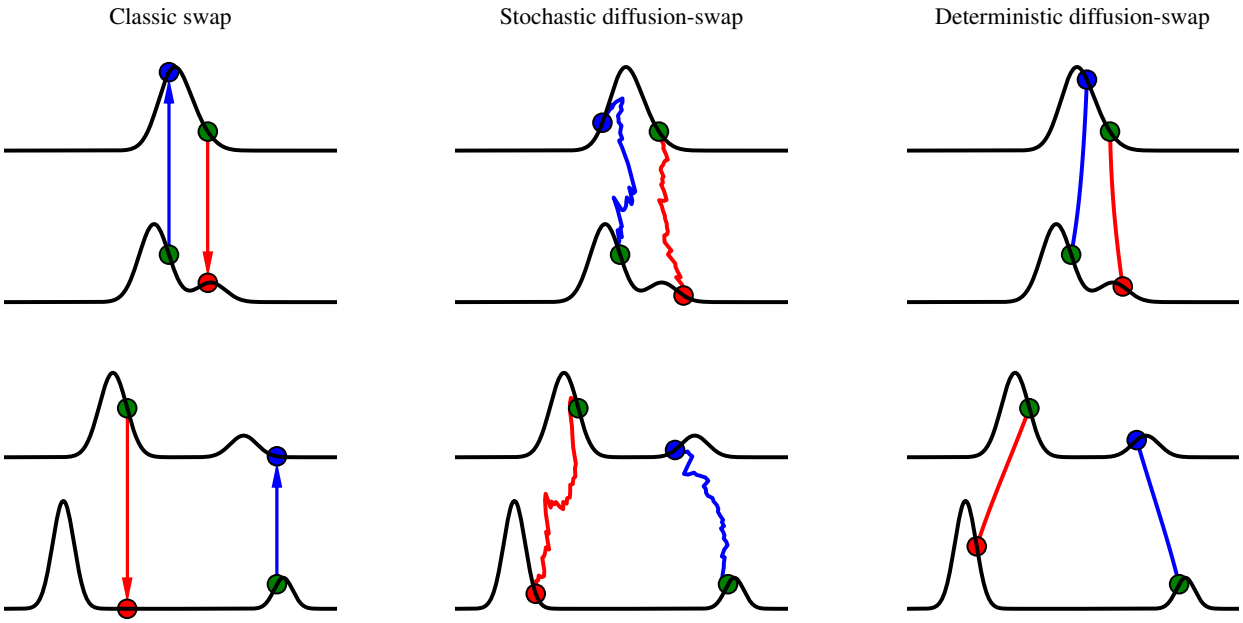

Figure 2: **Different diffusion-based swapping mechanisms for Replica Exchange. (Left)** standard swap scheme, see Section 2.2, where samples are exchanged directly across noise levels without guidance, potentially moving into low-probability regions. **(Middle)** DM-based swaps using forward and backward Markov kernels, as proposed by Zhang et al. (2025b) (coined Diff-APT), see Section 3.2. **(Right)** DM-based swaps using forward and backward transport maps under the deterministic framework introduced in Section 4.1. In each panel, black lines denote noise levels; green dots mark the original samples; blue and red paths indicate forward (low to high noise) and backward (high to low noise) trajectories, respectively; and the swapped samples are shown as the resulting blue and green dots. The DM-based swaps better preserve high-probability regions during exchange, enabling theoretically more effective sampling.

Using the identity $(\bar{\mathrm{T}}_k)_\# \bar{\pi}(x_{0:K}) = p_0(x_0)\dots(\mathrm{T}_{k|k+1})_\# p_{k+1}(x_k)(\mathrm{T}_{k+1|k})_\# p_k(x_{k+1})\dots p_K(x_K)$, we obtain the following acceptance probability:

$$\alpha_k^{\mathrm{RE}}(x_{0:K}) = \min\left(1, \frac{(\mathrm{T}_{k|k+1})_\# p_{k+1}(x_k)(\mathrm{T}_{k+1|k})_\# p_k(x_{k+1})}{p_k(x_k)p_{k+1}(x_{k+1})}\right) . \tag{22}$$

This swapping mechanism is illustrated in Figure 2. Note that setting both $\mathrm{T}_{k|k+1}$ and $\mathrm{T}_{k+1|k}$ as the identity map recovers the standard RE algorithm as a special case.

Effective application to diffusion models. For all aMC methods, the choice of the forward maps $\{\mathrm{T}_{k+1|k}\}_{k=0}^{K-1}$ and backward maps $\{\mathrm{T}_{k|k+1}\}_{k=0}^{K-1}$ is optimal if those verify the deterministic version of the Bayes rule (16) given by

$$(\mathrm{T}_{k+1|k})_\# p_k = p_{k+1}, \quad (\mathrm{T}_{k|k+1})_\# p_{k+1} = p_k, \qquad \forall k \in \{0, \dots, K-1\} .$$

Indeed, in the case of AIS/SMC samplers, satisfying this identity would enable to get zero-variance in the estimator, while this would ensure to maximize the acceptance rate in the RE sampler. Intuitively, this rule reflects the fact that the maps should be chosen so as to perfectly transport particles between adjacent levels to match their target distribution.

The next section discusses two key challenges that arise when implementing these methods in practice using DM's ingredients :

1. *How to design transition maps that verify the invertibility condition* (20) *?*
2. *Given those maps, how to compute efficiently the push-forward densities appearing in* (21) *and* (22) *?*

4.2 The key components needed for efficient implementation

In this section, we first describe how to construct invertible transport maps that approximately solve the probability flow ODE (6). We then present a practical methodology, inspired by residual NFs, for obtaining unbiased estimates of the push-forward density terms appearing in (21) and (22). As made explicit by the change-of-variables formula (1), this approach requires (i) the ability to evaluate the transport maps and (ii) the computation (or unbiased estimation) of their Jacobian determinants. The construction directly extends to the case where the score in the PF-ODE is replaced by an estimate, by simply substituting the estimated score throughout. Our main contributions are summarized in Table 1. In what follows, we focus on two adjacent noise levels k and $k+1$.

Building invertible transport maps. To guarantee that the forward map $\mathrm{T}_{k+1|k}$ and the backward map $\mathrm{T}_{k|k+1}$ are mutually inverse, see (20), we cannot simply rely on *explicit* ODE integrators of the form (11). Indeed, forward and backward maps inherited from such first-order approximations do not, in general, compose to the identity[5]. This motivates us to move towards the class of *implicit* integrators, which are extensively used to simulate Hamiltonian dynamics where trajectory invertibility is often a desirable feature. In particular, we propose to design our transition maps via the *Implicit Midpoint* (IM) integrator, as presented in Proposition 1 for the Euler scheme.

Proposition 1 (IM integrator with Euler scheme)**.** *Let* $\mathrm{T}_{k+1|k} : \mathbb{R}^d \to \mathbb{R}^d$ *and* $\mathrm{T}_{k|k+1} : \mathbb{R}^d \to \mathbb{R}^d$ *be implicitly defined as*

$$\mathrm{T}_{k+1|k} : x_k \mapsto x_k + \delta_k v\left(t_{k+1/2}, \frac{x_k + \mathrm{T}_{k+1|k}(x_k)}{2}\right),$$
$$\mathrm{T}_{k|k+1} : x_{k+1} \mapsto x_{k+1} - \delta_k v\left(t_{k+1/2}, \frac{\mathrm{T}_{k|k+1}(x_{k+1}) + x_{k+1}}{2}\right),$$

where $t_{k+1/2} = (t_k + t_{k+1})/2$, $\delta_k = t_{k+1} - t_k$ *and* $v(t,x) = f(t)x - (g(t)^2/2)\nabla \log p_t(x)$ *is the velocity field of PF-ODE* (6)*. Then, these maps are valid forward and backward integrators of PF-ODE* (6) *on time interval* $[t_k, t_{k+1}]$ *and satisfy the mutual invertibility condition* (20)*.*

In Appendix C, we provide the proof of the above proposition along with its generalization using the Exponential Integration scheme in Appendix B, which offers improved accuracy compared to the Euler scheme when δ_k is relatively large. Although the maps defined in Proposition 1 cannot be evaluated in closed form as they are by nature *implicit*, they can still be approximated in practice using fixed-point iterations as described in Proposition 2, which guarantees convergence of this scheme under certain assumptions detailed below. We refer to Appendix C for the proof of this result as well as Appendix B for its EI generalization.

Assumption 1 (Score smoothness & discretization error)**.** *(a) There exists* $L_k > 0$ *such that* $\nabla \log p_{t_{k+1/2}}$ *is* L_k*-Lipschitz and (b) the step-size* δ_k *is sufficiently small* [6]*, that is* $\delta_k = O(1/L_k)$*.*

Proposition 2 (Fixed-point approximation of the IM integrator)**.** *Following the same notation as in Proposition 1, under Assumption 1, for any inputs* x_k *and* x_{k+1}*, the sequences* $\{\mathrm{T}^{(n)}_{k+1|k}(x_k)\}_{n\in\mathbb{N}}$ *and* $\{\mathrm{T}^{(n)}_{k|k+1}(x_{k+1})\}_{n\in\mathbb{N}}$ *that are recursively defined as*

$$\mathrm{T}^{(0)}_{k+1|k}(x_k) = x_k, \quad \mathrm{T}^{(n+1)}_{k+1|k}(x_k) = x_k + \delta_k v\left(t_{k+1/2}, \frac{x_k + \mathrm{T}^{(n)}_{k+1|k}(x_k)}{2}\right), \tag{23}$$

$$\mathrm{T}^{(0)}_{k|k+1}(x_{k+1}) = x_{k+1}, \quad \mathrm{T}^{(n+1)}_{k|k+1}(x_{k+1}) = x_{k+1} - \delta_k v\left(t_{k+1/2}, \frac{\mathrm{T}^{(n)}_{k|k+1}(x_{k+1}) + x_{k+1}}{2}\right), \tag{24}$$

converge linearly to $\mathrm{T}_{k+1|k}(x_k)$ *and* $\mathrm{T}_{k|k+1}(x_{k+1})$*, respectively.*

[5]Note that this reasoning also applies in the case of EI-based first-order integrators.

[6]We provide the exact numerical constants related to this informal assumption in Appendix C.

In practice, we only compute the sequences from Proposition 2 up to a range $M \geq 1$ that ensures a prescribed fixed-point convergence tolerance $\varepsilon > 0$, that is, M is of the first order such that

$$\left\| \mathrm{T}_{k+1|k}^{(M+1)}(x_k) - \mathrm{T}_{k+1|k}^{(M)}(x_k) \right\|_2 \leq \varepsilon \text{ and } \left\| \mathrm{T}_{k|k+1}^{(M+1)}(x_{k+1}) - \mathrm{T}_{k|k+1}^{(M)}(x_{k+1}) \right\|_2 \leq \varepsilon, \tag{25}$$

and we approximate $\mathrm{T}_{k+1|k}(x_k)$, resp. $\mathrm{T}_{k|k+1}(x_{k+1})$, by the $(M+1)$-th term $\mathrm{T}_{k+1|k}^{(M)}(x_k)$, resp. $\mathrm{T}_{k|k+1}^{(M)}(x_{k+1})$. While this iterative scheme may introduce numerical errors, we note that potential violations of the invertibility property (20) could be mitigated through an additional optional rejection step as proposed by Noble et al. (2023). We leave the implementation of such a safeguard to future work.

Estimating the Jacobian log-determinants. We now address the second component of (1), namely the computation of the Jacobian determinants (more precisely, their logarithm). Since this quantity is generally intractable, we propose a numerical procedure that yields an *unbiased* estimator, thereby preserving the statistical guarantees of aMC. In the case of RE, one can for instance show that the resulting estimator of the MH acceptance rate still guarantees that the target distribution remains invariant, see Andrieu & Roberts (2009) for details. Our approach exploits the recursive structure of the Jacobian associated with the IM integrator, expressing it as a power series, a technique previously explored in the context of contractive residual normalizing flows (Behrmann et al., 2019; Chen et al., 2019). These considerations lead to the following proposition, whose proof is detailed in Appendix C.

Proposition 3 (Approximation of the Jacobian log-determinants via power series)**.** *Following the same notation as in Propositions 1 and 2, under Assumption 1, for any inputs x_k and x_{k+1}, and any prescribed fixed-point range $M \geq 1$ satisfying* (25)*, the following approximation is obtained*

$$\log \left| \det J_{\mathrm{T}_{k+1|k}}(x_k) \right| \approx \sum_{i=0}^{I} a_{k,i} \operatorname{Tr}([A^{(M)}(x_k)]^i) \,,$$
$$\log \left| \det J_{\mathrm{T}_{k|k+1}}(x_{k+1}) \right| \approx \sum_{i=0}^{I} b_{k,i} \operatorname{Tr}([B^{(M)}(x_{k+1})]^i) \,,$$

where $I \geq 1$ is a prescribed truncation order,

$$A^{(M)}(x_k) = \mathbf{H}_{t_{k+1/2}}\left(\frac{x_k + \mathrm{T}_{k+1|k}^{(M)}(x_k)}{2} \right), \quad B^{(M)}(x_{k+1}) = \mathbf{H}_{t_{k+1/2}}\left(\frac{x_{k+1} + \mathrm{T}_{k|k+1}^{(M)}(x_{k+1})}{2} \right) \,,$$

$\mathbf{H}_{t_{k+1/2}}$ is the Hessian of $\log p_{t_{k+1/2}}$ and $\{a_{k,i}, b_{k,i}\}_{i=0}^{I}$ are non-zero numerical coefficients given in Proposition 13 (see Appendix B.1).

As a consequence of Proposition 3, our deterministic approach formally requires evaluating the Hessian of the midpoint log-densities (more specifically, their trace) and can therefore be viewed, from a theoretical standpoint, as a second-order approach. Nevertheless, we stress that the proposed method remains tractable in realistic settings where these Hessians are not explicitly available. Indeed, their traces can be efficiently approximated using the Hutchinson estimator (Hutchinson, 1989; Avron & Toledo, 2011), which relies on the identity $\operatorname{Tr}(\mathrm{M}) = \mathbb{E}_v\left[v^\top \mathrm{M} v\right]$ for any matrix $\mathrm{M} \in \mathbb{R}^{d \times d}$ and any a random vector $v \in \mathbb{R}^d$ satisfying $\mathbb{E}[v] = 0$ and $\operatorname{Cov}[v] = \mathrm{I}_d$ (in practice, one often chooses $v \sim \mathrm{N}(0, \mathrm{I}_d)$). This approach only requires vector–Jacobian products, which can be computed efficiently via reverse-mode automatic differentiation, and provides an unbiased estimate of the trace, resulting in an *unbiased estimate of the Jacobian log-determinants*. While Behrmann et al. (2019) show that the error induced by truncating the expansion series can be safely bounded, Chen et al. (2019) propose using a Russian roulette estimator to further reduce the variance. We leave the exploration of such variance reduction techniques for future work.

Transition method	Forward design	Backward design	Needs $\nabla^2 \log p_k$
1st order kernel	Exact noising kernel (7)	EM/EI approx. (8)	✗
2nd order kernel	Exact noising kernel (7)	DDPM approx. (10)	✓
IM map via Hutchinson	Fixed-point approx. (23)	Fixed-point approx. (24)	✗
IM map via Hessian	Fixed-point approx. (23)	Fixed-point approx. (24)	✓

Table 1: **Summary of DM-based transitions used in annealed sampling methods.** This table presents stochastic transitions (top two lines), related to prior work, and deterministic transitions (bottom two lines), which we develop in Section 4, that can be built from a DM, to be then used within aMC samplers. The last column specifies whether access to the Hessians of the marginal log-densities is required. We recall that the acronym IM stands for *Implicit Midpoint*.

4.3 Empirical comparison between DM-based stochastic and deterministic transitions in aMC samplers

In Figure 3, we report the performance of the deterministic methodology applied to AIS, SMC and RE in idealized setting **(A)** for the *TwoModes* target, and compare it to methods that are based on stochastic kernels (first-order and second-order). In Appendix D.3, we provide further results in all considered multi-modal scenarios. Based on those results, we can make the following observations:

(i) *When the Hessian is available*, using deterministic transitions (pink bars) performs on par with the second-order stochastic approach (green bars). Interestingly, it appears that the deterministic method provides even better results for low values of K with AIS/SMC samplers, offering a significant computational advantage. In the case of RE, we however note a relative under-performance of the deterministic approach, which we believe to be due to the amplification of estimation error of the two Jacobian log-determinants (while AIS/SMC only requires the estimation of one term) with the expansion series.

(ii) *When the Hessian is not available*, the first-order deterministic variant relying solely on the score functions via the Hutchinson estimator (yellow bars) consistently improves over the standard baseline (red bars) and the use of first-order stochastic kernels (blue bars) presented in prior work, for each value of K. This highlights the promise of deterministic mappings in aMC samplers. Remarkably, we observe that the performance gap with the second-order deterministic scheme is relatively small in all multi-modal scenarios, proving the effectiveness of the Hutchinson estimation to fully exploit first-order information. Remarkably, the computational overhead induced by the computation of the log-determinant expansion series is very limited for AIS/SMC sampler; on the contrary, it incurs a non negligible burden for RE sampler.

Remark on second-order stochastic kernels. For the Gaussian kernels in (10), the Hessian appears in the covariance term. As a result, sampling only involves Jacobian–vector products, which can be handled with standard automatic differentiation tools. In contrast, likelihood evaluation additionally requires inverse–Jacobian–vector products through the term $[\Sigma_{s|t}(x_t)]^{-1}(x_s - m_{s|t}(x_t))$, which is substantially more challenging to implement efficiently. While recent work has begun to address this computational bottleneck (Siskind, 2019), developing practical implementations is an open and promising direction for future research.

5 Related Works

Normalizing flows into annealed sampling. Normalizing flows have been previously integrated into aMC frameworks. For instance, Arbel et al. (2021) and Matthews et al. (2022) incorporate flows as forward and backward kernels within AIS and SMC algorithms combined with tempering density paths Other works such as Midgley et al. (2021; 2023b;a) consider AIS schemes where the sequence of densities use π^{base} as a NF, allowing for better conditioned path. On the other hand, Invernizzi et al. (2022) propose an extension of RE of the form of (22), with the key difference being that their deterministic transformations are parameterized by NFs rather than derived from DM-based dynamics.

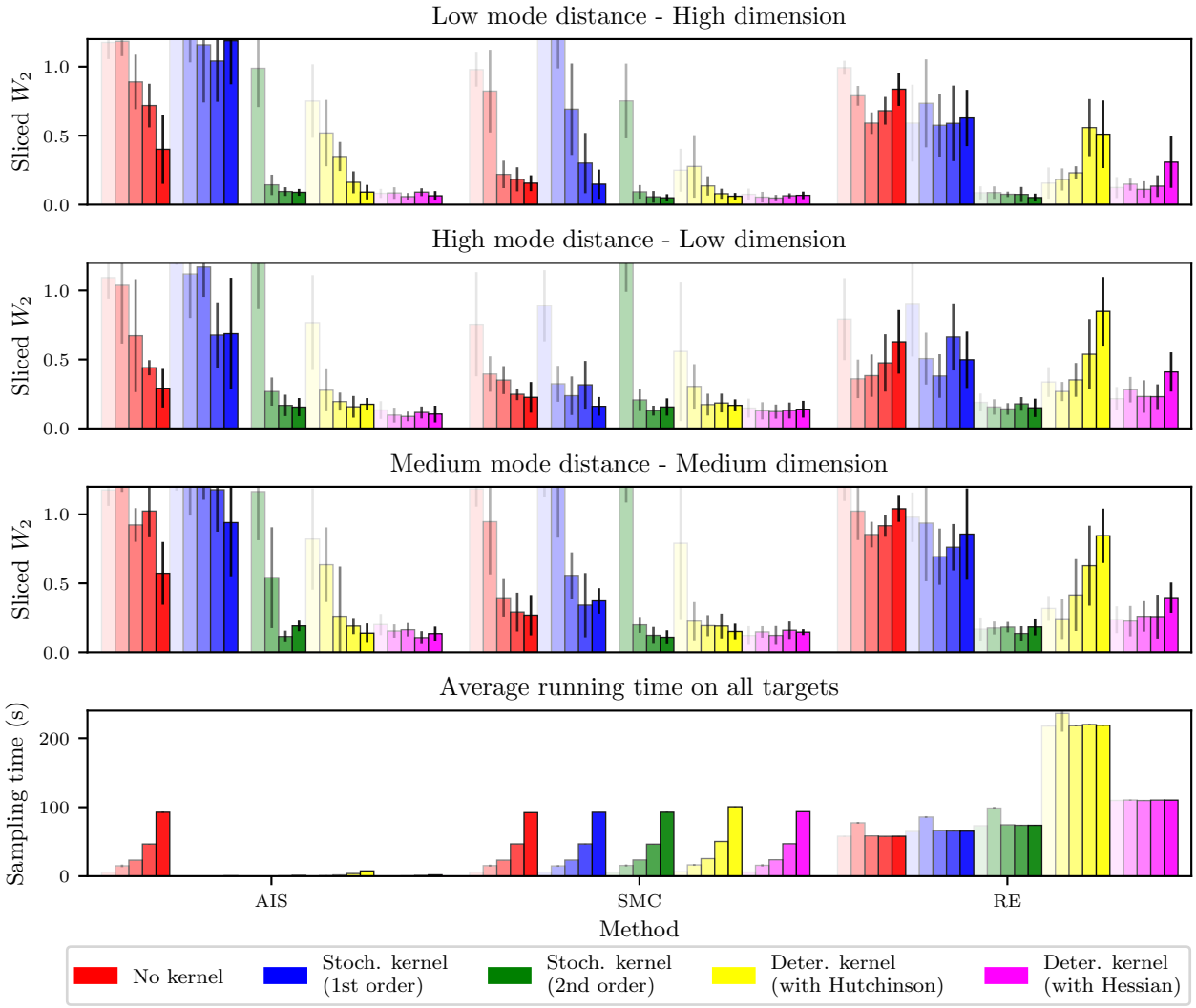

Figure 3: **DM-based aMC-BG results with annealed samplers using different mechanisms**, when targeting *TwoModes* distribution in idealized setting **(A)** : **(First row)** Low distance, high dimension, **(Second row)** High distance, low dimension, **(Third row)** Middle distance, middle dimension, **(Fourth row)** Average running time of each sampler over the three *TwoModes* variants. Each group of bars with the same color corresponds to a specific aMC method. Within each group, red bars refer to methods that do not use DM-based transition kernels (standard baseline); blue and green refer to methods that exploit 1st-order and 2nd-order stochastic kernels (prior work); pink and yellow correspond to variants with deterministic maps, where the log-determinant term is respectively computed from the ground truth diagonal Hessian or via the Hutchinson trick. For all settings, we display the results for all values of K : the darker the bar, the higher K (same range as Figure 1). In particular, configurations within each bar group do not share the same computational budget. Each result is averaged over 8 runs with 8,192 samples per run. We observe that using 1st-order stochastic kernels (in blue) does not always lead to better performance than the baseline (in red), while 2nd-order stochastic kernels (in green) provide consistent improvements. Moreover, using deterministic transitions combined with the Hutchinson trick (in yellow) brings better performance than 1st-order stochastic kernels (both only rely on the use of the scores), while having access to the Hessians (in pink) performs comparably to the 2nd-order stochastic methods. In the case of AIS/SMC sampler, using deterministic transitions incur only a little computational overhead compared to the stochastic case, but this cost is stringer for RE sampler. Similar results are given in Appendix D.3 for *ManyModes* targets.

Using DMs in aMC for sampling. This idea has recently seen a growing interest in the generative modeling community. Some works have built upon the AIS backbone with specific choices of transition operators. For instance, Zhang et al. (2024) propose to design both forward and backward stochastic kernels as a mix of exact noising kernels and first-order *explicit* integrators of the PF-ODE, in order to take advantage of the efficiency of deterministic mappings. On the other hand, Zhang et al. (2025a) design the backward transition kernels as Gaussian denoising kernels with a flexible scalar variance that is learned, in the same spirit as second-order kernels. Taking SMC as a sampling backbone, Phillips et al. (2024) present a end-to-end algorithm that aims to sample from a target distribution by learning the corresponding DM. This procedure alternates between (i) building a BG toward the target via DM-based SMC (here, the backward transitions are defined as first-order EI kernels) and (ii) updating this DM by minimizing a score matching objective with the samples from stage (i). To be able to evaluate the intermediate log-densities, the DM is parameterized as a multi-level energy-based model. More recently, Zhang et al. (2025b) explore the use of DM-based kernels as forward and backward stochastic transitions within a RE framework. Similarly to Phillips et al. (2024), they propose an iterative sampling approach , that involves RE combined with first-order stochastic kernels.

Combination of annealed sampling and DMs beyond BGs. Diffusion models have also been combined with aMC methods, though not primarily for building BGs. Instead, these approaches leverage DMs for various downstream tasks. For instance, SMC-based approaches have been proposed for conditional generation (Wu et al., 2023), posterior sampling in Bayesian inverse problems (Cardoso et al., 2024; Dou & Song, 2024; Janati et al., 2024; 2025), reward-guided generation and fine-tuning (Uehara et al., 2024; Kim et al., 2025; Singhal et al., 2025), as well as compositional and controlled generation tasks (Thornton et al., 2025; Skreta et al., 2025). While these methods rely on advanced sampling techniques, their primary focus lies in enhancing/extending generation capabilities rather than reweighting DMs with respect to a given target unnormalized density.

6 Numerical experiments in a realistic setting

In this section, we evaluate the performance of DM-based aMC-BGs in realistic setting **(B)**. This implies that the true dynamics are no longer available and are instead replaced by estimated dynamics driven by learned log-densities and scores. In particular, we assume that we do not have access to second-order information (*i.e.*, the Hessians of the log-densities), as it is often the case in practice. The purpose of this approach is to compare the practical performance of these samplers with their ideal counterparts described in Section 3 and Section 4, which are affected only by statistical and time-discretization errors. We detail precisely our experimental framework in Section 6.1, where we consider three versions of aMC methods : (a) using only learned log-densities (standard aMC setting), using learned log-densities and scores with (b) first-order stochastic transition kernels (as done in prior work) or (c) first-order deterministic transition maps (as proposed in Section 4).

6.1 Log-density and score learning framework

Architecture design. To evaluate DM-aMC BGs under realistic constraints, we first learn DM log-densities and scores simultaneously using available samples. To do so, we model the log-density $\log p_t(x)$ by a scalar-valued neural network $(t, x) \mapsto -\mathcal{E}_t^\theta(x)$, and deduce an approximation of the score function $\nabla \log p_t(x)$ by taking the negative gradient of $\mathcal{E}^\theta$, denoted by $\mathbf{s}_t^\theta(x) = -\nabla_x \mathcal{E}_t^\theta(x)$. To ensure correctness at $t = t_0$ close to 0, we compare two common architectures.

(a) *Pinned*: we first consider the pinned architecture (Phillips et al., 2024; Zhang et al., 2025b), defined as

$$\mathcal{E}_t^\theta(x) = (1 - f^\theta(t) + f^\theta(t_0))\mathcal{E}(x) + (f^\theta(t) - f^\theta(t_0))g_t^\theta(x) \ , \tag{26}$$

where $f^\theta : [0, T] \to \mathbb{R}$ is a neural network that solely takes time as input, and $g^\theta : [0, T] \times \mathbb{R}^d \to \mathbb{R}$ is another neural network conditioned on both t and x. While this setting ensures exact recovery of the target distribution π at t_0, it is known to be difficult to train (Du et al., 2025), motivating the consideration of the next architecture.

(b) *Hardcoded*: the second architecture is an unconstrained variant inspired by the preconditioned score network used in (Karras et al., 2022; Thornton et al., 2025). Since it does not enforce any boundary condition at $t = t_0$, we explicitly correct this during sampling by replacing $\mathbf{s}^\theta_{t_0}$ with $-\nabla\mathcal{E}$ and $\mathcal{E}^\theta_{t_0}$ with $\mathcal{E}$. While this approach offers more flexibility during training, it may lead to inaccurate behavior at inference.

Loss design. For each neural network, we consider six learning approaches. We restate their expression in Appendix A.4 and provide training details in Appendix D.2. These losses are denoted as follows:

- (DSM): *Denoising Score Matching* objective (Song et al., 2021; Karras et al., 2022),
- (TSM+DSM): *Target Score Matching* objective (Bortoli et al., 2024) with DSM regularization,
- (tSM+DSM): *Time Score Matching* objective (Yu et al., 2025; Guth et al., 2025a) with DSM regularization,
- (LFPE+DSM): DSM objective with *Log-density Fokker Planck Equation* regularization (Shi et al., 2024),
- (aLFPE+DSM): DSM objective with *approximated LFPE* regularization (Plainer et al., 2025),
- (RNE+DSM): DSM objective with *Radon-Nikodym Estimator* regularization (He et al., 2025).

6.2 DM-BGs via aMC seem inherently limited by log-density approximation

DM-BGs fail in practice. Figure 4 compares all first-order DM-based aMC-BGs in realistic setting **(B)**, for a subselection of DM training objectives presented above. It additionally displays

- tempering-based classic aMC samplers, which have been proved to be less accurate than DM-based classic aMC samplers in idealized setting **(A)** (see Section 3.1),
- ideal DM-base aMC-BGs from setting **(A)**, that only exploit first-order information, either with stochastic (see Section 3.2) or deterministic kernels (see Section 4),
- simulations of reverse DM stochastic and deterministic dynamics, which only use the score functions.

Within each setting, we present the best results obtained among all values of the number of levels $K \in \{16, 32, 64, 128, 256\}$ to improve clarity. Figure 4 notably demonstrates that the Harcoded architecture consistently and largely underperform compared (i) to their analogs of idealized setting **(A)** but also compared to (ii) the learned reverse SDE/ODE baseline (which nonetheless matches its ideal analog). Hence, these observations highlight that the poor performance of corresponding aMC methods is not imputable to the quality of the learned scores, but rather to the learned log-densities. In the case of the Pinned architecture, our findings are even more negative, since the simulated reverse SDE/ODE additionally produces highly biased samples, proving that the score functions are not well learned in the considered multi-modal scenario. Finally, we note that all DM-based aMC-BGs approaches often fail to improve (and sometimes degrade) results from standard tempering approaches that do not use any DM.

DM-BG failure cases could be attributed to mode switching in learned log-densities. We hypothesize that a primary contributor to the poor performance of learning-based approaches is the inherent *mode blindness* of presented DM training objectives, which likely leads to mode switching in the learned density path. This issue, well-documented in Wenliang & Kanagawa (2021); Zhang et al. (2022); Shi et al. (2024), is a known limitation of score-based methods to learn accurately log-densities, and affects all divergences derived from the Fisher divergence or Stein discrepancy. Indeed, when applied on multi-modal distributions with well-separated modes, these divergences fail to distinguish distributions that have the same mode locations but differ in mode proportions. This is because the score, being independent of the normalizing constant, is unaware of other modes when evaluated within a single mode. As a result, learning the log-density via score matching often produces the correct shape within each mode (*i.e.*, accurate gradients) but with incorrect relative weights. In Figure 6, we illustrate this phenomenon for the same subset of DM training objectives as in Figure 4 for 1D Gaussian mixtures (all results being given in Appendix D.3). The obtained results can be compared with the ground truth diffusion and tempering density paths displayed in Figure 5.

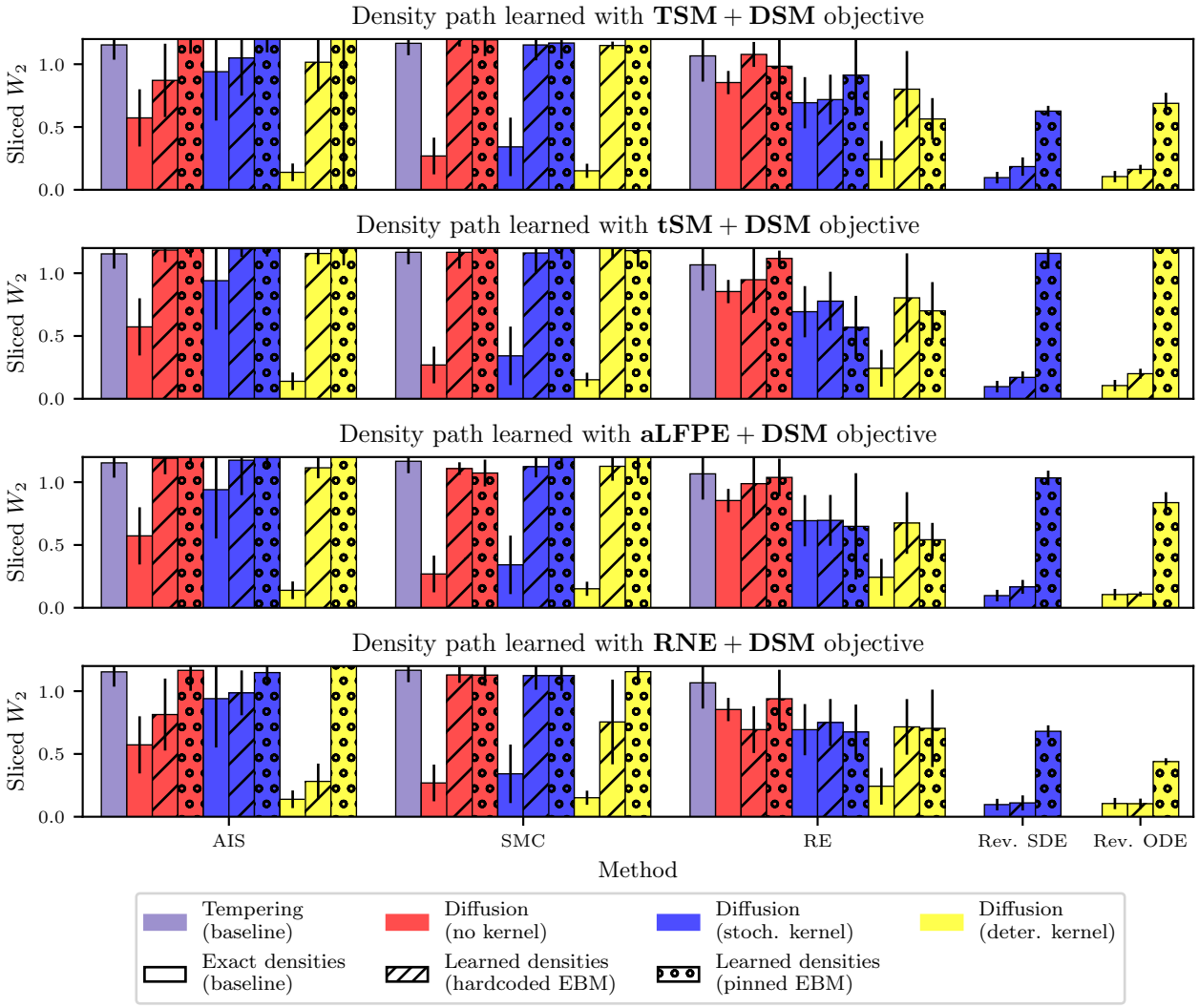

Figure 4: **Realistic results of DM-based aMC-BG**, when targeting *TwoModes* distribution with intermediate difficulty (middle mode distance, middle dimension) in setting **(B)** : **(From top to bottom)** the DM is trained via TSM+DSM, tSM+DSM, aLFPE+DSM or RNE+DSM objective with identical computational budget. Each group of bars with the same color corresponds to a specific aMC method, except for the last two groups on the right which shows the baseline obtained by directly simulating the reverse SDE (5) and reverse ODE (6). Bar colors indicate the type of density path used for the sampling method, and are consistent with those displayed in Figure 3: purple for the model-free tempering path (same baseline for all models), red for the diffusion density path solely exploiting the log-densities, blue, respectively yellow, for the diffusion density path additionally exploiting DM first-order stochastic kernels, respectively DM first-order deterministic kernels. On the other hand, hatching denotes the nature of log densities used in aMC (exact as in Figure 3, learned with hardcoded EBM or learned with pinned EBM). For each method and density path, the number of levels K was optimized individually so as to display the best expected result from the considered approach if K was carefully tuned. The results show a clear performance gap between learned and ideal paths. Notably, DM-BGs generally underperform compared to directly to the DM alone (*i.e.*simulating the reverse SDE/ODE), and rarely surpass the classic tempering methods. Similar results are given in Appendix D.3 for other *TwoModes* and *ManyModes* targets, and the other DM training objectives.

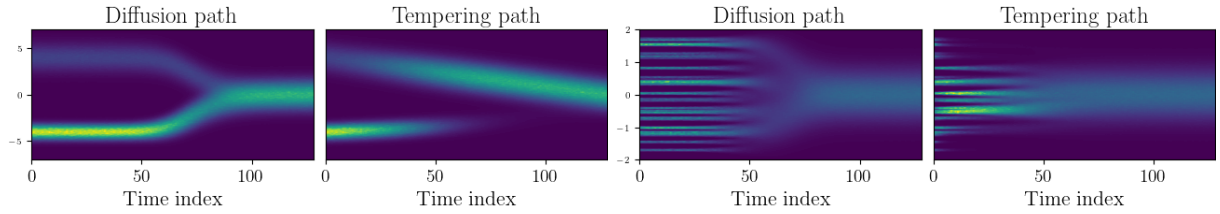

Figure 5: **Exact density paths bridging π^{base} (last time index) to 1D Gaussian mixtures (first time index). (Left)**: the target is an instance of *TwoModes* defined as $(3/4)\mathrm{N}(-4, 0.5^2) + (1/4)\mathrm{N}(+4, 1)$, **(Right)** the target is the 1D instance of *ManyModes* Noble et al. (2025) with 32 modes, **(First and third columns)** diffusion density path, **(Second and fourth columns)** tempering density path. Given a discretization of 128 timesteps in [0,1], we plot each of the induced marginal densities using importance sampling with 1,000,000 particles and a uniform proposal. We observe that the tempering path shows clear mode switching for both of the targets: in the case of the *TwoModes* target, the strongest mode emerges abruptly, while the weakest modes appear rapidly for the *ManyModes* target. On the other hand, the mode weights in the diffusion path remain stable over time, making it more favorable for aMC.

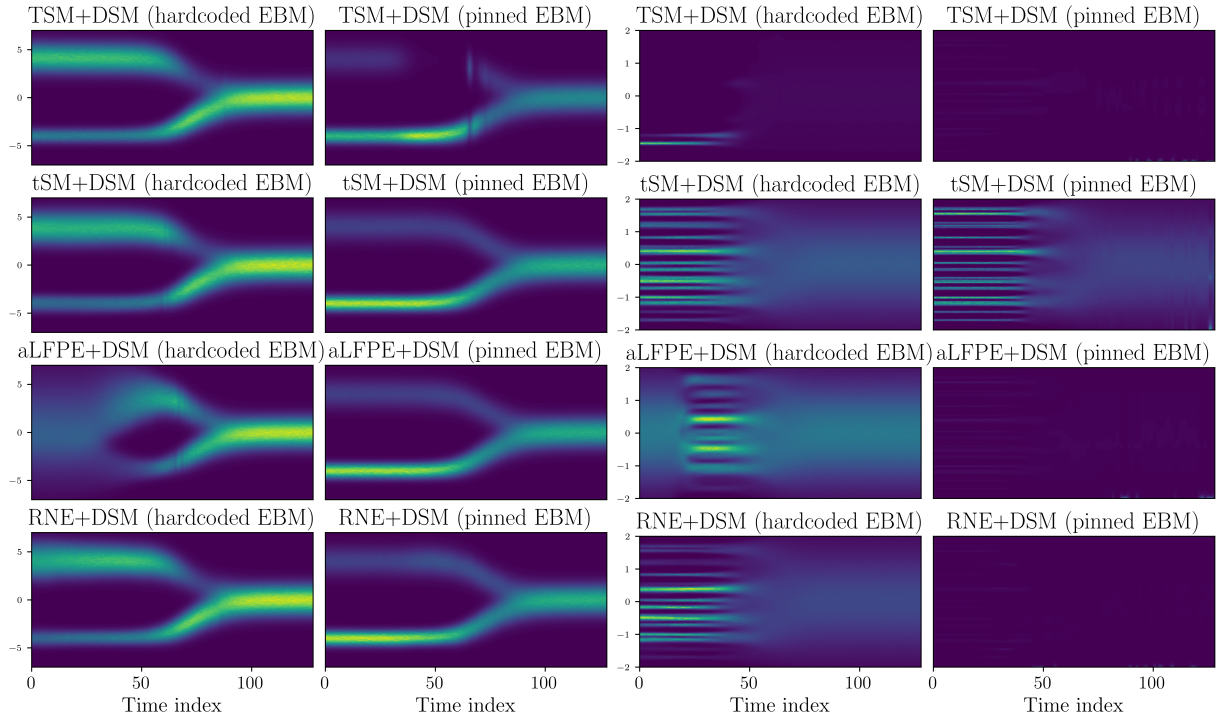

Figure 6: **Learned diffusion density paths bridging π^{base} (last time index) to the same targets as in Figure 5. (From top to bottom)** the DM is trained via TSM+DSM, tSM+DSM, aLFPE+DSM or RNE+DSM objective, with identical computational budget, **(Left)** *TwoModes* target, **(Right)** *ManyModes* target, **(First and third columns)** use of Hardcoded EBM, **(Second and fourth columns)** use of Pinned EBM. The same plotting configuration is used as in Figure 5. When using the Hardcoded architecture for both targets, we observe that the density is well learned for large times (near π^{base}), but systematically fails to recover the exact mode weights for small times, thus highlighting the mode blindness of related training objectives. While using the Pinned architecture enables to closely recover the exact diffusion path for the *TwoModes* target, we observe that this strategy is not successful when increasing the number of modes, with the notable exception of tSM+DSM objective. This under-performance of the Pinned architecture may notably explain its systematic failure in our experiments run in realistic setting **(B)**, where the targets are highly challenging.

The ubiquitous curse of mode blindness for log-density estimation ?

Although mode blindness has been empirically documented for the DSM objective, we emphasize that it also applies to the TSM objective too, as a consequence of its score-based formulation. Nevertheless, recent diffusion-based sampling methods have employed TSM to learn log-densities (Phillips et al., 2024; Zhang et al., 2025b) for using DMs within aMC. By the same reasoning, mode blindness also affects log-density estimation via score-based distillation losses (Thornton et al., 2025; Akhound-Sadegh et al., 2025), probably hurting the accuracy of SMC sampling for related inference-time alignment tasks. Based on the results displayed in Figure 6 and their score-based formulation (see Appendix A.4), we conjecture that mode blindness is prone to occur when using LFPE-based regularizations, despite their adoption in multiple diffusion-based sampling frameworks (Shi et al., 2024; Plainer et al., 2025; Sun et al., 2024b). Although mode blindness seems less straightforward for tSM and RNE objectives, we believe it is still a substantial barrier to accurate log-density estimation, as depicted by Figure 6, and leave the theoretical investigation of this question to future work.

Overall, we conjecture that mode switching significantly hinders aMC methods, as well for learned diffusion density paths as for tempering density paths. In SMC (including DM-enhanced variants), resampling must continually correct for imbalanced mode weights, which becomes increasingly challenging in high dimensions. Similarly, in RE, communication between chains is disrupted when mode alignment across levels is inconsistent, although we observe that it may be compensated for by the possibility of moving back and forth between levels during sampling procedure. This instability explains the poor performance of the learned path in Figure 4, even when the forward and backward transition kernels (both deterministic and stochastic) are accurate due to well-learned scores.

7 Conclusion & Limitations

This work revisits the design of Boltzmann Generators by replacing the standard normalizing-flow/importance-sampling backbone with a diffusion-model backbone embedded in annealed Monte Carlo. We first unify and review prior DM-aMC approaches, which exploit diffusion-induced *stochastic* denoising kernels to facilitate transitions between annealing levels, and we then introduce and study *deterministic* counterparts based on diffusion-derived transport maps. To compare these methods, we conduct an empirical study on multi-modal target distributions, emphasizing challenging characteristics such as inter-mode separation, number of modes, and dimensionality. Our analysis proceeds in two stages: we (i) isolate inference effects by assuming a perfectly learned DM, and (ii) turn to a realistic setting where the DM is trained from data.

In the idealized regime, empirical metrics reveal a non-zero discrepancy between the ground-truth target and the distribution induced by the resulting BG, despite perfect model knowledge. This indicates that aMC inference error alone can produce measurable bias. In this setting, *first-order* stochastic denoising kernels (score-only) often fail to improve over standard aMC baselines, whereas second-order kernels (incorporating Hessian information) and deterministic transitions yield substantially better results. Importantly, our deterministic construction based on a Hutchinson-type estimator remains competitive even without explicit Hessian access, suggesting that deterministic transport can recover much of the benefit of second-order information while relaxing its most demanding requirement.

In the learned regime (e.g., score-matching-like training objectives), the picture changes markedly: the resulting BGs systematically fail across our multi-modal benchmarks, even when the learned scores appear accurate. Our experiments point to inaccuracies in DM log-density estimation as the primary culprit. Specifically, the obtained estimates are mostly mode-blind, as they fail to accurately represent relative mode proportions along the diffusion path in regions where the modes are well separated. As a consequence, such errors directly disrupt sampling and can dominate any gains from improved transitions. In other words, high-quality score estimates are not sufficient to guarantee successful BG construction when the correction step relies on unreliable log-density approximations.

In the spirit of Grenioux et al. (2025), our goal is not to demonstrate scalability but to expose and analyze the fundamental limitations of diffusion-based aMC-BGs in a simple, fully controlled benchmark. The underlying rationale is that methods that do not succeed in these elementary multi-modal settings are unlikely to behave reliably on more complex targets with many modes or ill-conditioned energy landscapes. Accordingly, this work emphasizes failure mechanisms over performance claims, consistent with our largely negative conclusions.

Several extensions remain natural. First, moving beyond Gaussian mixtures to more challenging targets is an important next step. Second, we did not include adaptive SMC and RE-style algorithms in the main comparisons. With larger budgets (e.g., more intermediate distributions or carefully tuned tempering schedules), such methods could potentially narrow the observed gaps. Third, diffusion-based methods may depend on the noise schedule: for completeness we derive in Appendix B.3 the formulas for the VE noising scheme, and we leave a systematic numerical study of its impact to future work.

Finally, the promising behavior of deterministic (as opposed to stochastic) transitions motivates two concrete research directions. On the inference side, we will further study the deterministic construction of Section 4 to reduce sensitivity to Hessian-trace approximations, including the use of more robust estimators (Liu et al., 2025). On the modeling side, the main bottleneck in realistic settings is the mode blindness of current DM log-density estimation techniques; addressing this issue appears necessary for reliable sampling. Beyond building a single BG, if one can design training schemes that mitigate or eliminate mode blindness, the proposed aMC machinery could serve as an inner loop in iterative diffusion-based training procedures tailored for sampling, departing from the one-shot correction perspective, in the spirit of adaptive, data-free training strategies (Gabrié et al., 2022; Phillips et al., 2024; Akhound-Sadegh et al., 2024).

References

Josh Abramson, Jonas Adler, Jack Dunger, Richard Evans, Tim Green, Alexander Pritzel, Olaf Ronneberger, Lindsay Willmore, Andrew J. Ballard, Joshua Bambrick, Sebastian W. Bodenstein, David A. Evans, Chia-Chun Hung, Michael O'Neill, David Reiman, Kathryn Tunyasuvunakool, Zachary Wu, Akvilė Žemgulytė, Eirini Arvaniti, Charles Beattie, Ottavia Bertolli, Alex Bridgland, Alexey Cherepanov, Miles Congreve, Alexander I. Cowen-Rivers, Andrew Cowie, Michael Figurnov, Fabian B. Fuchs, Hannah Gladman, Rishub Jain, Yousuf A. Khan, Caroline M. R. Low, Kuba Perlin, Anna Potapenko, Pascal Savy, Sukhdeep Singh, Adrian Stecula, Ashok Thillaisundaram, Catherine Tong, Sergei Yakneen, Ellen D. Zhong, Michal Zielinski, Augustin Žídek, Victor Bapst, Pushmeet Kohli, Max Jaderberg, Demis Hassabis, and John M. Jumper. Accurate structure prediction of biomolecular interactions with alphafold 3. *Nature*, 630(8016):493–500, Jun 2024. ISSN 1476-4687. doi: 10.1038/s41586-024-07487-w. URL https://doi.org/10.1038/s41586-024-07487-w.

S. Agapiou, O. Papaspiliopoulos, D. Sanz-Alonso, and A. M. Stuart. Importance sampling: Intrinsic dimension and computational cost. *Statistical Science*, 32(3):405–431, 2017. ISSN 08834237, 21688745. URL http://www.jstor.org/stable/26408299.

Tara Akhound-Sadegh, Jarrid Rector-Brooks, Joey Bose, Sarthak Mittal, Pablo Lemos, Cheng-Hao Liu, Marcin Sendera, Siamak Ravanbakhsh, Gauthier Gidel, Yoshua Bengio, Nikolay Malkin, and Alexander Tong. Iterated denoising energy matching for sampling from boltzmann densities. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 760–786. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/akhound-sadegh24a.html.

Tara Akhound-Sadegh, Jungyoon Lee, Avishek Joey Bose, Valentin De Bortoli, Arnaud Doucet, Michael M. Bronstein, Dominique Beaini, Siamak Ravanbakhsh, Kirill Neklyudov, and Alexander Tong. Progressive inference-time annealing of diffusion models for sampling from boltzmann densities, 2025. URL https://arxiv.org/abs/2506.16471.

M. S. Albergo, G. Kanwar, and P. E. Shanahan. Flow-based generative models for markov chain monte carlo in lattice field theory. *Physical Review D*, 100(3):034515, 8 2019. ISSN 2470-0010, 2470-0029. doi: 10.1103/PhysRevD.100.034515. URL https://link.aps.org/doi/10.1103/PhysRevD.100.034515.

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.

Christophe Andrieu and Gareth O. Roberts. The pseudo-marginal approach for efficient Monte Carlo computations. *The Annals of Statistics*, 37(2):697 – 725, 2009. doi: 10.1214/07-AOS574. URL https://doi.org/10.1214/07-AOS574.

Michael Arbel, Alex Matthews, and Arnaud Doucet. Annealed flow transport monte carlo. In *International Conference on Machine Learning*, pp. 318–330. PMLR, 2021.

Haim Avron and Sivan Toledo. Randomized algorithms for estimating the trace of an implicit symmetric positive semi-definite matrix. *J. ACM*, 58(2), April 2011. ISSN 0004-5411. doi: 10.1145/1944345.1944349. URL https://doi.org/10.1145/1944345.1944349.

Fan Bao, Chongxuan Li, Jiacheng Sun, Jun Zhu, and Bo Zhang. Estimating the optimal covariance with imperfect mean in diffusion probabilistic models. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 1555–1584. PMLR, 17–23 Jul 2022a. URL https://proceedings.mlr.press/v162/bao22d.html.

Fan Bao, Chongxuan Li, Jun Zhu, and Bo Zhang. Analytic-DPM: an analytic estimate of the optimal reverse variance in diffusion probabilistic models. In *International Conference on Learning Representations*, 2022b. URL https://openreview.net/forum?id=0xiJLKH-ufZ.

Jens Behrmann, Will Grathwohl, Ricky T. Q. Chen, David Duvenaud, and Joern-Henrik Jacobsen. Invertible residual networks. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 573–582. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/behrmann19a.html.

Nicolas Bonneel, Julien Rabin, Gabriel Peyré, and Hanspeter Pfister. Sliced and radon wasserstein barycenters of measures. *Journal of Mathematical Imaging and Vision*, 51:22–45, 2015.

Valentin De Bortoli, Michael Hutchinson, Peter Wirnsberger, and Arnaud Doucet. Target score matching, 2024.

James Brofos, Marylou Gabrie, Marcus A. Brubaker, and Roy R. Lederman. Adaptation of the independent metropolis-hastings sampler with normalizing flow proposals. In Gustau Camps-Valls, Francisco J. R. Ruiz, and Isabel Valera (eds.), *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pp. 5949–5986. PMLR, 28–30 Mar 2022. URL https://proceedings.mlr.press/v151/brofos22a.html.

Alberto Cabezas, Louis Sharrock, and Christopher Nemeth. Markovian flow matching: Accelerating mcmc with continuous normalizing flows. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 104383–104411. Curran Associates, Inc., 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/bcd11db0b26d8fc2266b91d3ff982ed1-Paper-Conference.pdf.

Gabriel Cardoso, Yazid Janati el idrissi, Sylvain Le Corff, and Eric Moulines. Monte carlo guided denoising diffusion models for bayesian linear inverse problems. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=nHESwXvxWK.

Ricky T. Q. Chen, Jens Behrmann, David K Duvenaud, and Joern-Henrik Jacobsen. Residual flows for invertible generative modeling. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/5d0d5594d24f0f955548f0fc0ff83d10-Paper.pdf.

Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. *Advances in neural information processing systems*, 2018.

Nicolas Chopin and Omiros Papaspiliopoulos. *An introduction to sequential Monte Carlo*. Springer International Publishing, Cham, 2020. ISBN 978-3-030-47845-2. doi: 10.1007/978-3-030-47845-2_8. URL https://doi.org/10.1007/978-3-030-47845-2_8.

Luigi Del Debbio, Joe Marsh Rossney, and Michael Wilson. Machine Learning Trivializing Maps: A First Step Towards Understanding How Flow-Based Samplers Scale Up. *PoS*, LATTICE2021:059, 2022. doi: 10.22323/1.396.0059.

Pierre Del Moral, Arnaud Doucet, and Ajay Jasra. Sequential monte carlo samplers. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 68(3):411–436, 2006.

Zehao Dou and Yang Song. Diffusion posterior sampling for linear inverse problem solving: A filtering perspective. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=tplXNcHZs1.

Arnaud Doucet, Nando De Freitas, Neil James Gordon, et al. *Sequential Monte Carlo methods in practice*, volume 1. Springer, 2001.

Yilun Du, Conor Durkan, Robin Strudel, Joshua B Tenenbaum, Sander Dieleman, Rob Fergus, Jascha Sohl-Dickstein, Arnaud Doucet, and Will Sussman Grathwohl. Reduce, reuse, recycle: Compositional generation with energy-based diffusion models and mcmc. In *International conference on machine learning*, pp. 8489–8510. PMLR, 2023.

Yuanqi Du, Jiajun He, Francisco Vargas, Yuanqing Wang, Carla P Gomes, José Miguel Hernández-Lobato, and Eric Vanden-Eijnden. FEAT: Free energy estimators with adaptive transport. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=GQXeLGYMda.

Simon Duane, A. D. Kennedy, Brian J. Pendleton, and Duncan Roweth. Hybrid monte carlo. *Physics Letters B*, 195(2):216–222, 1987. ISSN 0370-2693. doi: https://doi.org/10.1016/0370-2693(87)91197-X. URL https://www.sciencedirect.com/science/article/pii/037026938791197X.

Daan Frenkel and Berend Smit. *Understanding Molecular Simulation: from Algorithms to Applications*. Elsevier, 2023.

Marylou Gabrié, Grant M. Rotskoff, and Eric Vanden-Eijnden. Adaptive monte carlo augmented with normalizing flows. *Proceedings of the National Academy of Sciences*, 119(10):e2109420119, 2022. doi: 10.1073/pnas.2109420119. URL https://www.pnas.org/doi/abs/10.1073/pnas.2109420119.

Ruiqi Gao, Yang Song, Ben Poole, Ying Nian Wu, and Diederik P Kingma. Learning energy-based models by diffusion recovery likelihood. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=v_1Soh8QUNc.

A. Gelman and X.-L. Meng. Simulating Normalizing Constants: From Importance Sampling to Bridge Sampling to Path Sampling. *Statistical Science*, 13:163–185, 1998.

Charles J Geyer et al. Markov chain monte carlo maximum likelihood. In *Computing science and statistics: Proceedings of the 23rd Symposium on the Interface*, volume 156163. New York, 1991.

Will Grathwohl, Ricky T. Q. Chen, Jesse Bettencourt, Ilya Sutskever, and David Duvenaud. Ffjord: Free-form continuous dynamics for scalable reversible generative models. *International Conference on Learning Representations (ICLR)*, 2019.

Louis Grenioux, Alain Oliviero Durmus, Eric Moulines, and Marylou Gabrié. On sampling with approximate transport maps. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 11698–11733. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/grenioux23a.html.

Louis Grenioux, Maxence Noble, Marylou Gabrié, and Alain Oliviero Durmus. Stochastic localization via iterative posterior sampling. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 16337–16376. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/grenioux24a.html.

Louis Grenioux, Maxence Noble, and Marylou Gabrié. Improving the evaluation of samplers on multi-modal targets. In *Frontiers in Probabilistic Inference: Learning meets Sampling*, 2025. URL https://openreview.net/forum?id=d91E9RhVFU.

Florentin Guth, Zahra Kadkhodaie, and Eero P Simoncelli. Learning normalized image densities via dual score matching. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025a. URL https://openreview.net/forum?id=wtYcS4kxpF.

Florentin Guth, Zahra Kadkhodaie, and Eero P Simoncelli. Learning normalized image densities via dual score matching, 2025b. URL https://arxiv.org/abs/2506.05310.

Brian C. Hall. An elementary introduction to groups and representations, 2000. URL https://arxiv.org/abs/math-ph/0005032.

Jiajun He, José Miguel Hernández-Lobato, Yuanqi Du, and Francisco Vargas. Rne: plug-and-play diffusion inference-time control and energy-based training, 2025. URL https://arxiv.org/abs/2506.05668.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P. Kingma, Ben Poole, Mohammad Norouzi, David J. Fleet, and Tim Salimans. Imagen video: High definition video generation with diffusion models, 2022. URL https://arxiv.org/abs/2210.02303.

Koji Hukushima and Koji Nemoto. Exchange monte carlo method and application to spin glass simulations. *Journal of the Physical Society of Japan*, 65(6):1604–1608, 1996. doi: 10.1143/JPSJ.65.1604. URL https://doi.org/10.1143/JPSJ.65.1604.

M.F. Hutchinson. A stochastic estimator of the trace of the influence matrix for laplacian smoothing splines. *Communications in Statistics - Simulation and Computation*, 18(3):1059–1076, 1989. doi: 10.1080/03610918908812806. URL https://doi.org/10.1080/03610918908812806.

Aapo Hyvärinen. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(24):695–709, 2005. URL http://jmlr.org/papers/v6/hyvarinen05a.html.

Michele Invernizzi, Andreas Krämer, Cecilia Clementi, and Frank Noé. Skipping the replica exchange ladder with normalizing flows. *The Journal of Physical Chemistry Letters*, 13(50):11643–11649, Dec 2022. doi: 10.1021/acs.jpclett.2c03327. URL https://doi.org/10.1021/acs.jpclett.2c03327.

Yazid Janati, Badr Moufad, Alain Durmus, Eric Moulines, and Jimmy Olsson. Divide-and-conquer posterior sampling for denoising diffusion priors. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 97408–97444. Curran Associates, Inc., 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/b0ae046e198a5e43141519868a959c74-Paper-Conference.pdf.

Yazid Janati, Eric Moulines, Jimmy Olsson, and Alain Oliviero-Durmus. Bridging diffusion posterior sampling and monte carlo methods: a survey. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 383(2299):20240331, 2025. doi: 10.1098/rsta.2024.0331. URL https://royalsocietypublishing.org/doi/abs/10.1098/rsta.2024.0331.

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *Advances in Neural Information Processing Systems*, 35:26565–26577, 2022.

Tero Karras, Miika Aittala, Jaakko Lehtinen, Janne Hellsten, Timo Aila, and Samuli Laine. Analyzing and improving the training dynamics of diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24174–24184, June 2024.

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

Leon Klein and Frank Noé. Transferable boltzmann generators. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 45281–45314. Curran Associates, Inc., 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/5035a409f5798e188079e236f437e522-Paper-Conference.pdf.

Leon Klein, Andreas Krämer, and Frank Noé. Equivariant flow matching. *Neural Information Processing Systems (NeurIPS)*, 2023.

Jonas Köhler, Leon Klein, and Frank Noé. Equivariant flows: exact likelihood generative learning for symmetric densities. *International Conference on Machine Learning (ICML)*, 2020.

Zhifeng Kong, Wei Ping, Jiaji Huang, Kexin Zhao, and Bryan Catanzaro. Diffwave: A versatile diffusion model for audio synthesis. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=a-xFK8Ymz5J.

Werner Krauth. Statistical mechanics: algorithms and computations. *OUP Oxford*, 13, 2006.

Chieh-Hsin Lai, Yuhta Takida, Naoki Murata, Toshimitsu Uesaka, Yuki Mitsufuji, and Stefano Ermon. Fp-diffusion: Improving score-based diffusion models by enforcing the underlying score fokker-planck equation. In *International Conference on Machine Learning*, pp. 18365–18398. PMLR, 2023.

Jun S Liu. *Monte Carlo Strategies in Scientific Computing*. Springer, 2001.

Xinyang Liu, Hengrong Du, Wei Deng, and Ruqi Zhang. Optimal stochastic trace estimation in generative modeling. In Yingzhen Li, Stephan Mandt, Shipra Agrawal, and Emtiyaz Khan (eds.), *Proceedings of The 28th International Conference on Artificial Intelligence and Statistics*, volume 258 of *Proceedings of Machine Learning Research*, pp. 4600–4608. PMLR, 03–05 May 2025. URL https://proceedings.mlr.press/v258/liu25k.html.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

Bálint Máté and François Fleuret. Learning interpolations between boltzmann densities. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL https://openreview.net/forum?id=TH6YrEcbth.

Alex Matthews, Michael Arbel, Danilo Jimenez Rezende, and Arnaud Doucet. Continual repeated annealed flow transport monte carlo. In *International Conference on Machine Learning*, pp. 15196–15219. PMLR, 2022.

Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21(6):1087–1092, 06 1953. ISSN 0021-9606. doi: 10.1063/1.1699114. URL https://doi.org/10.1063/1.1699114.

L. I. Midgley, V. Stimper, G. N. C. Simm, and J. M. Hernández-Lobato. Bootstrap your flow. In *1st ELLIS Machine Learning for Molecule Discovery Workshop*, December 2021. URL https://arxiv.org/abs/2111.11510.

Laurence Midgley, Vincent Stimper, Javier Antorán, Emile Mathieu, Bernhard Schölkopf, and José Miguel Hernández-Lobato. Se(3) equivariant augmented coupling flows. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 79200–79225. Curran Associates, Inc., 2023a. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/fa55eb802a531c8087e225ecf2dcfbca-Paper-Conference.pdf.

Laurence Illing Midgley, Vincent Stimper, Gregor N. C. Simm, Bernhard Schölkopf, and José Miguel Hernández-Lobato. Flow annealed importance sampling bootstrap. In *The Eleventh International Conference on Learning Representations*, 2023b. URL https://openreview.net/forum?id=XCTVFJwS9LJ.

Thomas Müller, Brian Mcwilliams, Fabrice Rousselle, Markus Gross, and Jan Novák. Neural importance sampling. *ACM Transactions on Graphics*, 38(5):1–19, 10 2019. ISSN 0730-0301, 1557-7368. doi: 10.1145/3341156. URL https://dl.acm.org/doi/10.1145/3341156.

Radford M Neal. Annealed importance sampling. *Statistics and computing*, 11:125–139, 2001.

Radford M Neal. MCMC using hamiltonian dynamics. *arXiv preprint arXiv:1206.1901*, 2012.

Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, pp. 8162–8171. PMLR, 2021.

Maxence Noble, Valentin De Bortoli, and Alain Durmus. Unbiased constrained sampling with self-concordant barrier hamiltonian monte carlo. *Advances in Neural Information Processing Systems*, 36:32672–32719, 2023.

Maxence Noble, Louis Grenioux, Marylou Gabrié, and Alain Oliviero Durmus. Learned reference-based diffusion sampler for multi-modal distributions. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=fmJUYgmMbL.

Frank Noé, Simon Olsson, Jonas Köhler, and Hao Wu. Boltzmann generators: Sampling equilibrium states of many-body systems with deep learning. *Science*, 365(6457):eaaw1147, 9 2019. ISSN 0036-8075, 1095-9203. doi: 10.1126/science.aaw1147. URL https://www.science.org/doi/10.1126/science.aaw1147.

Kaoru Ohno, Keivan Esfarjani, and Yoshiyuki Kawazoe. *Computational Materials Science: From Ab Initio to Monte Carlo Methods*. Springer, 2018.

Tsuneyasu Okabe, Masaaki Kawata, Yuko Okamoto, and Masuhiro Mikami. Replica-exchange monte carlo method for the isobaric–isothermal ensemble. *Chemical Physics Letters*, 335(5):435–439, 2001. ISSN 0009-2614. doi: https://doi.org/10.1016/S0009-2614(01)00055-0. URL https://www.sciencedirect.com/science/article/pii/S0009261401000550.

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

Zijing Ou, Mingtian Zhang, Andi Zhang, Tim Z. Xiao, Yingzhen Li, and David Barber. Improving probabilistic diffusion models with optimal diagonal covariance matching. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=fV0t65OBUu.

George Papamakarios, Eric Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed, and Balaji Lakshminarayanan. Normalizing flows for probabilistic modeling and inference. *Journal of Machine Learning Research*, 22(57):1–64, 2021. URL http://jmlr.org/papers/v22/19-1028.html.

Angus Phillips, Hai-Dang Dau, Michael John Hutchinson, Valentin De Bortoli, George Deligiannidis, and Arnaud Doucet. Particle denoising diffusion sampler. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 40688–40724. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/phillips24a.html.

Michael Plainer, Hao Wu, Leon Klein, Stephan Günnemann, and Frank Noe. Consistent sampling and simulation: Molecular dynamics with energy-based diffusion models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=gzYuvZg28E.

Danilo Rezende and Shakir Mohamed. Variational inference with normalizing flows. In *International conference on machine learning*, pp. 1530–1538. PMLR, 2015.

Lorenz Richter, Julius Berner, and Guan-Horng Liu. Improved sampling via learned diffusions. In *ICML Workshop on New Frontiers in Learning, Control, and Dynamical Systems*, 2023. URL https://openreview.net/forum?id=uLgYD7ie0O.

Gareth O. Roberts and Richard L. Tweedie. Exponential convergence of langevin distributions and their discrete approximations. *Bernoulli*, 2(4):341–363, 1996. ISSN 13507265. URL http://www.jstor.org/stable/3318418.

Tim Salimans and Jonathan Ho. Should EBMs model the energy or the score? In *Energy Based Models Workshop-ICLR 2021*, 2021.

Sergey Samsonov, Evgeny Lagutin, Marylou Gabrié, Alain Durmus, Alexey Naumov, and Eric Moulines. Local-global MCMC kernels: the best of both worlds. In *Advances in Neural Information Processing Systems*, 2022.

Zhekun Shi, Longlin Yu, Tianyu Xie, and Cheng Zhang. Diffusion-PINN sampler, 2024. URL https://arxiv.org/abs/2410.15336.

Raghav Singhal, Zachary Horvitz, Ryan Teehan, Mengye Ren, Zhou Yu, Kathleen McKeown, and Rajesh Ranganath. A general framework for inference-time scaling and steering of diffusion models. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=Jp988ELppQ.

Jeffrey Mark Siskind. Automatic differentiation: Inverse accumulation mode. In *Program Transformations for ML Workshop at NeurIPS 2019*, 2019. URL https://openreview.net/forum?id=Bygj2Ys6IS.

Marta Skreta, Tara Akhound-Sadegh, Viktor Ohanesian, Roberto Bondesan, Alan Aspuru-Guzik, Arnaud Doucet, Rob Brekelmans, Alexander Tong, and Kirill Neklyudov. Feynman-kac correctors in diffusion: Annealing, guidance, and product of experts. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=Vhc0KrcqWu.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pp. 2256–2265. PMLR, 2015.

Yang Song and Diederik P. Kingma. How to train your energy-based models, 2021. URL https://arxiv.org/abs/2101.03288.

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

Gabriel Stoltz, Mathias Rousset, and Tony Lelièvre. *Free Energy Computations: A Mathematical Perspective*. World Scientific, 2010.

Jingtong Sun, Julius Berner, Lorenz Richter, Marius Zeinhofer, Johannes Müller, Kamyar Azizzadenesheli, and Anima Anandkumar. Dynamical measure transport and neural pde solvers for sampling. *arXiv preprint arXiv:2407.07873*, 2024a.

Jingtong Sun, Julius Berner, Lorenz Richter, Marius Zeinhofer, Johannes Müller, Kamyar Azizzadenesheli, and Anima Anandkumar. Dynamical measure transport and neural pde solvers for sampling, 2024b. URL https://arxiv.org/abs/2407.07873.

Robert H. Swendsen and Jian-Sheng Wang. Replica monte carlo simulation of spin-glasses. *Physical Review Letters*, 57(21):2607–2609, 11 1986. doi: 10.1103/PhysRevLett.57.2607. URL https://link.aps.org/doi/10.1103/PhysRevLett.57.2607. Publisher: American Physical Society.

Saifuddin Syed, Vittorio Romaniello, Trevor Campbell, and Alexandre Bouchard-Cote. Parallel tempering on optimized paths. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 10033–10042. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/syed21a.html.

Saifuddin Syed, Alexandre Bouchard-Côté, George Deligiannidis, and Arnaud Doucet. Non-reversible parallel tempering: A scalable highly parallel mcmc scheme. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 84(2):321–350, 2022. doi: https://doi.org/10.1111/rssb.12464. URL https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/rssb.12464.

Saifuddin Syed, Alexandre Bouchard-Côté, Kevin Chern, and Arnaud Doucet. Optimised annealed sequential monte carlo samplers, 2025. URL https://arxiv.org/abs/2408.12057.

James Thornton, Louis Béthune, Ruixiang ZHANG, Arwen Bradley, Preetum Nakkiran, and Shuangfei Zhai. Controlled generation with distilled diffusion energy models and sequential monte carlo. In *The 28th International Conference on Artificial Intelligence and Statistics*, 2025. URL https://openreview.net/forum?id=6GyX0YRw8P.

Masatoshi Uehara, Yulai Zhao, Tommaso Biancalani, and Sergey Levine. Understanding reinforcement learning-based fine-tuning of diffusion models: A tutorial and review, 2024. URL https://arxiv.org/abs/2407.13734.

Pascal Vincent. A connection between score matching and denoising autoencoders. *Neural computation*, 23 (7):1661–1674, 2011.

Li K. Wenliang and Heishiro Kanagawa. Blindness of score-based methods to isolated components and mixing proportions, 2021. URL https://arxiv.org/abs/2008.10087.

Luhuan Wu, Brian L. Trippe, Christian A Naesseth, John Patrick Cunningham, and David Blei. Practical and asymptotically exact conditional sampling in diffusion models. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=eWKqr1zcRv.

Shahar Yadin, Noam Elata, and Tomer Michaeli. Classification diffusion models: Revitalizing density ratio estimation. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=d99yCfOnwK.

Hanlin Yu, Arto Klami, Aapo Hyvarinen, Anna Korba, and Omar Chehab. Density ratio estimation with conditional probability paths. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=Gn2izAiYzZ.

Fengzhe Zhang, Jiajun He, Laurence I Midgley, Javier Antorán, and José Miguel Hernández-Lobato. Efficient and unbiased sampling of boltzmann distributions via consistency models. *arXiv preprint arXiv:2409.07323*, 2024.

Fengzhe Zhang, Laurence I. Midgley, and José Miguel Hernández-Lobato. Efficient and unbiased sampling from boltzmann distributions via variance-tuned diffusion models, 2025a. URL https://arxiv.org/abs/2505.21005.

Leo Zhang, Peter Potaptchik, Jiajun He, Yuanqi Du, Arnaud Doucet, Francisco Vargas, Hai-Dang Dau, and Saifuddin Syed. Accelerated parallel tempering via neural transports, 2025b. URL https://arxiv.org/abs/2502.10328.

Mingtian Zhang, Oscar Key, Peter Hayes, David Barber, Brooks Paige, and Francois-Xavier Briol. Towards healing the blindness of score matching. In *NeurIPS 2022 Workshop on Score-Based Methods*, 2022. URL https://openreview.net/forum?id=Ij8G_k0iuL.

Xinwei Zhang, Zhiqiang Tan, and Zhijian Ou. Persistently trained, diffusion-assisted energy-based models. *Stat*, 12(1):e625, 2023. doi: https://doi.org/10.1002/sta4.625. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/sta4.625.

Yan Zhou, Adam M. Johansen, and John A.D. Aston. Toward automatic model comparison: An adaptive sequential monte carlo approach. *Journal of Computational and Graphical Statistics*, 25(3):701–726, 2016. doi: 10.1080/10618600.2015.1060885. URL https://doi.org/10.1080/10618600.2015.1060885.

Yaxuan Zhu, Jianwen Xie, Ying Nian Wu, and Ruiqi Gao. Learning energy-based models by cooperative diffusion recovery likelihood. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=AyzkDpuqcl.

Organization of the supplementary

The appendix is organized as follows. Appendix A summarizes general facts that will be useful for proofs and corresponding computations. In Appendix B, we describe the general framework of noising diffusion processes, as well as the particular case of Variance-Preserving (consistently used in our experiments) and Variance-Exploding schemes: we notably detail there the formulas related to SDE/ODE integrators and to the computation of the log-determinant terms arising from the use of deterministic transitions in DM-aMC samplers (see Section 4). In Appendix C, we provide the proofs of all theoretical results dispensed in Section 4. Finally, we precisely detail our experimental setting in Appendix D, along with additional numerical results.

A Preliminaries

A.1 Useful lemmas

Lemma 4 (Power series expansion of the matrix logarithm)**.** *Let* $(\alpha, \beta) \in \mathbb{R}^2$*. For any matrix* $\mathrm{M} \in \mathbb{R}^{d\times d}$ *satisfying* $\|\mathrm{M}\| < \min(1/|\alpha|, 1/|\beta|)$*, the following identities hold*

$$\log\left[(\mathrm{I}_d - \beta\mathrm{M})^{-1}(\mathrm{I}_d + \alpha\mathrm{M})\right] = \sum_{i=1}^{\infty} \frac{\beta^i - (-1)^i\alpha^i}{i}\mathrm{M}^i, \quad \log\left[(\mathrm{I}_d + \beta\mathrm{M})^{-1}(\mathrm{I}_d - \alpha\mathrm{M})\right] = \sum_{i=1}^{\infty} \frac{(-1)^i\beta^i - \alpha^i}{i}\mathrm{M}^i,$$

where log *denotes the matrix logarithm.*

Proof. This is an immediate corollary from (Hall, 2000, Theorem 3.6). □

Corollary 5. *Let* $(c_1, c_2, c_3) \in \mathbb{R}^3$ *with* $c_1 \neq 0$*. Let* $\mathrm{M} \in \mathbb{R}^{d\times d}$ *be a matrix satisfying* $\|\mathrm{M}\| < \min(1/|c_2|, 1/|c_3|)$*. Define the matrices*

$$\mathrm{M}_1 = c_1(\mathrm{I}_d + c_2\mathrm{M})^{-1}(\mathrm{I}_d - c_3\mathrm{M}) \ , \ \mathrm{M}_2 = c_1^{-1}(\mathrm{I}_d - c_3\mathrm{M})^{-1}(\mathrm{I}_d + c_2\mathrm{M}) \ .$$

Then we have

$$\log|\det \mathrm{M}_1| = d\log|c_1| + \sum_{i=1}^{\infty} \frac{(-1)^i c_2^i - c_3^i}{i} \mathrm{Tr}[\mathrm{M}^i] \ , \ \log|\det \mathrm{M}_2| = -d\log|c_1| + \sum_{i=1}^{\infty} \frac{c_3^i - (-1)^i c_2^i}{i} \mathrm{Tr}[\mathrm{M}^i] \ .$$

Proof. Consider such (c_1, c_2, c_3) and such matrix M. Note that the assumption on c_2 and c_3 guarantees the invertibility of $\mathrm{I}_d + c_2\mathrm{M}$ and $\mathrm{I}_d - c_3\mathrm{M}$. Regarding M_1, we have

$$\begin{aligned}
\log|\det M_1| &= d\log|c_1| + \log\left|\det\left((\mathrm{I}_d + c_2\mathrm{M})^{-1}(\mathrm{I}_d - c_3\mathrm{M})\right)\right| \\
&= d\log|c_1| + \log\left|\det(\mathrm{I}_d + c_2\mathrm{M})^{-1}\det(\mathrm{I}_d - c_3\mathrm{M})\right| \\
&= d\log|c_1| - \log|\det(\mathrm{I}_d + c_2\mathrm{M})| + \log|\det(\mathrm{I}_d - c_3\mathrm{M})| \\
&= d\log|c_1| - \log\det(\mathrm{I}_d + c_2\mathrm{M}) + \log\det(\mathrm{I}_d - c_3\mathrm{M}) \qquad \text{(Behrmann et al., 2019, Lemma 6)} \\
&= d\log|c_1| + \log\det\left((\mathrm{I}_d + c_2\mathrm{M})^{-1}(\mathrm{I}_d - c_3\mathrm{M})\right) \\
&= d\log|c_1| + \mathrm{Tr}\log\left((\mathrm{I}_d + c_2\mathrm{M})^{-1}(\mathrm{I}_d - c_3\mathrm{M})\right) . \qquad \text{(Hall, 2000, Theorem 3.10)}
\end{aligned}$$

Hence, we obtain the first result of Corollary 5 by using the second statement of Lemma 4 with $\beta = c_2$ and $\alpha = c_3$. Similar computations with M_2 lead to the second result. □

A.2 Discrete time setting for diffusion models

Following Karras et al. (2024); Grenioux et al. (2024), we define the time discretization $\{t_k\}_{k=0}^K \subset [0, T]$ accordingly to the growth (in log-scale) of the noise level $t \mapsto \sigma(t)$.

Given fixed boundary values $\sigma_{\min} > 0$ and $\sigma_{\max} > 0$, we define for any $k \in \{0, \dots, K\}$ the timestep t_k by

$$\log \sigma(t_k)^2 = \log \sigma_{\min}^2 + \frac{k}{K} \log \left(\frac{\sigma_{\max}^2}{\sigma_{\min}^2} \right) \iff \sigma(t_k)^2 = \sigma_{\min}^2 \left(\frac{\sigma_{\max}^2}{\sigma_{\min}^2} \right)^{k/K} \tag{27}$$

$$\iff t_k = \sigma^{-1} \left(\sigma_{\min} \left(\frac{\sigma_{\max}}{\sigma_{\min}} \right)^{k/K} \right) . \tag{28}$$

Depending on the type of noising, the map σ^{-1} may be explicit or not; in the latter case, we approximate it via a dichotomy scheme. In practice, $\sigma_{\max} = \sigma(T)$ and $\sigma_{\min}$ is chosen very close to 0 such that $t_K \approx T$ and $t_0 \approx 0$; hence, we have $p_{t_K} \approx \pi^{\text{base}}$ and $p_{t_0} \approx \pi$. We denote $\delta_k = t_{k+1} - t_k$.

A.3 General results on SDE/ODE integration

Assumption 2 (Integrability conditions on f and g)**.** *Coefficients $f : [0,T] \to \mathbb{R}$ and $g : [0,T] \to (0,\infty)$ are such that (a) f is integrable on $(0,T)$ and (b) g is integrable on $(0,T)$.*

Lemma 6 (SDE exponential integration)**.** *Let $T > 0$ and $b \in \mathbb{R}^d$. Consider the SDE defined on $[0,T]$ by $\mathrm{d}Y_t = f(t)(Y_t + b)\mathrm{d}t + g(t)\mathrm{d}B_t$, where coefficients f and g verify Assumption 2. Then, for any pair of time-steps (s,t) such that $T \geq t > s \geq 0$, the conditional distribution of Y_t given $Y_s = y_s \in \mathbb{R}^d$, denoted by $q_{t|s}(\cdot|y_s)$, verifies*

$$q_{t|s}(\cdot|y_s) = \mathrm{N}\left(\exp(\textstyle\int_s^t f(u)\mathrm{d}u) y_s + \left(\exp(\int_s^t f(u)\mathrm{d}u) - 1 \right) b, \int_s^t g^2(u) \exp(2 \textstyle\int_u^t f(r)\mathrm{d}r)\mathrm{d}u \, \mathrm{I}_d \right) .$$

Proof. Assume Assumption 2. Define the function $\zeta : t \in [0,T] \to \exp(-\int_0^t f(u)\mathrm{d}u)$ and consider the stochastic process $(Z_t)_{t\in[0,T]}$ defined by $Z_t = \zeta(t)Y_t$ for any $t \in [0,T]$. By Îto's formula, we have $\mathrm{d}Z_t = f(t)\zeta(t)b\mathrm{d}t + \zeta(t)g(t)\mathrm{d}B_t = -\dot{\zeta}(t)b\mathrm{d}t + \zeta(t)g(t)\mathrm{d}B_t$. Therefore, for any time-steps (s,t) such that $T \geq t > s \geq 0$, we have

$$\zeta(t)Y_t - \zeta(s)Y_s = \{\zeta(s) - \zeta(t)\}b + \textstyle\int_s^t \zeta(u)g(u)\mathrm{d}B_u \ ,$$

and then

$$Y_t = \exp(\textstyle\int_s^t f(u)\mathrm{d}u) Y_s + \left(\exp(\int_s^t f(u)\mathrm{d}u) - 1 \right) b + \int_s^t g(u) \exp(\textstyle\int_u^t f(r)\mathrm{d}r)\mathrm{d}B_u \ ,$$

which gives the result using Îto's isometry and that Y_s is independent from $(B_t - B_s)_{t\in[s,T]}$. □

The following lemma can be seen as the limit of Lemma 6 in the deterministic regime, *i.e.*, when $g(t) = 0$ for any $t \in [0,T]$.

Lemma 7 (ODE exponential integration)**.** *Let $T > 0$ and $b \in \mathbb{R}^d$. Consider the ODE defined on $[0,T]$ by $\mathrm{d}Y_t = f(t)[Y_t + b]\mathrm{d}t$, where coefficient f verifies Assumption 2. Then, for any pair of time-steps (s,t) such that $T \geq t > s \geq 0$, the ODE solution Y_t given $Y_s = y_s \in \mathbb{R}^d$ is defined by*

$$Y_t = \exp\left(\textstyle\int_s^t f(u)\mathrm{d}u \right) y_s + \left(\exp\left(\int_s^t f(u)\mathrm{d}u \right) - 1 \right) b \ .$$

Proof. Let $0 \leq s < t \leq T$, set $Z_t = \exp(-F(t))Y_t$, where $F(t) = \int_0^t f(u)\mathrm{d}u$, then

$$\mathrm{d}Z_t = f(t) \exp(-F(t))b\mathrm{d}t \ ,$$

which implies that

$$Z_t = Z + (\exp(-F(s)) - \exp(-F(t)))\, b \ ,$$

which gives the result. □

A.4 Review of score and energy matching methods

Consider the noising diffusion process given by the SDE (2). In this section, we review a selection of methods used to learn the scores, the time scores and/or the log-densities (*i.e.*, energies) of the marginal distributions $(p_t)_{t\in[0,T]}$ associated to this process, based on samples from the target distribution π with unnormalized density γ. While the presented score matching (and time score matching) techniques have widely been experimented within the diffusion model community, the evoked log-density estimation (also referred to as energy matching) approaches are much more recent, and only provide a small glimpse into the pretty young field of research to which they belong. In the following, we will denote $r(t) = S(t)\sigma(t)$, where coefficients S and σ are introduced in (3) to marginally characterize diffusion models. We adopt a consistent notation for neural networks: $\mathbf{U}_t^\theta$ is used to learn the log-density $\log p_t$, $\mathbf{s}_t^\theta$ to learn the score $\nabla \log p_t$, and $\mathbf{u}_t^\theta$ to learn the time score $\partial_t \log p_t$.

Denoising Score Matching (DSM) (Song et al., 2021). This is the standard score matching loss used in the diffusion-based generative modeling literature. It relies on the so-called Tweedie identity

$$\nabla \log p_t(x_t) = \mathrm{E}[\nabla \log q_{t|0}(x_t|X_0)] \ , \ X_0 \sim q_{0|t}(\cdot|x_t) \ ,$$

where $q_{t|0}$ is the *tractable* noising transition kernel between times 0 and t, see (7), and $q_{0|t}$ is the related denoising transition kernel, which verifies by Bayes property $q_{0|t}(x_0|x_t) \propto \gamma(x_0) q_{t|0}(x_t|x_0)$. This gives rise to the following objective for estimating the score function $(t,x) \mapsto \nabla \log p_t(x)$ by a neural network $(t,x) \mapsto \mathbf{s}_t^\theta(x)$

$$\mathcal{L}_{\mathrm{DSM}}(\theta) = \mathrm{E}\left[\left\| \mathbf{s}_t^\theta(X_t) + \frac{Z}{r(t)} \right\|_2^2 \right] \ , \ X_t = S(t)X_0 + r(t)Z \ ,$$

where $t \sim \mathrm{U}(0,T)$, $X_0 \sim \pi$ and $Z \sim \mathrm{N}(0, \mathrm{I}_d)$. In practice, one rather uses a reweighted version of this objective given by

$$\tilde{\mathcal{L}}_{\mathrm{DSM}}(\theta) = \mathrm{E}\left[r^2(t) \left\| \mathbf{s}_t^\theta(X_t) + \frac{Z}{r(t)} \right\|_2^2 \right] = \mathrm{E}\left[\left\| r(t)\mathbf{s}_t^\theta(X_t) + Z \right\|_2^2 \right] \ .$$

To further reduce its variance with respect to the noise variable, one may consider applying the antithetic trick on Z variable and thus obtain the loss function

$$\tilde{\mathcal{L}}_{\mathrm{DSM}}^{\mathrm{anti}}(\theta) = \mathrm{E}\left[\frac{1}{2} \left\| r(t)\mathbf{s}_t^\theta(X_t) + Z \right\|_2^2 + \frac{1}{2} \left\| r(t)\mathbf{s}_t^\theta\left(X_t^-\right) - Z \right\|_2^2 \right] \ ,$$
$$\text{with } X_t = S(t)X_0 + r(t)Z \ , \ X_t^- = S(t)X_0 - r(t)Z.$$

In practice, the DSM objective described above still exhibits high variance. To ensure its robustness and stability for large-scale applications, an equivalent objective, coined EDM, was proposed by Karras et al. (2022), which specifically relies on preconditioning guidelines for the neural network $\mathbf{s}^\theta$. In our experiments, the "DSM objective" will systematically refer to this specific EDM training loss, enhanced with the antithetic trick, whose success has been widely proven over the last few years for generative tasks.

Target Score Matching (TSM) (Bortoli et al., 2024). Alternatively, by operating a change-of-variable in the Tweedie's formula, the following identity also holds

$$\nabla \log p_t(x_t) = \frac{1}{S(t)} \mathrm{E}[\nabla \log \gamma(X_0)] \ , \ X_0 \sim q_{0|t}(\cdot|x_t) \ .$$

This gives rise to the following objective for estimating the score function $(t,x) \mapsto \nabla \log p_t(x)$ by a neural network $(t,x) \mapsto \mathbf{s}_t^\theta(x)$

$$\mathcal{L}_{\mathrm{TSM}}(\theta) = \mathrm{E}\left[\left\| \mathbf{s}_t^\theta(X_t) - \frac{\nabla \log \gamma(X_0)}{S(t)} \right\|_2^2 \right] \ , \ X_t = S(t)X_0 + r(t)Z \ ,$$

where $t \sim \mathrm{U}(0,T)$, $X_0 \sim \pi$ and $Z \sim \mathrm{N}(0, \mathrm{I}_d)$. In practice, one rather uses a reweighted version of this objective given by

$$\tilde{\mathcal{L}}_{\mathrm{TSM}}(\theta) = \mathrm{E}\left[S^2(t)\left\|\mathbf{s}_t^\theta\left(X_t\right) - \frac{\nabla \log \gamma(X_0)}{S(t)}\right\|_2^2\right] = \mathrm{E}\left[\left\|S(t)\mathbf{s}_t^\theta\left(X_t\right) - \nabla \log \gamma(X_0)\right\|_2^2\right] ,$$

which itself can be improved via the antithetic trick as

$$\tilde{\mathcal{L}}^{\mathrm{anti}}_{\mathrm{TSM}}(\theta) = \mathrm{E}\left[\frac{1}{2}\left\|S(t)\mathbf{s}_t^\theta\left(X_t\right) - \nabla \log \gamma(X_0)\right\|_2^2 + \frac{1}{2}\left\|S(t)\mathbf{s}_t^\theta\left(X_t^-\right) - \nabla \log \gamma(X_0)\right\|_2^2\right] ,$$
$$\text{with } X_t = S(t)X_0 + r(t)Z \ , \ X_t^- = S(t)X_0 - r(t)Z.$$

In our experiments, the "TSM objective" will systematically refer to the training loss function $\tilde{\mathcal{L}}^{\mathrm{anti}}_{\mathrm{TSM}}$. As originally proposed by Bortoli et al. (2024), this loss can also be combined with preconditioning schemes to reduce its variance in practice; however, since those are not compatible with the preconditioning directives from Karras et al. (2022), we do not integrate them in our numerical experiments.

Time Score Matching (tSM)(Guth et al., 2025b; Yu et al., 2025). Interestingly, the time score function has a similar decomposition

$$\partial_t \log p_t(x_t) = \mathrm{E}[\partial_t \log q_{t|0}(x_t|X_0)] \ , \ X_0 \sim q_{0|t}(\cdot|x_t) \ .$$

Since the conditional time derivative $\partial_t \log q_{t|0}$ is as tractable as the conditional score $\nabla \log q_{t|0}$, this gives rise to the following objective for estimating the time score function $(t,x) \mapsto \partial_t \log p_t(x)$ by a neural network $(t,x) \mapsto \mathbf{u}_t^\theta\left(x\right)$

$$\mathcal{L}_{\mathrm{tSM}}(\theta) = \mathrm{E}\left[\left\|\mathbf{u}_t^\theta\left(X_t\right) - u^{\mathrm{target}}_{\mathrm{tSM}}(t, X_0, Z)\right\|_2^2\right] \ , \ X_t = S(t)X_0 + r(t)Z \ ,$$

where $u^{\mathrm{target}}_{\mathrm{tSM}}(t,x_0,z) = -[\dot{r}(t)/r(t)]\left(d - \|z\|^2\right) + [\dot{S}(t)/r(t)]x_0^\top z$, $t \sim \mathrm{U}(0,T)$, $X_0 \sim \pi$ and $Z \sim \mathrm{N}(0,\mathrm{I}_d)$. Similarly, this objective may include antithetic trick and rewrite as

$$\mathcal{L}^{\mathrm{anti}}_{\mathrm{tSM}}(\theta) = \mathrm{E}\left[\frac{1}{2}\left\|\mathbf{u}_t^\theta\left(X_t\right) - u^{\mathrm{target}}_{\mathrm{tSM}}(t, X_0, Z)\right\|_2^2 + \frac{1}{2}\left\|\mathbf{u}_t^\theta\left(X_t^-\right) - u^{\mathrm{target}}_{\mathrm{tSM}}(t, X_0, -Z)\right\|_2^2\right] ,$$
$$\text{with } X_t = S(t)X_0 + r(t)Z \ , \ X_t^- = S(t)X_0 - r(t)Z.$$

As such, it has been observed that the derived objective exhibits very high variance, even more than score matching methods. While Guth et al. (2025b) explore a reweighting precisely adjusted to the VE noising scheme, Yu et al. (2025) propose an alternative reweighting in the context of the VP noising scheme; we implement the latter formulation with the antithetic trick, to which the "tSM objective" will systematically refer in our experiments. Note that by including a change-of-variable into the expression of the time score, we may obtain a target-like version of the tSM objective given by

$$\mathcal{L}_{\mathrm{tTSM}}(\theta) = \mathrm{E}\left[\left\|\mathbf{u}_t^\theta\left(X_t\right) - u^{\mathrm{target}}_{\mathrm{tTSM}}(t, X_0, Z)\right\|_2^2\right] \ , \ X_t = S(t)X_0 + r(t)Z \ ,$$

where $u^{\mathrm{target}}_{\mathrm{tTSM}}(t,x_0,z) = -\left\{[\dot{S}(t)/S(t)]x_0 + [\sigma(t)\dot{r}(t)/r(t)]z\right\}^\top \nabla \log \gamma(x_0)$, $t \sim \mathrm{U}(0,T)$, $X_0 \sim \pi$ and $Z \sim \mathrm{N}(0,\mathrm{I}_d)$, along with its antithetic-like version

$$\mathcal{L}^{\mathrm{anti}}_{\mathrm{tTSM}}(\theta) = \mathrm{E}\left[\tfrac{1}{2}\left\|\mathbf{u}_t^\theta\left(X_t\right) - u^{\mathrm{target}}_{\mathrm{tTSM}}(t, X_0, Z)\right\|_2^2 + \tfrac{1}{2}\left\|\mathbf{u}_t^\theta\left(X_t^-\right) - u^{\mathrm{target}}_{\mathrm{tTSM}}(t, X_0, -Z)\right\|_2^2\right] ,$$
$$\text{with } X_t = S(t)X_0 + r(t)Z \ , \ X_t^- = S(t)X_0 - r(t)Z.$$

While this objective is enriched with the information of the target score $\nabla \log \gamma$, we did not use this objective in our experiments due to its variance instability during training procedure.

Log-density Fokker-Planck-Equation (LFPE) (Lai et al., 2023; Shi et al., 2024; Sun et al., 2024a). A key property of the noising SDE (2) is that the induced log-densities $(p_t)_{t\in[0,1]}$ can be described by a partial differential equation called the *Fokker-Planck equation* (Øksendal, 2003), whose formulation can be written as

$$\partial_t \log p_t(x) = \mathcal{F}[\log p](t,x) \overset{\text{def}}{=} \frac{1}{2}g^2(t)\left[\operatorname{div}(\nabla \log p_t)(x) + \|\nabla \log p_t(x)\|_2^2\right] - f(t)\{d + x^\top \nabla \log p_t(x)\}\ ,$$

where div denotes the divergence operator defined by $\operatorname{div} \mathrm{F} = \operatorname{Tr}[\nabla \mathrm{F}]$. This gives rise to the following objective for estimating the log-density $(t,x) \mapsto \log p_t(x)$ by a neural network $(t,x) \mapsto \mathbf{U}_t^\theta(x)$

$$\mathcal{L}_{\text{LFPE}}(\theta) = \mathrm{E}\left[\left\|\partial_t \mathbf{U}_t^\theta(X_t) - \operatorname{sg}\left\{\mathcal{F}[\mathbf{U}^\theta](t,X_t)\right\}\right\|_2^2\right]\ ,\ X_t = S(t)X_0 + r(t)Z\ ,$$

where sg denotes the stop-gradient[7] operator with respect to parameter θ, $t \sim \mathrm{U}(0,T)$, $X_0 \sim \pi$ and $Z \sim \mathrm{N}(0, \mathrm{I}_d)$. In this case too, we can derive an objective based on the antithetic trick

$$\mathcal{L}^{\text{anti}}_{\text{LFPE}}(\theta) = \mathrm{E}\left[\frac{1}{2}\left\|\partial_t \mathbf{U}_t^\theta(X_t) - \operatorname{sg}\left\{\mathcal{F}[\mathbf{U}^\theta](t,X_t)\right\}\right\|_2^2 + \frac{1}{2}\left\|\partial_t \mathbf{U}_t^\theta(X_t^-) - \operatorname{sg}\left\{\mathcal{F}[\mathbf{U}^\theta](t,X_t^-)\right\}\right\|_2^2\right]\ ,$$
$$\text{with } X_t = S(t)X_0 + r(t)Z\ ,\ X_t^- = S(t)X_0 - r(t)Z.$$

In our experiments, the "LFPE objective" will always refer to the training loss $\mathcal{L}^{\text{anti}}_{\text{LFPE}}$.

Approximate LFPE (aLFPE) (Plainer et al., 2025). In the case where the target term $\mathcal{F}[\mathbf{U}^\theta]$ is not detached with respect to θ in $\mathcal{L}_{\text{LFPE}}$, the main numerical burden lies in the computation of the divergence term $\operatorname{div}(\nabla \mathbf{U}^\theta)$ when the dimension is large. To reduce this overhead, Plainer et al. (2025) propose to instead consider a first-order statistical estimation of the residual term $\mathcal{R}^\theta(t,x) = \mathcal{F}[\mathbf{U}^\theta](t,x) - \partial_t \mathbf{U}_t^\theta(x)$ given by $\tilde{\mathcal{R}}^\theta(t,x) = \mathrm{E}_v[\tilde{\mathcal{R}}^\theta(t,x;v)]$, with $v \sim \mathrm{N}(0,\sigma^2\mathrm{I}_d)$ for a small $\sigma > 0$, where[8]

$$\begin{aligned}\tilde{\mathcal{R}}^\theta(t,x;v) = {}& \frac{1}{2}g^2(t)\left[\left(\frac{v}{\sigma}\right)^\top \frac{\nabla \mathbf{U}_t^\theta(x+v) - \nabla \mathbf{U}_t^\theta(x-v)}{2\sigma}\right] \\ & + \frac{1}{2}\left[\frac{1}{2}g^2(t)\left\|\nabla \mathbf{U}_t^\theta(x+v)\right\|_2^2 - f(t)\{d + (x+v)^\top \nabla \mathbf{U}_t^\theta(x+v)\} - \partial_t \mathbf{U}_t^\theta(x+v)\right] \\ & + \frac{1}{2}\left[\frac{1}{2}g^2(t)\left\|\nabla \mathbf{U}_t^\theta(x-v)\right\|_2^2 - f(t)\{d + (x-v)^\top \nabla \mathbf{U}_t^\theta(x-v)\} - \partial_t \mathbf{U}_t^\theta(x-v)\right]\ .\end{aligned}$$

This gives rise to the following objective

$$\mathcal{L}_{\text{aLFPE}}(\theta) = \mathrm{E}\left[\left(\sum_{i=1}^N \tilde{\mathcal{R}}^\theta(t,X_t;v_i^{X_t})\right)\left(\sum_{j=1}^N \tilde{\mathcal{R}}^\theta(t,X_t;v_j^{X_t})\right)\right]\ ,\ X_t = S(t)X_0 + r(t)Z\ ,$$

where $t \sim \mathrm{U}(0,T)$, $X_0 \sim \pi$, $Z \sim \mathrm{N}(0,\mathrm{I}_d)$ and $\{v_i^{X_t}, v_j^{X_t}\}_{j=1}^N$ are $2N$ independent samples from $\mathrm{N}(0,\sigma^2\mathrm{I}_d)$ defined for each input X_t. Overall, this formulation avoids the need of the divergence computation (while maintaining backpropagation through the scores), at the cost of non-negligible statistical error. Following the guidelines from Plainer et al. (2025), we consistently set in our experiments $\sigma = 0.0001$, but choose $N = 64$ (instead of $N = 1$ as originally proposed) to reduce the variance of the loss, and bring it into the most favorable setting. We also choose to keep the use of auto-differentiation to compute the time derivative $\partial_t \mathbf{U}^\theta$ instead of using finite difference approximation as suggested by Plainer et al. (2025), as it brings more stability during training. In our experiments, we will systematically refer to this version of $\mathcal{L}_{\text{aLFPE}}$ as the "aLFPE objective".

[7]While cited related works did not consider detaching the term $\mathcal{F}[\mathbf{U}^\theta]$ with respect to θ in their respective formulation, we made this choice to avoid backpropagation through both first and second-order derivatives of $\mathbf{U}^\theta$, which was computationally infeasible in the high-dimensional settings considered in this paper. Nonetheless, we emphasize that, in our early experiments, we observed unchanged results on pure log-density estimation tasks for small dimensional settings, thereby suggesting that our methodology remains sound.

[8]Even though this objective features an additional term compared to the one stated in Equation (12) from Plainer et al. (2025), it is consistent with the related code available at `https://github.com/noegroup/ScoreMD`. This extra term actually originates from the use of the antithetic trick on the Gaussian variable v.

Radon-Nikodym Estimator (RNE) (He et al., 2025). Alternatively, a discrete-time formulation of the LFPE objective has been proposed to learn the log-densities $(\log p_t)_{t\in[0,T]}$, based on the Bayes's rule (ideally satisfied by DMs) stating that for any times $(s,t) \in [0,T]^2$ and any inputs x_s and x_t, we have $p_t(x_t)q_{s|t}(x_s|x_t) = p_s(x_s)q_{t|s}(x_t|x_s)$, where $q_{s|t}$ and $q_{t|s}$ correspond to related stochastic transition kernels (see Section 2.1). Enforcing this consistency with log-densities can thus be translated into the following objective for estimating the log-density $(t,x) \mapsto \log p_t(x)$ by a neural network $(t,x) \mapsto \mathbf{U}_t^\theta(x)$

$$\mathcal{L}_{\mathrm{RNE}}(\theta) = \mathrm{E}\left[\left\|\mathbf{U}_t^\theta(X_t) - \mathbf{U}_s^\theta(X_s) - \mathrm{sg}\{\log q_{t|s}(X_t|X_s) - \log q_{s|t}^\theta(X_s|X_t)\}\right\|_2^2\right] ,$$
$$\text{with } X_s = S(s)X_0 + r(s)Z \ , \ X_t = [S(t)/S(s)]X_s + r(t)\{1-\sigma^2(s)/\sigma^2(t)\}^{1/2}\tilde{Z} \ ,$$

where sg denotes the stop-gradient operator with respect to parameter θ, $(s,t) \sim \mathrm{U}(\{(t_k,t_{k+1}) : k \in [0,K-1]\})$ with $\{t_k\}_{k=0}^{K-1}$ being a discretization of time interval $[0,T]$, $X_0 \sim \pi$ and $(Z,\tilde{Z}) \sim \mathrm{N}(0,\mathrm{I}_d) \otimes \mathrm{N}(0,\mathrm{I}_d)$. While $q_{t|s}$ denotes a *noising* transition kernel[9], that is tractable by (7), $q_{s|t}^\theta$ is an approximate *denoising* transition kernel, that may be computed via the learned score $\nabla\mathbf{U}^\theta$, see for instance (8). In this case too, one may consider the variant featuring the antithetic trick

$$\mathcal{L}_{\mathrm{RNE}}^{\mathrm{anti}}(\theta) = \mathrm{E}\Bigg[\frac{1}{2}\left\|\mathbf{U}_t^\theta(X_t) - \mathbf{U}_s^\theta(X_s) - \mathrm{sg}\{\log q_{t|s}(X_t|X_s) - \log q_{s|t}^\theta(X_s|X_t)\}\right\|_2^2$$
$$+ \frac{1}{2}\left\|\mathbf{U}_t^\theta(X_t^-) - \mathbf{U}_s^\theta(X_s^-) - \mathrm{sg}\{\log q_{t|s}(X_t^-|X_s^-) - \log q_{s|t}^\theta(X_s^-|X_t^-)\}\right\|_2^2\Bigg] ,$$
$$\text{with } X_s = S(s)X_0 + r(s)Z \ , \ X_t = [S(t)/S(s)]X_s + r(t)\{1-\sigma^2(s)/\sigma^2(t)\}^{1/2}\tilde{Z} \ ,$$
$$\text{and } X_s^- = S(s)X_0 - r(s)Z \ , \ X_t^- = [S(t)/S(s)]X_s^- + r(t)\{1-\sigma^2(s)/\sigma^2(t)\}^{1/2}\tilde{Z} \ .$$

Contrary to the LFPE objective, the obtained loss function does not require backpropagation through the time derivative $\partial_t\mathbf{U}^\theta$, which represents a significant computational advantage. However, $\mathcal{L}_{\mathrm{RNE}}$ suffers from a severe bias-variance tradeoff with respect to the time gap $\delta = t-s$ for selected times s and t: if δ is too large, then the denoising approximation obtained via $q_{s|t}^\theta$ may be ineffective and bring much bias; on the other hand, if δ is too small, the resulting objective may be prone to high variance. While He et al. (2025) propose to use the Euler-Maruyama estimation for $q_{s|t}^\theta$, see (8), we rather consider the Exponential Integration, see Appendix B.2 and Appendix B.3 for the formulas, which provides better accuracy for larger gap δ. In our experiments, we will systematically refer to this version of $\mathcal{L}_{\mathrm{RNE}}^{\mathrm{anti}}$ as the "RNE objective".

B Details on (de)noising diffusion processes

In this section, we consider a target probability distribution $\pi \in \mathcal{P}(\mathbb{R}^d)$ and a pair of time points (s,t) satisfying $T \geq t > s \geq 0$. We present technical derivations related to the integration of (de)noising diffusion processes under a unified framework, covering the generic setting (Appendix B.1), the Variance-Preserving scheme (Appendix B.2), and the Variance-Exploding scheme (Appendix B.3). Throughout, the notation $\mathbf{s}_t(x)$ and $\mathbf{H}_t(x)$ denotes, respectively, exact or approximate evaluation of the score $\nabla\log p_t(x)$ and the Hessian $\nabla^2\log p_t(x)$. This unified formulation allows our computations to encompass both idealized and practical regimes considered in this paper.

B.1 General noising scheme

Here, we consider the most general form of SDE (2), where f and g both verify Assumption 2.

Lemma 8 (Exact noising SDE integration - General case)**.** *The conditional distribution of X_t given $X_s = x_s \in \mathbb{R}^d$ is defined by the Gaussian kernel*

$$q_{t|s}(\cdot|x_s) = \mathrm{N}\left(\{S(t)/S(s)\}x_s, S(t)^2\{\sigma^2(t)-\sigma^2(s)\}\,\mathrm{I}_d\right) \ ,$$

[9]While He et al. (2025) propose to replace $q_{t|s}$, though tractable, by its Euler-Maruyama estimation, our implementation relies rather on its exact formulation to avoid bringing additional approximation error into the loss.

where $S(t) = \exp(\int_0^t f(u)\mathrm{d}u)$ *and* $\sigma^2(t) = \int_0^t g^2(u)/S(u)^2\mathrm{d}u$.

Proof. This is an immediate corollary of Lemma 6. □

Lemma 9 (Approximate denoising SDE integration - General case)**.** *Denote* $\delta = t - s$. *Then, the conditional distribution of* X_t *given* $X_s = x_s \in \mathbb{R}^d$ *may be approximated by the Gaussian kernel*

$$q_{s|t}(\cdot|x_t) = \mathrm{N}\left((1 - f(t)\delta)x_t + g^2(t)\delta\,\mathbf{s}_t\left(x_t\right), g^2(t)\delta\,\mathrm{I}_d\right) \ ,$$

Proof. This result is a straightforward application of the Euler-Maruyama scheme applied to SDE (5). □

Lemma 10 (Approximate noising ODE integration - General case)**.** *Denote* $\delta = t - s$. *Then, the solution at time* t *of the forward probability flow ODE* (6) *starting from* $x_s \in \mathbb{R}^d$ *at time* s *may be approximated in two ways:*

$$\tilde{\mathrm{T}}_{t|s}(x_s) = x_s + \delta v(s, x_s) \qquad \textit{(Euler method: explicit, 1st order)}$$

$$\mathrm{T}_{t|s}(x_s) = x_s + \delta v\left(\frac{s+t}{2}, \frac{x_s + \mathrm{T}_{t|s}(x_s)}{2}\right) \qquad \textit{(Midpoint method : implicit, 2nd order)}$$

$$\textit{where } v(u, x) = f(u)x - \frac{g^2(u)}{2}\mathbf{s}_u\left(x\right) \ .$$

Lemma 11 (Approximate denoising ODE integration - General case)**.** *Denote* $\delta = t - s$. *Then, the solution at time* s *of the backward probability flow ODE* (6) *starting from* $x_t \in \mathbb{R}^d$ *at time* t *may be approximated in two ways:*

$$\tilde{\mathrm{T}}_{s|t}(x_t) = x_t - \delta v(t, x_t) \qquad \textit{(Euler method : explicit, 1st order)}$$

$$\mathrm{T}_{s|t}(x_t) = x_t - \delta v\left(\frac{s+t}{2}, \frac{\mathrm{T}_{s|t}(x_t) + x_t}{2}\right) \qquad \textit{(Midpoint method: implicit, 2nd order)}$$

$$\textit{where } v(u, x) = f(u)x - \frac{g^2(u)}{2}\mathbf{s}_u\left(x\right) \ .$$

Remark on the mutual invertibility of the ODE integrators. It is easy to verify that the noising and denoising implicit Midpoint integrators described above are mutual inversible maps, *i.e.*, we have $\mathrm{T}_{s|t} \circ \mathrm{T}_{t|s} = \mathrm{T}_{t|s} \circ \mathrm{T}_{s|t} = \mathrm{Id}$. However, this is not the case for the Euler maps $\tilde{\mathrm{T}}_{s|t}$ and $\tilde{\mathrm{T}}_{t|s}$.

Lemma 12 (Formula for the Jacobian of the Midpoint integrators)**.** *Let* $\delta > 0$, *and let define the numerical constants* $c_1(\delta)$, $c_2(\delta)$ *and* $c_3(\delta)$ *as*

$$c_1(\delta) = \frac{1 + (\delta/2)f((s+t)/2)}{1 - (\delta/2)f((s+t)/2)} \ , \ c_2(\delta) = \frac{\delta}{4}\frac{g^2\left(\frac{s+t}{2}\right)}{1 - \frac{\delta}{2}f\left(\frac{s+t}{2}\right)} \ , \ c_3(\delta) = \frac{\delta}{4}\frac{g^2\left(\frac{s+t}{2}\right)}{1 + \frac{\delta}{2}f\left(\frac{s+t}{2}\right)} \ .$$

Consider the same notation as in Lemma 10 and Lemma 11. Assume that there exists $L > 0$ *such that* $\mathbf{s}_{(s+t)/2}$ *is* L*-Lipschitz. Then for any positive step-size* $\delta = t - s$ *such that* $\max\left(|c_2(\delta)|, |c_3(\delta)|\right) < 1/L$, *the Jacobians of Midpoint integration maps* $\mathrm{T}_{t|s}$ *and* $\mathrm{T}_{s|t}$, *respectively denoted by* $J_{t|s}$ *and* $J_{s|t}$, *verify for any inputs* $x_s \in \mathbb{R}^d$ *and* $x_t \in \mathbb{R}^d$

$$J_{t|s}(x_s) = c_1(\delta)\left(\mathrm{I}_d + c_2(\delta)A(x_s)\right)^{-1}\left(\mathrm{I}_d - c_3(\delta)A(x_s)\right) \ ,$$
$$J_{s|t}(x_t) = c_1(\delta)^{-1}\left(\mathrm{I}_d - c_3(\delta)B(x_t)\right)^{-1}\left(\mathrm{I}_d + c_2(\delta)B(x_t)\right) \ ,$$

where $A(x_s) = \mathbf{H}_{(s+t)/2}\left(\frac{x_s + \mathrm{T}_{t|s}(x_s)}{2}\right)$ *and* $B(x_t) = \mathbf{H}_{(s+t)/2}\left(\frac{x_t + \mathrm{T}_{s|t}(x_t)}{2}\right)$.

Proof. The result from Lemma 12 follows from the factorization of the following identities, inherited from the implicit expressions of $\mathrm{T}_{t|s}$ and $\mathrm{T}_{s|t}$,

$$J_{t|s}(x_s) = \left(\left(1 - \frac{\delta}{2}f\left(\frac{s+t}{2}\right)\right)\mathrm{I}_d + \frac{\delta}{4}g^2\left(\frac{s+t}{2}\right)A(x_s)\right)^{-1}\left(\left(1 + \frac{\delta}{2}f\left(\frac{s+t}{2}\right)\right)\mathrm{I}_d - \frac{\delta}{4}g^2\left(\frac{s+t}{2}\right)A(x_s)\right) \ ,$$

$$J_{s|t}(x_t) = \left(\left(1 + \frac{\delta}{2}f\left(\frac{s+t}{2}\right)\right)\mathrm{I}_d - \frac{\delta}{4}g^2\left(\frac{s+t}{2}\right)B(x_t)\right)^{-1}\left(\left(1 - \frac{\delta}{2}f\left(\frac{s+t}{2}\right)\right)\mathrm{I}_d + \frac{\delta}{4}g^2\left(\frac{s+t}{2}\right)B(x_t)\right) \ .$$

Here, the assumption on δ guarantees the invertibility of the matrices $\mathrm{I}_d + c_2(\delta)A(x_s)$ and $\mathrm{I}_d - c_3(\delta)B(x_t)$. $\square$

Proposition 13 (Exact expression of the Jacobian log-determinants of the Midpoint integrators via power series)**.** *Consider the same notation as in Lemma 12. Assume that there exists $L > 0$ such that $\mathbf{s}_{(s+t)/2}$ is L-Lipschitz. Then, for any positive step-size $\delta = t - s$ such that $\max(|c_2(\delta)|, |c_3(\delta)|) < 1/L$, for any inputs $x_s \in \mathbb{R}^d$ and $x_t \in \mathbb{R}^d$, we have*

$$\log\left|\det J_{t|s}(x_s)\right| = \sum_{i=0}^{\infty} a_i(s,t)\,\mathrm{Tr}([A(x_s)]^i)\ ,$$
$$\log\left|\det J_{s|t}(x_t)\right| = \sum_{i=0}^{\infty} b_i(s,t)\,\mathrm{Tr}([B(x_t)]^i)\ ,$$

where $\{a_i(s,t), b_i(s,t)\}_{i=0}^{\infty}$ are numerical coefficients defined by

$$a_0(s,t) = -b_0(s,t) = d\log\left[\left|\frac{1+(\delta/2)f((s+t)/2)}{1-(\delta/2)f((s+t)/2)}\right|\right]\ ,$$
$$a_i(s,t) = \frac{\delta^i}{4^i}g^{2i}\left(\frac{s+t}{2}\right)\frac{(-1)^i\left(1+\frac{\delta}{2}f\left(\frac{s+t}{2}\right)\right)^i - \left(1-\frac{\delta}{2}f\left(\frac{s+t}{2}\right)\right)^i}{i\left(1-\frac{\delta^2}{4}f^2\left(\frac{s+t}{2}\right)\right)^i} \text{ for any } i \geq 1\ ,$$
$$b_i(s,t) = \frac{\delta^i}{4^i}g^{2i}\left(\frac{s+t}{2}\right)\frac{\left(1-\frac{\delta}{2}f\left(\frac{s+t}{2}\right)\right)^i - (-1)^i\left(1+\frac{\delta}{2}f\left(\frac{s+t}{2}\right)\right)^i}{i\left(1-\frac{\delta^2}{4}f^2\left(\frac{s+t}{2}\right)\right)^i} \text{ for any } i \geq 1\ .$$

Proof. Consider the Jacobian matrices $J_{t|s}(x_s)$ and $J_{s|t}(x_t)$ introduced in Lemma 12. Note that we have $\|A(x_s)\| < \min(1/|c_2(\delta)|, 1/|c_3(\delta)|)$ and $\|B(x_t)\| < \min(1/|c_2(\delta)|, 1/|c_3(\delta)|)$ based on the assumptions on $\mathbf{s}_{(s+t)/2}$ and δ. This allows us to apply Corollary 5 on $J_{t|s}(x_s)$ and $J_{s|t}(x_t)$, respectively with $\mathrm{M} = A(x_s)$ and $\mathrm{M} = B(x_t)$, to obtain their expansion series in a straightforward manner. $\square$

Remark on the δ-assumption in Lemma 12 and Proposition 13. For any general noise schedule defined by coefficients f and g, the assumption $\max(|c_2(\delta)|, |c_3(\delta)|) < 1/L$ can be rephrased into $\delta = O(1/L)$, by considering limit approximations of coefficients $c_2(\delta)$ and $c_3(\delta)$ in the asymptotic regime $\delta \to 0$. Below, we present a rigorous expression of this upper bound on δ for the noising schemes considered in this paper, that is the *Variance-Preserving* approach (see Appendix B.2) and the *Variance-Exploding* approach (see Appendix B.3).

B.2 Variance-Preserving diffusion

Consider the noising SDE (2) where $f(t) = -g^2(t)/2$ and g being such that $\int_0^T g^2(s)\mathrm{d}s \gg 1$, with arbitrary volatility coefficient $\sigma > 0$,

$$\mathrm{d}X_t = -\frac{g^2(t)X_t}{2}\mathrm{d}t + \sigma g(t)\mathrm{d}W_t\ ,\ X_0 \sim \pi\ . \tag{29}$$

This noising scheme, known as the *Variance-Preserving* (VP) scheme (Song et al., 2021), is largely used in score-based generative models. In the following, we denote $\alpha_t = \int_0^t g^2(t)\mathrm{d}t$ for any $t \in [0,T]$.

On the choice of the g-schedule. Previous works have considered a linear schedule $g^2(t) = \beta_{\min}(1 - t/T) + \beta_{\max}(t/T)$ where $\beta_{\min} = 0.1$, $\beta_{\max} \in \{10, 20\}$ and $T = 1$, see e.g., Song et al. (2021) or cosine parameterization (Nichol & Dhariwal, 2021), which has been proved to perform better in generative modeling. In our sampling experiments, we did not observe any significant difference between these two settings. Hence, we fix the linear schedule to be the default setting for our numerics, and let σ be arbitrarily chosen.

Lemma 14 (Exact noising SDE integration - VP case)**.** *The conditional distribution of X_t given $X_s = x_s \in \mathbb{R}^d$ is defined by the Gaussian kernel*

$$q_{t|s}(\cdot|x_s) = \mathrm{N}\left(\sqrt{1-\lambda^f_{s,t}}x_s, \sigma^2\lambda^f_{s,t}\mathrm{I}_d\right)\ ,$$

with $\lambda_{s,t}^f = 1 - \exp(\alpha_s - \alpha_t)$. *Since* $p_T(x) = \int_{\mathbb{R}^d} p_{T|0}(x|x_0)\mathrm{d}\pi(x_0)$, *it results that* $p_T \approx \mathrm{N}(0, \sigma^2 \mathrm{I}_d)$.

Proof. Lemma 6 applied on noising SDE (29). □

Based on the previous lemma, the interpolation coefficients in Equation (3) are given by

$$S(t) = \exp(-\alpha_t/2) \text{ and } \sigma(t) = \sigma\sqrt{1 - \exp(-\alpha_t)}.$$

In particular, $t \mapsto \sigma(t)$ is not explicitly invertible. Following Lemma 14, the VP scheme is an 'ergodic' noising scheme, converging exponentially fast to the Gaussian distribution $\mathrm{N}(0, \sigma^2 \mathrm{I}_d)$; therefore, we have $\pi^{\text{base}} = \mathrm{N}(0, \sigma^2 \mathrm{I}_d)$ in this setting. Moreover, under mild assumptions on π, the denoising SDE (5) writes as

$$\mathrm{d}X_t = -\frac{g^2(t)}{2}\{X_t + 2\sigma^2 \nabla \log p_t(X_t)\}\mathrm{d}t + \sigma g(t)\mathrm{d}\tilde{B}_t,\ X_T \sim \pi^{\text{base}}\ . \tag{30}$$

To integrate this SDE (or the equivalent probability flow ODE), one could turn to the formulas introduced in Appendix B.2, by replacing general coefficients with VP coefficients. Instead, we propose to rely on Exponential Integration (EI) formulas dispensed in Lemma 6 (SDE case) and Lemma 7 (ODE case), that make exact the integration of the linear part of the drift.

Lemma 15 (Approximate denoising SDE EI-based integration - VP case)**.** *The conditional distribution of* X_t *given* $X_s = x_s \in \mathbb{R}^d$ *may be approximated by the Gaussian kernel*

$$q_{s|t}(\cdot|x_t) = \mathrm{N}\left(\sqrt{1 + \lambda_{s,t}^b}x_t + 2\sigma^2\left\{\sqrt{1 + \lambda_{s,t}^b} - 1\right\}\mathbf{s}_t\left(x_t\right), \sigma^2\lambda_{s,t}^b \mathrm{I}_d\right)\ ,$$

with $\lambda_{s,t}^b = \exp(\alpha_t - \alpha_s) - 1$.

Proof. Lemma 6 applied on denoising SDE (30). □

Lemma 16 (Approximate noising ODE EI-based integration - VP case)**.** *The solution at time* t *of the forward probability flow ODE* (6) *starting from* $x_s \in \mathbb{R}^d$ *at time* s *may be approximated in two ways:*

$$\tilde{\mathrm{T}}_{t|s}(x_s) = \sqrt{1 - \lambda_{s,t}^f}x_s + \sigma^2\left\{\sqrt{1 - \lambda_{s,t}^f} - 1\right\}\mathbf{s}_s\left(x_s\right) \qquad \textit{(Euler method : explicit)}$$

$$\mathrm{T}_{t|s}(x_s) = \sqrt{1 - \lambda_{s,t}^f}x_s + \sigma^2\left\{\sqrt{1 - \lambda_{s,t}^f} - 1\right\}\mathbf{s}_{(s+t)/2}\left(\frac{x_s + \mathrm{T}_{t|s}(x_s)}{2}\right) \qquad \textit{(Midpoint method : implicit)}$$

Proof. Lemma 7 applied on forward time ODE (6). □

Lemma 17 (Approximate denoising ODE EI-based integration - VP case)**.** *The solution at time* s *of the probability flow ODE* (6) *starting from* $x_t \in \mathbb{R}^d$ *at time* t *may be approximated in two ways:*

$$\tilde{\mathrm{T}}_{s|t}(x_t) = \sqrt{1 + \lambda_{s,t}^b}x_t + \sigma^2\left\{\sqrt{1 + \lambda_{s,t}^b} - 1\right\}\mathbf{s}_t\left(x_t\right) \qquad \textit{(Euler method: explicit)}$$

$$\mathrm{T}_{s|t}(x_t) = \sqrt{1 + \lambda_{s,t}^b}x_t + \sigma^2\left\{\sqrt{1 + \lambda_{s,t}^b} - 1\right\}\mathbf{s}_{(s+t)/2}\left(\frac{\mathrm{T}_{s|t}(x_t) + x_t}{2}\right) \qquad \textit{(Midpoint : implicit)}$$

Proof. Lemma 7 applied on backward time ODE (6). □

Remark on the mutual invertibility of the ODE integrators. The noising and denoising implicit Midpoint integrators described above are mutual inversible maps, *i.e.*, $\mathrm{T}_{s|t} \circ \mathrm{T}_{t|s} = \mathrm{T}_{t|s} \circ \mathrm{T}_{s|t} = \mathrm{Id}$. This is due to the identity $(1 + \lambda_{s,t}^b)^{-1} = 1 - \lambda_{s,t}^f$. This is not the case for the Euler maps $\tilde{\mathrm{T}}_{s|t}$ and $\tilde{\mathrm{T}}_{t|s}$.

Simplification of δ-assumption in Lemma 12 and Proposition 13. Following the notation introduced in Lemma 12, we obtain simplifications of $c_2(\delta)$ and $c_3(\delta)$ in the specific VP case, for any positive step-size δ, that are given by

$$c_2(\delta) = \sigma^2 \left(\frac{4}{\delta g^2\left((s+t)/2\right)} + 1 \right)^{-1} \ , \ c_3(\delta) = \sigma^2 \left(\frac{4}{\delta g^2\left((s+t)/2\right)} - 1 \right)^{-1} \ .$$

Hence, for any given $L > 0$, if we have $\delta < 4/\{(\sigma^2 L + 1) g^2\left((s+t)/2\right)\}$, then it comes that $\max\left(|c_2(\delta)|, |c_3(\delta)|\right) < 1/L$. In particular, we may use this upper bound on δ as a more readable δ-assumption in Lemma 12 and Proposition 13.

Lemma 18 (Formula for the Jacobian of the Midpoint integrators - VP case)**.** *Let $\delta > 0$, and let define the numerical constants $c_1(\delta)$, $c_2(\delta)$ and $c_3(\delta)$ as*

$$c_1(\delta) = \sqrt{1 - \lambda_{s,t}^f} \ , \ c_2(\delta) = \frac{\sigma^2}{2} \left\{ 1 - \exp\left(\frac{\alpha_s - \alpha_t}{2} \right) \right\} \ , \ c_3(\delta) = \frac{\sigma^2}{2} \left\{ \exp\left(\frac{\alpha_t - \alpha_s}{2} \right) - 1 \right\} \ .$$

Consider the same notation as in Lemma 16 and Lemma 17. Assume that there exists $L > 0$ such that $\mathbf{s}_{(s+t)/2}$ is L-Lipschitz. If we further assume that $(\alpha_t - \alpha_s) < 2\log\left(1 + 2/(L\sigma^2)\right)$, then the Jacobians of Midpoint integration maps $\mathrm{T}_{t|s}$ and $\mathrm{T}_{s|t}$, respectively denoted by $J_{t|s}$ and $J_{s|t}$, verify for any inputs $x_s \in \mathbb{R}^d$ and $x_t \in \mathbb{R}^d$

$$\begin{aligned} J_{t|s}(x_s) &= c_1(\delta) \left(\mathrm{I}_d + c_2(\delta) A(x_s) \right)^{-1} \left(\mathrm{I}_d - c_3(\delta) A(x_s) \right) \ , \\ J_{s|t}(x_t) &= c_1(\delta)^{-1} \left(\mathrm{I}_d - c_3(\delta) B(x_t) \right)^{-1} \left(\mathrm{I}_d + c_2(\delta) B(x_t) \right) \ , \end{aligned}$$

where $A(x_s) = \mathbf{H}_{(s+t)/2}\left(\frac{x_s + \mathrm{T}_{t|s}(x_s)}{2} \right)$ and $B(x_t) = \mathbf{H}_{(s+t)/2}\left(\frac{x_t + \mathrm{T}_{s|t}(x_t)}{2} \right)$.

Proof. The result from Lemma 18 follows from the factorization of the following identities, inherited from the implicit expressions of $\mathrm{T}_{t|s}$ and $\mathrm{T}_{s|t}$,

$$\begin{aligned} J_{t|s}(x_s) &= \left(\mathrm{I}_d - \frac{\sigma^2}{2} \left\{ \sqrt{1 - \lambda_{s,t}^f} - 1 \right\} A(x_s) \right)^{-1} \left(\sqrt{1 - \lambda_{s,t}^f} \mathrm{I}_d + \frac{\sigma^2}{2} \left\{ \sqrt{1 - \lambda_{s,t}^f} - 1 \right\} A(x_s) \right) \ , \\ J_{s|t}(x_t) &= \left(\mathrm{I}_d - \frac{\sigma^2}{2} \left\{ \sqrt{1 + \lambda_{s,t}^b} - 1 \right\} B(x_t) \right)^{-1} \left(\sqrt{1 + \lambda_{s,t}^b} \mathrm{I}_d + \frac{\sigma^2}{2} \left\{ \sqrt{1 + \lambda_{s,t}^b} - 1 \right\} B(x_t) \right) \ . \end{aligned}$$

Here, the additional assumption on the term $(\alpha_t - \alpha_s)$ may be seen as the EI-based analog to the assumption on the step size $\delta = t - s$ in Lemma 12. Indeed, if we have $(\alpha_t - \alpha_s) < 2\log\left(1 + 2/(L\sigma^2)\right)$, then it comes that $\max(|c_2(\delta)|, |c_3(\delta)|) < 1/L$, which thus guarantees the invertibility of the matrices $\mathrm{I}_d + c_2(\delta) A(x_s)$ and $\mathrm{I}_d - c_3(\delta) B(x_t)$. □

Proposition 19 (Exact expression of the Jacobian log-determinants of the Midpoint integrators via power series - VP case)**.** *Consider the same notation as in Lemma 18. Assume that there exists $L > 0$ such that $\mathbf{s}_{(s+t)/2}$ is L-Lipschitz. If we further assume that $(\alpha_t - \alpha_s) < 2\log\left(1 + 2/(L\sigma^2)\right)$, then, for any inputs $x_s \in \mathbb{R}^d$ and $x_t \in \mathbb{R}^d$, we have*

$$\begin{aligned} \log\left|\det J_{t|s}(x_s)\right| &= \sum_{i=0}^{\infty} a_i(s,t) \, \mathrm{Tr}([A(x_s)]^i) \ , \\ \log\left|\det J_{s|t}(x_t)\right| &= \sum_{i=0}^{\infty} b_i(s,t) \, \mathrm{Tr}([B(x_t)]^i) \ , \end{aligned}$$

where $\{a_i(s,t), b_i(s,t)\}_{i=0}^{\infty}$ *are numerical coefficients defined by*

$$a_0(s,t) = -b_0(s,t) = \frac{d}{2}(\alpha_s - \alpha_t) \ ,$$
$$a_i(s,t) = \frac{\sigma^{2i}}{2^i i}\left(\exp\left(\frac{\alpha_s-\alpha_t}{2}\right)-1\right)^i\left(1-(-1)^i\exp\left(-i\frac{\alpha_s-\alpha_t}{2}\right)\right) \ \textit{for any } i \geq 1 \ ,$$
$$b_i(s,t) = \frac{\sigma^{2i}}{2^i i}\left(\exp\left(-\frac{\alpha_s-\alpha_t}{2}\right)-1\right)^i\left(1-(-1)^i\exp\left(i\frac{\alpha_s-\alpha_t}{2}\right)\right) \ \textit{for any } i \geq 1 \ .$$

Proof. Similarly to the proof of Proposition 13, we combine the results of Lemma 18 and Corollary 5 to get the final result. Intermediary simplifications of the terms are omitted here to help the reading. □

B.3 Variance-Exploding diffusion

Consider the case where $f(t) = 0$. Then, SDE (2) simply writes as

$$\mathrm{d}X_t = g(t)\mathrm{d}W_t, \ X_0 \sim \pi \ . \tag{31}$$

This noising scheme is known as the *Variance-Exploding* (VE) scheme (Song et al., 2021).

On the choice of the g-schedule. Following the guidelines from (Karras et al., 2022), we consider the geometric schedule

$$g^2(t) = \sigma_{\min}^2\left(\frac{\sigma_{\max}^2}{\sigma_{\min}^2}\right)^t \log\left(\frac{\sigma_{\max}^2}{\sigma_{\min}^2}\right) \ ,$$

where $\sigma_{\min} \approx 0$ and $\sigma_{\max} \gg 1$ can be arbitrarily chosen.

Lemma 20 (Exact noising SDE integration - VE case)**.** *The conditional distribution of X_t given $X_s = x_s \in \mathbb{R}^d$ is defined by the Gaussian kernel*

$$q_{t|s}(\cdot|x_s) = \mathrm{N}(x_s, \lambda_{s,t}\mathrm{I}_d) \ , \ \textit{with } \lambda_{s,t} = \sigma_{min}^2\left(\frac{\sigma_{max}}{\sigma_{min}}\right)^{2s}\left(\left(\frac{\sigma_{max}}{\sigma_{min}}\right)^{2(t-s)}-1\right)$$

Since $p_T(x) = \int_{\mathbb{R}^d} q_{T|0}(x|x_0)\mathrm{d}\pi(x_0)$, it results that $p_T \approx \mathrm{N}(0, \sigma_{max}^2\,\mathrm{I}_d)$.

Proof. Lemma 6 applied on noising SDE (31). □

Based on the previous lemma, the interpolation coefficients in (3) are given by

$$S(t) = 1 \text{ and } \sigma(t) = \sigma_{\min}\sqrt{\left(\frac{\sigma_{\max}}{\sigma_{\min}}\right)^{2t} - 1}.$$

In particular, $t \mapsto \sigma(t)$ is explicitly invertible, since we have

$$\sigma^{-1}(\sigma) = \frac{\log\left(\left(\frac{\sigma}{\sigma_{\min}}\right)^2 + 1\right)}{2\log\left(\frac{\sigma_{\max}}{\sigma_{\min}}\right)} \ .$$

Time discretization. We define $t_{\text{start}} = \sigma^{-1}(\sigma_{\min})$ and $t_{\text{end}} = \sigma^{-1}(\sigma_{\max})$. With proper choices of $\sigma_{\min}$ and $\sigma_{\max}$, we have $t_{\text{start}} \approx 0$ and $t_{\text{end}} \approx T$. Following (27), the SNR-adapted time discretization $\{t_k\}_{k=0}^K$ between t_{start} and t_{end} is given by

$$t_k = \frac{\log\left(\left(\frac{\sigma_{\max}}{\sigma_{\min}}\right)^{\frac{2k}{K}} + 1\right)}{2\log\left(\frac{\sigma_{\max}}{\sigma_{\min}}\right)} \ .$$

Under mild assumptions on π, the denoising SDE (5) writes as

$$\mathrm{d}X_t = -g^2(t)\nabla \log p_t(X_t)\mathrm{d}t + g^2(t)\mathrm{d}\tilde{B}_t,\ X_T \sim \pi^{\text{base}}\ . \tag{32}$$

Similarly to the VP case (see Appendix B.2), we present below approximate transition kernels and maps based on the Exponential Integration (EI). Since the linear drift term is 0 here, the EI strategy amounts to exactly integrate the time-dependent coefficient associated to the (unknown) score drift term.

Lemma 21 (Approximate denoising SDE EI-based integration - VE case)**.** *The conditional distribution of X_t given $X_s = x_s \in \mathbb{R}^d$ may be approximated by the Gaussian kernel*

$$q_{s|t}(\cdot|x_t) = \mathrm{N}\left(x_t + \lambda_{s,t}\mathbf{s}_t\left(x_t\right), \lambda_{s,t}\,\mathrm{I}_d\right)\ .$$

Proof. Lemma 6 applied on denoising SDE (32). □

Lemma 22 (Approximate noising ODE EI-based integration - VE case)**.** *The solution at time t of the forward probability flow ODE* (6) *starting from $x_s \in \mathbb{R}^d$ at time s may be approximated in two ways:*

$$\begin{aligned}
&\tilde{\mathrm{T}}_{t|s}(x_s) = x_s - \frac{\lambda_{s,t}}{2}\mathbf{s}_s\left(x_s\right) && \textit{(Euler method : explicit)}\\
&\mathrm{T}_{t|s}(x_s) = x_s - \frac{\lambda_{s,t}}{2}\mathbf{s}_{(s+t)/2}\left(\frac{x_s + \mathrm{T}_{t|s}(x_s)}{2}\right) && \textit{(Midpoint method : implicit)}
\end{aligned}$$

Proof. Lemma 7 applied on forward time ODE (6). □

Lemma 23 (Approximate denoising ODE EI-based integration - VE case)**.** *The solution at time s of the backward probability flow ODE* (6) *starting from $x_t \in \mathbb{R}^d$ at time t may be approximated in two ways:*

$$\begin{aligned}
&\tilde{\mathrm{T}}_{s|t}(x_t) = x_t + \frac{\lambda_{s,t}}{2}\mathbf{s}_t\left(x_t\right) && \textit{(Euler method : explicit)}\\
&\mathrm{T}_{s|t}(x_t) = x_t + \frac{\lambda_{s,t}}{2}\mathbf{s}_{(s+t)/2}\left(\frac{\mathrm{T}_{s|t}(x_t) + x_t}{2}\right) && \textit{(Midpoint method : implicit)}
\end{aligned}$$

Proof. Lemma 7 applied on backward time ODE (6). □

Remark on the mutual invertibility of the ODE integrators. The noising and denoising implicit Midpoint integrators described above are mutual inversible maps, *i.e.*, $\mathrm{T}_{s|t} \circ \mathrm{T}_{t|s} = \mathrm{T}_{t|s} \circ \mathrm{T}_{s|t} = \mathrm{Id}$. This is not the case for the Euler maps $\tilde{\mathrm{T}}_{s|t}$ and $\tilde{\mathrm{T}}_{t|s}$.

Simplification of δ-assumption in Lemma 12 and Proposition 13. Following the notation introduced in Lemma 12, we obtain simplifications of $c_2(\delta)$ and $c_3(\delta)$ in the specific VE case, for any positive step-size δ, that are given by

$$c_2(\delta) = c_3(\delta) = \frac{\delta}{4}g^2\left(\frac{s+t}{2}\right)\ .$$

Hence, for any given $L > 0$, if we have $\delta < 4/\{Lg^2\left((s+t)/2\right)\}$, then it comes that $\max\left(|c_2(\delta)|, |c_3(\delta)|\right) < 1/L$. In particular, we may use this upper bound on δ as a more readable δ-assumption in Lemma 12 and Proposition 13.

Lemma 24 (Formula for the Jacobian of the Midpoint integrators - VE case)**.** *Let $\delta > 0$, and let define the numerical constants $c_1(\delta)$, $c_2(\delta)$ and $c_3(\delta)$ as*

$$c_1(\delta) = 1\ ,\ c_2(\delta) = c_3(\delta) = \frac{\lambda_{s,t}}{4}\ .$$

Consider the same notation as in Lemma 22 and Lemma 23. Assume that there exists $L > 0$ such that $\mathbf{s}_{(s+t)/2}$ is L-Lipschitz. If we further assume that $\lambda_{s,t} < 4/L$, then the Jacobians of Midpoint integration maps $\mathrm{T}_{t|s}$ and $\mathrm{T}_{s|t}$, respectively denoted by $J_{t|s}$ and $J_{s|t}$, verify for any inputs $x_s \in \mathbb{R}^d$ and $x_t \in \mathbb{R}^d$

$$J_{t|s}(x_s) = c_1(\delta) \left(\mathrm{I}_d + c_2(\delta) A(x_s)\right)^{-1} \left(\mathrm{I}_d - c_3(\delta) A(x_s)\right) \ ,$$
$$J_{s|t}(x_t) = c_1(\delta)^{-1} \left(\mathrm{I}_d - c_3(\delta) B(x_t)\right)^{-1} \left(\mathrm{I}_d + c_2(\delta) B(x_t)\right) \ ,$$

where $A(x_s) = \mathbf{H}_{(s+t)/2}\left(\frac{x_s + \mathrm{T}_{t|s}(x_s)}{2}\right)$ *and* $B(x_t) = \mathbf{H}_{(s+t)/2}\left(\frac{x_t + \mathrm{T}_{s|t}(x_t)}{2}\right)$.

Proof. The result from Lemma 24 follows from the factorization of the following identities, inherited from the implicit expressions of $\mathrm{T}_{t|s}$ and $\mathrm{T}_{s|t}$,

$$J_{t|s}(x_s) = \left(\mathrm{I}_d + \frac{\lambda_{s,t}}{4} A(x_s)\right)^{-1} \left(\mathrm{I}_d - \frac{\lambda_{s,t}}{4} A(x_s)\right) \ ,$$
$$J_{s|t}(x_t) = \left(\mathrm{I}_d - \frac{\lambda_{s,t}}{4} B(x_t)\right)^{-1} \left(\mathrm{I}_d + \frac{\lambda_{s,t}}{4} B(x_t)\right) \ .$$

Here, the additional assumption on the term $\lambda_{s,t}$ may be seen as the EI-based analog to the assumption on the step size $\delta = t - s$ in Lemma 12. Indeed, if we have $\lambda_{s,t} < 4/L$, then it comes that $\max(|c_2(\delta)|, |c_3(\delta)|) < 1/L$, which thus guarantees the invertibility of the matrices $\mathrm{I}_d + c_2(\delta) A(x_s)$ and $\mathrm{I}_d - c_3(\delta) B(x_t)$. □

Proposition 25 (Exact expression of the Jacobian log-determinants of the Midpoint integrators via power series - VE case)**.** *Consider the same notation as in Lemma 24. Assume that there exists $L > 0$ such that $\mathbf{s}_{(s+t)/2}$ is L-Lipschitz. If we further assume that $\lambda_{s,t} < 4/L$, then, for any inputs $x_s \in \mathbb{R}^d$ and $x_t \in \mathbb{R}^d$, we have*

$$\log \left|\det J_{t|s}(x_s)\right| = \sum_{i=0}^{\infty} a_i(s,t) \operatorname{Tr}([A(x_s)]^i) \ ,$$
$$\log \left|\det J_{s|t}(x_t)\right| = \sum_{i=0}^{\infty} b_i(s,t) \operatorname{Tr}([B(x_t)]^i) \ ,$$

where $\{a_i(s,t), b_i(s,t)\}_{i=0}^{\infty}$ are numerical coefficients defined by

$$a_i(s,t) = b_i(s,t) = 0 \text{ for any even } i \in \mathbb{N} \ ,$$
$$a_i(s,t) = -b_i(s,t) = -2\frac{\lambda_{s,t}^i}{4^i} \text{ for any odd } i \in \mathbb{N} \ .$$

Proof. Similarly to the proof of Proposition 13, we combine the results of Lemma 24 and Corollary 5 to get the final result. Intermediary simplifications of the terms are omitted here to help the reading. □

C Proofs of Section 4

In the main paper, we present our methodology to design diffusion-based aMC-BGs via deterministic transitions between noise levels, by relying on the general noising framework presented in Appendix B.1 to maintain a certain generality. We highlight that these results still hold within the specific EI-based framework of VP noising (see Appendix B.2) and VE noising (see Appendix B.3), based on the formulas introduced in the respective sections. We leave the proof for the reader.

Proof of Proposition 1. This is a restatement of the results of Lemma 10 and Lemma 11 in the case where $s = t_k$, $t = t_{k+1}$ and $\mathbf{s}_t = \nabla \log p_t$. The mutual invertibility property is immediate. □

We give below a formal version of Assumption 1.

Assumption 3 (Score smoothness & discretization error - Formal version of Assumption 1)**.** *(a) There exists $L_k > 0$ such that $\nabla \log p_{t_{k+1/2}}$ is L_k-Lipschitz and (b) the step-size δ_k verifies*

$$\max\left(L_k \left|c_2(\delta_k)\right|, L_k \left|c_3(\delta_k)\right|, c_4(\delta_k, L_k)\right) < 1 \ ,$$

where c_2 and c_3 are given in Lemma 12, and $c_4(\delta, L) = \frac{\delta}{2}\left(\left|f(t_{k+1/2}\right| + Lg^2(t_{k+1/2})/2\right)$.

Proof of Proposition 2. Assume Assumption 3. Fix current states x_k and x_{k+1}. Respectively, denote the sequences $\{\mathrm{T}^{(n)}_{k+1|k}(x_k)\}_{n\in\mathbb{N}}$ and $\{\mathrm{T}^{(n)}_{k|k+1}(x_{k+1})\}_{n\in\mathbb{N}}$ by $\{y_n\}_{n\in\mathbb{N}}$ and $\{z_n\}_{n\in\mathbb{N}}$. Respectively define the *forward* map $\Psi_{k+1|k} : \mathbb{R}^d \to \mathbb{R}^d$ and the *backward* map $\Psi_{k|k+1} : \mathbb{R}^d \to \mathbb{R}^d$ by

$$\Psi_{k+1|k}(y) = x_k + \delta_k v\left(t_{k+1/2}, \frac{x_k + y}{2}\right) \ , \ \Psi_{k|k+1}(z) = x_{k+1} - \delta_k v\left(t_{k+1/2}, \frac{z + x_{k+1}}{2}\right) \ ,$$

such that $y_{n+1} = \Psi_{k+1|k}(y_n)$ and $z_{n+1} = \Psi_{k|k+1}(z_n)$ for any $n \in \mathbb{N}$. By combining Assumption 3-(a) and $c_4(\delta_k, L_k) < 1$ from Assumption 3-(b), it is easy to see that both maps $\Psi_{k+1|k}$ and $\Psi_{k|k+1}$ are contractive Lipschitz mappings. We directly obtain the result by application of Banach fixed-point theorem. □

Lemma 26 (Formula for the Jacobian of the IM integrator)**.** *Following the same notation as in Proposition 1, under Assumption 1, the Jacobians of $\mathrm{T}_{k|k+1}$ and $\mathrm{T}_{k+1|k}$ verify*

$$J_{\mathrm{T}_{k+1|k}}(x_k) = c_1\left(\mathrm{I}_d + c_2 A(x_k)\right)^{-1}\left(\mathrm{I}_d - c_3 A(x_k)\right) \ ,$$
$$J_{\mathrm{T}_{k|k+1}}(x_{k+1}) = c_1^{-1}\left(\mathrm{I}_d - c_3 B(x_{k+1})\right)^{-1}\left(\mathrm{I}_d + c_2 B(x_{k+1})\right) \ ,$$

where

$$A(x_k) = \mathbf{H}_{t_{k+1/2}}\left(\frac{x_k + \mathrm{T}_{k+1|k}(x_k)}{2}\right) \ , \ B(x_{k+1}) = \mathbf{H}_{t_{k+1/2}}\left(\frac{x_{k+1} + \mathrm{T}_{k|k+1}(x_{k+1})}{2}\right) \ ,$$

$\mathbf{H}_{t_{k+1/2}}$ is the Hessian of $\log p_{t_{k+1/2}}$, and c_1, c_2, c_3 are the numerical constants given in Lemma 12.

Proof of Lemma 26. This is a restatement of Lemma 12 in the case where $s = t_k$, $t = t_{k+1}$, with exact score and Hessian functions ($\nabla \log p_{t_{k+1/2}}$ and $\mathbf{H}_{t_{k+1/2}}$) used for $\mathbf{s}$ and $\mathbf{H}$. In particular, the assumptions that are needed for this result are verified by Assumption 3. □

Note that similar constants to those introduced in Proposition 1 are derived for VP, respectively VE, noising scheme combined with exponential integration in Lemma 18, respectively Lemma 24, under small change in the assumption.

Proof of Proposition 3. Assume Assumption 3. Consider a prescribed fixed-point range $M \geq 1$ satisfying (25). We first consider the approximations of $A(x_k)$ and $B(x_{k+1})$, introduced in Lemma 26, obtained by replacing the intractable implicit map evaluations with their fixed-point estimations, see (23) and (24). This leads to the following expressions

$$A^{(M)}(x_k) = \mathbf{H}_{t_{k+1/2}}\left(\frac{x_k + \mathrm{T}^{(M)}_{k+1|k}(x_k)}{2}\right) \ , \ B^{(M)}(x_{k+1}) = \mathbf{H}_{t_{k+1/2}}\left(\frac{x_{k+1} + \mathrm{T}^{(M)}_{k|k+1}(x_{k+1})}{2}\right) \ .$$

By Assumption 3(a)-(b) we verify that we have $\|A(x_k)\| < \min(1/\left|c_2(\delta_k)\right|, 1/\left|c_3(\delta_k)\right|)$ and $\|B(x_{k+1})\| < \min(1/\left|c_2(\delta_k)\right|, 1/\left|c_3(\delta_k)\right|)$. Therefore, we encounter the same theoretical requirements as in the proof of Proposition 13 where $s = t_k$, $t = t_{k+1}$, with exact score and Hessian functions ($\nabla \log p_{t_{k+1/2}}$ and $\mathbf{H}_{t_{k+1/2}}$) used for $\mathbf{s}$ and $\mathbf{H}$, which allows us to define the *exact* expression of the Jacobian log-determinants

$$\log\left|\det J_{\mathrm{T}_{k+1|k}}(x_k)\right| = \sum_{i=0}^{\infty} a_i(t_k, t_{k+1})\,\mathrm{Tr}([A(x_k)]^i) \ , \tag{33}$$

$$\log\left|\det J_{\mathrm{T}_{k+1|k}}(x_k)\right| = \sum_{i=0}^{\infty} b_i(t_k, t_{k+1})\,\mathrm{Tr}([B(x_{k+1})]^i) \ , \tag{34}$$

using the numerical coefficients introduced in Proposition 13. On the other hand, we also have that $\left\|A^{(M)}(x_k)\right\| < \min(1/\left|c_2(\delta_k)\right|, 1/\left|c_3(\delta_k)\right|)$ and $\left\|B^{(M)}(x_{k+1})\right\| < \min(1/\left|c_2(\delta_k)\right|, 1/\left|c_3(\delta_k)\right|)$, which allows us to *exactly* define the following expansion series based on replacing Jacobian terms $A^{(M)}(x_k)$ and $B^{(M)}(x_{k+1})$

$$\sum_{i=0}^{\infty} a_i(t_k, t_{k+1}) \operatorname{Tr}([A^{(M)}(x_k)]^i) \text{ and } \sum_{i=0}^{\infty} b_i(t_k, t_{k+1}) \operatorname{Tr}([B^{(M)}(x_{k+1})]^i) \,. \tag{35}$$

Since we expect to have $A(x_k) \approx A^{(M)}(x_k)$ and $B(x_{k+1}) \approx B^{(M)}(x_{k+1})$, we may substitute the trace terms in (33) and (34) by those in (35) to approximate $\log\left|\det J_{\mathrm{T}_{k+1|k}}(x_k)\right|$ and $\log\left|\det J_{\mathrm{T}_{k|k+1}}(x_{k+1})\right|$. Finally, we obtain the result from Proposition 3 under additional approximation induced by the truncation of the power series at a given order $I \geq 1$, letting $a_{k,i} = a_i(t_k, t_{k+1})$ and $b_{k,i} = b_i(t_k, t_{k+1})$. □

In our experiments based on the VP noising scheme combined with exponential integration, we adapt the result from Proposition 3 by using the coefficients introduced in Proposition 19. Similarly, one could use the coefficients from Proposition 25 for the VE noising scheme combined with exponential integration.

D Experimental details

D.1 Target details

Definition of the *TwoModes* target distribution. For our target π, we first consider the Gaussian mixture introduced in Grenioux et al. (2025), whose density is defined over $\mathbb{R}^d$ as

$$\gamma(x) = \tfrac{2}{3}\mathrm{N}(x; -a\mathbf{1}_d, \Sigma_1) + \tfrac{1}{3}\mathrm{N}(x; a\mathbf{1}_d, \Sigma_2) \,,$$

where $\Sigma_1, \Sigma_2 \in \mathbb{R}^{d\times d}$ are diagonal covariance matrices. The diagonal entries of Σ_1 are given by $(\Sigma_1)_{i,i} = \frac{i}{d}\sigma^2_{\max} + \frac{d-i}{d}\sigma^2_{\min}$, and those of Σ_2 are the reverse of Σ_1: $(\Sigma_2)_{i,i} = (\Sigma_1)_{d-i,d-i}$, with $\sigma^2_{\max} = 0.2$ and $\sigma^2_{\min} = 0.01$ (hence, the conditioning number of each covariance matrix is 20). We vary the separation parameter $a \in \{1.0, 2.5, 5.0, 10.0\}$ and the dimension $d \in \{16, 32, 64, 128\}$. In our experiments, we rather consider the "standardized" version of γ (*i.e.*, with zero mean and unit covariance), given by the unnormalized density $x \mapsto \gamma(\Sigma_\pi^{1/2}x + \mathbf{m}_\pi)$, where $\mathbf{m}_\pi$ is the exact mean of π, and Σ_π is a diagonal covariance matrix whose entries correspond to the exact marginal variances of π along each coordinate. The mode weight absolute error metric corresponds to the absolute error $|\hat{w} - 2/3|$ when estimating the strongest mode weight via a Monte Carlo estimator $\hat{w}$ based on generated samples, see (Grenioux et al., 2025, Section 3.1) for more details.

Definition of the *ManyModes* target distribution. We also consider the d-dimensional Gaussian mixture with $L > 2$ components introduced in (Noble et al., 2025, Appendix H.1) defined for any $x \in \mathbb{R}^d$ by its density $\gamma(x) = \sum_{\ell=1}^{L} w_\ell \mathrm{N}(x; \mathbf{m}_\ell, 0.5\mathrm{I}_d)$, where the means $\{\mathbf{m}_\ell\}_{\ell=1}^L$ are sampled independently from $\mathrm{U}([-L, L]^d)$, and the weights $\{w_\ell\}_{\ell=1}^L$ form a strictly increasing geometric sequence such that $w_L/w_1 = 3$. We will consider $L \in \{16, 32, 64\}$ with fixed dimension $d = 32$. Moreover, we apply the same standardization procedure as for the *TwoModes* targets. To evaluate how well mode weights are recovered, we compute the Total Variation (TV) distance between the true mode weight histogram and its Monte Carlo estimate.

D.2 Training and sampling parameters

Diffusion model training details. As explained in Section 6.1, we consider two types of architectures $\mathcal{E}^\theta$ to learn the log-densities of DMs: (i) a pinned architecture, ensuring exact recovery of the target distribution π at t_0 and (ii) an hardcoded architecture, without any boundary condition fixed at training stage. For both of these models, we rely on an enhanced version of the score-like architecture advocated by Richter et al. (2023), denoted by $\mathbf{s}^\theta : (t, x) \in \mathbb{R}^{d+1} \to \mathbf{s}^\theta_t(x) \in \mathbb{R}^d$, which is a 4-layer 128-width fully connected network with GeLU activations, position-input preconditioning (based on target mean and scalar variance), time-input preconditioning (based on Fourier embedding) and time-input skip-connections at every layer. Our models are the following: (a) *Pinned*: given (26), we set $g^\theta_t(x) = \frac{1}{2}\left\|\mathbf{s}^\theta_t(x)\right\|_2^2$ and f^θ to be a scalar-to-scalar 4-layer 64-width fully connected network with GeLU activations and the same time-input preconditioning as in $\mathbf{s}^\theta$; (b) *Hardcoded*: we adopt the network preconditioning strategy proposed by Thornton et al. (2025) on $\mathbf{s}^\theta$.

Training method	LFPE+DSM	aLFPE+DSM	RNE+DSM
Value of $\lambda_{\mathbf{reg}}$	$\{10^{-2}, 10^{-3}, 10^{-4}\}$	$\{10^{-2}, 10^{-3}, 10^{-4}\}$	$\{10, 1, 10^{-1}\}$

Table 2: **Values of $\lambda_{\mathbf{reg}}$ for regularizing DSM objective with LFPE, aLFPE or RNE objective.**

For each setting of *TwoModes* and *ManyModes* distributions (varying by dimension, mode spacing or number of modes), we train both network architectures with the six objectives described in Appendix A.4, using parameterized log-density $\mathbf{U}_t^\theta = -\mathcal{E}_t^\theta$, parameterized score $\mathbf{s}_t^\theta = -\nabla_x \mathcal{E}_t^\theta$ and parameterized time score $\mathbf{u}_t^\theta = -\partial_t \mathcal{E}_t^\theta$. For TSM and tSM objectives, we consistently apply a score matching regularization by simply adding up the DSM objective term (hence with scale 1). Following the guidelines of cited energy matching works, we respectively implement LFPE+DSM, aLFPE+DSM and RNE+DSM objectives as a linear combination of the DSM objective (with scale 1) and the LFPE, aLFPE or RNE objective, multiplied by an hyperparameter λ_{reg} chosen in the range of values described in Table 2. For the latter three methods, we systematically display, for each sampling/target setting, the best sampling results obtained among the considered values of λ_{reg}.

Regardless of the target distribution, all diffusion models are trained on datasets of size 60,000 for 1,000 epochs with a batch size of 1024. We use AdamW optimizer (Loshchilov & Hutter, 2019) and set the default learning rate as 10^{-4}, multiplied by a factor d^{-1} for score matching approaches (DSM, TSM, tSM) and d^{-2} for energy matching techniques (LFPE, aLFPE, RNE), following guidelines of related works. In particular, when training networks with TSM and RNE objectives, we initialize $\mathbf{U}^\theta$ based on the output of DSM training procedure. In the case of tSM, LFPE and aLPFE obectives, we observed that this DSM pretraining strategy was not beneficial, and even led to training instability (certainly due to arbitrary values taken by $\partial_t \mathbf{U}^\theta$ at initialization); to maintain a fair computational comparison between all methods, networks trained with tSM, LFPE and aLPFE objectives benefit from 1,000 additional training epochs (hence, 2,000 epochs in total). For each value of K considered in the sampling results, training is performed over a discrete time grid of $[0, T]$ with K levels defined by the SNR-discretization scheme evoked in Appendix A.2, rather than by uniform sampling in $[0, T]$. This means that a separate network is trained for each value of K, ensuring precise learning at the exact times where inference is conducted.

General remarks on annealed sampling methods. Since the considered targets are systematically standardized, we set the base distribution as their Gaussian approximation $\pi^{\text{base}} = \mathrm{N}(0, \mathrm{I}_d)$, for both tempering and diffusion-based approaches. For tempering paths, the sequence of densities defined by (18) is employed with a logarithmic schedule $\beta_k = 1 - \varepsilon^{k/K}$ and $\varepsilon = 10^{-5}$, following the recommendations of Grenioux et al. (2025). For the standard SMC approach, we implemented the adaptive level selection strategy of Zhou et al. (2016), as also used in Grenioux et al. (2025). However, within our computational upper bound of 256 intermediate levels, we did not observe any improvement over a fixed, logarithmic schedule. We therefore use the simpler fixed-schedule approach, which is also computationally cheaper. Similar adaptive strategies for optimizing the annealing schedule in RE have been proposed in the literature (Syed et al., 2021; 2022), but we restrict ourselves here to non-adaptive settings to maintain comparability and investigate the impact of the number of intermediate levels directly. When using second-order approaches, *i.e.*, methods that require access to the Hessians of the bridging log-densities, we only exploit the diagonal of these Hessians to ensure a good compromise between accuracy and computational efficiency in high dimensional scenarios.

All SMC variants (also including diffusion-enhanced SMC samplers), as well as the standard AIS sampler, apply 160 MCMC steps (including 128 warm-up steps) for local exploration at each level $k \in \{0, \dots, K\}$, using *Metropolis-Adjusted Langevin Algorithm* (MALA) (Roberts & Tweedie, 1996). Following the recommendations of Chopin & Papaspiliopoulos (2020), we do not perform resampling systematically, but instead apply it adaptively based on the current IS weights, using an effective sample size threshold of 30%.

For RE-based sampling methods, we perform a total of 24,576 MCMC steps (including 8,192 warm-up steps), with local exploration made via MALA and swaps occurring every 8 steps, thereby defining the computational budget of RE (with or without transition kernels) to be comparable to the footprint of the SMC setting[10]

[10]We however note that the version RE with deterministic transitions based on Hutchinson estimation requires twice as much sampling time as its other RE counterparts, making it the most demanding sampling method.

with the largest number of levels (*i.e.*, $K = 256$ where SMC performs the best), see the last row of Figure 3. When we target the diffusion density path, we leverage prior knowledge brought by the pre-trained diffusion model to initialize each intermediate level by simply simulating the denoising SDE (5), thus populating each level with target-informed particles. In Appendix D.4, we provide an ablation study in idealized setting **(A)** comparing the results of the related RE samplers when rather initially populating the levels with samples from π^{base} and observe that our score-informed strategy is more beneficial to RE.

For all variants of annealed samplers based on deterministic transitions, we use by default $M = 4$ fixed-point iterations, truncate the power series at order $I = 3$, and use 32 samples in the Hutchinson estimator when applicable. In Appendix D.4, we provide a precise ablation study of these hyperparameters in idealized setting **(A)**, to evaluate their individual impact on sampling performance. Finally, all local MALA steps are performed with an initial step size of 0.01; then, its is geometrically adapted during both warm-up and effective sampling based on local MH acceptance rates, targeting 70% acceptance.

Inference and sampling details. For diffusion-based methods, whether the path is learned or fixed, we adopt the SNR-adapted discretization from Appendix A.2 to establish the annealing levels : when combined with a learned path, this ensures consistency between learning and inference stages. For all AIS/SMC samplers, we use 8192 particles, and keep, for each particle, when it is available, the last 32 MCMC samples generated at the last level (properly reweighted using the associated importance weights) to compute the metrics. For all RE methods, we use 4 parallel RE chains; once the fixed global number of MCMC steps is reached, each of thse chains is subsampled by retaining only the last local MCMC state before each swap. For all annealed samplers, we repeat the sampling run 8 times to produce averaged results in the plots.

D.3 Additional targets and metrics

This section presents a series of extended experiments that complement the core findings of the paper by evaluating additional configurations and metrics. Specifically, Figures 8 and 9 expand upon Figure 1 on the *TwoModes* benchmark, considering various mode spacings and dimensions, and reporting sliced Wasserstein distance and absolute mode weight error, respectively. Figure 7 extends this comparison to the *ManyModes* setting using the total variation distance between mode weight histograms. Similarly, Figures 10, 12, 14 and 16 generalize Figure 3 over more *TwoModes* settings by displaying the sliced Wasserstein distance results, while their analog for the mode weight error metric are respectively provided in Figures 11, 13, 15 and 17. To complete Figure 4, we illustrate, for the exact same settings, the failure of the annealed samplers when the DM log-densities are trained via the score/energy matching objectives introduced in Section 6.1 by reporting the sliced Wasserstein distance in Figures 18 to 41. Similarly, we display the results obtained for *ManyModes* settings in idealized setting **(A)** with Figure 42 (sliced Wasserstein distance) and Figure 43 (mode weight TV metric), and in realistic setting **(B)** for all considered score/energy matching objectives by reporting the related sliced Wasserstein distance in Figures 44 to 49. Finally, we compare in Figures 50 and 51 all learned density paths for 1D Gaussian mixtures introduced in Figure 5. We highlight that all of these additional experiments are fully consistent with the conclusions and observations from Sections 4 and 6.

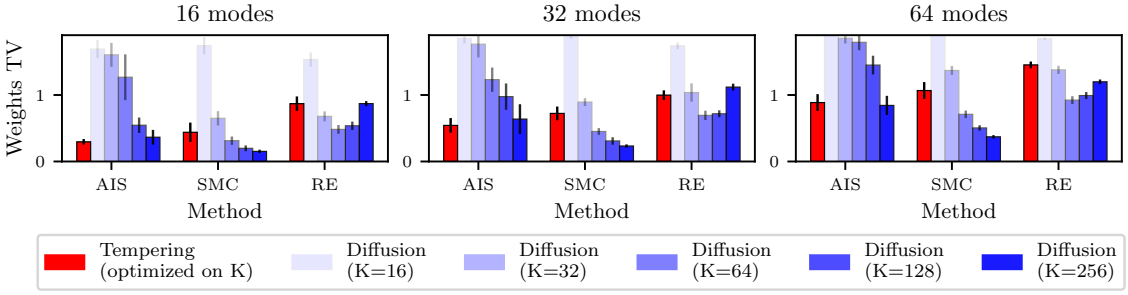

Figure 7: **Sampling results via the total variation distance of the weight histograms for classic annealed samplers with diffusion (blue) and tempering (red) density paths**, when targeting *ManyModes*. This is complementary to Figure 1.

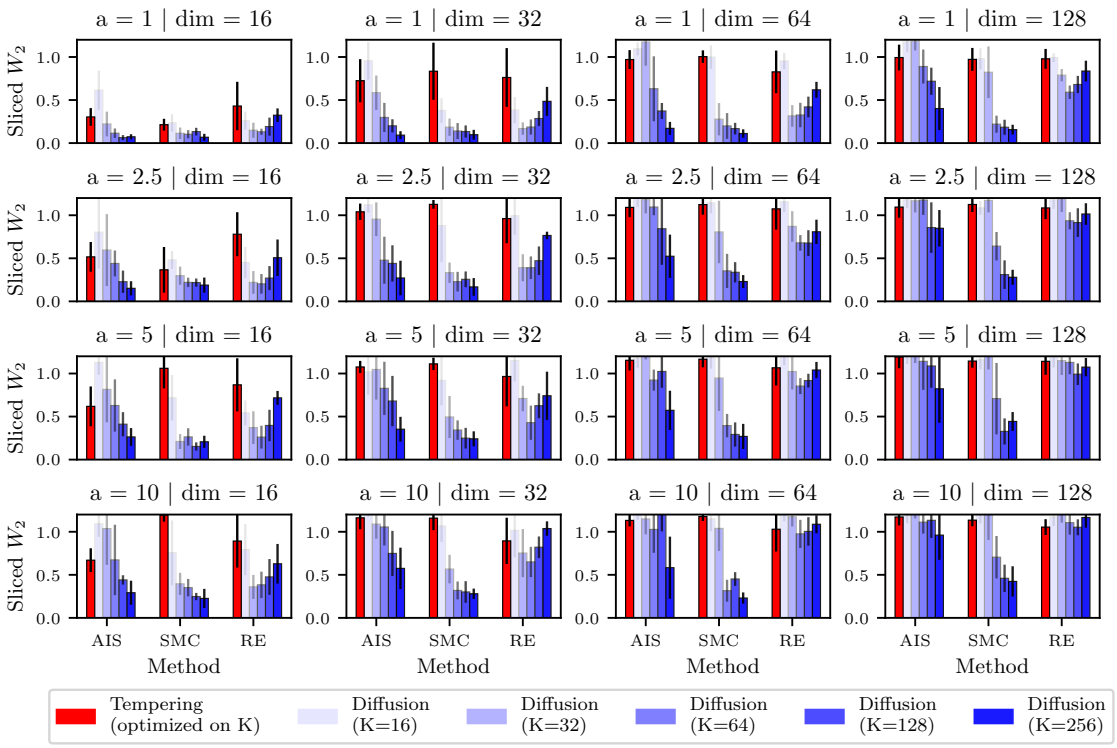

Figure 8: **Sampling results via sliced Wasserstein distance for classic annealed samplers with diffusion (blue) and tempering (red) density paths**, when targeting *TwoModes* in different settings of mode spacing and dimension. This is complementary to Figure 1.

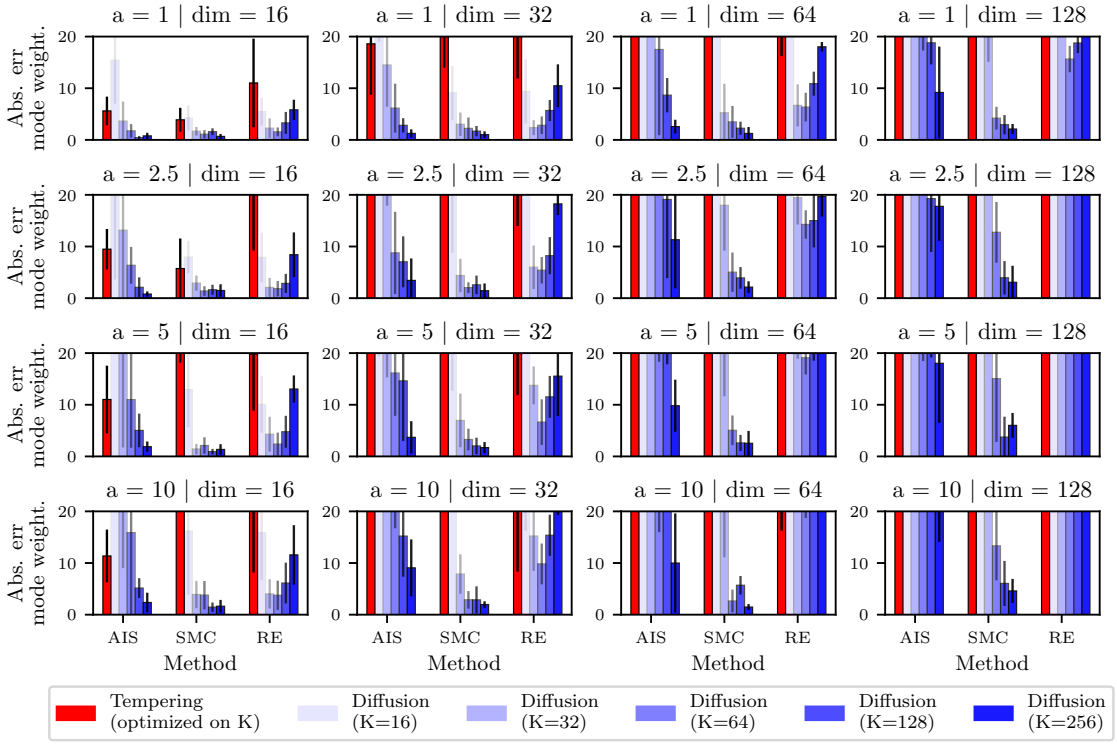

Figure 9: **Sampling results via mode weight absolute error for classic annealed samplers with diffusion (blue) and tempering (red) density paths**, when targeting *TwoModes* in different settings of mode spacing and dimension. This is complementary to Figure 1 and Figure 8.

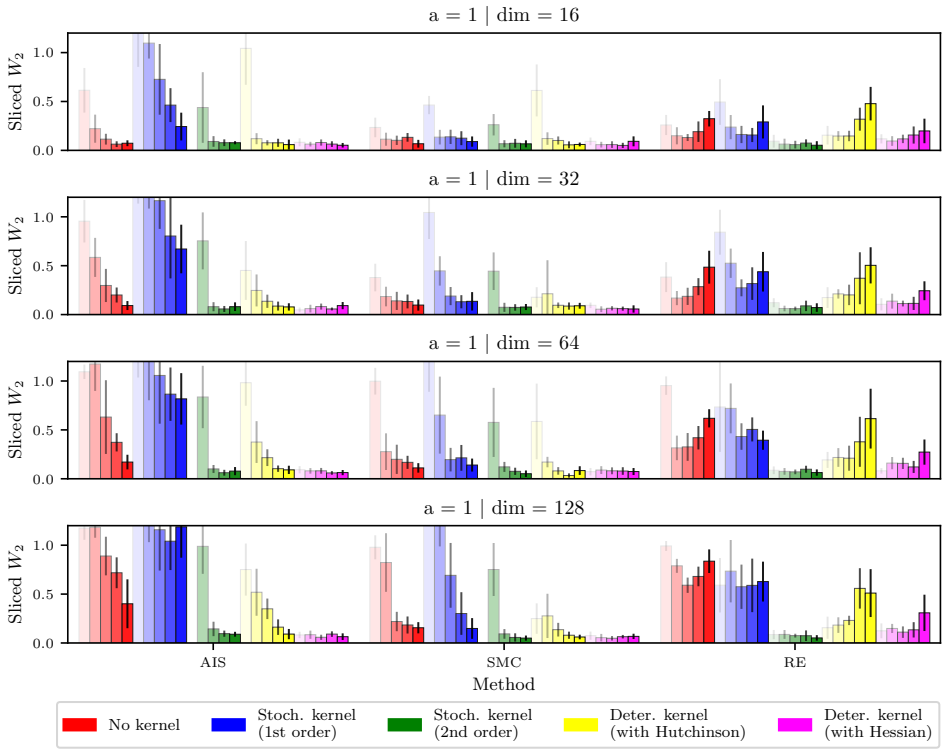

Figure 10: **Diffusion-based aMC-BG results via sliced Wasserstein distance in idealized setting (A)**, when targeting *TwoModes* distribution with $a = 1$, for all dimensional settings. This is complementary to Figure 3.

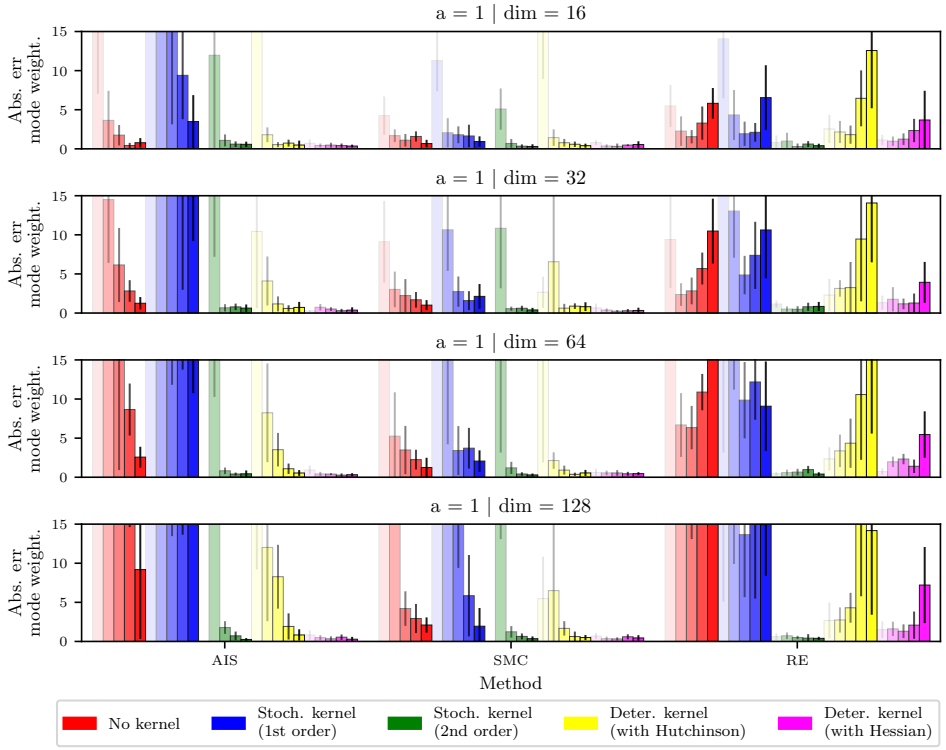

Figure 11: **Diffusion-based aMC-BG results via mode weight absolute error in idealized setting (A)**, when targeting *TwoModes* distribution with $a = 1$, for all dimensional settings. This is complementary to Figure 3 and Figure 10.

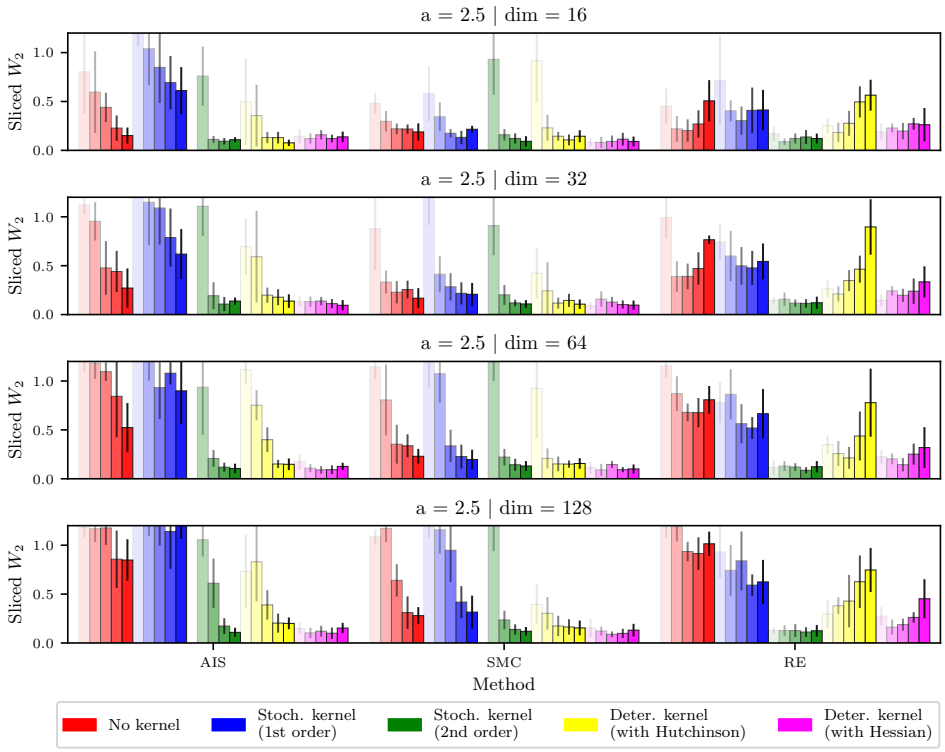

Figure 12: **Diffusion-based aMC-BG results via sliced Wasserstein distance in idealized setting (A)**, when targeting *TwoModes* distribution with $a = 2.5$, for all dimensional settings. This is complementary to Figure 3.

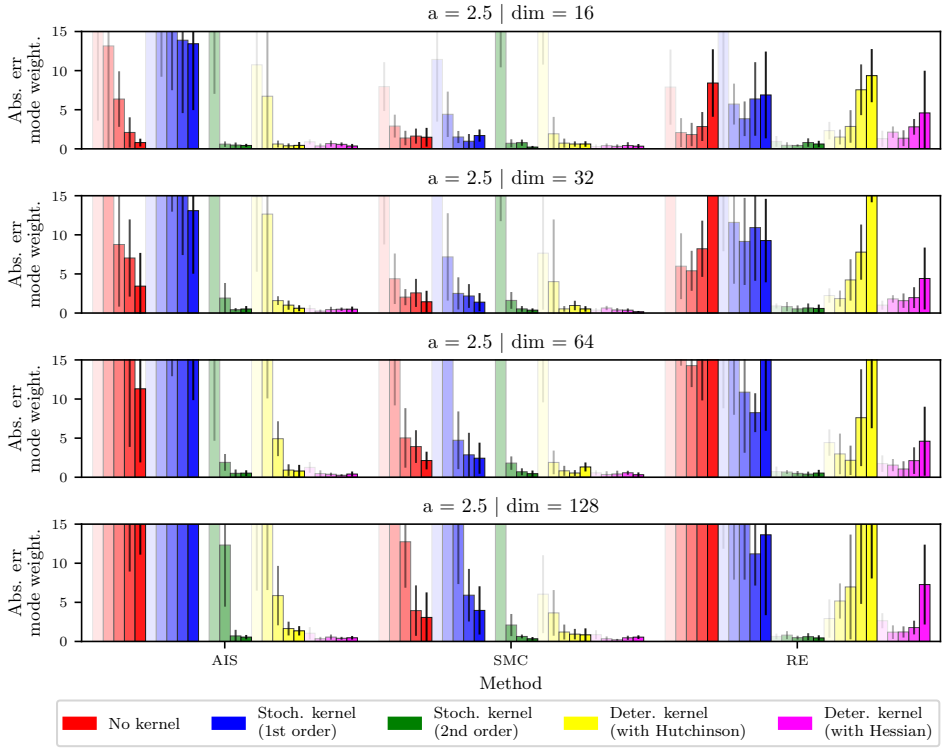

Figure 13: **Diffusion-based aMC-BG results via mode weight absolute error in idealized setting (A)**, when targeting *TwoModes* distribution with $a = 2.5$, for all dimensional settings. This is complementary to Figure 3 and Figure 12.

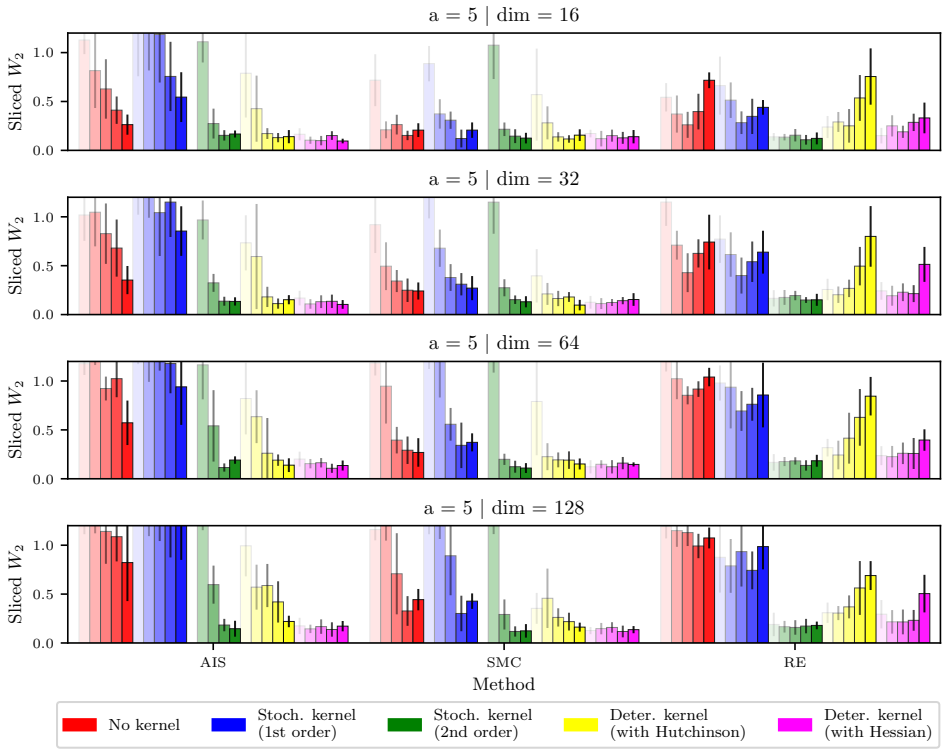

Figure 14: **Diffusion-based aMC-BG results via sliced Wasserstein distance in idealized setting (A)**, when targeting *TwoModes* distribution with $a = 5$, for all dimensional settings. This is complementary to Figure 3.

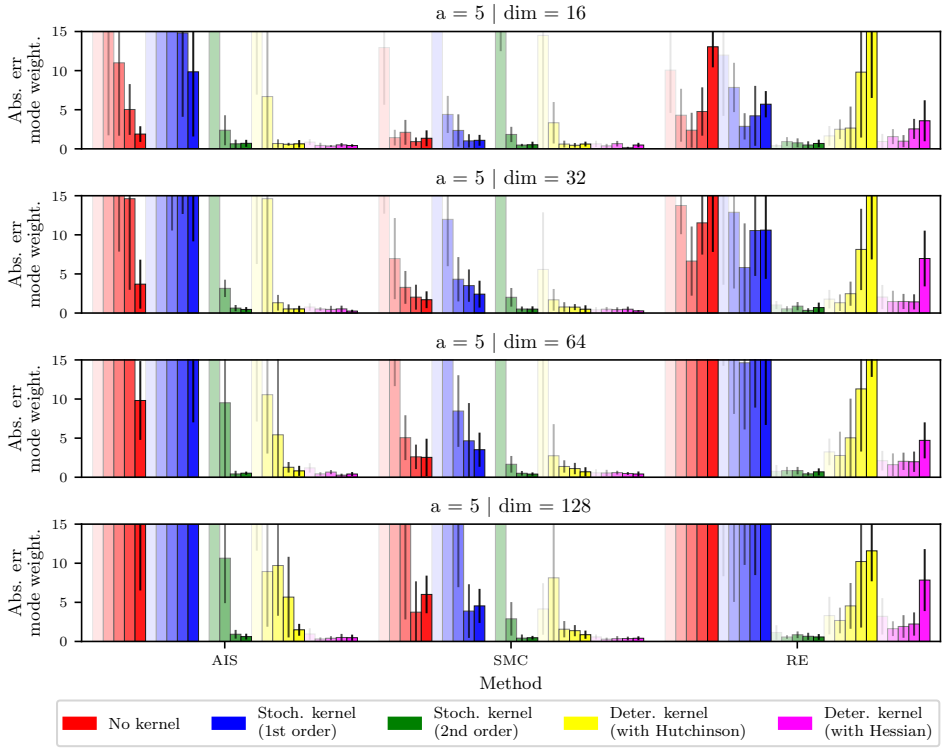

Figure 15: **Diffusion-based aMC-BG results via mode weight absolute error in idealized setting (A)**, when targeting *TwoModes* distribution with $a = 5$, for all dimensional settings. This is complementary to Figure 3 and Figure 14.

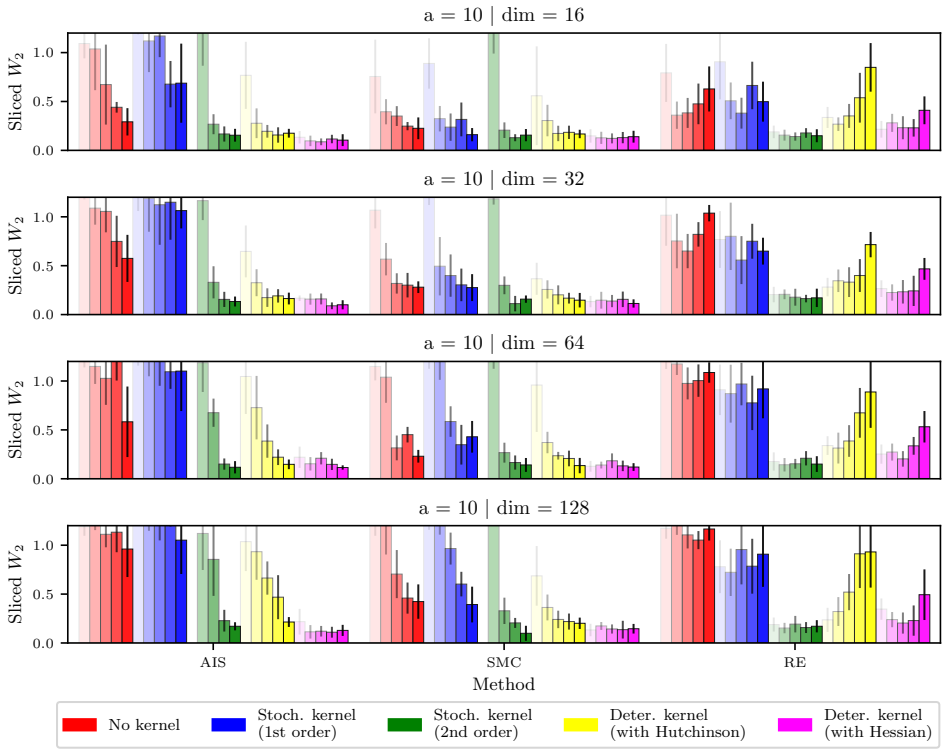

Figure 16: **Diffusion-based aMC-BG results via sliced Wasserstein distance in idealized setting (A)**, when targeting *TwoModes* distribution with $a = 10$, for all dimensional settings. This is complementary to Figure 3.

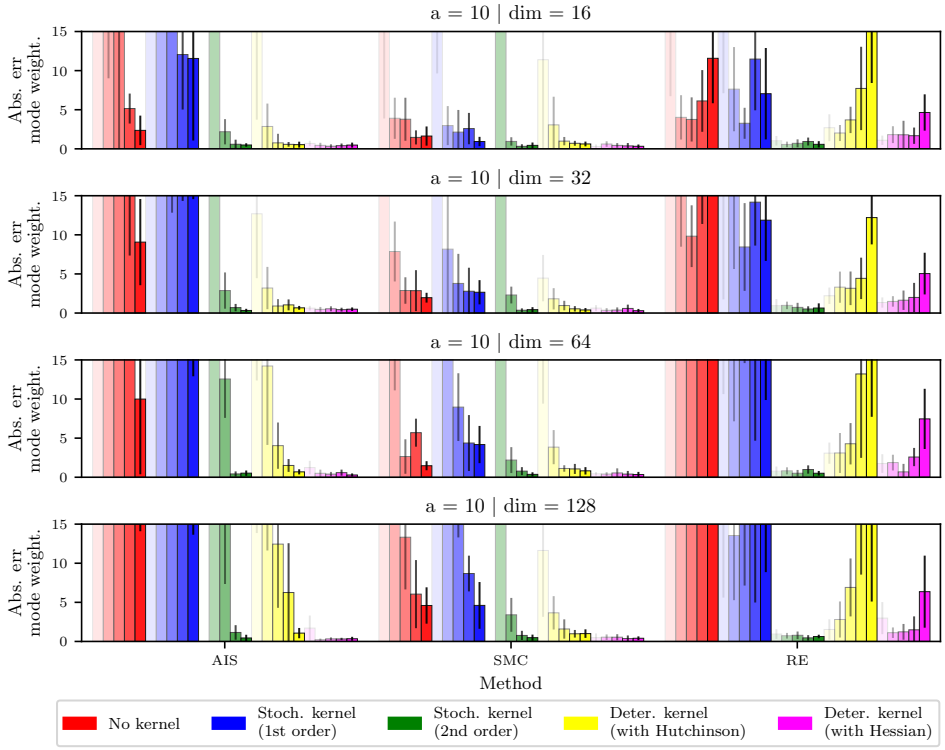

Figure 17: **Diffusion-based aMC-BG results via mode weight absolute error in idealized setting (A)**, when targeting *TwoModes* distribution with $a = 10$, for all dimensional settings. This is complementary to Figure 3 and Figure 16.

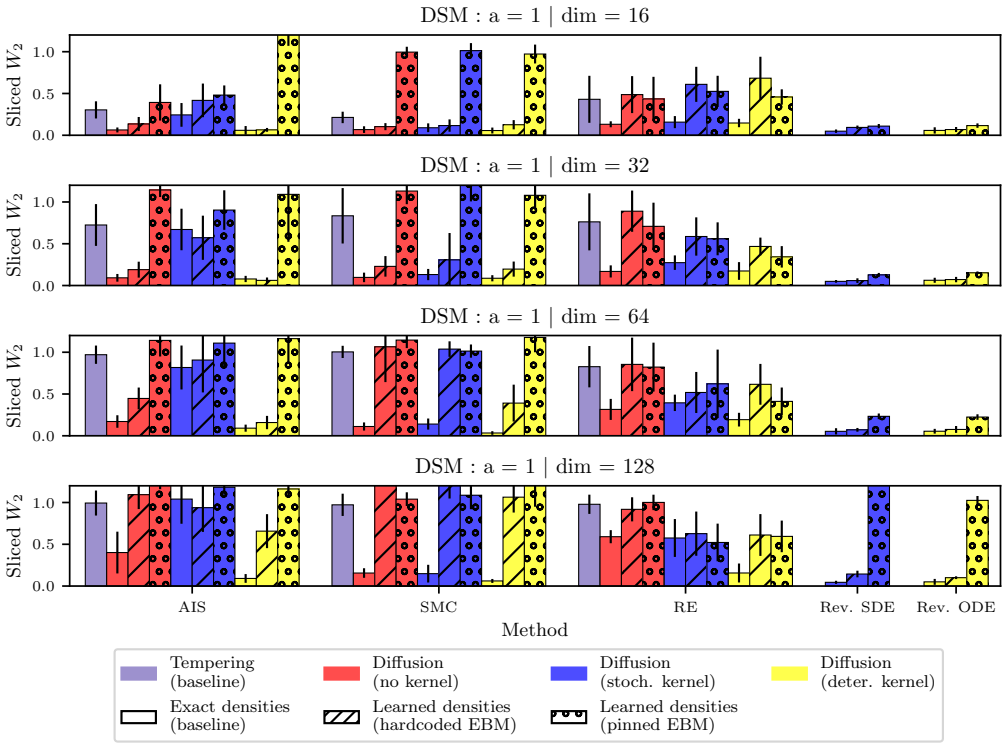

Figure 18: **Diffusion-based aMC-BG results via sliced Wasserstein distance in realistic setting (B) with DSM objective**, when targeting *TwoModes* distribution with $a = 1$, for all dimensional settings. This is complementary to Figure 4.

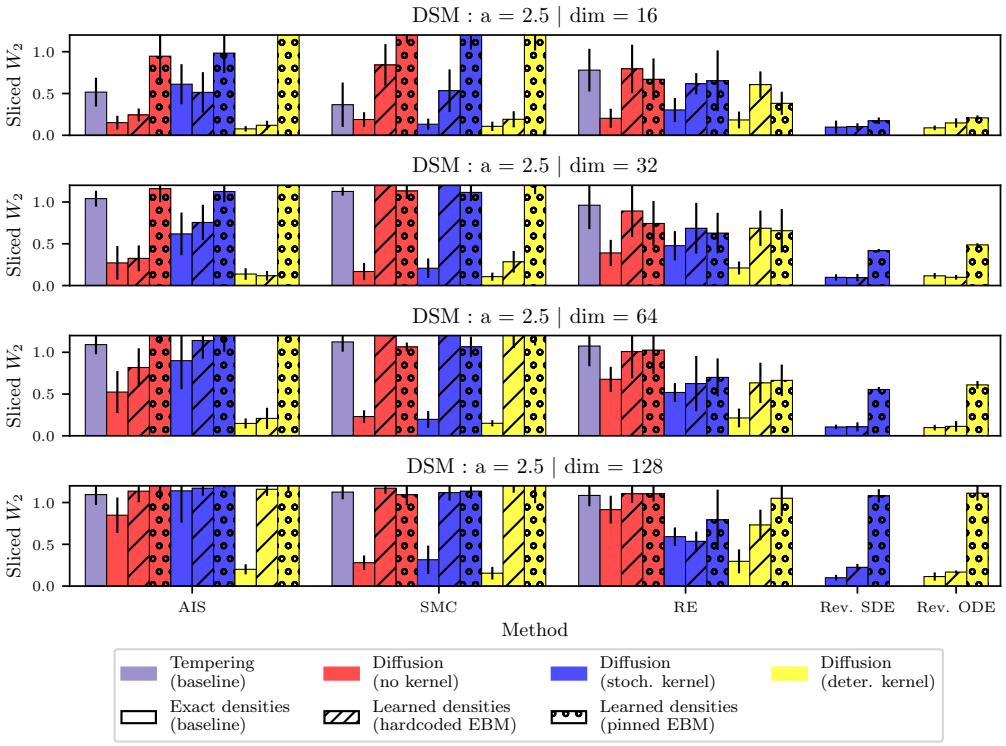

Figure 19: **Diffusion-based aMC-BG results via sliced Wasserstein distance in realistic setting (B) with DSM objective**, when targeting *TwoModes* distribution with $a = 2.5$, for all dimensional settings. This is complementary to Figure 4.

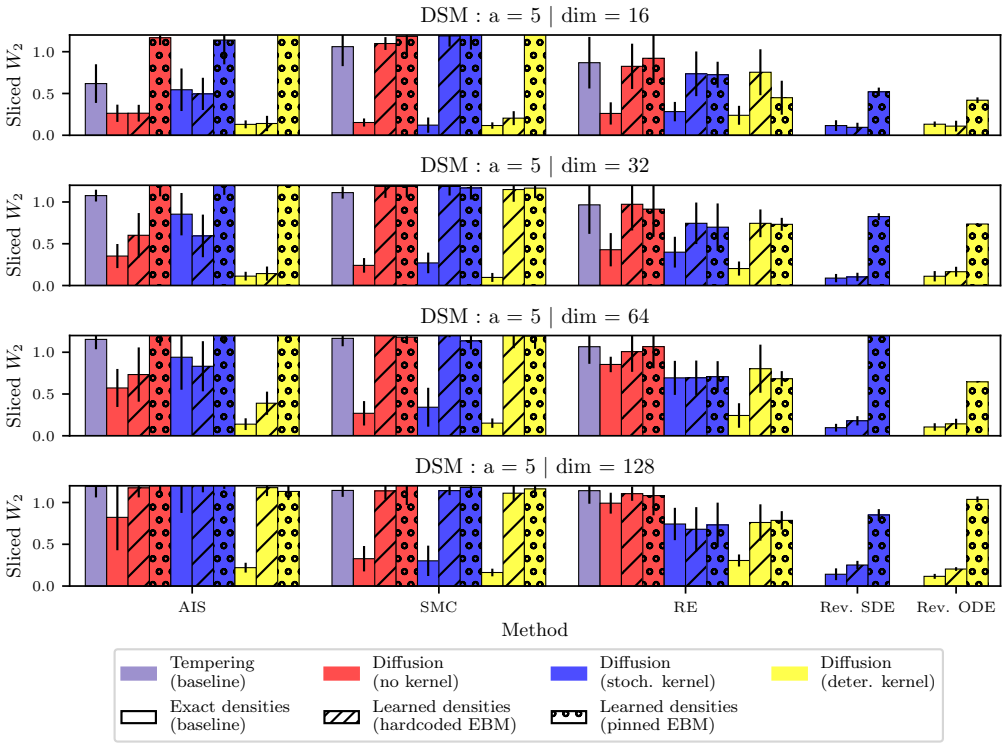

Figure 20: **Diffusion-based aMC-BG results via sliced Wasserstein distance in realistic setting (B) with DSM objective**, when targeting *TwoModes* distribution with $a = 5$, for all dimensional settings. This is complementary to Figure 4.

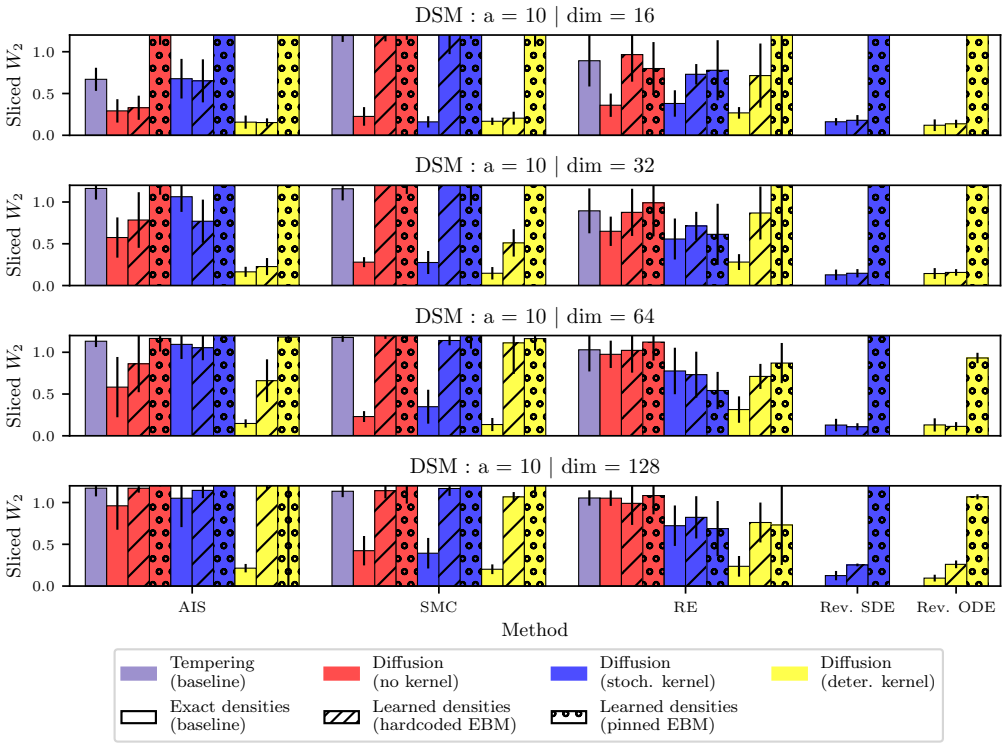

Figure 21: **Diffusion-based aMC-BG results via sliced Wasserstein distance in realistic setting (B) with DSM objective**, when targeting *TwoModes* distribution with $a = 10$, for all dimensional settings. This is complementary to Figure 4.

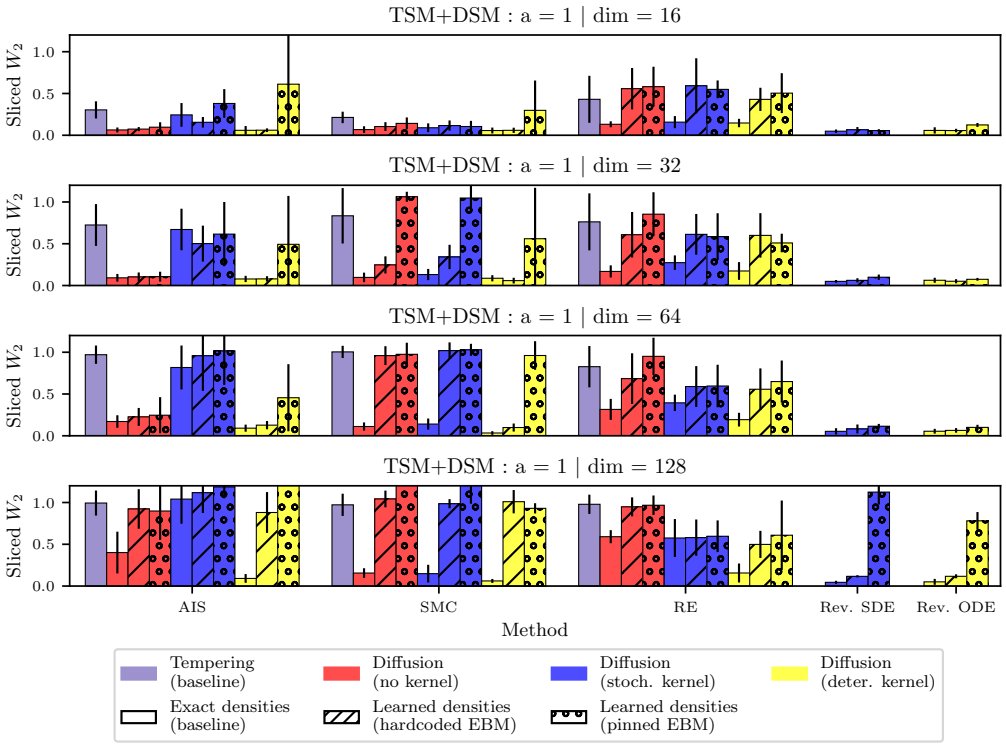

Figure 22: **Diffusion-based aMC-BG results via sliced Wasserstein distance in realistic setting (B) with TSM+DSM objective**, when targeting *TwoModes* distribution with $a = 1$, for all dimensional settings. This is complementary to Figure 4.

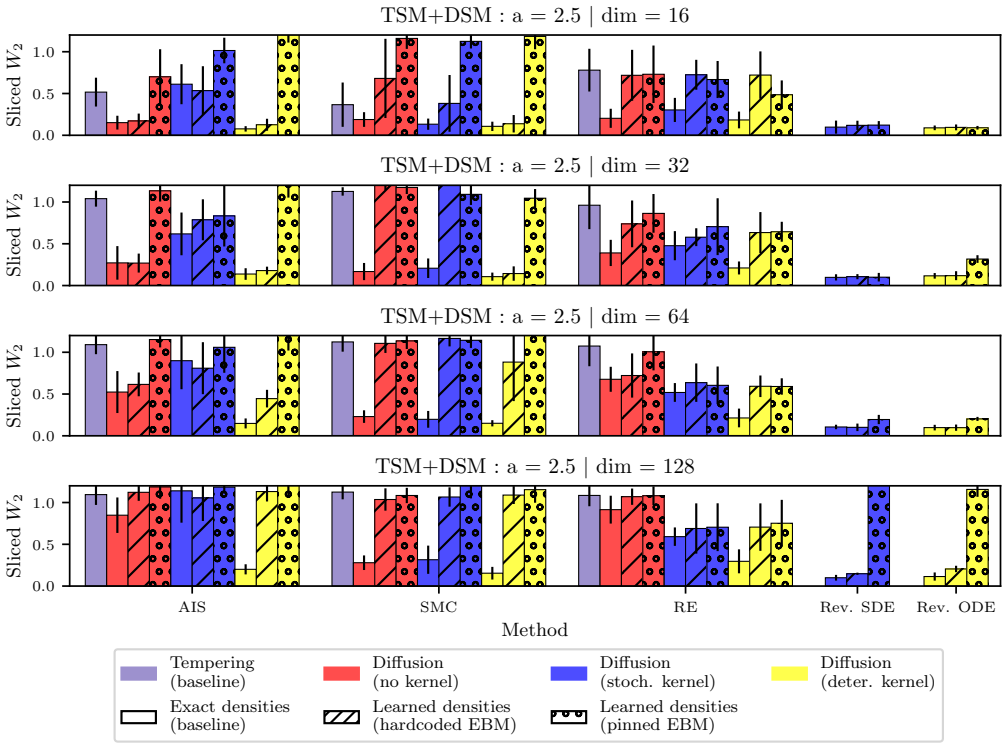

Figure 23: **Diffusion-based aMC-BG results via sliced Wasserstein distance in realistic setting (B) with TSM+DSM objective**, when targeting *TwoModes* distribution with $a = 2.5$, for all dimensional settings. This is complementary to Figure 4.

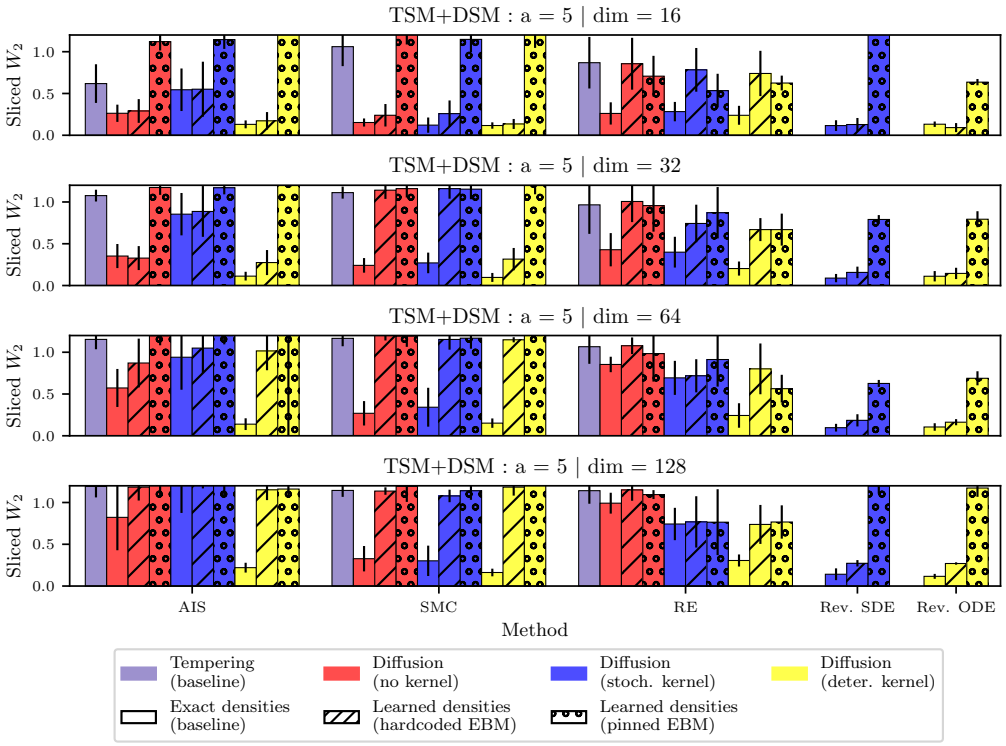

Figure 24: **Diffusion-based aMC-BG results via sliced Wasserstein distance in realistic setting (B) with TSM+DSM objective**, when targeting *TwoModes* distribution with $a = 5$, for all dimensional settings. This is complementary to Figure 4.

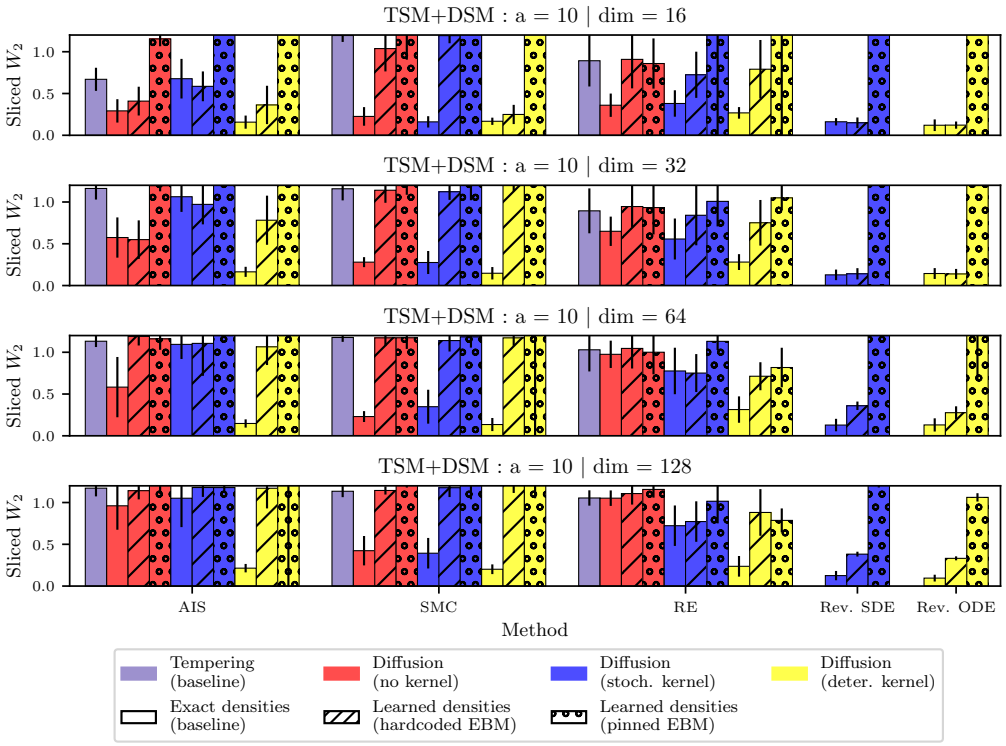

Figure 25: **Diffusion-based aMC-BG results via sliced Wasserstein distance in realistic setting (B) with TSM+DSM objective**, when targeting *TwoModes* distribution with $a = 10$, for all dimensional settings. This is complementary to Figure 4.

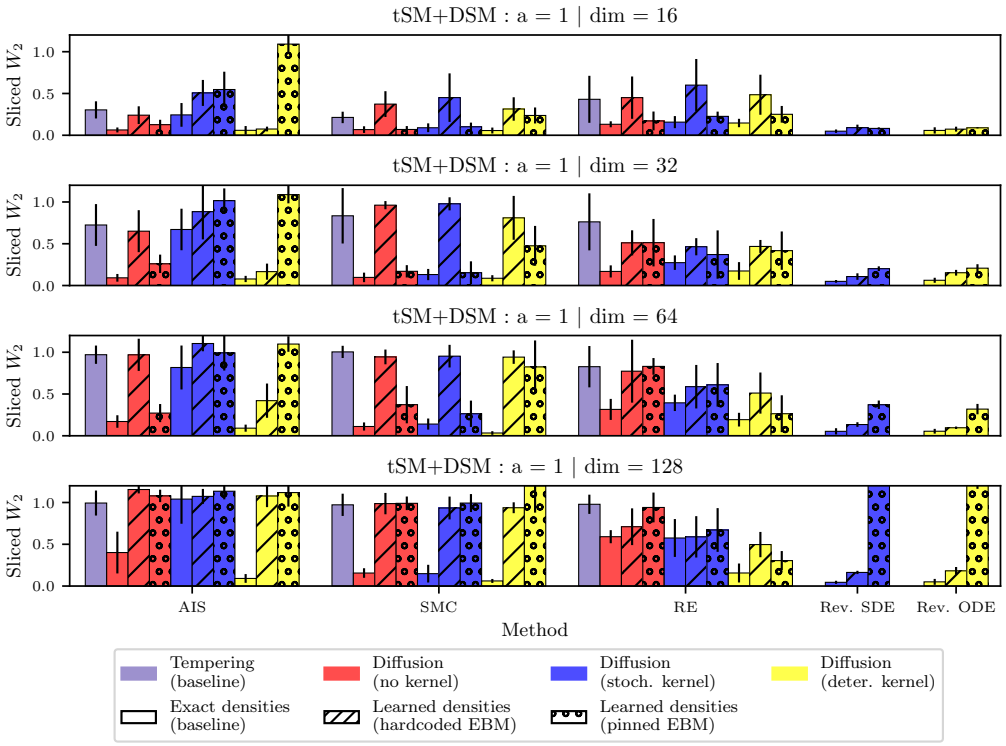

Figure 26: **Diffusion-based aMC-BG results via sliced Wasserstein distance in realistic setting (B) with tSM+DSM objective**, when targeting *TwoModes* distribution with $a = 1$, for all dimensional settings. This is complementary to Figure 4.

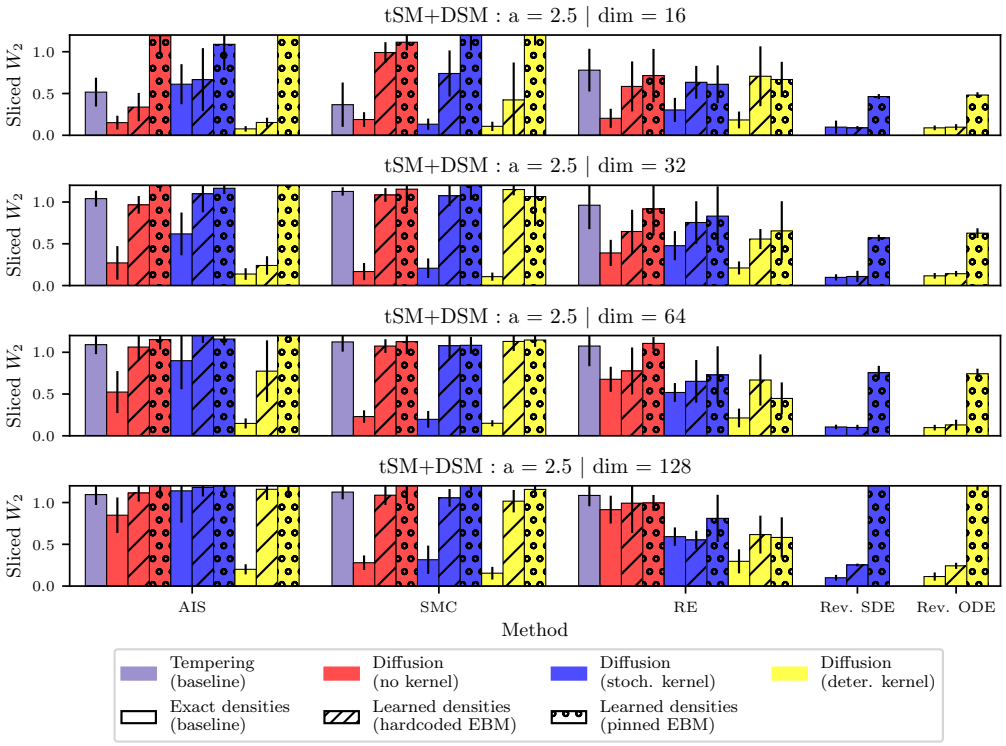

Figure 27: **Diffusion-based aMC-BG results via sliced Wasserstein distance in realistic setting (B) with tSM+DSM objective**, when targeting *TwoModes* distribution with $a = 2.5$, for all dimensional settings. This is complementary to Figure 4.

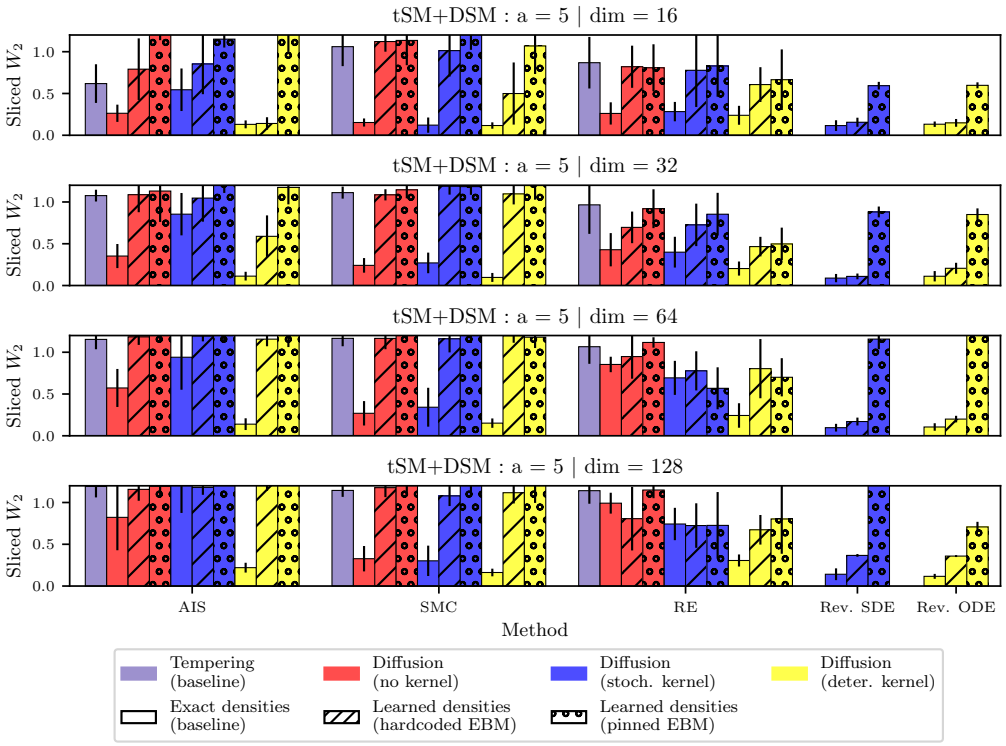

Figure 28: **Diffusion-based aMC-BG results via sliced Wasserstein distance in realistic setting (B) with tSM+DSM objective**, when targeting *TwoModes* distribution with $a = 5$, for all dimensional settings. This is complementary to Figure 4.

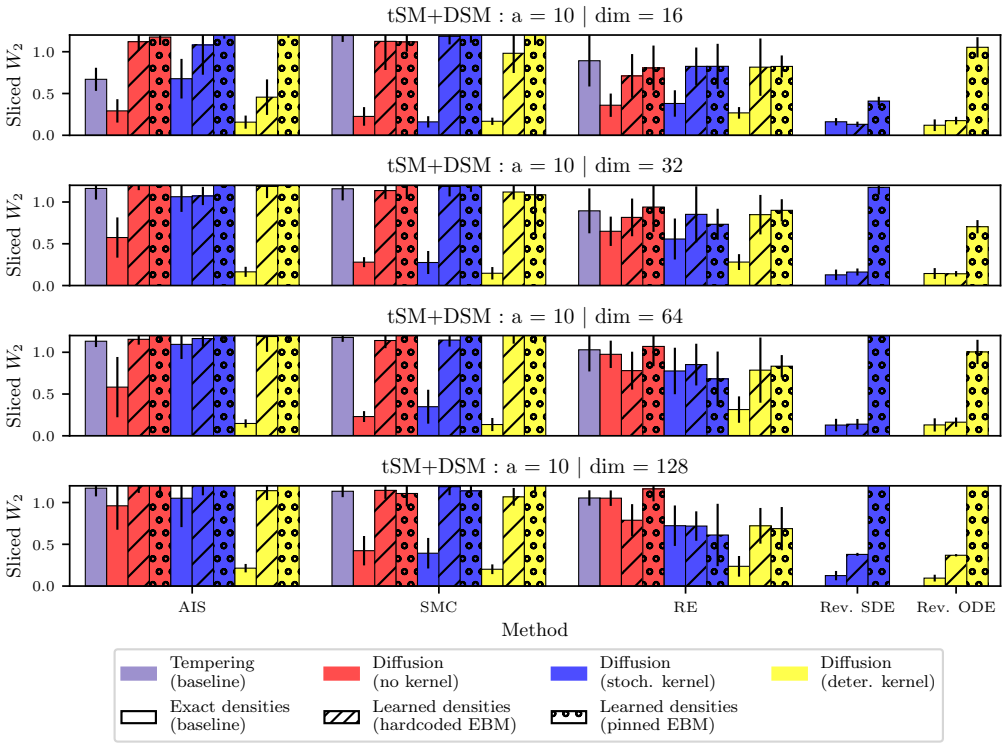

Figure 29: **Diffusion-based aMC-BG results via sliced Wasserstein distance in realistic setting (B) with tSM+DSM objective**, when targeting *TwoModes* distribution with $a = 10$, for all dimensional settings. This is complementary to Figure 4.

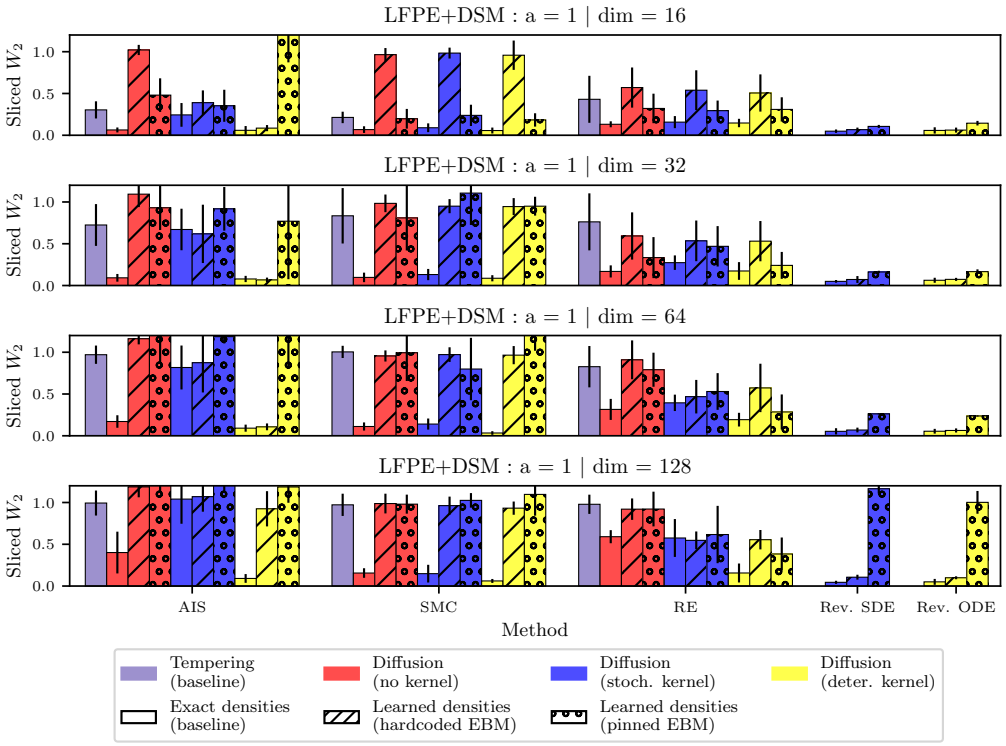

Figure 30: **Diffusion-based aMC-BG results via sliced Wasserstein distance in realistic setting (B) with LFPE+DSM objective**, when targeting *TwoModes* distribution with $a = 1$, for all dimensional settings. This is complementary to Figure 4.

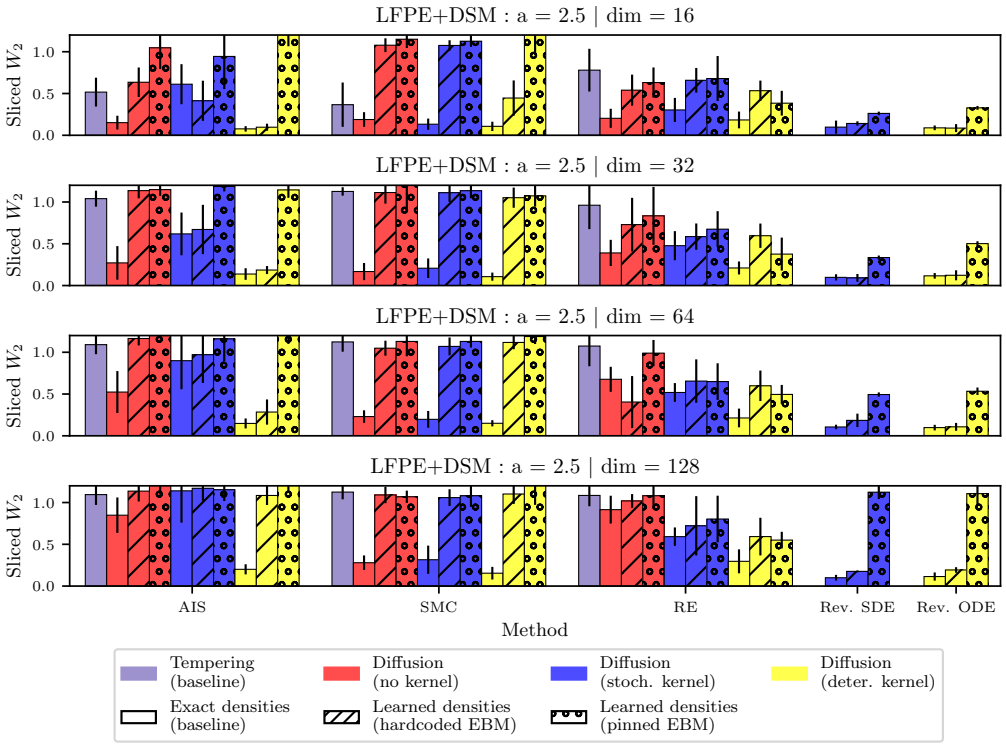

Figure 31: **Diffusion-based aMC-BG results via sliced Wasserstein distance in realistic setting (B) with LFPE+DSM objective**, when targeting *TwoModes* distribution with $a = 2.5$, for all dimensional settings. This is complementary to Figure 4.

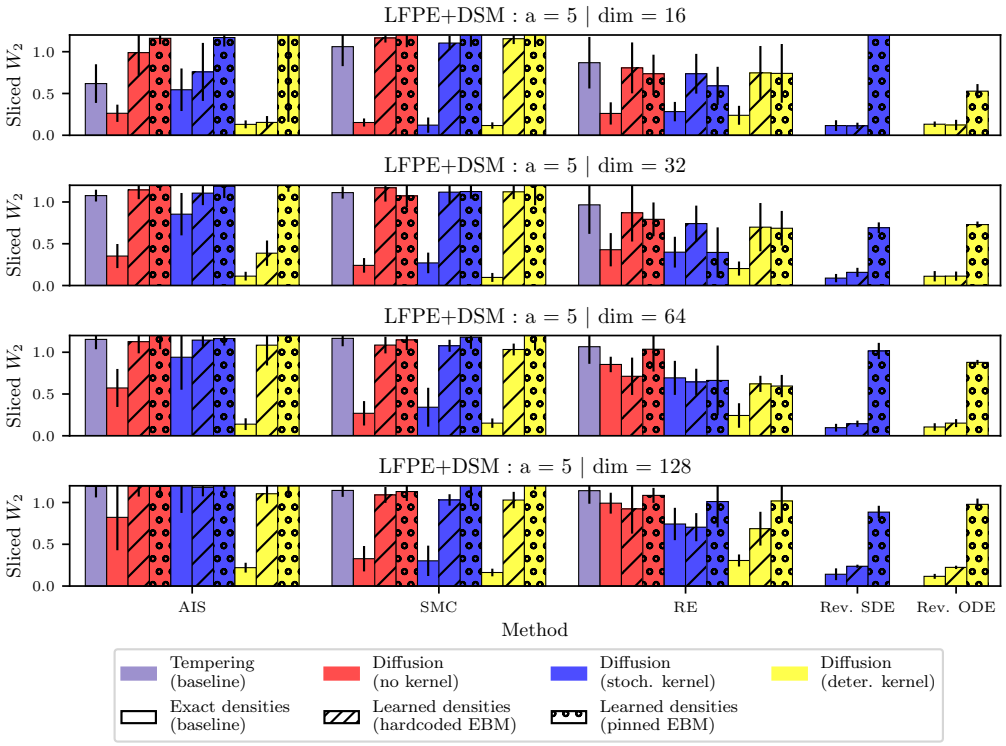

Figure 32: **Diffusion-based aMC-BG results via sliced Wasserstein distance in realistic setting (B) with LFPE+DSM objective**, when targeting *TwoModes* distribution with $a = 5$, for all dimensional settings. This is complementary to Figure 4.

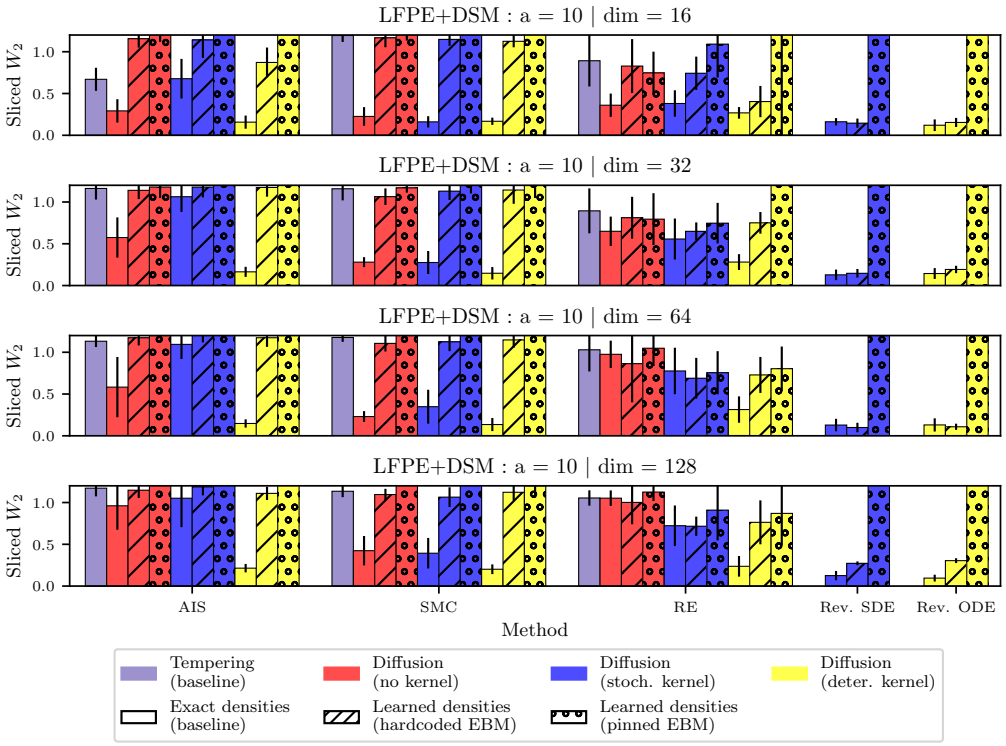

Figure 33: **Diffusion-based aMC-BG results via sliced Wasserstein distance in realistic setting (B) with LFPE+DSM objective**, when targeting *TwoModes* distribution with $a = 10$, for all dimensional settings. This is complementary to Figure 4.

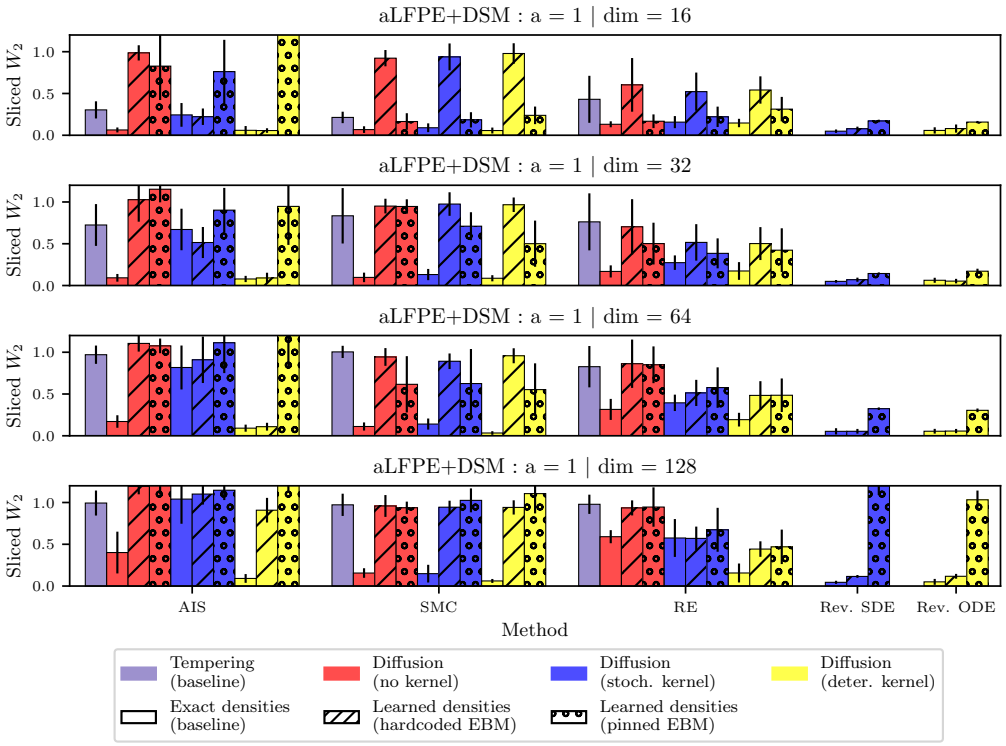

Figure 34: **Diffusion-based aMC-BG results via sliced Wasserstein distance in realistic setting (B) with aLFPE+DSM objective**, when targeting *TwoModes* distribution with $a = 1$, for all dimensional settings. This is complementary to Figure 4.

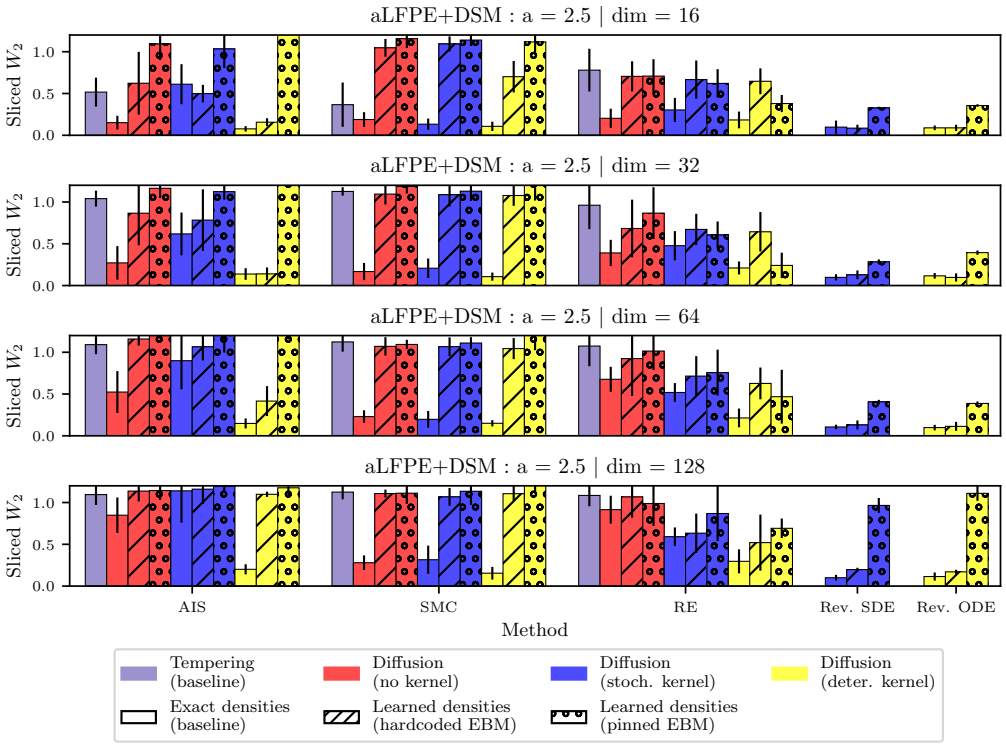

Figure 35: **Diffusion-based aMC-BG results via sliced Wasserstein distance in realistic setting (B) with aLFPE+DSM objective**, when targeting *TwoModes* distribution with $a = 2.5$, for all dimensional settings. This is complementary to Figure 4.

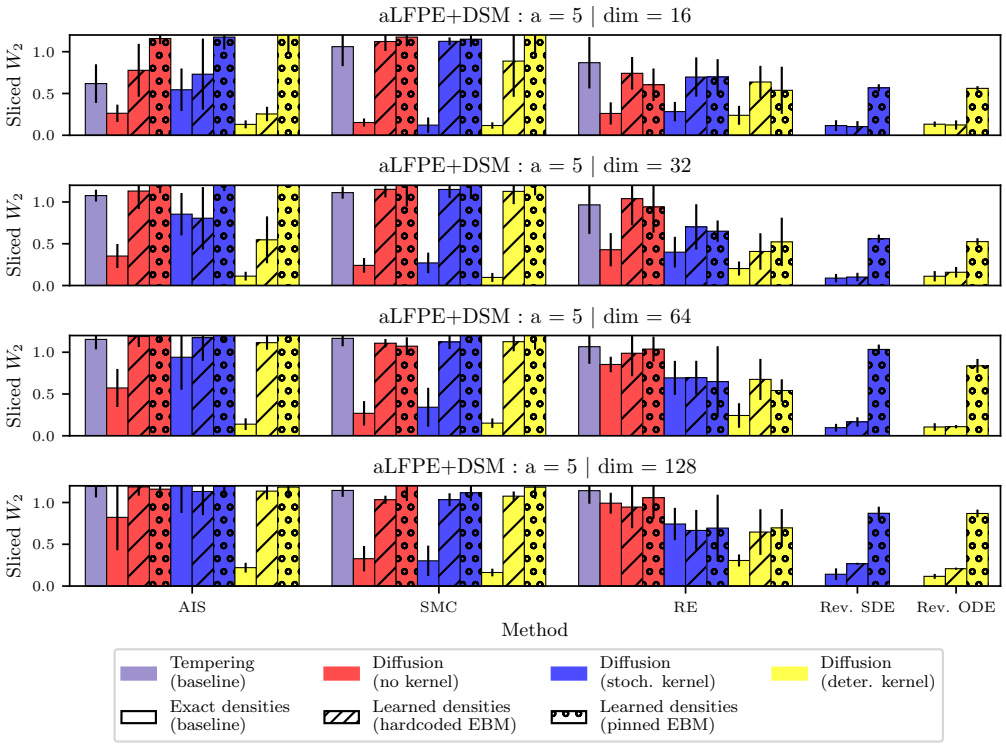

Figure 36: **Diffusion-based aMC-BG results via sliced Wasserstein distance in realistic setting (B) with aLFPE+DSM objective**, when targeting *TwoModes* distribution with $a = 5$, for all dimensional settings. This is complementary to Figure 4.

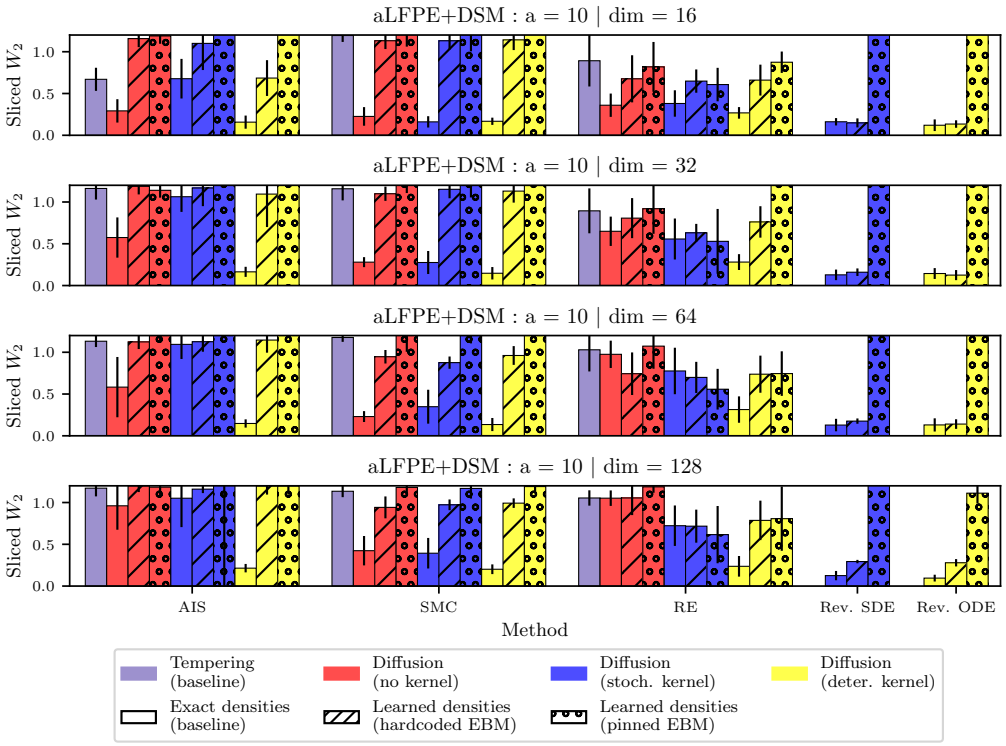

Figure 37: **Diffusion-based aMC-BG results via sliced Wasserstein distance in realistic setting (B) with aLFPE+DSM objective**, when targeting *TwoModes* distribution with $a = 10$, for all dimensional settings. This is complementary to Figure 4.

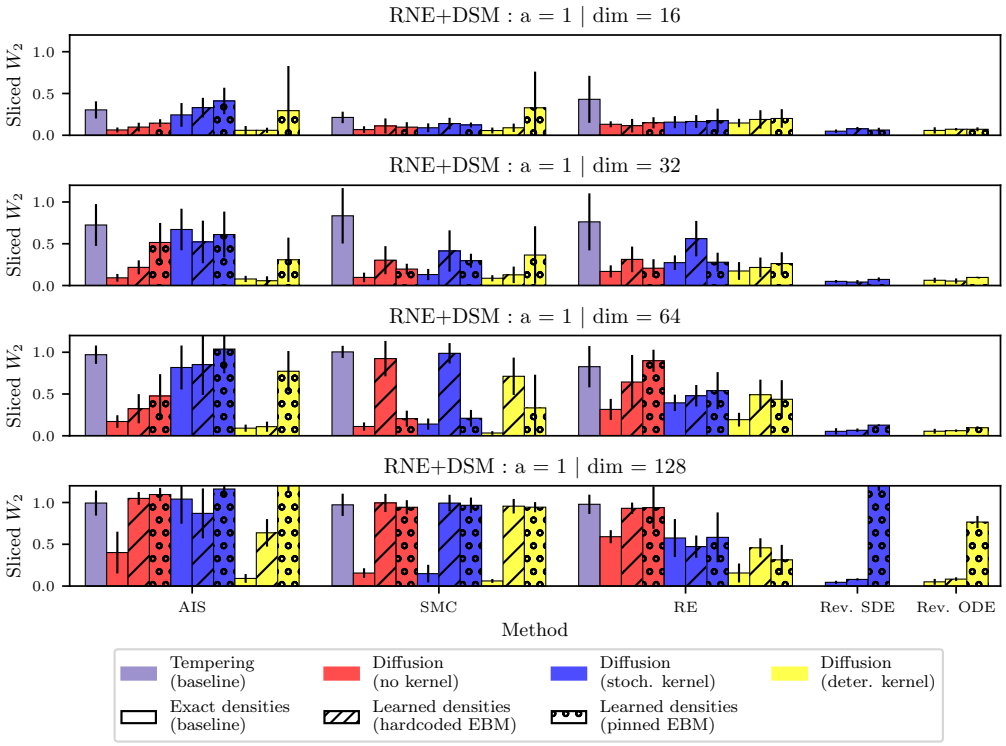

Figure 38: **Diffusion-based aMC-BG results via sliced Wasserstein distance in realistic setting (B) with RNE+DSM objective**, when targeting *TwoModes* distribution with $a = 1$, for all dimensional settings. This is complementary to Figure 4.

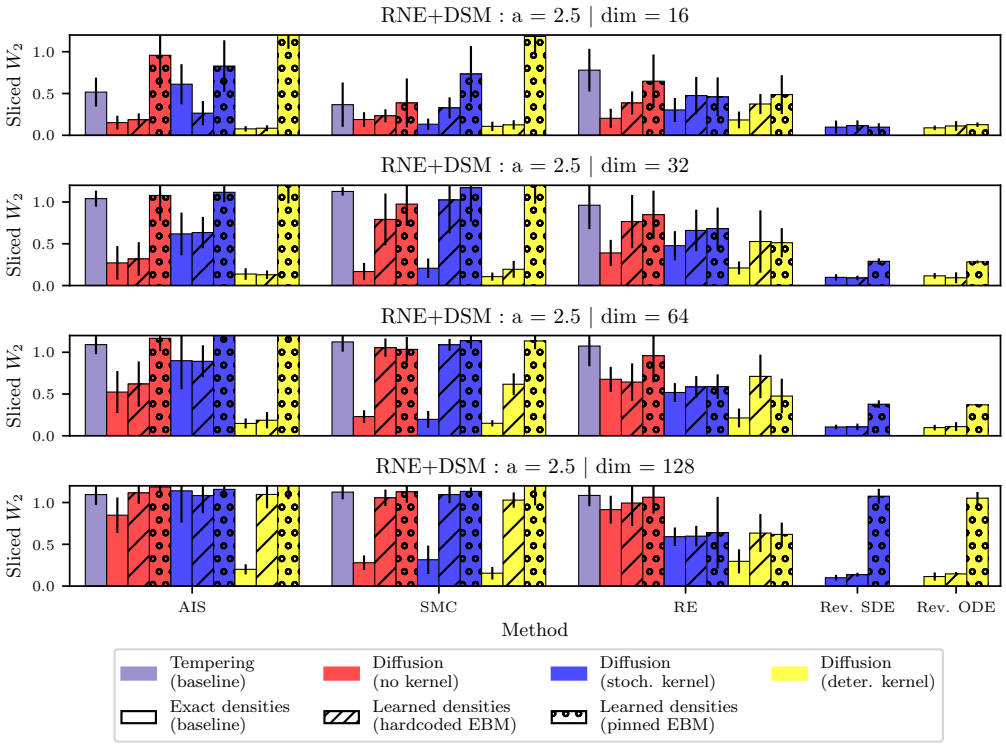

Figure 39: **Diffusion-based aMC-BG results via sliced Wasserstein distance in realistic setting (B) with RNE+DSM objective**, when targeting *TwoModes* distribution with $a = 2.5$, for all dimensional settings. This is complementary to Figure 4.

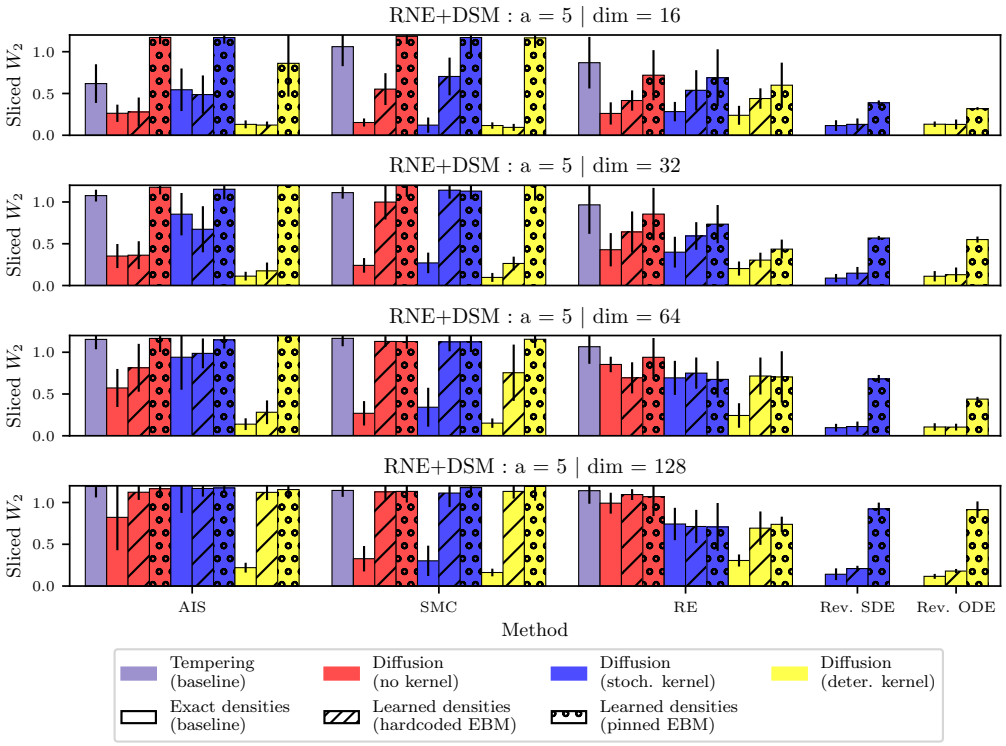

Figure 40: **Diffusion-based aMC-BG results via sliced Wasserstein distance in realistic setting (B) with RNE+DSM objective**, when targeting *TwoModes* distribution with $a = 5$, for all dimensional settings. This is complementary to Figure 4.

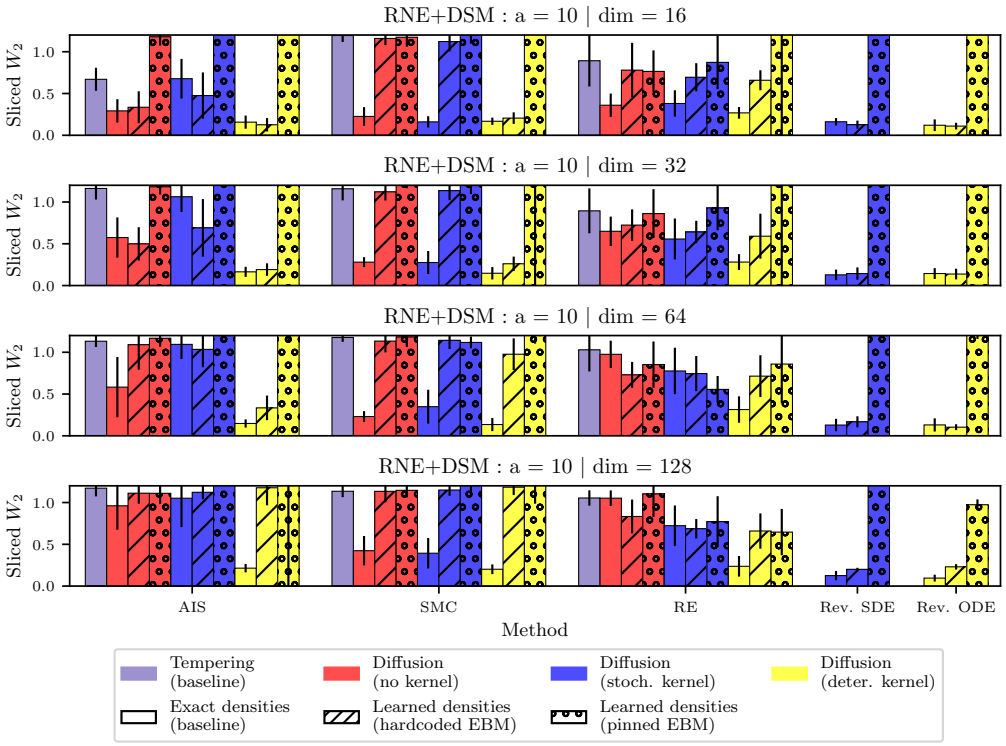

Figure 41: **Diffusion-based aMC-BG results via sliced Wasserstein distance in realistic setting (B) with RNE+DSM objective**, when targeting *TwoModes* distribution with $a = 10$, for all dimensional settings. This is complementary to Figure 4.

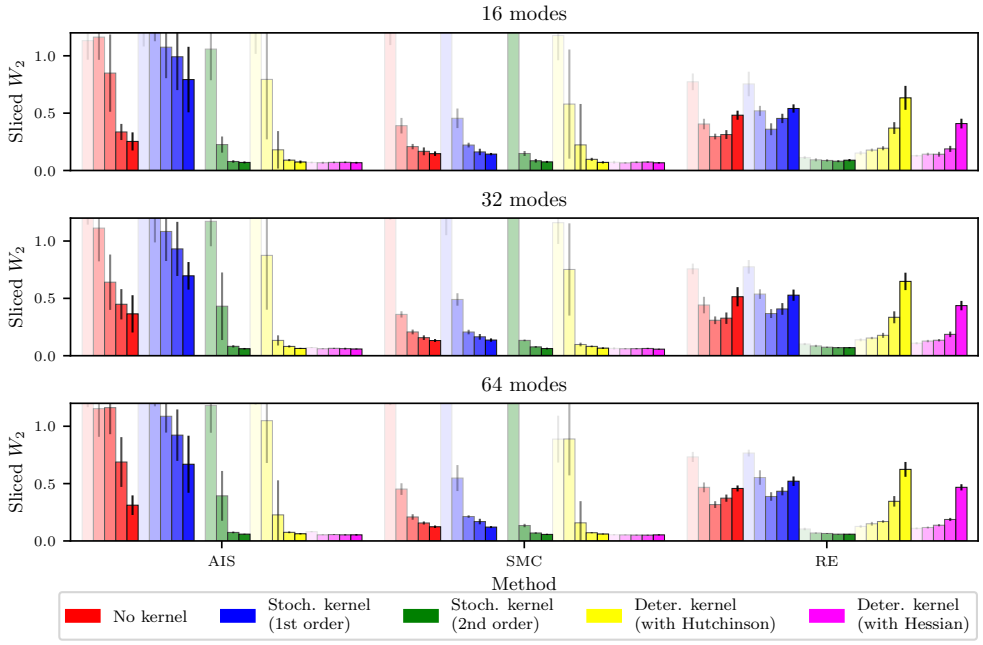

Figure 42: **Diffusion-based aMC-BG results via sliced Wasserstein distance in idealized setting (A)**, when targeting *ManyModes* distribution in all possible settings. This is complementary to Figure 3.

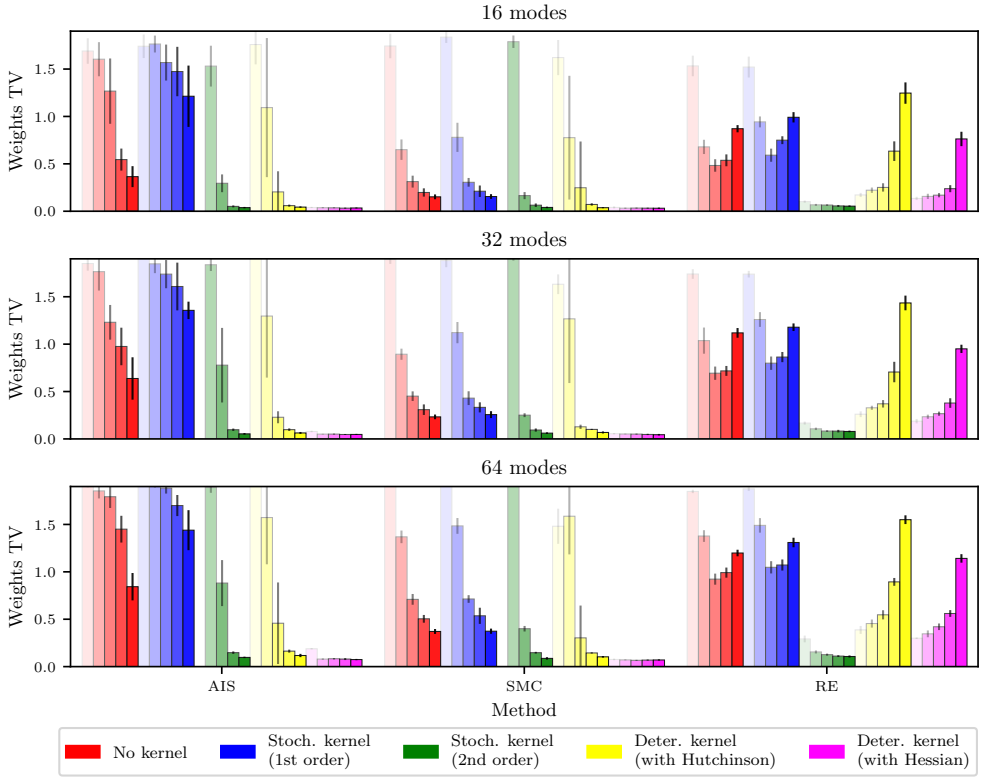

Figure 43: **Diffusion-based aMC-BG results via weight histogram total variation distance in idealized setting (A)**, when targeting *ManyModes* distribution in all possible settings. This is complementary to Figure 3.

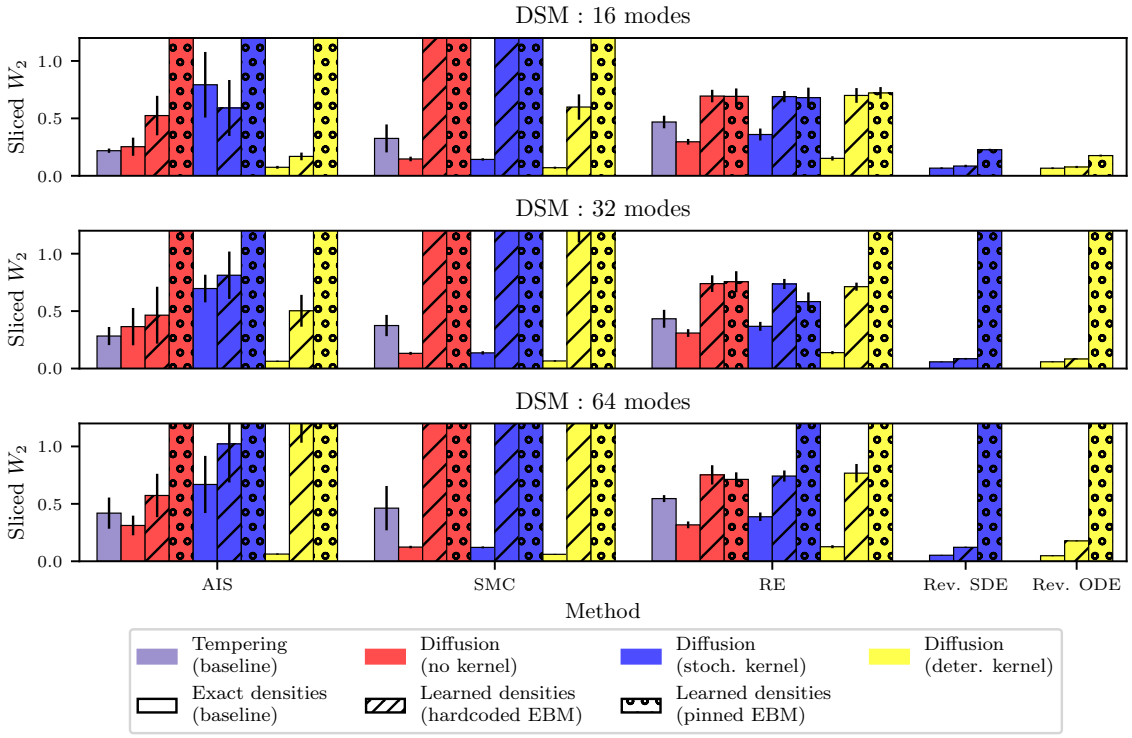

Figure 44: **Diffusion-based aMC-BG results via sliced Wasserstein distance in realistic setting (B) with DSM objective**, when targeting *ManyModes* distribution in all possible settings. This is complementary to Figure 4.

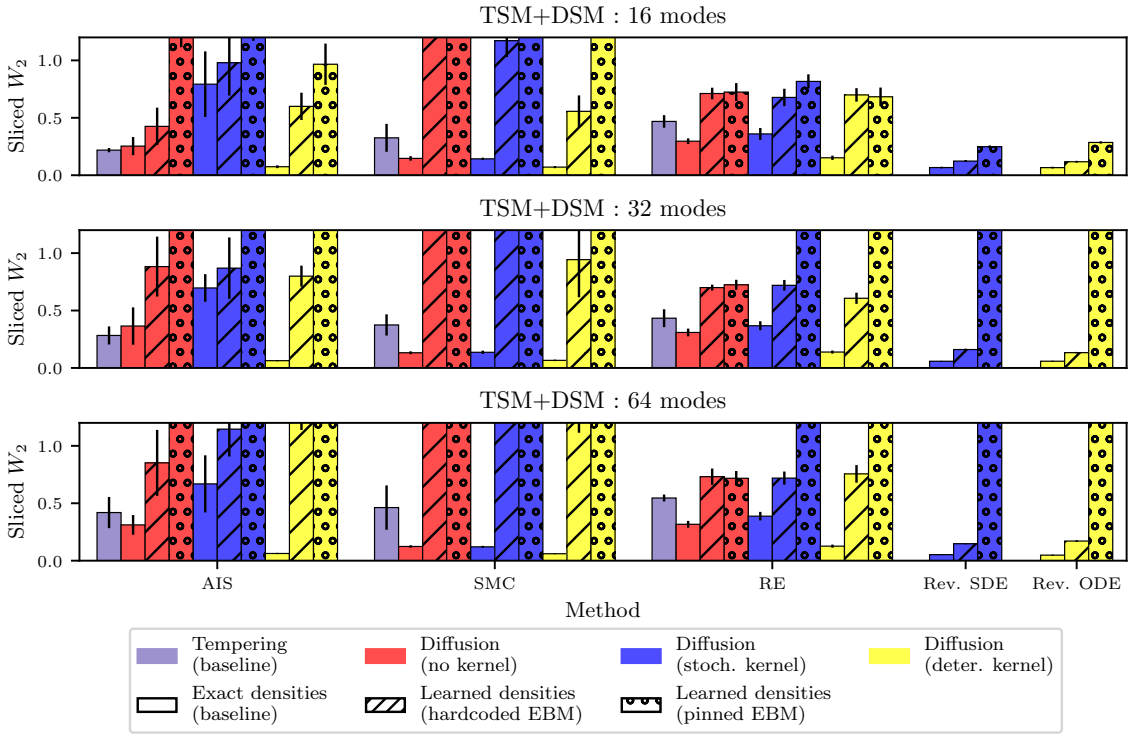

Figure 45: **Diffusion-based aMC-BG results via sliced Wasserstein distance in realistic setting (B) with TSM+DSM objective**, when targeting *ManyModes* distribution in all possible settings. This is complementary to Figure 4.

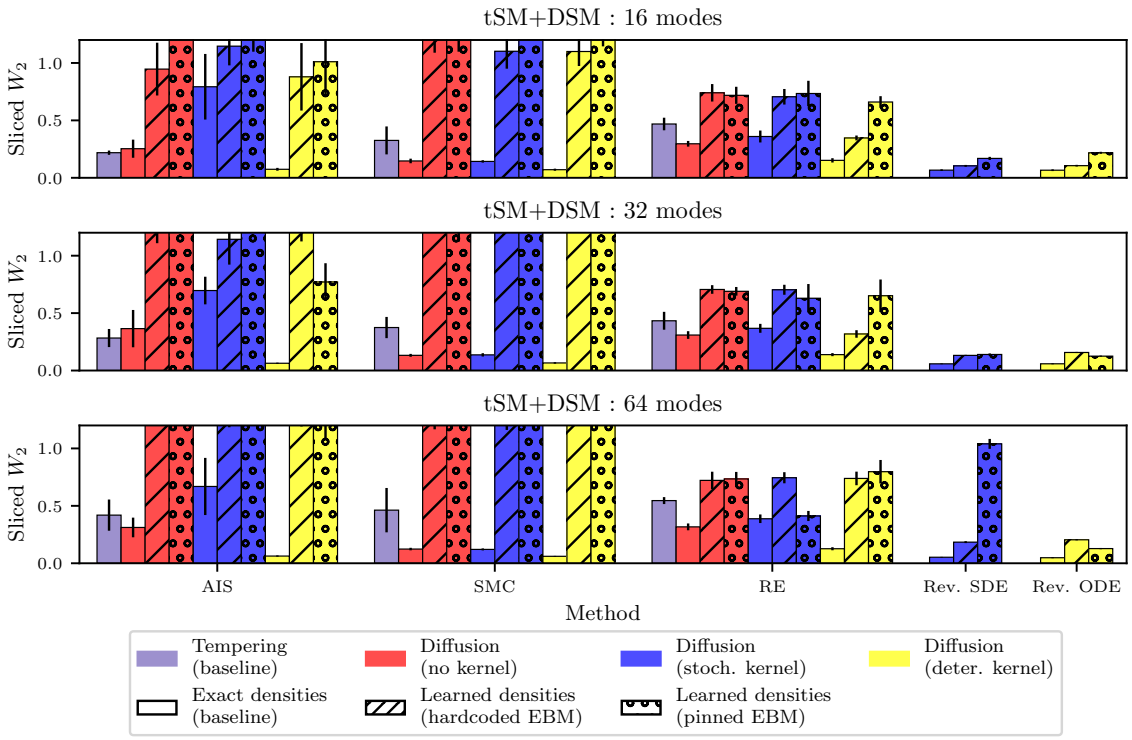

Figure 46: **Diffusion-based aMC-BG results via sliced Wasserstein distance in realistic setting (B) with tSM+DSM objective**, when targeting *ManyModes* distribution in all possible settings. This is complementary to Figure 4.

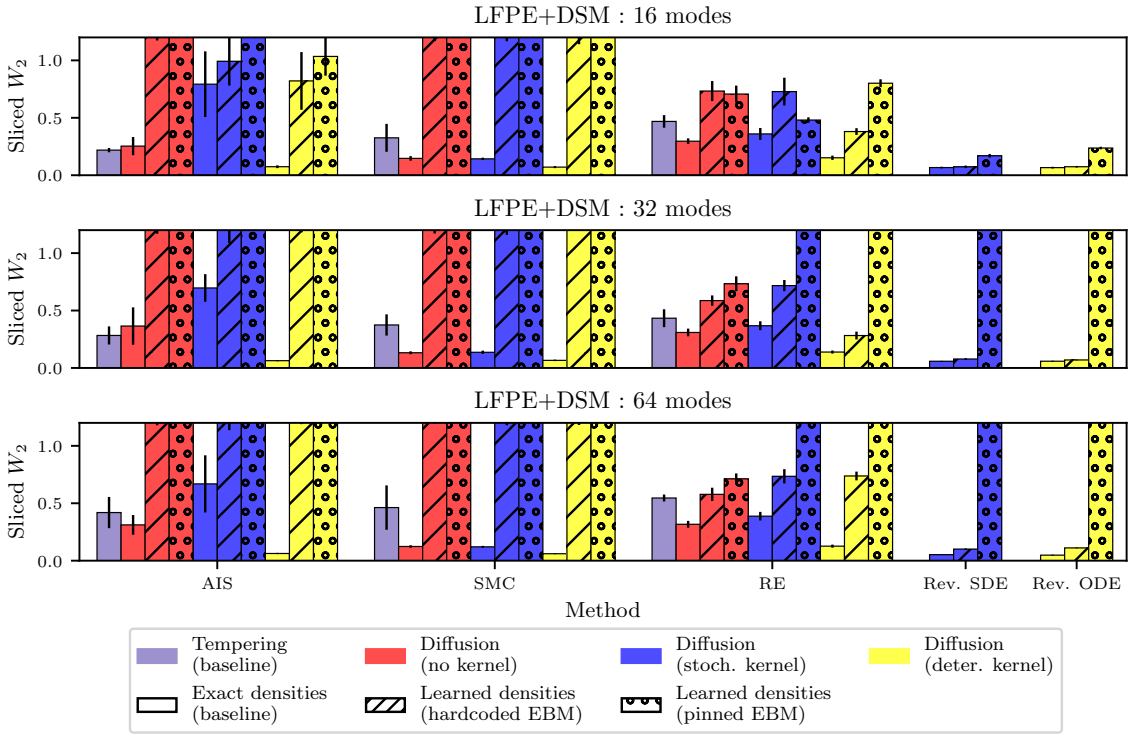

Figure 47: **Diffusion-based aMC-BG results via sliced Wasserstein distance in realistic setting (B) with LFPE+DSM objective**, when targeting *ManyModes* distribution in all possible settings. This is complementary to Figure 4.

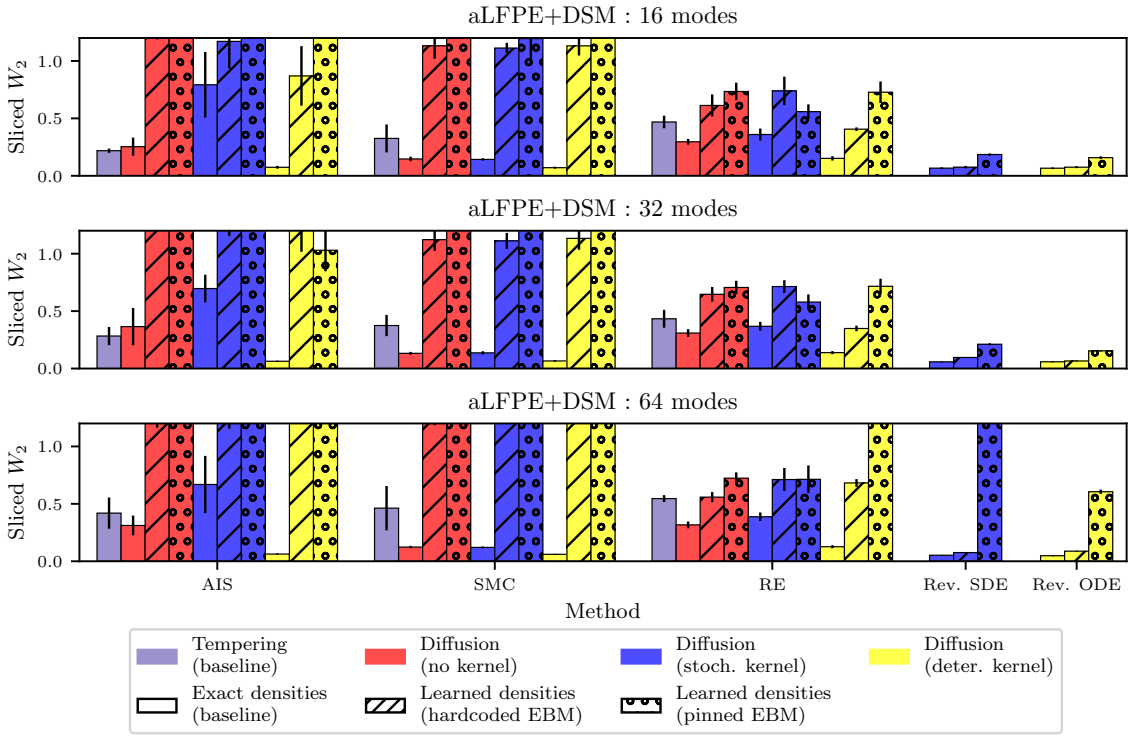

Figure 48: **Diffusion-based aMC-BG results via sliced Wasserstein distance in realistic setting (B) with aLFPE+DSM objective**, when targeting *ManyModes* distribution in all possible settings. This is complementary to Figure 4.

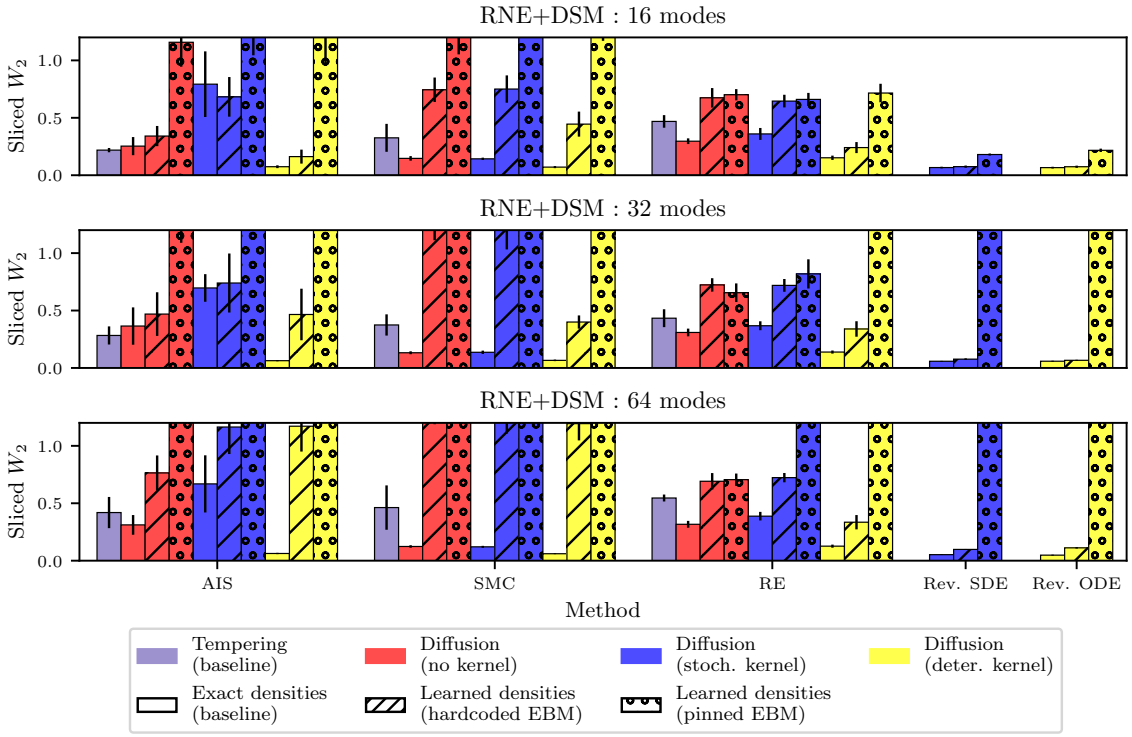

Figure 49: **Diffusion-based aMC-BG results via sliced Wasserstein distance in realistic setting (B) with RNE+DSM objective**, when targeting *ManyModes* distribution in all possible settings. This is complementary to Figure 4.

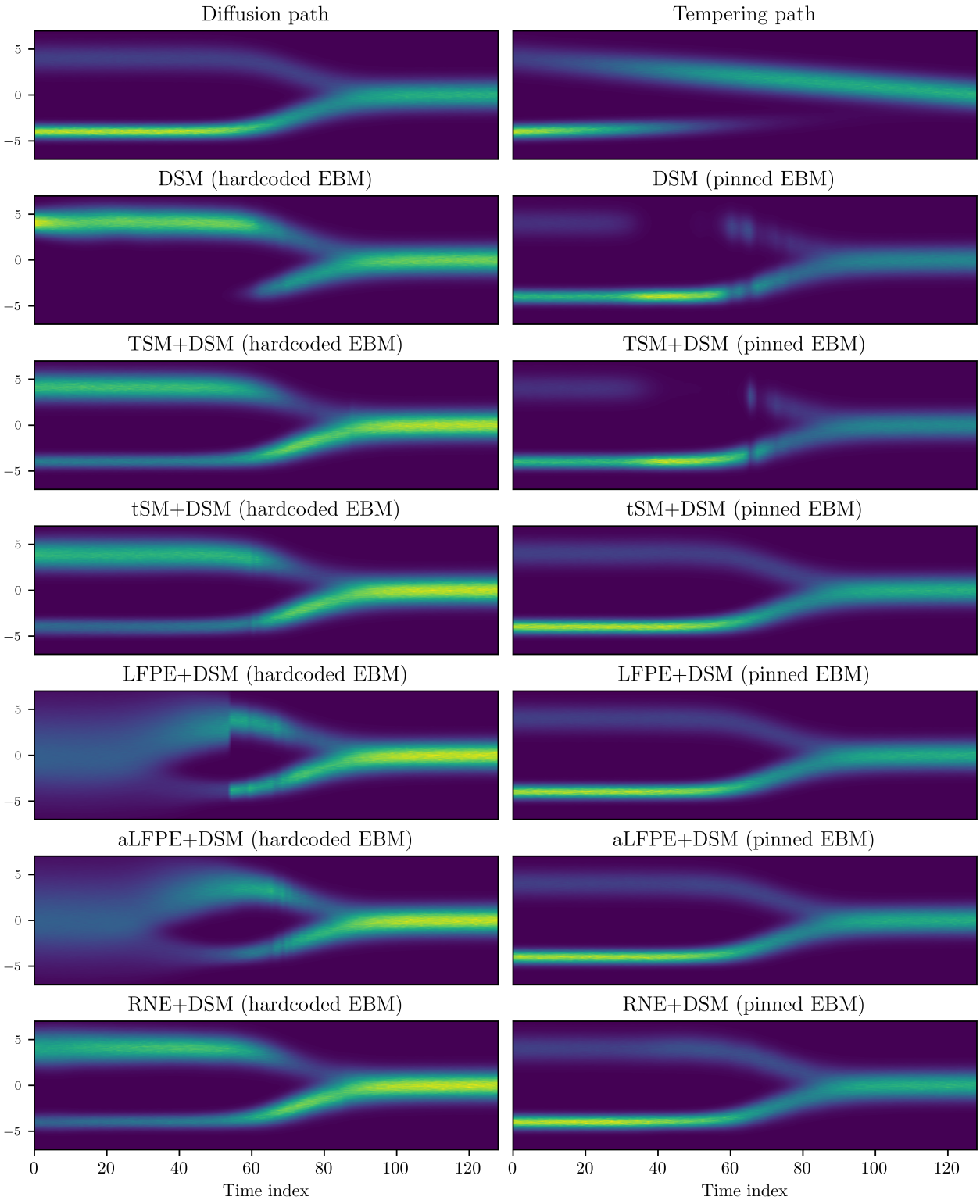

Figure 50: **Diffusion density paths bridging π^{base} (last time index) to the same *TwoModes* target as in Figure 5 (first time index). (First row)** the exact diffusion path is displayed on the left, the exact tempering path on the right, **(From second to last row)** we display the learned density path when using the DSM, TSM+DSM, tSM+DSM, LFPE+DSM, aLFPE+DSM or RNE+DSM objective, with identical computational budget, **(Left)** use of Hardcoded EBM, **(Right)** use of Pinned EBM. This is complementary to Figure 6. Zoom in to get more details.

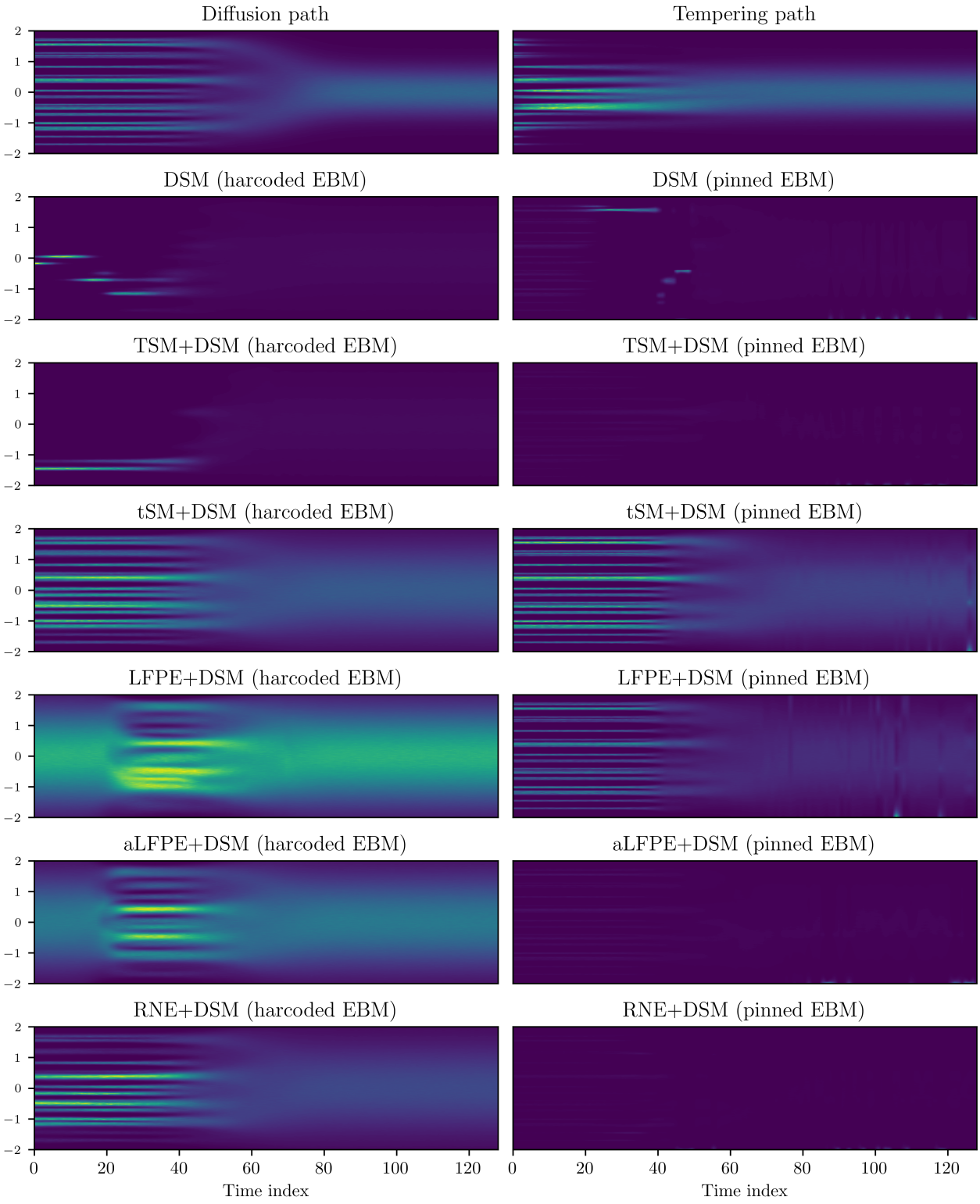

Figure 51: **Diffusion density paths bridging π^{base} (last time index) to the same *ManyModes* target as in Figure 5 (first time index). (First row)** the exact diffusion path is displayed on the left, the exact tempering path on the right, **(From second to last row)** we display the learned density path when using the DSM, TSM+DSM, tSM+DSM, LFPE+DSM, aLFPE+DSM or RNE+DSM objective, with identical computational budget, **(Left)** use of Hardcoded EBM, **(Right)** use of Pinned EBM. This is complementary to Figure 6. Zoom in to get more details.

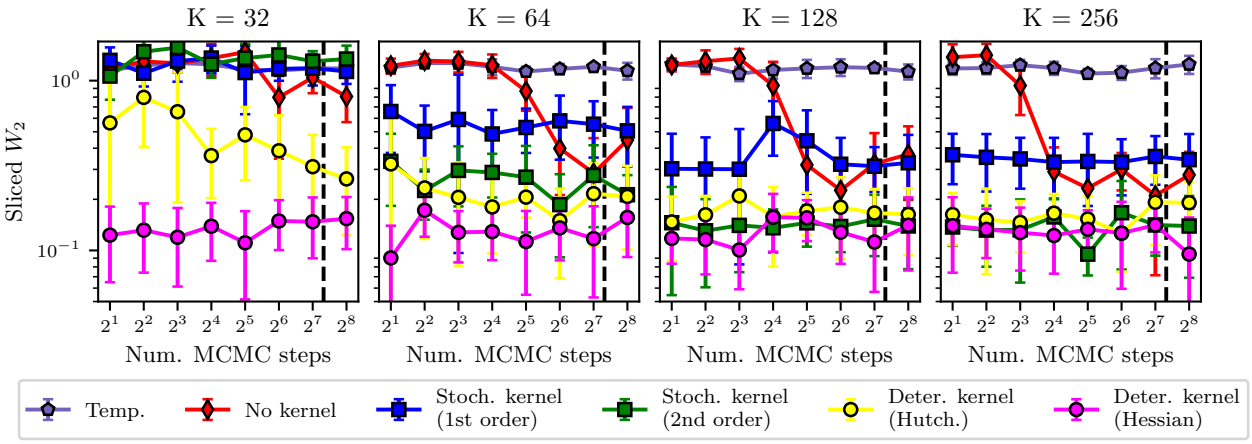

Figure 52: **Sensitivity of SMC methods (including diffusion-based BGs) with respect to the number of local MCMC steps across different values of K.** We do not consider any MCMC warm-up procedure here. The default setting of the main experiments is represented by a dashed line.

D.4 Ablation studies

In this section, we investigate the sensitivity of the annealed samplers considered in this work with respect to their respective hyperparameters, with a particular focus on diffusion-based aMC-BGs featuring deterministic transitions, see Section 4. To keep the analysis focused and interpretable, all experiments are conducted in idealized setting **(A)** on a representative configuration, the so-called "medium case", of the *TwoModes* distribution with mode spacing $a = 5.0$ and dimension $d = 64$.

Dependence of SMC with respect to the number of local MCMC steps. Figure 52 examines the sensitivity of all SMC-based methods to the number of local MCMC steps performed (with MALA sampler) at each intermediate global step $k \in \{0, \dots, K\}$, where the number of annealing levels K is selected in $\{32, 64, 128, 256\}$. Note that, unlike the main experiments, we do not use any MCMC warm-up here, thereby making the corresponding sampling runs much less computationally intensive on average; for comparison, we display with a dashed line the MCMC budget used in the main experiments (*i.e.*, 160 steps). We first observe that, in the case of the tempering path, increasing the number of local sampling steps does not bring better results for each value of K, thereby proving the unfavorable sampling conditions of this density path. For diffusion-based methods, we remark that increasing the number of MCMC steps is actually beneficial when no kernel is used, *i.e.*, specifically when the score information is not used at a global scale (between transitions) but a local scale (with MALA sampler); in particular, the no-kernel variant reaches the performance of the version using a first-order denoising kernel after a certain threshold. The other methods show little sensitivity to the number of MCMC steps in this range, suggesting robustness to this hyperparameter choice.

Dependence of RE with respect to the swap period. In Figure 53, we investigate the sensitivity of all RE-based methods to the choice of the swap period, which corresponds to the number of local MCMC steps performed between swap operations (selected in $\{2, 4, 8, 16, 32, 64\}$), for the range of annealing levels $K \in \{32, 64, 128, 256\}$. Note that the overall MCMC budget is the same as in the main experiments in order to isolate the impact of the swap period on the sampling results; for comparison, we display with a dashed line the swap period used by default (set to 8). Based on these results, we may identify two groups of samplers : (i) methods with poor performance (tempering path, diffusion path without kernel or with first-order stochastic kernel), for which the results are globally independent to the setting of the swap period, and (ii) methods with better performance (diffusion path with second-order stochastic kernel or deterministic kernels), where we observe a sweet spot for the swap period. For the latter methods, this result clearly validates the intuitive trade-off between global exploration and local exploration in RE, induced by the number of local MCMC steps between swaps under a fixed global sampling budget. Indeed, while increasing the swap period tends to prioritize local exploration, decreasing the swap period rather favors global exploration, thus hindering the overall performance of RE in both cases.

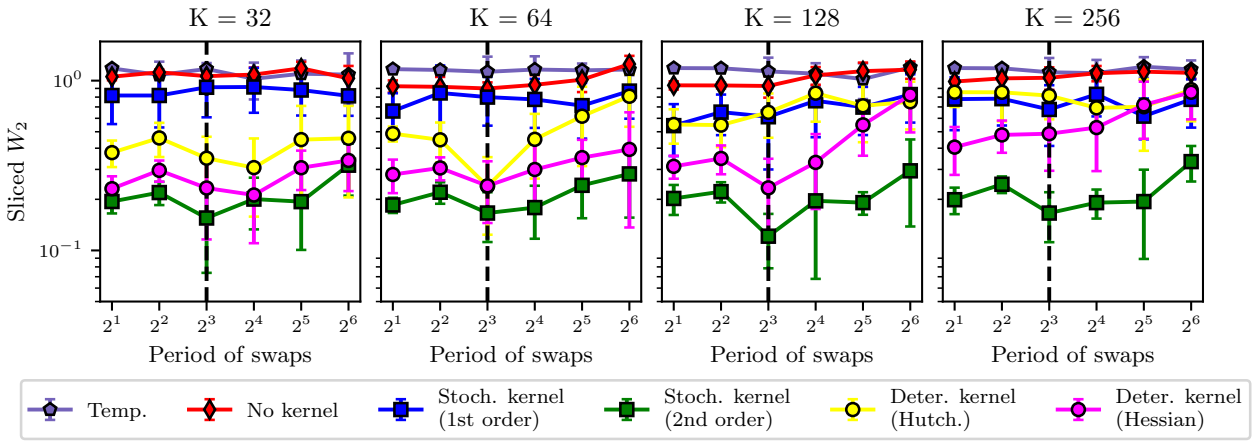

Figure 53: **Sensitivity of RE methods (including diffusion-based BGs) with respect to the swap period across different values of K.** The global number of MCMC steps is the same as in the main experiments. The default setting of the main experiments is represented by a dashed line. For the three best performing methods, we observe a sweet spot when varying the swap period.

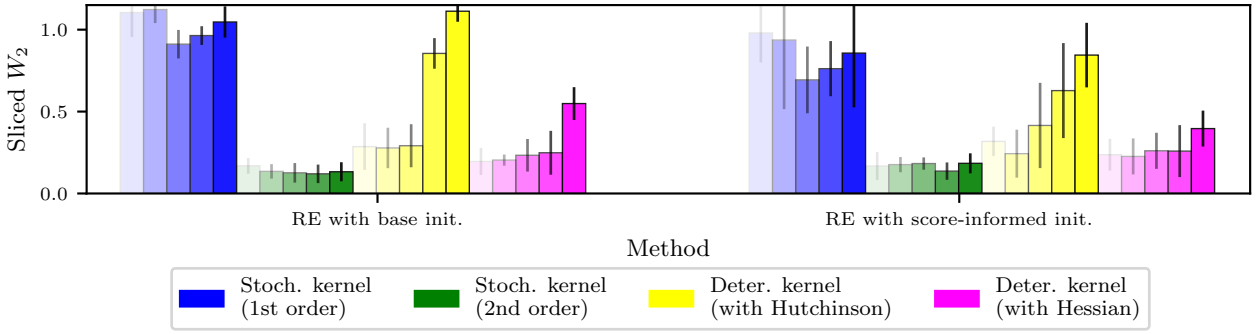

Figure 54: **Sensitivity of diffusion-based RE-BGs methods with respect to the per-level initialization across different values of $K \in \{16, 32, 64, 128, 256\}$: the darker the bar, the higher K.** The group of bars displayed on the right of the figure ("RE with score-informed init") exactly corresponds to the RE bars from the third row of Figure 3. Compared to the naive initialization ("RE with base init"), populating the levels by simulating the reverse SDE performs on par or better for each setting.

Dependence of RE with respect to the per-level initialization. When targeting the diffusion path with RE samplers that exploit at least first-order information (*i.e.*, the scores of the intermediate marginal distributions), one may leverage this knowledge to wisely initialize the annealing levels. Indeed, by simulating the reverse SDE starting from π^{base}, we are able to populate each level with samples that are approximately distributed according to the related target marginal distribution, thereby offering a favorable setting for faster MCMC convergence. In our main experiments, we systematically adopt this strategy, and compare it to the naive initialization (that is performed in the standard version of RE) with samples from π^{base} at each level. The results displayed in Figure 54 (which use the same color code as Figure 3) show that the score-informed warm-up initialization clearly benefits to all RE variants, especially when the number of levels is high.

Analysis of diffusion-based aMC-BGs with deterministic transitions. Finally, we propose three ablation studies focused on the diffusion-based variants of AIS, SMC and RE introduced in Section 4. For each annealed sampling method, we investigate for the range of annealing levels $K \in \{32, 64, 128\}$, the following :

• their inter-dependence to the number of fixed-point iterations $M \in \{2, 4, 8, 16\}$ and the truncation order index $I \in \{1, 2, 3, 4, 5\}$, see Figure 55. For all samplers, we observe that the performance of the Hessian-based variant is globally independent to the setting of (M, I), thereby proving its robust accuracy under restricted computational budgets (especially lower than in the main experiments). For AIS/SMC methods, we observe that the performance gap between Hutchinson and Hessian-based variants diminishes when taking K larger with fixed (M, I), and is relatively constant with respect to (M, I) with large values of K; on the other hand, when K is low, we observe that a large number of fixed-point iterations is needed to close this gap. Inversely, this gap is very small at low K for RE sampler, and gets larger as K increases; interestingly, we do not note a significant dependence to (M, I) in RE variants.

• their inter-dependence to the truncation order index $I \in \{1, 2, 3, 4, 5\}$ and the number of Hutchinson samples $N_H \in \{2, 4, 8, 16, 32, 64, 128\}$ (here, the Hessian is used as the baseline), see Figure 56. For AIS/SMC samplers, we observe that, for fixed values of N_H and K, taking a large truncation order often degrades the performance of the Hutchinson-based approach, especially when K is low. We attribute this behavior to the growing error accumulation of the expansion series terms, when increasing the value of I. For those samplers, it seems more advantageous to rather consider a large value of K, as well as intermediary values for I and N_H, in order to reach a good compromise in the estimation of the Jacobian log-determinants via expansion series. On the other hand, for RE sampler, the gap between the Hutchinson-based method and its baseline is remarkably constant with the choice of (I, N_H), but tends to increase when K is large.

• their inter-dependence to the number of fixed-point iterations $M \in \{2, 4, 8, 16\}$ and the number of Hutchinson samples $N_H \in \{2, 4, 8, 16, 32, 64, 128\}$ (here, the Hessian is used as the baseline), see Figure 57. Based on the obtained results, we may draw consistent conclusions with respect to the two previous ablation studies: increasing the number of fixed-point iterations does not seem to specifically improve the accuracy of all annealed samplers with deterministic transitions (with or without access to Hessians), setting N_H with an intermediary value (for instance, 32 as we do in our main experiments) provides a good compromise between accuracy and computational effort, as long as K is large for AIS/SMC samplers or low for RE sampler.

Note that we display within each figure a dashed line to represent the setting used in our main experiments, given by $(M, I, N_H) = (4, 3, 32)$. In all presented ablation studies, we observe that it systematically embodies a positive balance between computational cost and empirical performance, thus motivating its use it as default setting.

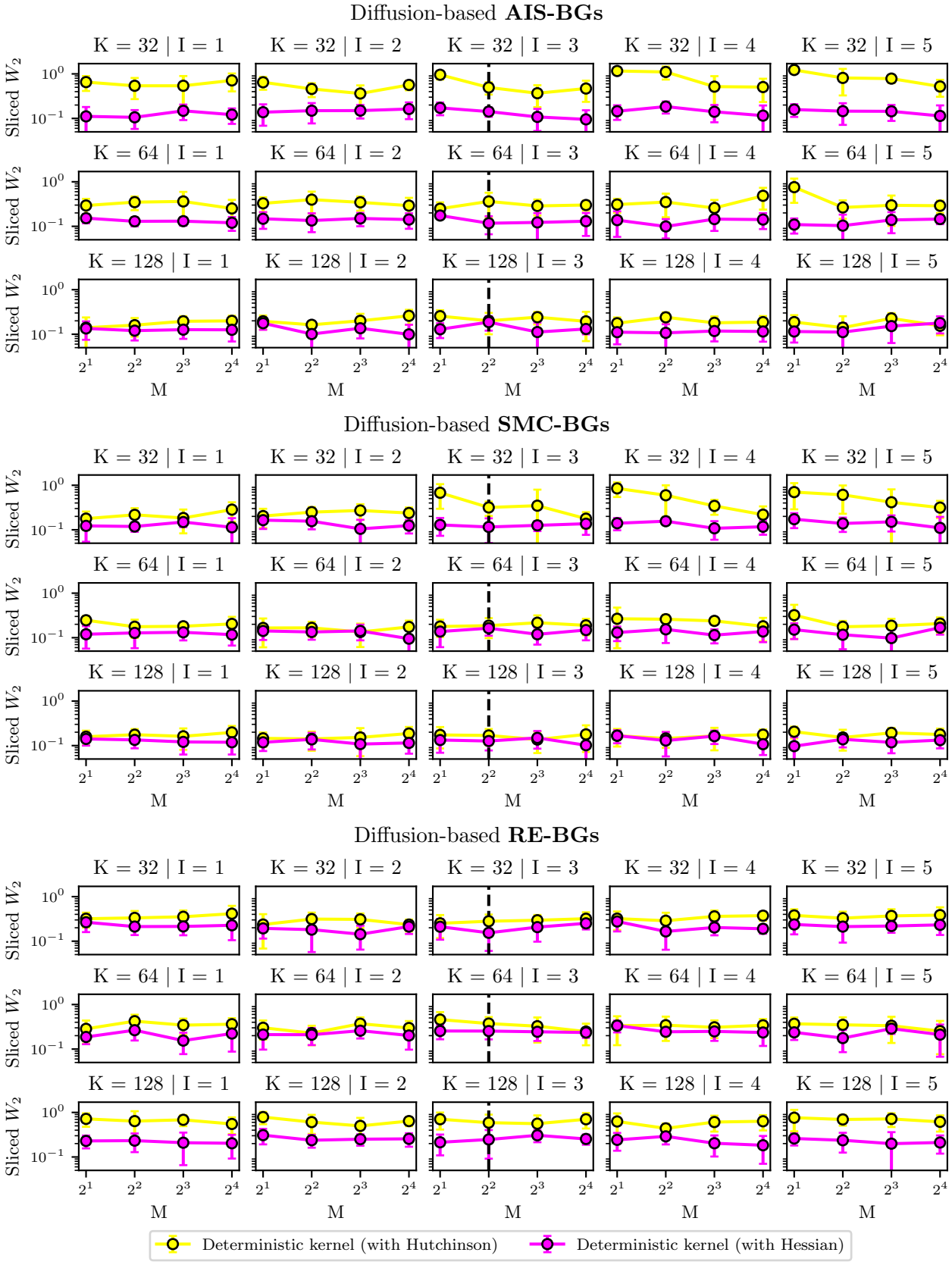

Figure 55: **Sensitivity of all diffusion-based aMC-BGs with deterministic transitions with respect to the number of fixed-point iterations (x-axis) and the truncation order I, across different values of K. (Top)** AIS variant, **(Middle)** SMC variant, **(Bottom)** RE variant. We fix $N_H = 32$.

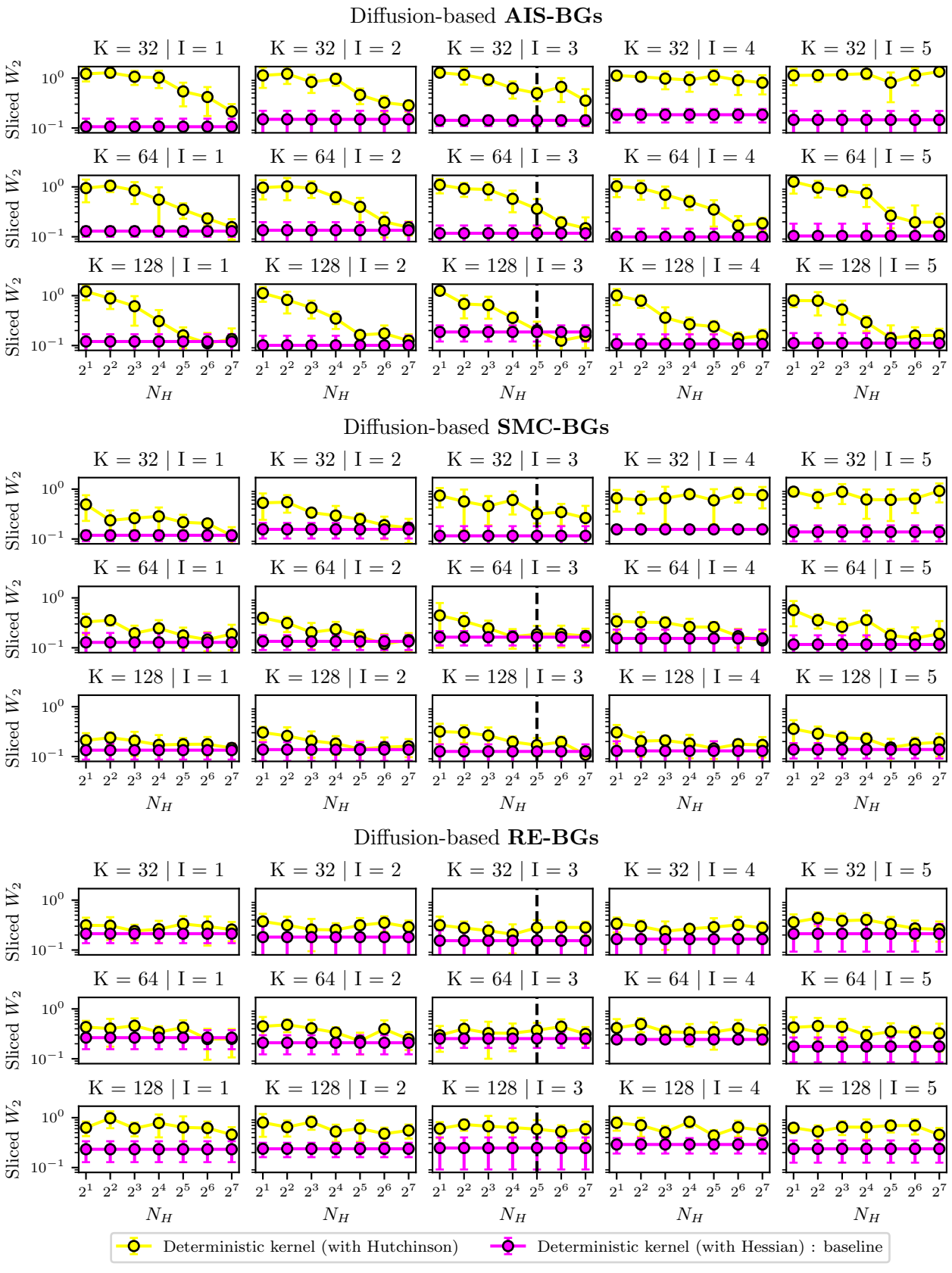

Figure 56: **Sensitivity of all diffusion-based aMC-BGs with deterministic transitions with respect to the number of Hutchinson samples N_H (x-axis) and the truncation order I, across different values of K. (Top)** AIS variant, **(Middle)** SMC variant, **(Bottom)** RE variant. We fix $M = 4$.

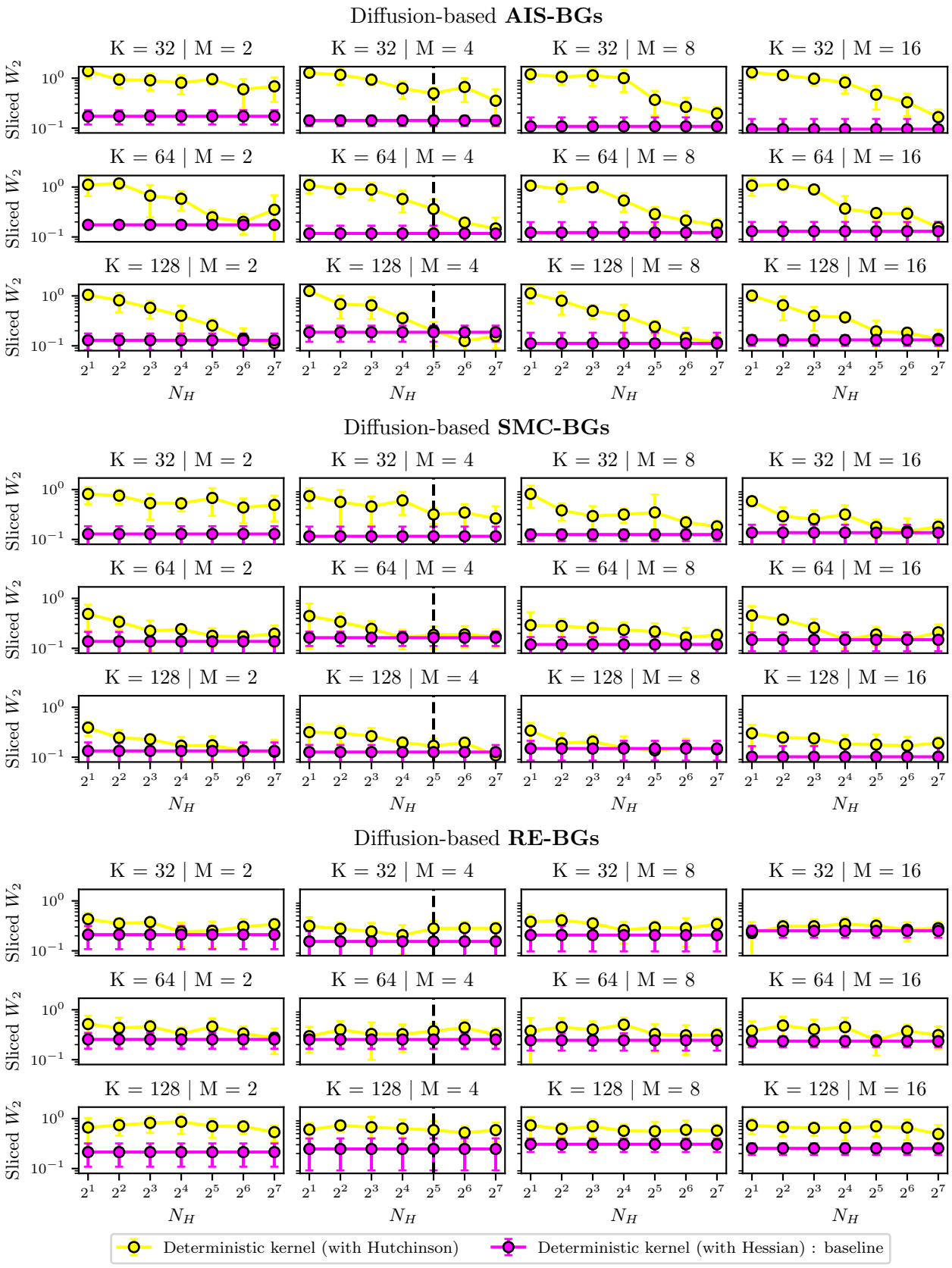

Figure 57: **Sensitivity of all diffusion-based aMC-BGs with deterministic transitions with respect to the number of Hutchinson samples N_H (x-axis) and fixed-point iterations M, across different values of K. (Top)** AIS variant, **(Middle)** SMC variant, **(Bottom)** RE variant. We fix $I = 3$.